# Table of Contents


- Introduction
- General Information
- Getting Started
- What's New in this Release
- Version History
- Programming Interface Overview
  - Building Blocks
  - Data Series and Binding
  - Dashboard Layout
  - Styling and Themes
  - CSS Styling
  - Chart Controls
  - Gauges
    - Overview
    - Scales
    - Ticks and Labels
    - Pointers
    - Ranges
- Tutorials
  - Building Your First Chart
  - Building A Chart With Axes
  - Data Binding
  - Using DateTime Values
  - Add, Style and Customize Axes
  - Labels in Charts
  - Labels on Axes
  - Styling Chart Series
  - Add and Style Legends
  - Grid
  - Tooltips
  - Dashboard Layout
- Reference
  - MindFusion.Charting
    - AnnotationRenderer Class
    - AreaRenderer Class
    - AreaStackRenderer Class
    - Axis Class
    - AxisInfo Class
    - AxisRenderer Class
    - BarContainer Interface
    - BarLayout Enumeration
    - BarModel3D Enumeration
    - BarOverlayRenderer Class
    - BarOverlayRenderer3D Class
    - BarRenderer Class
    - BarRenderer3D Class
    - BarSeries Class
    - BarStackRenderer Class
    - BarStackRenderer3D Class
    - BubbleLabelAlignment Enumeration
    - BubbleRenderer Class
    - CandlestickRenderer Class
    - CandlestickSeriesStyle Class
    - ChartEvent Class

CurveAreaRenderer Class
CurveAreaStackRenderer Class
CurveRenderer Class
DataBoundSeries Class
DateTimeFormat Enumeration
DateTimeSeries Class
EventSeries Class
FunctionSeries Class
FunnelRenderer Class
GridType Enumeration
HitResult Class
LabelKinds Enumeration
LegendController Class
LegendRenderer Class
LineRenderer Class
LineType Enumeration
Margins Class
MixedSeriesStyle Class
Pair Class
PanController Class
PanControllerAnimation Class
PerElementSeriesStyle Class
PerSeriesStyle Class
PieRadarRenderer Class
PieRenderer Class
PieSeries Class
Plot Class
Plot2D Class
Plot2DController Class
Plot3D Class
PlotController Class
Point3D Class
PointSeries Class
PointSeries3D Class
PolarPlot Class
RadarAxisOptions Class
RadarGridType Enumeration
RadarPlot Class
RadarRenderer Class
RadarScatterRenderer Class
RadarType Enumeration
RenderContext Class
RenderContext2D Class
Renderer2D Class
Renderer3D Interface
RotationController Class
RotationControllerAnimation Class
ScatterRenderer Class
ScatterType Enumeration
Series Interface
Series2D Class
Series3D Class
SeriesContainer Interface
SeriesRenderer Class
SeriesStyle Interface
SimpleSeries Class
StackRenderer Class
StepAreaRenderer Class
StepRenderer Class
StockPrice Class

StockPriceSeries Class
TextRenderer Class
TextStyle Class
TextStyleHint Enumeration
Theme Class
ToolTip Class
ToolTip Class
TowerLayout Enumeration
TowerRenderer Class
TowerSegmentShape Enumeration
UniformSeriesStyle Class
Vector Class
WebComponents Class
XAxisRenderer Class
XmlPersistContext Class
YAxisRenderer Class
ZoomController Class
MindFusion.Charting.Commands
Command Class
CompositeCommand Class
DisposableCommand Class
TrackChangesCommand Class
UndoEventArgs Class
MindFusion.Charting.Components
BorderComponent Class
ButtonComponent Class
ButtonDrawEventArgs Class
ButtonDrawEventDispatcher Class
Component Class
ComponentAnimation Interface
ComponentController Interface
ComponentVisitor Class
CursorHint Enumeration
GridColumn Class
GridPanel Class
GridRow Class
HitTestVisibility Enumeration
ImageComponent Class
LayoutAlignment Enumeration
LengthType Enumeration
Orientation Enumeration
Panel Class
RangeSelector Class
RootControl Interface
SimplePanel Class
StackPanel Class
TextComponent Class
Visibility Enumeration
MindFusion.Charting.Controls
AreaChart Class
BarChart Class
BarChart3D Class
BiaxialChart Class
BubbleChart Class
CandlestickChart Class
Chart Class
ChartBase Class
FunnelChart Class
LayoutBuilder Class
LineChart Class

- PieChart Class
- RadarChart Class
- ScatterChart Class
- TowerChart Class

MindFusion.Charting.Gauges
- GaugeRenderer Class
- LinearGaugeController Class
- LinearGaugeRenderer Class
- OvalGaugeController Class
- OvalGaugeRenderer Class

MindFusion.Charting.ThreeD
- Label3D Class
- LabelProjection Class
- Matrix3D Class
- Mesh3D Class
- Model3D Class
- Polygon3D Class
- Projection Interface
- Scene3D Class
- Vector3D Class

MindFusion.Common
- Control Class
- DateFormats Class
- DateSettings Class
- EventDispatcher Class
- IdGenerator Class
- Locale Class
- NotifyCollectionChangedAction Class
- NotifyCollectionChangedEventArgs Class
- NotifyCollectionChangingEventArgs Class
- PropertyEventArgs Class
- UIControl Class
- Unit Class
- UnitType Enumeration

MindFusion.Common.Collections
- IEnumerable Class
- List Class
- ObservableCollection Class

MindFusion.Controls
- Alignment Enumeration
- CancelEventArgs Class
- Canvas Class
- CanvasControl Class
- Control Class
- Disposable Class
- DomUtils Class
- EventArgs Class
- MouseButton Enumeration
- MouseCursors Enumeration
- TickPosition Enumeration
- ZoomControl Class

MindFusion.Drawing
- Border3D Class
- Brush Class
- Color Class
- Color.knownColors Class
- Component Class
- ComponentBase Class
- Container Class
- DashStyle Enumeration

- DrawingUtils Class
- Ellipse Class
- Font Class
- FontStyle Enumeration
- Graphics Class
- GraphicsUnit Class
- Image Class
- ImageAlign Enumeration
- LayoutAlignment Enumeration
- LinearGradientBrush Class
- LineJoin Enumeration
- Path Class
- Pen Class
- Point Class
- Rect Class
- Size Class
- StringAlignment Enumeration
- StringFormat Class
- Text Class
- Thickness Class
- Vector Class
- Video Class
- Visibility Enumeration

MindFusion.Gauges

- Alignment Enumeration
- ArcArea Class
- ArcSegment Class
- BaseGauge Class
- BaseScale Class
- CenterPanel Class
- CustomInterval Class
- DisplayType Enumeration
- Ellipse Class
- Events Class
- FunctionType Enumeration
- GeometryFactory Class
- LabelRotation Enumeration
- Length Class
- LengthType Enumeration
- LinearGauge Class
- LinearScale Class
- LineSegment Class
- MajorTickSettings Class
- MiddleTickSettings Class
- MinorTickSettings Class
- MouseAction Enumeration
- MouseInfo Class
- Orientation Enumeration
- OvalGauge Class
- OvalGaugeStyle Enumeration
- OvalScale Class
- PaintEventArgs Class
- PathFigure Class
- Pointer Class
- PointerShape Enumeration
- PrepaintEventArgs Class
- Range Class
- RoundRectangle Class
- Segment Class
- Tick Class

TickLabel Class
TickSettings Class
TickShape Enumeration
TickType Enumeration
Utils Class
ValueChangedEventArgs Class
ValueChangingEventArgs Class
VisualElement Class
VisualElementContainer Class
WebComponents Class
Online Resources
Technical Support
Licensing
MindFusion Products

# Introduction

MindFusion.Charting for JavaScript is a versatile data visualization library that draws various types of charts, graphs, and gauges. It is implemented in TypeScript/JavaScript and renders graphics objects using the HTML 5 Canvas element. For Canvas support and heavy-duty geometric and layout calculations done in JavaScript, MindFusion.Charting for JavaScript requires a latest-generation browser. By providing your software with ready to use capabilities for laying out, rendering and feeding data into dashboards, MindFusion.Charting saves you hundreds of hours coding and debugging.

With MindFusion.Charting for JavaScript you can:

- Draw bar, line, area, scatter, bubble, pie, doughnut, funnel, radar and polar charts;
- Draw highly customizable oval and linear gauges;
- Draw multiple plots;
- Draw more than one graphics per plot;
- Draw multiple X and Y axes;
- Draw a customizable background grid;
- Draw 3D bar charts in stack, overlay or side-by-side format;
- Bind lists or objects to chart's properties;
- Add multiple chart legends and choose their settings;
- Show multiple image or text components;
- Add label prefixes and suffixes to chart numbers;
- Set the minimum and maximum value for any of the chart's axes;
- Set up dynamic layout of dashboard components;

MindFusion.Charting for JavaScript, as a component, has been specially designed to be very easy to use, with intuitive programming interface and simple, elegant software architecture.

# General Information

### The MindFusion.Charting for JavaScript component

- Redistribution is royalty free
- Can be purchased with source code
- Is written in JavaScript
- Draws via the HTML5 Canvas element

### Dashboard

- Arrange plots and gauges in a dashboard
- Dynamic layout system
- Show multiple axes on each side of a plot
- Text and image components
- Each component can be placed inside border
- Show multiple legends

### Line/Area charts

- No limits for the count of graphics per chart
- Straight lines or curves
- Separate as well stacked graphics
- Step charts
- Scatter graphics with shapes at data points
- Multiple axes
- Associate data series with axis

### Bar charts

- Drawing of bar series in rows or columns
- Stacked bars
- Groups and clusters of bars
- Candlestick charts
- Bar width relative to available space
- Labels inside and/or at the top of bars
- 3D bar charts
- Five different axes labels types
- Multiple axes

### Pie charts

- Doughnut charts
- Several options for count and position of detached slices
- Customizable start angle
- Different label types
- Labels inside and outside pie segments
- Adjustment of position of labels inside pies

### Radar charts

- Polar charts
- Spider web background
- Three combinable appearance styles
- Pie-radar charts
- Custom labels inside and outside the chart
- Labels at chart segments
- Pens for all inner and outer axes
- Brushes for chart segments
- Custom-set background sections and brushes for them
- Customizable position of the inner axis

### Other chart types

- Candlestick charts
- Bubble charts
- Scatter charts
- Funnel charts

### Gauge types

- Oval gauges
- Horizontal and vertical linear gauges

### Legend

- Adjustable position and alignment
- Possibility to set the legend width
- Customizable background
- Custom-set font and brush for the labels

### Grid

- Several different grid types
- Customizable grid width
- Custom-set pens at grid lines/stripes
- Custom-set pens/brushes for grid lines/stripes
- Grid labels with pointers at chart axes

### Programming

- Numerous methods and properties for customizing the chart's appearance
- Strong design-time support to facilitate appearance customization
- Tested programming examples with all properties and methods

### User interaction

- Pan and zoom
- Rotate pie and radar charts
- Hit testing
- Move the legend

### Appearance

- Customize all pens and brushes
- CSS styling
- Support for themes and a theme editor tool
- 3D charts
- Horizontal and vertical charts
- Customizable fonts
- Customizable tooltips
- Possibility to set the chart's position in bigger graphics

### Chart data

- Set with numerical data arrays or json formatted database-like objects
- Feed data to charts by implementing Series interface instead of copying values
- Custom annotations
- Date values supported through DateTimeSeries class

### Chart axes

- Several different types of labels
- Specify minimum and maximum axis values or auto scale
- Custom text
- Several rows/columns of labels for multi-graphics charts
- Division markers
- Several positions and orientations of axis labels
- Explanation labels
- Customize colors of all axis elements

# Getting Started

### HTML elements

The various charting controls rely on the HTML5 Canvas element to render graphics on web page.

**HTML**

```
<div style="position: absolute; left: 0px; top: 0px; bottom: 0px; right: 0px;">
    <canvas id="areaChart" style="width: 100%; height: 100%; display: block;">
    </canvas>
</div>
```

### Modules and namespaces

The charting library is available in three different distributions - ES modules, CommonJs modules and UMD. Each of them splits the code into six modules or namespaces. The UMD module exposes a global MindFusion namespace object, containing child namespaces that match the modules from the other two distributions, namely Controls, Drawing, Common, Common.Collections, **Charting** and Gauges.

For more information on JavaScript modules see https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Modules.

### Loading UMD namespaced scripts

UMD is a pattern of universal module definition for JavaScript modules. These modules are capable of working irrespective of the executing environment. In web browsers, it exposes the module functionality to the global scope.

In order to load the UMD scripts, add references to files from charting distrbution's /umd folder using <script> tags. Note that the order of the scripts matters: controls.js must be loaded after drawing, and charting.js must be loaded after gauges.

**HTML**

```
<script src="umd/drawing.js" type="text/javascript"></script>
<script src="umd/controls.js" type="text/javascript"></script>

<script src="umd/common.js" type="text/javascript"></script>
<script src="umd/common-collections.js" type="text/javascript"></script>

<script src="umd/gauges.js" type="text/javascript"></script>
<script src="umd/charting.js" type="text/javascript"></script>
```

After these scripts are loaded, you can access the global MindFusion namespace object from your code:

**JavaScript**

```
var AreaChart = MindFusion.Charting.Controls.AreaChart;

var myChart = new AreaChart(document.getElementById('areaChart'));
```

### Loading ES6 modules via importmap

JsChart's ES6 modules conform to modern JavaScript module definition that supports native import and export statements. You could import these modules directly, without using any additional module/package libraries or tools, by defining your script file as a module via <script type="module"> tag:

**HTML**

```
<script src="app.js" type="module"></script>
```

In order to make the chart's ES6 modules available for importing, add the following importmap, referencing the scripts, contained in distribution's /esm folder. The purpose of the importmap is to define a map of module names which allow using import specifiers such as import {x} from "@mindfusion/charting" (instead of a relative path to the corresponding script file), both internally and in in your code.

**HTML**

```
<script type="importmap">
{
    "imports":
    {
        "@mindfusion/controls":"./esm/controls.js",
        "@mindfusion/drawing":"./esm/drawing.js",
        "@mindfusion/common":"./esm/common.js",
        "@mindfusion/common-collections":"./esm/common-collections.js",
        "@mindfusion/gauges":"./esm/gauges.js",
        "@mindfusion/charting":"./esm/charting.js"
    }
}
</script>
```

Now you can consume the modules you need by importing them into your app.js:

**JavaScript**

```
import * as Charting from '@mindfusion/charting';
import * as Drawing from '@mindfusion/drawing';
```

```
var barChart = new Charting.Controls.BarChart(
    document.getElementById("barChart"));
```

Note that at the time of writing this, support for importmap is yet to be implemented in all browsers (https://caniuse.com/import-maps). Notably, importmap is not yet supported by Firefox, but it is under development from what we know.

### Loading npm packages

There are six npm packages that provide the functionality of MindFusion.Charting, each containing the corresponding module. To use JsChart in a project, you only need to install the @mindfusion/charting package, the rest will be automatically added as dependencies:

**npm**

```
npm install @mindfusion/charting
```

Each of the packages contains three distributions - ES module, CommonJs module and UMD, that you can use depending on your project setup.

To consume a module as a CommonJS module, use the 'require' statement. The CommonJs version of the module will be loaded from the /dist/cjs folder, as specified in the package.json's main field.

**JavaScript**

```
const Charting = require('@mindfusion/charting');
const { LineChart } = require('@mindfusion/charting');
```

To consume a module as an ES6 module, use the 'import' statement. The ES6 version of the module will be loaded from the /dist/esm folder, as specified in the package.json's module field.

**JavaScript**

```
import * as Charting from '@mindfusion/charting';
import { LineChart, AreaChart } from '@mindfusion/charting';
```

### TypeScript support

A d.ts declaration file is available for every package. For more information on declaration files see https://www.typescriptlang.org/docs/handbook/2/type-declarations.html#dts-files.

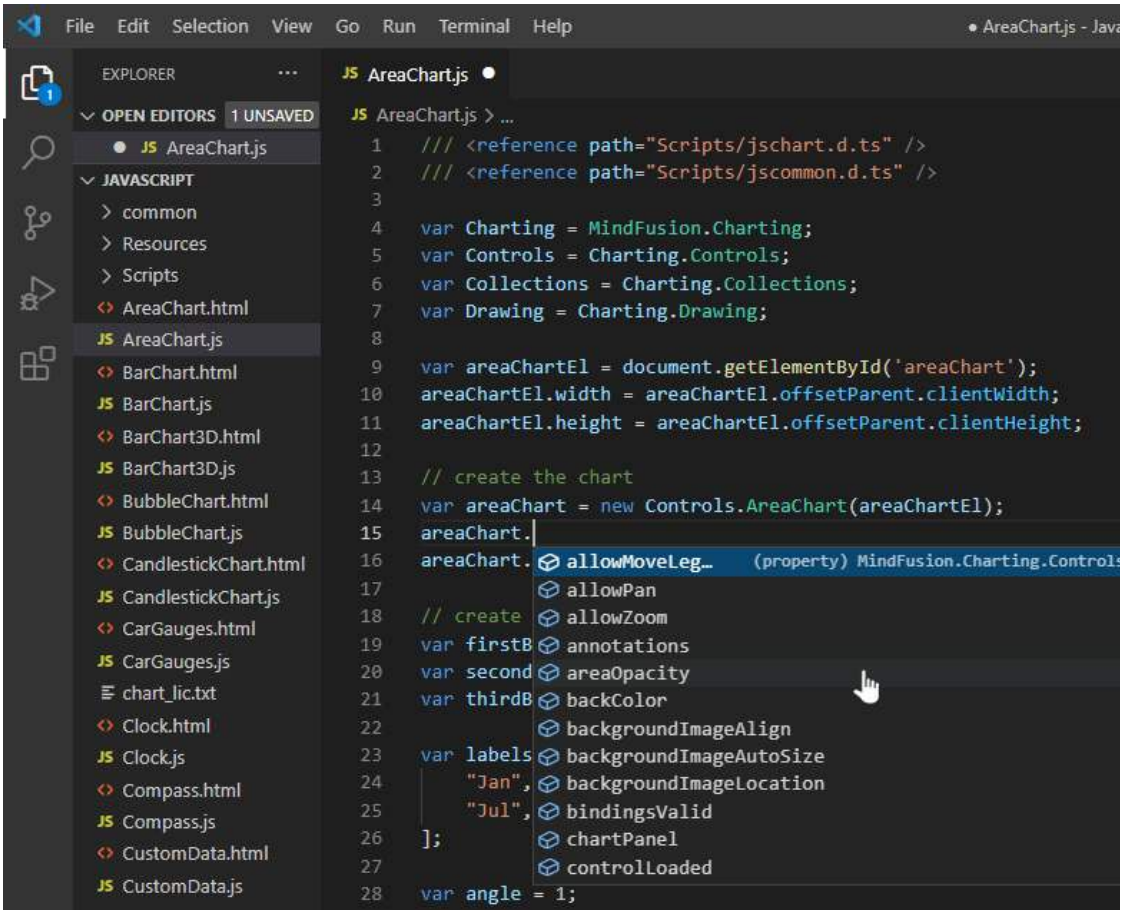
### Set up code completion

The library provides code completion and hints through d.ts declaration files, supported by most popular IDEs. In order to enable the feature, you need to add a commented reference to the TypeScript *.d.ts files that are included in the library distribution. The most commonly used API members are in the charting file, but we advise that you add references to the auxiliary files as well:

**JavaScript**

```
/// <reference path="Scripts/charting.d.ts" />
/// <reference path="Scripts/gauges.d.ts" />
/// <reference path="Scripts/controls.d.ts" />
/// <reference path="Scripts/drawing.d.ts" />
/// <reference path="Scripts/common.d.ts" />
/// <reference path="Scripts/common-collections.d.ts" />
```

Once the references are loaded, you should be able to see code suggestions for the applicable members together with their type. All members should be visible: methods, properties, classes etc.




The distribution also includes JsChart-vsdoc.js file providing IntelliSense documentation in VSDoc format. If your programming environment supports it, you could add a reference to that file instead of d.ts ones:

**JavaScript**

```
/// <reference path="Scripts/JsChart-vsdoc.js" />
```

# What's New in this Release

The list below describes recent changes and additions to MindFusion.Charting for JavaScript:

## New in version 2.1

### CSS styling

The library now features CSS styling system that lets you customize the appearance of charts, graphs, and gauges entirely through external stylesheets. By leveraging CSS Custom Properties (Variables), you can define backgrounds, axes, fonts, labels, and even complex series color palettes directly in CSS (e.g., **--mfc-plot-background** or **--mfc-common-series-fill-0**). This eliminates verbose JavaScript appearance configurations via Theme class, and lets you define chart appearance in application-wide stylesheets.

### Miscellaneous

- The numberFormat property lets you format axis coordinates via **Intl.NumberFormat** object.
- LineChart and LineRenderer rendering improvements.

### API changes

To avoid naming conflicts with our upcoming Dashboard library, which provides a broader set of dashboard widgets and functionality, the charting library's **Dashboard** control has been renamed to ChartBase.

## New in version 2

### ES6 modules

JsChart distribution now includes a set of ES6 modules, in *Scripts/esm* folder, that lets you import the classes from respective scripts. The distribution still includes ES5-compatible scripts located in Scripts/umd folder, which are transpiled from current ES6 code, and whose classes are exposed as members of global MindFusion namespace object as in previous versions.

### Miscellaneous

- The FunctionSeries class -represents a series that calculates its values from provided formula.
- Area charts are now rendered relatively to axis' Origin too.

### API changes

MindFusion.Charting source code has been refactored and is now dependent on the following shared MindFusion libraries: drawing, controls, common, common.collections and gauges.

- For projects that access MindFusion namespace from the global namespace, check the "Loading UMD namespaced scripts" section in Getting Started guide.
- For projects that use module loaders, check the various "Loading ... modules" sections in **Getting Started** guide.

# Version History

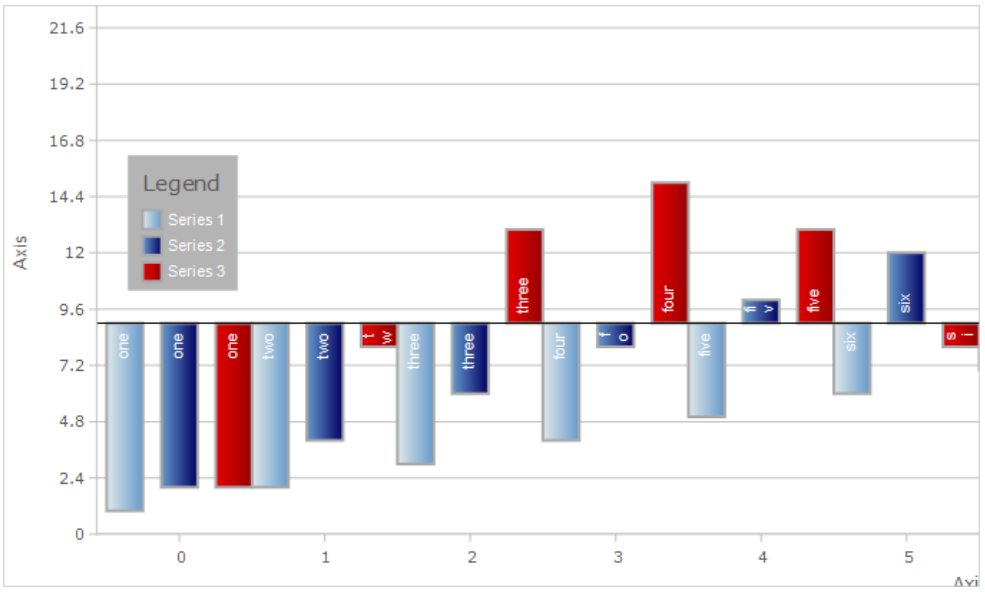
The list below describes past changes and additions to MindFusion.Charting for JavaScript:

## New in version 1.2

#### Axis origin

The origin property of the Axis class lets you specify the origin of an axis. If set to a non-null value, the component renders an axis line inside the plot. Bars are drawn below that axis line if their values are smaller than the origin. For stacked bar charts, each bar length corresponds to the difference between bar's value and origin.

The Plot.drawOrigins and Plot2D.drawOrigins methods allow you to draw the axis origin lines when rendering the chart plot by yourself.



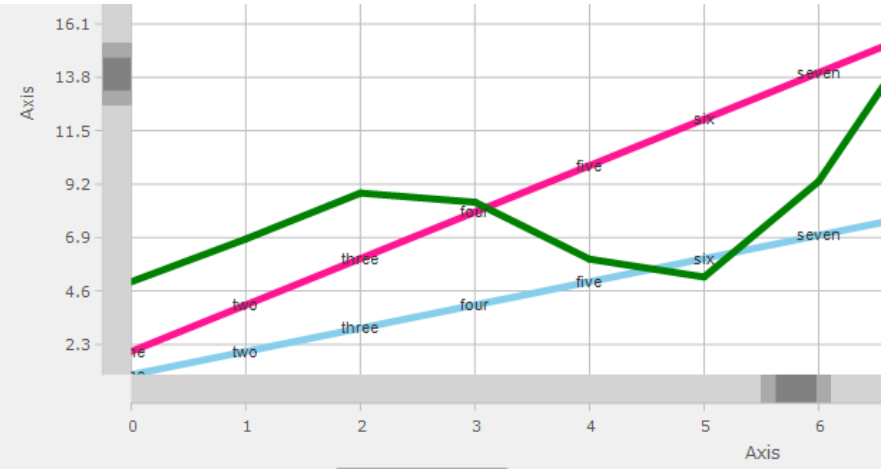

### Miscellaneous

- A new SimpleSeries constructor override allows you to specify a list with tooltips for the newly created Series.
- The new drawLeftFromPoint method allows you to render text, which is left-aligned compared to a given Point.
- The new methods fromObject and toObject of the Font class allow you to deserialize and serialize Font instances from/to Json objects.
- innerLabelRotation, outerLabelRotation and stackOuterLabels properties give you better control over the rendering of labels in bar charts
- All **Series** can accept now simple JavaScript array-s as arguments instead of Collections.List instances
- The ToolTip class is greatly extended with many new properties that allow you to customize the apparance and position of tooltips
- The Color.knownColors field lists all standard CSS color names
- Brush and Pen instances can be created with simple strings that specify the HTML code of the color as argument instead of Color objects.
- The yLabelAlignment property of BiaxialChart specifies horizontal alignment of Y-axis labels.
- Texts are now properly underlined when FontStyle.Underline is set.
- Fixed a bug that caused a stacked BarChart sometimes to crash.

## New in version 1.1.1

### Interactive range selection

- RangeSelector is a scrollbar-like component that lets users scroll or resize currently visible range of chart's data. BiaxialChart shows a horizontal range selector when its showXRangeSelector property is enabled, and a vertical one when showYRangeSelector is enabled. Scroll ranges are specified using xScrollRangeMin and xScrollRangeMax properties for the X axis, and yScrollRangeMin and yScrollRangeMax properties for Y axis.




### Miscellaneous

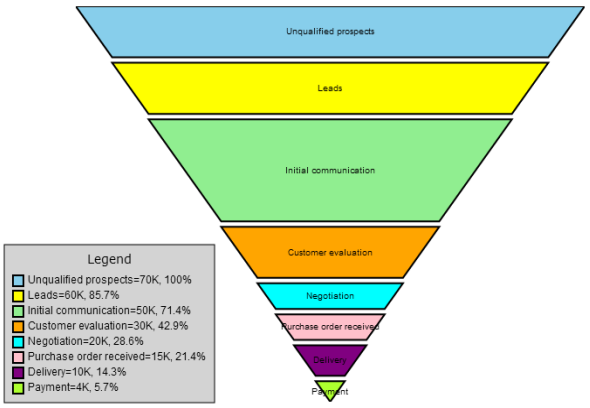
- Legends can display all elements of a series instead of series title values, which is useful for pie and funnel charts. To enable that mode, set the showSeriesElements of LegendRenderer to true, and assign the kind of labels to display to the elementLabelKind property.
- FunnelRenderer/FunnelChart bottomWidth property has been renamed to stemWidth.
- Event listeners' signatures used by IEventDispatcher implementations now include a sender argument.
- showHighlight and showToolTips properties let you disable highlights and tooltips.

## New in version 1.1

### FunnelChart

Funnel charts represent the stages of a process and show relative values or measurements associated with each stage. In MindFusion.Charting for JavaScript funnel charts can be created by utilizing the FunnelChart or FunnelRenderer classes. **FunnelChart** inherits from the base Chart class and offers additional customization through its segmentSpacing and stemBase properties.

### Theme XML serialization




Themes now can be serialized to and from XML through the use of the respective loadFrom and saveTo methods.

Miscellaneous

- labelRotationAngle property added to AxisRenderer. xAxisLabelRotationAngle and yAxisLabelRotationAngle properties added to BiaxialChart.
- PieChart now properly shows outer labels.
- LegendRenderer now properly shows border.
- Fixed hit-testing in **pie** and radar charts.
- Bubbles in bubble charts are now properly centered on their values.
- Charts' zoom buttons scaling issue fixed.

# Programming Interface Overview

# Building Blocks

The MindFusion Charting library lets you build dashboards by composing various kinds of elements in a hierarchy of Component objects. The main types are:

- Panel and its derived classes for defining dynamic layout.
- Plot and its derived classes for drawing chart graphics.
- AxisRenderer and derived classes for drawing chart axes.
- GaugeRenderer and derived classes for drawing gauges.
- LegendRenderer displays titles and colors of specified series.

In addition, you can use TextComponent, ImageComponent and BorderComponent to add decorative elements to the dashboard, such as chart title or company logo.

**Plot** components can display multiple SeriesRenderer objects for matching coordinate system types. For example, you can mix bar and line graphics in 2D Cartesian system by adding a LineRenderer and a BarRenderer to a Plot2D, while a RadarPlot can contain PieRadarRenderer to draw pie-radars and a RadarRenderer to draw polygons in polar coordinate system. Different plots also display different types of grids, as listed in GridType and RadarGridType enumerations.

A single instance of a **SeriesRenderer** -derived class can usually draw data from multiple Series, so you need just one **BarRenderer** to display bars from several series. Series renderers arrange the elements from several series in appropriate groups. For example, **BarRenderer** draws elements at same index from multiple series in side-by-side groups, while BarStackRenderer arranges them in a stack.

# Data Series and Binding

SeriesRenderer classes used to draw chart graphics read data through the Series interface. You could implement this interface in your own model classes to avoid duplicating data. Alternatively, if a model class is sealed but provides API to access its data, you could implement the interface in an adapter class.

The library includes several pre-defined series classes that let you specify data via List or array objects. For example, BarSeries lets you specify lists of values, inner labels and top labels, and PointSeries lets you specify data as a list of points.

The DataBoundSeries class can take a data-source JSON formatted object or a list of JavaScript objects, and read their values using reflection on fields specified via properties such as XDataField and InnerLabelsDataField. Setting the DataSource property of a Chart control will automatically create **DataBoundSeries** for all fields listed in the chart's ***Fields** properties such as XDataFields. You can freely mix explicitly created **Series** objects with ones generated automatically from **DataSource**.

# Dashboard Layout

The ChartBase control implements dashboard layout via Panel classes that can contain multiple child Component objects. SimplePanel arranges its children on top of each other and can be useful for hosting overlapping components with transparent regions. StackPanel arranges its children in horizontal or vertical stacks. GridPanel arranges its children in a grid of rows and columns. The target cell of a component inside a **GridPanel** is specified via its GridRow and GridColumn properties.

The hierarchy of dashboard's components starts from its RootPanel. By default, **RootPanel** contains an ImageComponent for drawing BackgroundImage, and the LayoutPanel where you should add objects that participate in layout and should be placed relatively to each other. Chart and its derived classes automatically add PlotPanel to **LayoutPanel** to host Plot, AxisRenderer and **ImageComponent** for PlotImage. You can add your own components and panels to any of the pre-defined panels listed above.

The LayoutBuilder class and LayoutBuilder property of the **ChartBase** let you add commonly used layout fragments to the component hierarchy. It also takes care of setting various alignment and orientation properties, such as PlotLeftSide of YAxisRenderer passed as left axis to CreatePlotAndAxes method.

# Styling and Themes

Values of appearance properties can come from several places in the component hierarchy. SeriesRenderer -derived objects can use attributes from their local SeriesStyle, from plot's **SeriesStyle**, or from the ***Series** properties in current Theme. Component classes use either their local properties or ones defined in the theme. By default, appearance properties in **SeriesRenderer** and **Component** classes have **null** values, which makes the drawing code use values from the theme.

The **SeriesStyle** interface defines how **SeriesRenderer** objects get appearance attributes for series elements, where each element of each series can generally be drawn in distinct style from others. You could implement that interface in your own class to calculate stroke and fill colors dynamically based on series and element indices. For example, you could paint climbing values in green and falling values in red if drawing financial charts, or calculate a blue-red gradient if drawing temperatures.

There are several **SeriesStyle** classes provided out-of-the-box that implement various levels of distinctness for graphical elements drawn for each data item. UniformSeriesStyle uses same attributes for all elements of all series. PerSeriesStyle contains different attributes for different series, where all elements in the series are drawn with same attributes. PerElementSeriesStyle allows customization of each element of each series. MixedSeriesStyle allows a mix of uniform, per-element and per-series attributes.

# CSS Styling

MindFusion.Charting Theme properties can be set using CSS Custom Properties (Variables). Every property of the internal  class is automatically mapped to its CSS variable equivalent by converting its camelCase name to kebab-case, with added "--mfc-" prefix. For example:

- plotBackground: --mfc-plot-background
- axisLabelsFontSize: --mfc-axis-labels-font-size
- uniformSeriesStroke: --mfc-uniform-series-stroke

## Supported Property Types

The CSS bridge translates CSS property values into internal JS engine types:

| CSS Value Example | Internal Type | Notes |
|---|---|---|
| red, #ff0000, rgb(255,0,0) | Brush / Color | Handled natively. Supports gradients and transparency. |
| 12px, 4mm, 10pt, 0.5in | GraphicsUnit | Automatically scaled using GraphicsUnit convert method. |
| bold, italic, underline | FontStyle | Combined as flags. |
| Dash, Dot, Solid | DashStyle | Resolved via internal DashStyle enum mappings. |

## Styling Series

For properties containing lists (like Series collections), the engine uses an indexed naming convention:

### Common Series Palettes

Applied to elements sequentially across series:

- --mfc-common-series-fill-0: Brush for Series 1.
- --mfc-common-series-fill-1: Brush for Series 2.
- --mfc-common-series-fill-2: Brush for Series 3.

### Multi-Dimensional Series Palettes

For nested lists, use the multi-indexed format --mfc-series-fill-[SeriesIndex]-[ElementIndex]:

- --mfc-series-fill-0-0: First bar of the first series.
- --mfc-series-fill-0-1: Second bar of the first series.
- --mfc-series-fill-1-0: Third bar of the second series.

### Complete Example

CSS

```
#areaChart
{
    --mfc-plot-background: #f0f0f0;
    --mfc-legend-background: #fff0e0;
    --mfc-legend-border-stroke: #333333;
    --mfc-legend-title-font-size: 40px;
    --mfc-axis-title-font-name: 'Verdana';
    --mfc-axis-labels-font-name: 'Verdana';
    --mfc-data-labels-font-name: 'Verdana';
    --mfc-axis-title-font-size: 16px;
    --mfc-axis-labels-font-size: 14px;
    --mfc-data-labels-font-size: 14px;

    /* Series palette */
    --mfc-common-series-fill-0: #ff6666;
    --mfc-common-series-fill-1: #66ff66;
    --mfc-common-series-fill-2: #6666ff;

    --mfc-common-series-stroke-0: #cc0000;
    --mfc-common-series-stroke-1: #00cc00;
    --mfc-common-series-stroke-2: #0000cc;

    --mfc-common-series-stroke-thickness-0: 2px;
    --mfc-common-series-stroke-thickness-1: 2px;
    --mfc-common-series-stroke-thickness-2: 2px;
}
```

# Chart Controls

The library includes several pre-defined JavaScript controls deriving from the Chart class. Each concrete class is a ChartBase with a single Plot inside and appropriate default SeriesRenderer. Charts that draw in Cartesian coordinate system also include an XAxisRenderer and YAxisRenderer.

You can feed data into **Chart** controls by setting either their Series or DataSource properties. The former lets you assign one or more series that provide data via the Series interface. The latter is used in conjunction with properties such as XDataFields, InnerLabelsDataFields etc.

Concrete chart types available at this time include:

- AreaChart
- BarChart
- BarChart3D
- BubbleChart
- CandlestickChart
- FunnelChart
- LineChart
- PieChart
- RadarChart
- ScatterChart
- TowerChart

# Gauges

## Overview

MindFusion.Charting for JavaScript provides two type of gauge controls – oval and linear. The two types of gauges are represented by the OvalGauge and LinearGauge classes respectively. The purpose of the gauge controls is to visualize various quantities in two layouts – circular and linear.

### Structure

The most important component of each gauge control are the scales contained within it. The scales are collections of gauge elements such as ticks, labels, pointers, and so on. The gauges are merely shells, which provide visual decoration to the shells nested within them. Scales can be added to the gauges programmatically or in design-time through the Scales collection. The image below demonstrates an oval gauge with a single scale added to it.




Removing the scale from the above gauge will result in an empty gauge control.

### Scales

As previously mentioned the scales can contain ticks, labels, pointers, ranges and other elements. There are two types of scale corresponding to the two types of gauges – OvalScale and LinearScale, which share a common base class – BaseScale. The oval scale distributes its elements in a circle and the linear scale distributes its elements along a horizontal or vertical axis. The scales are responsible for arranging their contents. They also provide user interaction.

The following example adds a pointer to the gauge defined in the previous section:




To learn more about gauge scales, refer to the Scales topic.

### Composition

Gauge controls can contain more than just one scale. A common scenario is an oval gauge containing several concentric scales displaying different measures. The following code creates an oval gauge control with two concentric scales:

**JavaScript**

```
var scale = new OvalScale();
ovalGauge.addScale(scale);

scale = new OvalScale(ovalGauge);
scale.fill = "transparent";
scale.stroke = "transparent";
scale.scaleRelativeRedius = 0.25;
scale.functionType = FunctionType.Logarithmic;
scale.middleTickSettings.showLabels = false;
scale.middleTickSettings.showTicks = false;
scale.majorTickSettings.numberPrecision = 0;
scale.majorTickSettings.tickShape = TickShape.Rectangle;
scale.majorTickSettings.labelOffset = new Length(0);
scale.majorTickSettings.fontSize = new Length(10, LengthType.Relative);
scale.majorTickSettings.fill = "gray";
```

The following image demonstrates the gauge:



## Customizing Gauges

The customization of the gauge controls is achieved by performing custom painting of various gauge elements. This is done by handling one or more of the paint events. Two events raised for each gauge element – a **Prepaint*** event and a ***Paint** event. The first event is raised before the default rendering of the element and can be used to paint before and/or cancel the default rendering. The second event is raised after the default rendering has completed. For example, the events raised for the gauge background are PrepaintBackground and PaintBackground. For a complete list of the available paint events, see Events.

## Oval Gauge Styles

There are several built-in gauge styles for quadratic and semicircular gauges. These styles can be enabled through the Style property of the OvalGauge class. The image below illustrates the QuadraticNW style:



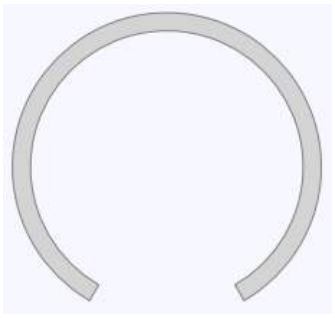
# Scales

As discussed in the Overview topic, the scales are the most important component of a gauge. The scales display ticks and labels, they arrange pointers and also maintain ranges and other auxiliary elements. There are two types of scales – oval and linear. The oval scale distributes its elements in a circle, the linear scale distributes its element on a vertical or horizontal axis depending on its orientation. The oval and linear scales are represented by the OvalScale and LinearScale classes respectively. These classes share a common base class – BaseScale and apart from the difference in their layout logic they are very similar.

## Customization

### Geometry

The scales display a geometry according to its type. The elements displayed in the scale are aligned and arranged relative to the scale geometry. The geometry is either circular, for oval scale, or trapezoidal, for linear scales. The following figures show the geometries of oval and horizontal and vertical linear scales.

| Oval | Horizontal Linear | Vertical Linear |
| --- | --- | --- |
|  |  |  |

The scale geometry can be customized through various properties. The fill and stroke of the geometry are defined by the Fill and Stroke properties. The start and end width are specified through the StartWidth and EndWidth properties. These properties are of type Length and can express relative values. See the **Metrics** section at the end of this topic for additional information regarding this. The start and end angles of the oval geometry can be specified through the StartAngle and EndAngle properties of the OvalScale class respectively. Both properties are expressed in degrees.

### Data

The MinValue and MaxValue properties can be used to define the range of values displayed on the scale. The distribution of the values along the scale geometry is determined by the FunctionType property and can be Linear, **Logarithmic** or **Custom**. The base of the logarithmic distribution is specified through the LogarithmBase property. The custom value distribution function is supplied through the CustomFunction and ReversedCustomFunction properties.

### Custom Painting

The appearance of the scale can be further customized by handling one of the PrepaintScale and PaintScale events. The former is raised before the default scale rendering, the latter is raised afterwards. The following code demonstrates a sample **PaintScale** event handler which draws a star on top of the scale:

**JavaScript**

```
var path = new PathFigure(
    "M0.5,0.02 L0.62,0.36 L0.98,0.36 L0.7,0.58 L0.8,0.92 L0.5,0.72 L0.2,0.92 L0.3,0.58
L0.02,0.36 L0.38,0.36 LM0.5,0.02");
path.stroke = "black";
path.fill = "lime";

var size = e.element.renderSize;
var shapeSize = 32;
path.margin = new Thickness(size.width / 2 - shapeSize / 2, size.height / 4,
```

```
    size.width / 2 - shapeSize / 2, size.height - size.height / 4 - shapeSize, false);

e.paintVisualElement(path, e.element.renderSize);
```

The above method will render a scale similar to the one below:




## Scale Elements

Various elements can be added to a scale, including ticks, labels and pointers.

### Ticks and Labels

The ticks and labels represent cornerstones along the scale, usually round values. The ticks and labels are created and added to the scales implicitly according to the specified settings. There are three categories of ticks and labels – major, middle and minor. Each category has corresponding settings, exposed respectively by the MajorTickSettings, MiddleTickSettings and MinorTickSettings properties of the BaseScale class. These settings provide properties for specifying the number of ticks along the scale, their alignment and appearance, the format of their associated label, and more. For additional information about ticks and labels refer to the Ticks and Labels topic.

### Pointers

The pointers represent needles or similar types of indicators within the scale. They are used to visualize a current measurement within the scale. For example, in a car speedometer the pointers are used to indicate the current speed. Pointers can be added to the scale through the Pointers property. Pointers expose numerous properties that can be used to customize their appearance. For additional information check the Pointers topic.

### Ranges

The ranges emphasize portions of the scale which represent particular interest. An example of a range is the red zone of a car cyclometer. Ranges are added to the scale through the Ranges property. For extensive information on how to use and setup ranges, refer to the Ranges topic.

## Interaction

Users can interactively adjust the value of the pointers in the scale by clicking anywhere within the scale bounds. Interactivity only applies to pointers with IsInteractive property set to **true**. When the value of a pointer is changed interactively, the ValueChanged of the pointer is raised.

## Metrics

The sizes of various scale elements can be specified either relatively to the size of the scale or as absolute values. This applies to all properties of type Length. The following example defines a pointer with relative width and height:

**JavaScript**

```
var pointer = new Pointer();
pointer.pointerWidth = new Length(100, LengthType.Relative);
```

```
pointer.pointerHeight = new Length(50, LengthType.Relative);
```

Relative sizes are calculated differently depending on the type of scale. In oval scales relative sizes are calculated based on the radius of the scale. Therefore the above pointer will extend from the center to the outer edge of the scale. In linear scales sizes are based on the width or height of the scale depending on the scale orientation.

# Ticks and Labels

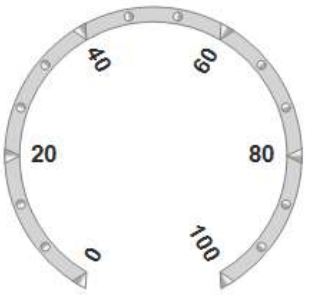
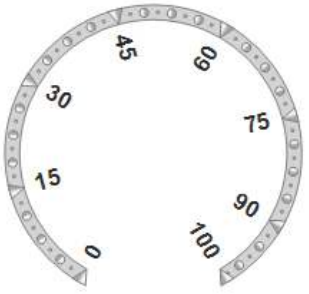
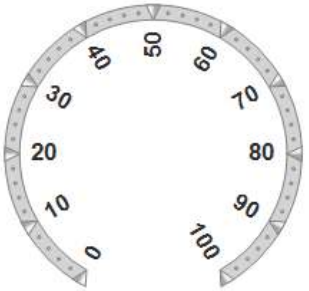
There are three categories of ticks and labels in a scale – major, middle and minor. Each category is governed by a separate settings object in the BaseScale class – MajorTickSettings, MiddleTickSettings and MinorTickSettings.

The number of ticks from a specific category can be specified in either of two ways:

- By setting the Count property to the desired number of ticks.
- By setting the Step property to the desired quantity between adjacent ticks.

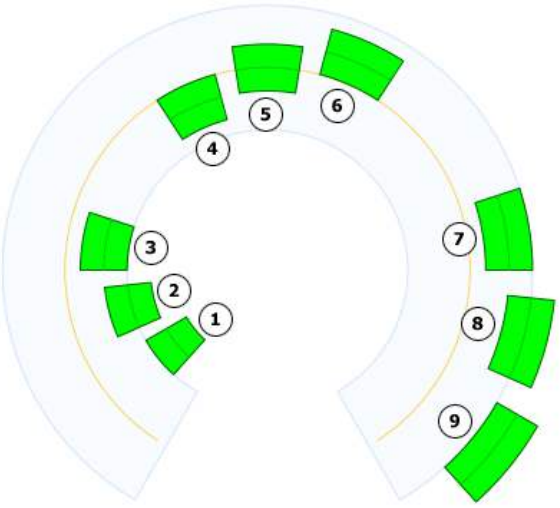
If both of the above properties are specified **Step** will take precedence. The ticks and/or labels in a particular category can be hidden through the ShowTicks and ShowLabels properties of the respective settings objects.

Here is a table with some examples illustrating various combinations of the above properties, applied on an oval scale with MinValue set to 0 and MaxValue set to 100:

| Settings | Scale |
|---|---|
| MajorTickSettings<br>Count = 5<br><br>MiddleTickSettings<br>Count = 3<br>ShowLabels = false<br><br>MinorTickSettings<br>ShowTicks = false |  |
| MajorTickSettings<br>Step = 15<br><br>MiddleTickSettings<br>Count = 4<br>ShowLabels = false<br><br>MinorTickSettings<br>Count = 2 |  |
| MajorTickSettings<br>Count = 10<br><br>MiddleTickSettings<br>Count = 1<br>ShowLabels = False<br><br>MinorTickSettings<br>Step = 2 |  |

### Alignment

The alignment of ticks and labels relative to the scale geometry is specified through the TickAlignment and LabelAlignment properties. The following scheme illustrates the possible alignment modes in an oval scale:



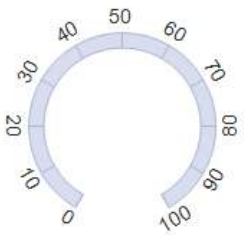
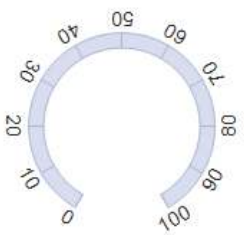
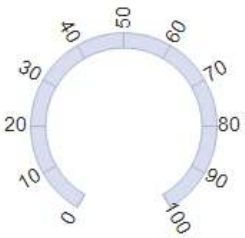
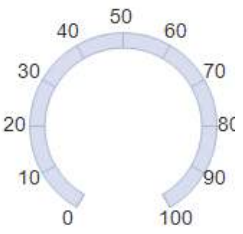

The pale blue lane at the background represents an oval scale, the orange line represents the center of the scale and the green objects are the elements being aligned to the scale using different alignment modes. The center of each element is illustrated in light green color. The alignment modes used for each of the elements in the scheme is specified below:

- InnerInside
- **InnerCenter**
- **InnerOutside**
- **CenterInside**
- **TrueCenter**
- **CenterOutside**
- **OuterInside**
- **OuterCenter**
- **OuterOutside**

The position of the aligned elements can be further adjusted through their offset, specified respectively by the TickOffset and LabelOffset properties.

## Label Rotation

The following schemes illustrates all possible label rotation modes in an oval scale.

| Auto | BaselineAwayFromCenter | BaselineToCenter | None |
|---|---|---|---|
|  |  |  |  |



## Appearance

The fill and stroke of the ticks are specified through the Fill and Stroke properties. The TickShape specifies the shape of the ticks. There are several predefined shapes available. The tick appearance can can be customized by handling the PrepaintTick or PaintTick events. The size of the ticks is controlled by the TickWidth and TickHeight properties and can be specified as either relative or absolute value.

The foreground color of the labels can be specified through the LabelForeground property. The size of the label font is set through the FontSize property. **FontSize** is of type Length and can express relative sizes. The other font attributes of the labels are specified from the FontFamily, **FontSize** and FontStyle properties of the containing scale.

### Custom Intervals

It is possible to customize the appearance of the ticks in a specific interval by using CustomInterval objects. To initialize the custom interval, set its MinValue and MaxValue properties to reflect the range of ticks this interval will apply to, then use its Fill and Stroke properties to specify explicit fill and stroke for the ticks in this interval. Finally add the custom interval to the CustomIntervals collection of the respective TickSettings.

Below is an example of a custom interval added to an existing scale. The custom interval specifies red background for all major ticks in the range [10..50].

**JavaScript**

```
var interval = new CustomInterval();
interval.stroke = "red";
interval.minValue = 10;
interval.maxValue = 50;
scale.majorTickSettings.addCustomInterval(interval);
```

# Pointers

The pointers represent needles or similar type of indicators within the scale. They are used to visualize a current measurement within the scale. The most important member of the pointer object is its Value. The pointer is automatically adjusted by its containing scale so that it points to the scale location, corresponding to its value.

The size of the pointer is defined by the PointerWidth and PointerHeight properties. These properties can express the size as relative or absolute value. The location of the pointer can be adjusted through the PointerOffset property and its alignment in a linear scale can be specified through the Alignment property.

The pointer can be made interactive by setting the IsInteractive property to true. Clicking with the mouse within a scale causes all its interactive pointers to be adjusted to the value corresponding to the clicked point. A pointer can be made discrete by setting IsDiscrete to true. The value of a discrete pointer is always integer.

### Usage

The sample code below illustrates how to define and add pointers to an existing scale:

**JavaScript**

```
var pointer = new Point();
pointer.isInteractive = true;
scale.addPointer(pointer);
```

### Custom Appearance

The appearance of the pointer can be customized by using the PrepaintPointer and PaintPointer events. The following example demonstrates an oval gauge with a compass needle type of pointer:

**JavaScript**

```
var ovalScale = new OvalScale(ovalGauge);
ovalScale.fill = "transparent";
ovalScale.stroke = "transparent";
ovalScale.minValue = 0;
ovalScale.maxValue = 360;
ovalScale.majorTickSettings.showTicks = false;
ovalScale.majorTickSettings.showLabels = false;
```

```
ovalScale.middleTickSettings.showTicks = false;
ovalScale.middleTickSettings.showLabels = false;

var pointer = new Pointer();
pointer.pointerWidth = new Length(200, LengthType.Relative);
pointer.pointerHeight = new Length(20, LengthType.Relative);
pointer.pointerOffset = new Length(90, LengthType.Relative);
pointer.isInteractive = true;
pointer.shape = MindFusion.Gauges.PointerShape.Custom;
ovalScale.addPointer(pointer);

ovalGauge.addEventListener(Events.paintPointer, function (sender, args) {
    var context = args.context;
    var element = args.element;
    var size = element.renderSize;

    var figure = new PathFigure('M0.5, 0 L1, 0.5 L0.5, 1 Z');
    figure.fill = Utils.createRadialGradient(new Point(0.5, 0.1), 0, [0, 'rgba(193,0,0,1)', 0.3,
'rgba(255,0,0,1)', 1, 'rgba(102,0,0,1)']);
    figure.stroke = 'rgba(32,32,32,1)';
    args.paintVisualElement(figure, size);

    var figure = new PathFigure('M0.5, 0 L0, 0.5 L0.5, 1 Z');
    figure.fill = Utils.createRadialGradient(new Point(0.5, 0.1), 0, [0, 'rgba(193,193,193,1)',
0.3, 'rgba(204,204,204,1)', 1, 'rgba(102,102,102,1)']);
    figure.stroke = 'rgba(32,32,32,1)';
    args.paintVisualElement(figure, size);

    context.save();
    context.beginPath();
    context.arc(size.width / 2, size.height / 2, 3, 0, 2 * Math.PI, false);
    context.fillStyle = 'gray';
    context.strokeStyle = 'rgba(32,32,32,1)';
    context.fill();
    context.stroke();
    context.restore();
});
```

The above gauge will look similar to the following image:



# Ranges

The range is a visual representation of an interval within a gauge. Ranges are commonly used to mark or emphasize specific zones on the scale. Often the end of the scale is a danger zone and a range can be used to indicate this. The following image illustrates an oval gauge with three ranges (red, yellow and green) specifying different zones on the scale.



The range interval is specified through the MinValue and MaxValue properties. The Alignment and Offset properties control the positioning and the StartWidth and EndWidth properties specify the size of the range. The range can be auto-sized to the scale by setting AutoSize to **true**. To learn more about the different alignment modes check the Ticks and Labels topic.

### Appearance

The Fill and Stroke properties can be used to specify the appearance of the range. The stroke on the different sides of the range can be turned off by using the CapStart, CapEnd, StrokeInside and StrokeOutside properties.

### Usage

The sample code below illustrates how to define and add ranges to an existing linear scale:

**JavaScript**

```
var range1 = new Range();
range1.minValue = 70;
range1.maxValue = 80;
range1.startWidth = new Length(0);
range1.endWidth = new Length(5, LengthType.Relative);
range1.fill = "yellow";
range1.stroke = "DimGray";
range1.alignment = Alignment.OuterOutside;
range1.offset = new Length(-2);
range1.capEnd = false;

var range2 = new Range();
range2.minValue = 80;
range2.maxValue = 90;
range2.startWidth = new Length(5, LengthType.Relative);
range2.endWidth = new Length(10, LengthType.Relative);
range2.fill = "orange";
range2.stroke = "DimGray";
range2.alignment = Alignment.OuterOutside;
range2.offset = new Length(-2);
range2.capStart = false;
range2.capEnd = false;

var range3 = new Range();
range3.minValue = 90;
range3.macValue = 100;
range3.startWidth = new Length(10, LengthType.Relative);
range3.endWidth = new Length(15, LengthType.Relative);
range3.fill = "red";
range3.stroke = "DimGray";
range3.alignment = Alignment.OuterOutside;
range3.offset = new Length(-2);
range3.capEnd = false;

linearScale.addRange(range1);
linearScale.addRange(range2);
linearScale.addRange(range3);
```

The linear gauge defined above will look like this:




# Tutorials

# Building Your First Chart

This tutorial aims to a provide step-by-step guide on how to implement, customize and render a chart with the library. It assumes you have basic knowledge of JavaScript and are absolutely new to MindFusion.Charting for JavaScript. Feel free to skip content you are already familiar with.

## I. Set up of The Project

Follow the steps outlined in the Getting Started topic and set up your project either by loading the necessary charting modules (ES6/CommonJs) or by loading the namespaced UMD scripts.

### 1.1. Loading UMD namespaced scripts

Add the following references to your web page:

**HTML**

```
<script src="umd/drawing.js" type="text/javascript"></script>
<script src="umd/controls.js" type="text/javascript"></script>
<script src="umd/common.js" type="text/javascript"></script>
<script src="umd/common-collections.js" type="text/javascript"></script>
<script src="umd/gauges.js" type="text/javascript"></script>
<script src="umd/charting.js" type="text/javascript"></script>
```

We will write the code for the sample in a separate file called piechart.js, so we need to reference it as well:

**JavaScript**

```
<script type="text/javascript" src="piechart.js"></script>
```

### 1.2. Using ES6 modules with importmap

**JavaScript**

```
<script type="importmap">
{
    "imports":
    {
        "@mindfusion/controls":"./esm/controls.js",
        "@mindfusion/drawing":"./esm/drawing.js",
        "@mindfusion/common":"./esm/common.js",
  "@mindfusion/common-collections":"./esm/common-collections.js",
        "@mindfusion/gauges":"./esm/gauges.js",
        "@mindfusion/charting":"./esm/charting.js"
    }
}
</script>

<script src="piechart.js" type="module"></script>
```

### 1.3. Using npm packages

**HTML**

```
npm install @mindfusion/charting
```

### 2. Adding the canvas element to your HTML page

In your HTML file you should add a code like this in order to initialize the Canvas needed by the chart library to render itself onto:

JavaScript

```
<canvas id="myChart" width="400" height="400"></canvas>
```

The *id* would let you access the Canvas later in the *.js file, so be sure to include it.

## II. General Settings

### 1. If You Have Referenced the UMD Chart/Gauge Libraries Directly

We recommend that you add some namespace mappings to make writing code shorter:

JavaScript

```
var Charting = MindFusion.Charting;
var Controls = Charting.Controls;
var Collections = MindFusion.Common.Collections;
var Drawing = MindFusion.Drawing;
```

Beside that you don't need to do anything more.

### 2. If You've Used Modules

Add the necessary imports to your code-behind:

For ES6 Modules

```
import * as Charting from '@mindfusion/charting';
import * as Gauges from '@mindfusion/gauges';
import * as Drawing from '@mindfusion/drawing';
```

For CommonJS Modules

```
const Charting = require('@mindfusion/charting');
const Gauges = require('@mindfusion/gauges');
const { List } = require('@mindfusion/common-collections');
```

### 3. Create the Chart

The chart is created with the constructor of the chosen chart type:

JavaScript

```
var myChart = new Controls.PieChart(document.getElementById('myChart'));
```

You can look at the Charting.Controls namespace for a reference of the available predefined chart controls. The title and title font are set as follows:

JavaScript

```
myChart.title = "Corporate Sales";
myChart.theme.titleFontSize = 16;
```

The chart.theme property is very important for adjusting chart appearance. It contains most of the properties you'll need to customize the look of your chart.

## III. Series and Data

Generally charts have **series** property, which holds a Series with the data, title and labels. Let's assume you are building a pie chart. We create the data with a simple array:

JavaScript

```
var values = new Collections.List([20, 30, 10, 40, 35]);
var labels = new Collections.List(["January", "March", "May", "July", "September"]);
```

Then we create a series:

**JavaScript**

```
pieChart.series = new Charting.SimpleSeries(values, labels);
```

Once the chart series is assigned we can draw the chart. To actually render the chart, call:

**JavaScript**

```
chart.draw();
```

The control offers a variety of **Series** classes which you can use depending on your needs.

## IV. Appearance

The appearance of the chart can be changed with direct properties like commonSeriesFills, uniformSeriesFill etc. or with special SeriesStyle objects like PerElementSeriesStyle or PerSeriesStyle. Loading a theme is also an option as well are separate styling properties found on different chart elements. Here we'll use the properties that are likely to be set in the most common-case scenario. When you use a **SeriesStyle** object you assign it to the plot of the chart.

For a pie chart we use PerElementSeriesStyle. Let's define the brushes for the pie slices:

**JavaScript**

```
var brushes = new Collections.List([

        new Drawing.Brush("#193e4e"),
        new Drawing.Brush("#5a7444"),
        new Drawing.Brush("#8eb848"),
        new Drawing.Brush("#678b99"),
        new Drawing.Brush("#a1d0d8"),
        new Drawing.Brush("#c5b28a"),

 ]);

var seriesBrushes = new Collections.List();
seriesBrushes.add(brushes);
```

We repeat the code for the strokes:

**JavaScript**

```
var strokes = new Collections.List([

     new Drawing.Brush("#f2ebcf"),

 ]);

var seriesStrokes = new Collections.List();
seriesStrokes.add(strokes);
```

And for the stroke thickness:

**JavaScript**

```
var strokeThicknesses = new Collections.List([

     10,

 ]);

var serieStrokeThicknesses = new Collections.List();
serieStrokeThicknesses.add(strokeThicknesses);
```

**JavaScript**

```
myChart.plot.seriesStyle = new PerElementSeriesStyle(seriesBrushes, seriesStrokes,
seriesStrokeThicknesses);
```

All arguments are nested lists, each list is responsible for setting the appearance of the elements in a single series in the chart.

In short this is all you need to know to create a chart with the MindFusion.Charting for JavaScript library. For more advanced and creative scenarios, refer to the other tutorials:

Legend
Grid
Dashboard Layout

# Building A Chart With Axes

Follow the steps from first tutorial to load the diagram scripts, and add a "lineChart" DIV element. Now create the JavaScript chart instance:

**JavaScript**

```
// create the chart
var lineChart = new Controls.LineChart(document.getElementById('lineChart'));
```

If we want to show the zoom buttons, here is how to do it:

**JavaScript**

```
lineChart.showZoomWidgets = true;
```

For ticks on the axes use these properties:

**JavaScript**

```
lineChart.showXTicks = true;
lineChart.showYTicks = true;
```

Each predefined chart type e.g. line, bar, pie, radar etc. has series property, where you can add series. In our sample chart, we'll create two series. They are of type Series2D and can be used with any chart type that needs to draw 2D graphics:

**JavaScript**

```
// create sample data series
    var labels = new Collections.List([
        "one", "two", "three", "four", "five", "six",
        "seven", "eight", "nine", "ten", "eleven", "twelve"
    ]);
```

The constructor of the Series2D class requires three arguments. The first two are the data for the X and Y axis, the last one is for the labels. If you don't need labels, you can specify null.

**JavaScript**

```
var series1 = new Charting.Series2D(
    new Collections.List([0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11]),
    new Collections.List([1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]),
    labels);
lineChart.series.add(series1);

var series2 = new Charting.Series2D(new Collections.List(
    [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11]),
    new Collections.List([2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24]),
```

```
    labels);
lineChart.series.add(series2);
```

The line chart uses the Cartesian coordinate system. The LineChart class provides properties, which you can use to customize the scale of your axis:

**JavaScript**

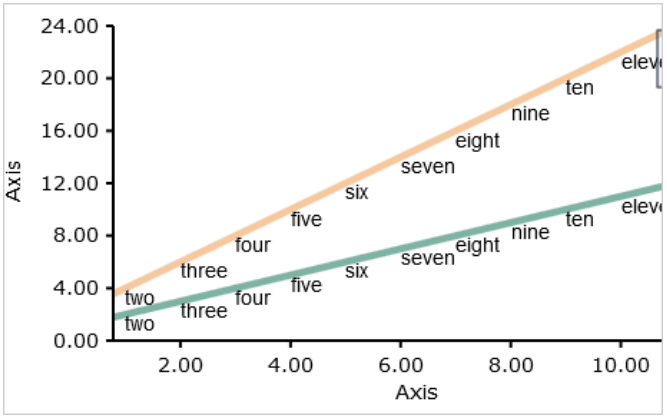
```
lineChart.xAxis.minValue = 0;
lineChart.xAxis.maxValue = 24;
lineChart.xAxis.interval = 4;
```

When you are ready with the settings, don't forget to draw the chart:

**JavaScript**

```
lineChart.draw();
```

And here is the chart:




# Data Binding

In this tutorial we'll use a list of objects for data provider. The objects are instances of a custom class and expose various properties. We will bind data properties of a radar chart with property fields of our custom class.

**I. Initialize the StudentData class:**

**JavaScript**

```
var StudentData = (function () {
        function StudentData(subject, gpa1, gpa2, gpa3) {
            this.Subject = subject;
            this.GPA1 = gpa1;
            this.GPA2 = gpa2;
            this.GPA3 = gpa3;
        }
        return StudentData;
    }());
```

**II. Create a collection of StudenData objects:**

**JavaScript**

```
var data = new Collections.List();
        var subjects = new Array("Math", "Physics", "Chemistry", "Biology", "IT");
```

```
        var gpa1 = new Array(2.3, 1.7, 2, 4, 3.7);
        var gpa2 = new Array(3.3, 2.7, 1.7, 2, 3.3);
        var gpa3 = new Array(4, 3.7, 2, 1, 1.3);
        for (var i = 0; i < 5; i++)
            data.add(new StudentData(subjects[i], gpa1[i], gpa2[i], gpa3[i]));
```

**III. Add a Radar Chart:**

**JavaScript**

```
//get the HTML Canvas
var chartEl = document.getElementById('radarChart');
var radarChart = new Controls.RadarChart(chartEl);

//specify the data source
radarChart.dataSource = data;
```

**IV. Bind Data Fields to Data Properties of the Control:**

**JavaScript**

```
radarChart.xDataFields = radarChart.yDataFields = new Collections.ObservableCollection(["GPA1",
"GPA2", "GPA3"]);
radarChart.innerLabelsDataFields = new Collections.ObservableCollection(["GPA1", "GPA2",
"GPA3"]);
radarChart.toolTipsDataFields = new Collections.ObservableCollection(["Subject", "Subject",
"Subject"]);
```

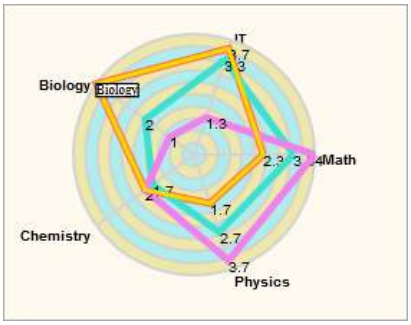
Here three series, each one using X-data, Y-data and tool tips text from the respective data field are created. Note that the control creates these DataBoundSeries objects automatically, we don't initialize them. Each data field results in one **DataBoundSeries** being created.

Don't forget to call the dataBind() method for it to work:

**JavaScript**

```
radarChart.dataBind();
```

Here is the chart:




# Using DateTime Values

The DateTimeSeries class implements the necessary functionality that allows using DateTime values as chart data. Let's create a list with DateTime values:

**JavaScript**

```
var years = new Collections.List();
   var dt = new Date(new Date().getFullYear() - 10, 11, 32);
```

```
  for (;;) {
     if (dt.getFullYear() > new Date().getFullYear())
         break;
     years.add(new Date(dt.setFullYear(dt.getFullYear() + 1)));
  }
```

We'll need some numeric data:

**JavaScript**

```
var income = new Collections.List([13.2, 15.6, 17.8, 39, 20, 29, 79, 101, 120, 122]);
```

Now we can create the DateTime series. The constructor takes as parameters a list of numbers, a list of DateTime values and the start and end DateTime of the series:

**JavaScript**

```
var series = new Charting.DateTimeSeries(years, income, years.item(0), years.item(years.count()
- 1));
```

The minValue and maxValue properties specify the numbers which correspond to the minDate and maxDate properties in the series:

**JavaScript**

```
series.minValue = 0;
series.maxValue = 1;
```

There are many ways to format a DateTime value and the dateTimeFormat property lets you specify one. In our case we want custom DateTime formatting and we set:

**JavaScript**

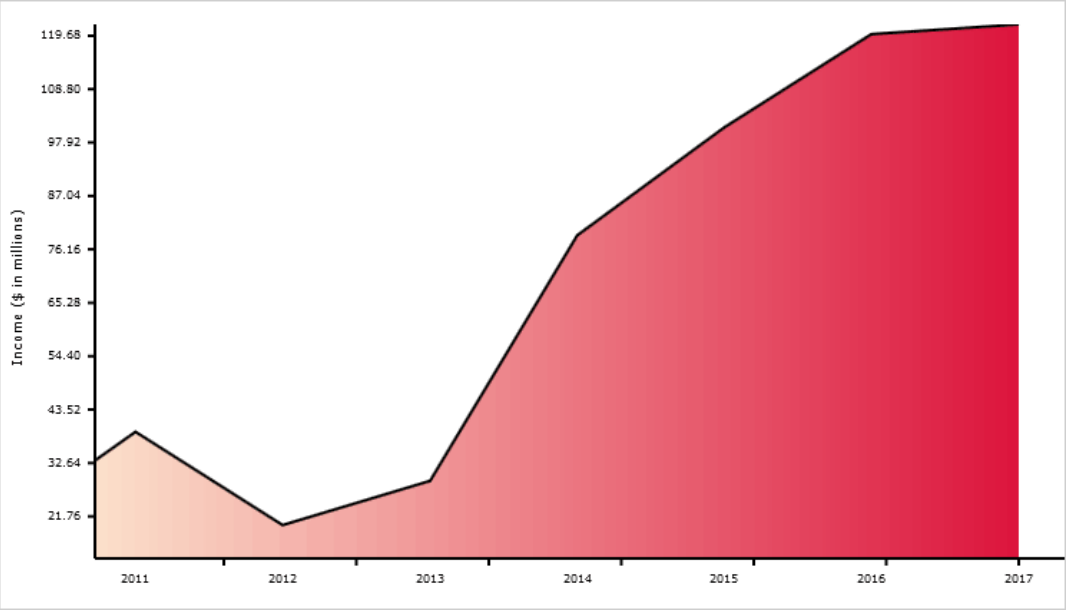
```
series.dateTimeFormat = Charting.DateTimeFormat.CustomDateTime;
series.customDateTimeFormat = '{"year":"numeric"}';
```

Finally we add the newly created series:

**JavaScript**

```
chart.series.add(series);
```

Here is the chart with some styling:




# Add, Style and Customize Axes

The chart control supports unlimited number of axes. They can be added and customized in several ways.

**I. Axes in the Chart Components**

Each ready-to-use chart component that uses the Cartesian coordinate system (line chart, bar chart) exposes two properties: xAxis and yAxis. You can use them to specify the basic settings of an axes:

**JavaScript**

```
lineChart.yAxis.minValue = 0;
lineChart.yAxis.maxValue = 24;
lineChart.yAxis.interval = 4;
lineChart.yAxis.title = "Years 2014 - 2016";
```

The Axis class exposes also numberFormat property, which you can use to specify how the labels will appear using the standard format strings.

In order to style an axis specified this way you must use the properties of the theme class:

**JavaScript**

```
lineChart.theme.axisLabelsFontSize = 14;
lineChart.theme.axisTitleFontSize = 14;
lineChart.theme.axisLabelsBrush = new Drawing.Brush("#333333");
lineChart.theme.axisLabelsBrush = lineChart.theme.axisStroke = new Drawing.Brush("#333333");
```

The Theme class exposes more than 10 properties that are about the axis: axisStrokeDashStyle, axisTitleBrush etc. - each one that starts with axis*.

**II. Custom Axis**

In case you are building your dashboard or plot that uses custom axes you must follow these simple steps:

**1. Create the axis.**

**JavaScript**

```
var commonAxis = new Charting.Axis();
commonAxis.minValue = 0;
commonAxis.interval = 0.5;
commonAxis.title = "Values";
```

**2. Create an AxisRenderer.**

**JavaScript**

```
var commonAxisRenderer1 = new Charting.XAxisRenderer(commonAxis);
```

The AxisRenderer class is the one where you find the styling options. Here is a XAxisRenderer that specifies the color of the axis, axis title and labels.

**JavaScript**

```
var commonAxisRenderer1 = new Charting.XAxisRenderer(commonAxis);
commonAxisRenderer1.gridColumn = 1;
commonAxisRenderer1.gridRow = 1;
commonAxisRenderer1.axisStroke = new Drawing.Brush("#737373");
commonAxisRenderer1.labelBrush = new Drawing.Brush("#737373");
commonAxisRenderer1.titleBrush = new Drawing.Brush("#737373");
```

The AxisRenderer must be added to a layout panel, where it will be measured and arranged. For instance, a GridPanel:

**JavaScript**

```
// set up grid panel
var grid = new Components.GridPanel();
   grid.columns.add(new Components.GridColumn());
   for (var i = 0; i < 3; i++)
      grid.rows.add(new Components.GridRow());

grid.children.add(commonAxisRenderer1);
```

**3. Assign the axis (Optional).**

You might want to designate the axis to be a particular axis in a plot:

**JavaScript**

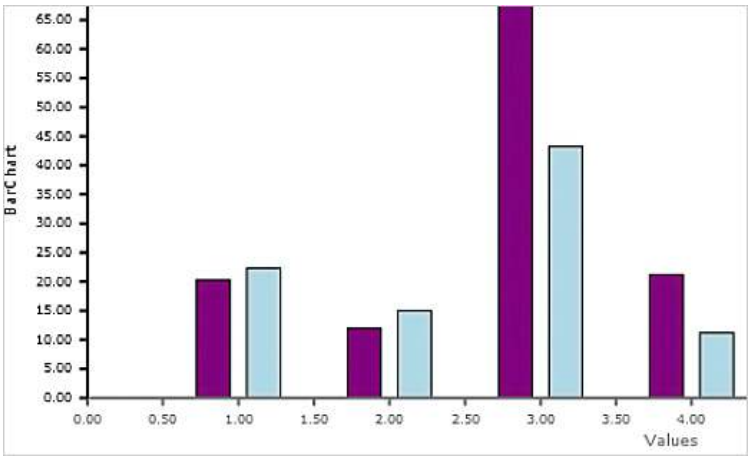
```
var plot1 = new Charting.Plot2D();
plot1.xAxis = commonAxis;
```

The plot then can be added to another element, for instance a grid.

**JavaScript**

```
plot1.gridColumn = 1;
plot1.gridRow = 0;
grid.children.add(plot1);
```

The custom axis we created is visible in this image:




# Labels in Charts

**I. Line and Area Charts**

In those types of charts you will typically set the labels that appear on chart data points in the constructor of the series that you use:

**JavaScript**

```
var labels = new Collections.List([
        "one", "two", "three", "four", "five", "six",
        "seven", "eight", "nine", "ten", "eleven", "twelve"
    ]);
    var series1 = new Charting.Series2D(new Collections.List([0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10,
11]), new Collections.List([1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]), labels);
 lineChart.series.add(series1);
```

**II. Bubble Charts**

The Bubble chart control requires series of type Series3D, where you an also set the labels:

**JavaScript**

```
var series3D1 = new Charting.PointSeries3D(new Collections.List([
        new Charting.Point3D(0.32, 81, 9.5 * 10),
        new Charting.Point3D(0.39, 66, 7.8 * 10),
        new Charting.Point3D(0.75, 65, 7.6 * 10),
        new Charting.Point3D(0.49, 60, 7.1 * 10)
    ]), new Collections.List(["Germany", "France", "UK", "Italy"]));

var data = new Collections.ObservableCollection();
    data.add(series3D1);
    chart.series = data;
```

**III. Pie Charts**

Pie charts use inner and outer labels. If you don't want certain labels to appear provide null to the PieSeries constructor:

**JavaScript**

```
pieChart.series = new Charting.PieSeries(values, new Collections.List(["20", "30", "10", "40"]),
new Collections.List(["twenty", "thirty", "ten", "forty"]));
```

### IV. Bar Charts

Bar charts can have top and inner labels, which you set in the constructor of the BarSeries class. If you don't want any of those labels provide an empty list or null.

**JavaScript**

```
var labels = new Collections.List([
        "one", "two", "three", "four", "five", "six",
        "seven", "eight", "nine", "ten", "eleven", "twelve"
    ]);

var topLabels = new Collections.List([
        "Jan", "Feb", "Mar", "Apr", "May", "Jun",
        "Jul", "Aug", "Sep", "Oct", "Nov", "Dec"
    ]);

    // create sample data series
    var series = new Collections.ObservableCollection(new Array(new Charting.BarSeries(new
Collections.List([1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]), labels, topLabels)));
```

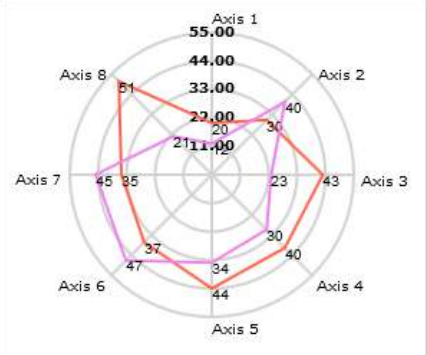
### V. Radar Charts

Radar charts also get their labels from the labels property of the Series class used to provide data. Here is how you'll set labels at data points in a radar using a SimpleSeries class:

**JavaScript**

```
var data = new Collections.ObservableCollection();
    var series1 = new Charting.SimpleSeries(new Collections.List([20, 30, 43, 40, 44, 37, 35,
51]), new Collections.List(["20", "30", "43", "40", "44", "37", "35", "51"]));
    var series2 = new Charting.SimpleSeries(new Collections.List([12, 40, 23, 30, 34, 47, 45,
21]), new Collections.List(["12", "40", "23", "30", "34", "47", "45", "21"]));
    data.add(series1);
    data.add(series2);
    radarChart.series = data;
```

The radar chart with all types of labels:




In all chart types you can hide the labels by using:

**JavaScript**

```
chart.showDataLabels = false;
```

# Labels on Axes

## I. Radar Charts

Radar charts support unlimited number of axes. You add as many Axis instances as you want to see on the radar:

**JavaScript**

```
for (var i = 0; i < 8; i++) {
        var axis = new Charting.Axis();
        axis.title = "Axis " + (i + 1).toString();
        radarChart.axes.add(axis);
    }
```

Labels on radar chart axes are set with the title property of the axes. If you don't want a label just leave an empty string.

The labels on the scale of the main radar axes depend on the intervals. You set them this way:

**JavaScript**

```
radarChart.defaultAxis.minValue = 0;
radarChart.defaultAxis.maxValue = 55;
```

In case you don't want labels on this default axis just hide them:

**JavaScript**

```
radarChart.showCoordinates = false;
```

## II. Axes and Labels on Axes That Use The Cartesian Coordinate System

By default each axis shows its scale, which is set with the minValue, maxValue and interval properties.

**JavaScript**

```
lineChart.xAxis.minValue = 1;
lineChart.xAxis.maxValue = 10;
lineChart.xAxis.interval = 1;
```

If you don't want to show these labels you can set

**JavaScript**

```
lineChart.showXCoordinates = false;
```

or

**JavaScript**

```
lineChart.showYCoordinates  = false;
```

If you are using an AxisRenderer the property is:

**JavaScript**

```
axisRenderer.showCoordinates = false;
```

#### 1. Custom Labels in BarCharts

Charts that use the Cartesian coordinate system can show custom labels on all of their axis.

**JavaScript**

```
barSeries.xAxisLabels = new Collections.List([
        "one", "two", "three", "four", "five", "six",
        "seven", "eight", "nine", "ten", "eleven", "twelve"
    ]);
```

#### 2. Custom Labels on Axes

You can set the labels from any series as labels on the axis using the supportedLabels property.

Let's assume you have created a Series2D like that:

**JavaScript**

```
var labels = new Collections.List([
        "one", "two", "three", "four", "five", "six",
        "seven", "eight", "nine", "ten", "eleven", "twelve"
    ]);
 var series1 = new Charting.Series2D(new Collections.List([0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10,
11]), new Collections.List([1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]), labels);
    series1.title = "Series 1";
```

In order to show the labels on the Y-axis you should set:

**JavaScript**

```
series1.supportedLabels = Charting.LabelKinds.YAxisLabel;
```

Of course, to disable the scale labels add:

**JavaScript**

```
lineChart.showYCoordinates  = false;
```

BarSeries classes support special xAxisLabels property to set the labels under the bars:

**JavaScript**

```
var labels = new Collections.List([
        "one", "two", "three", "four", "five", "six",
        "seven", "eight", "nine", "ten", "eleven", "twelve"
    ]);
barSeries.xAxisLabels = labels;
```

# Styling Chart Series

When you want different styling for each element in a single series e.g. each pie piece, bubble, scatter etc, the right way to do is to use the PerElementSeriesStyle class. The constructor can take up to four arguments, all of them optional.

**Note**

Each argument in the PerElementSeriesStyle is of type List that holds other List-s.

The logic is that the **PerElementSeriesStyle** class serves charts that might use multiple series. Since the style property is one for all series, it must provide a way to specify the style settings for all elements in all

series. The style settings are taken consequently in  the order they were added and applied to the series at the same index  as the index of the array with the styling properties.

Initialize brushes for each element in the series:

**JavaScript**

```
 var brushes = new Collections.List([
        new Drawing.Brush("#f5ded0"),
        new Drawing.Brush("#e9ca91"),
        new Drawing.Brush("#87b3ab"),
        new Drawing.Brush("#ea684f"),
        new Drawing.Brush("#c0c0bc"),
 new Drawing.Brush("#877d75")
    ]);

var seriesBrushes = new Collections.List();
    seriesBrushes.add(brushes);
```

Now the seriesBrushes variable is ready to be passed to the **PerElementSeriesStyle** constructor. Let's initialize one Brush for the outlining of each element. When the count of elements in the style settings array is less than the count of data in the series, the control cycles the styling settings. For the outlining, we'll use a single Bursh. Nonetheless we must put it in a nested array as well:

**JavaScript**

```
var pens = new Collections.List([
        new Drawing.Brush("white")
    ]);

var seriesPens = new Collections.List();
  seriesPens.add(pens);
```

We repeat the same initialization for the thickness of the pens. We use  a pie chart and we want a thick pen:

**JavaScript**

```
var thicknesses = new Collections.List([
        10
    ]);
var seriesThicknesses = new Collections.List();
    seriesThicknesses.add(thicknesses);
```

If you want to set a special dash style, you can set it this way:

**JavaScript**

```
var dashStyles = new Collections.List([
        Drawing.DashStyle.Dash
    ]);
var seriesDashStyles = new Collections.List();
    seriesDashStyles.add(dashStyles);
```

The DashStyle enumeration lists different dash styles, which you can use. Now, let's assign the style to the chart.

**JavaScript**

```

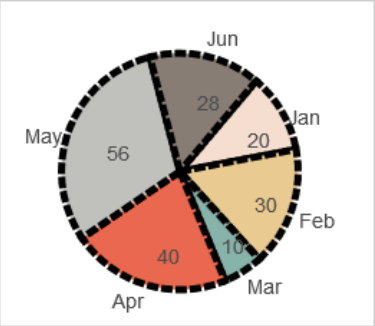
pieChart.plot.seriesStyle = new Charting.PerElementSeriesStyle
(seriesBrushes, seriesPens, seriesThicknesses, seriesDashStyles);
```

Don't forget to draw the chart:

**JavaScript**

```
pieChart.draw();
```

Here is the result:




# Add and Style Legends

**I. Add a Legend to a Simple Chart Type**

On a basic chart type - PieChart, LineChart etc. setting the showLegend property to true is enough to render the legend with default appearance
settings. In these cases the legend is always only one and can have as many items as the count of chart series.

**JavaScript**

```
pieChart.showLegend = true;
```

The legend labels are taken from the title property of the series that are added to the chart. The brush is the brush for this series. If there are multiple brushes used, the first one is considered.

**JavaScript**

```
pieChart.series.title = "Monthly Sales";
```

The appearance settings for the legend are in the theme property:

**JavaScript**

```
pieChart.theme.legendBackground = new Drawing.Brush("#f0f0f0");
pieChart.theme.legendTitleFontName = "Arial";
pieChart.theme.legendTitleFontSize = 14;
```

and many more. The title of the legend is set with:

**JavaScript**

```
pieChart.legendTitle = "Year 2016";
```

If you don't want a legend title just set:

**JavaScript**

```
pieChart.showLegendTitle = "false";
```

By default the legend shows as many labels are there are series in the chart. The color box before each chart label is painted with the first brush assigned to this series. Legend drag is enabled by default so you can position it wherever you want. If you want to render your legend at a given location from the very start use the properties of the chart class: legendHorizontalAlignment, legendVerticalAlignment, legendSpacing, and legendMargin.

**II. Using a LegendRenderer**

The LegendRenderer is the class that you should use when you want to add a legend in a more advanced chart scenario.

**JavaScript**

```
var legend = new Charting.LegendRenderer();
legend.content = new Collections.ObservableCollection
  ([lineRenderer]);
legend.background = new Drawing.Brush
(Drawing.Color.knownColors.lightyellow);
legend.horizontalAlignment = Components.LayoutAlignment.Far;
dashboard.rootPanel.children.add(legend);
```

A **LegendRenderer** has properties for legend positioning and styling. The content property ties the **LegendRenderer** to a collection of SeriesRender objects, each of which provides a series title to display as a legend item. You can add as many **LegendRenderer** objects as you wish and arrange them according to your needs. As a rule you just need to add your **LegendRenderer** to the children of your panel as you do with any other chart element.

If you are building a dashboard, add the **LegendRenderer** directly to the root panel:

**JavaScript**

```
dashboard.rootPanel.children.add(legend1);
```

The **LegendRenderer** class exposes numerous properties for styling your legend, its labels, title, border etc. Check them here.

# Grid

In the basic chart controls - LineChart, BarChart etc. you access the gridType property directly:

**Grid**

```
chart.gridType = Charting.GridType.Vertical;
```

Styling is done through the theme property of the control:

**JavaScript**

```
chart.theme.gridColor1 = Drawing.Color.fromArgb(0.4, 0, 0, 0);
chart.theme.gridColor2 = Drawing.Color.fromArgb(1, 40, 40, 40);
```

The Theme class offers a lot of styling options for grids: gridLineColor, gridLineThickness and gridLineStyle help you style the grid lines. By default no grid lines are drawn.

Grid in complex charts is easy to set with the gridType property that is exposed by the container:

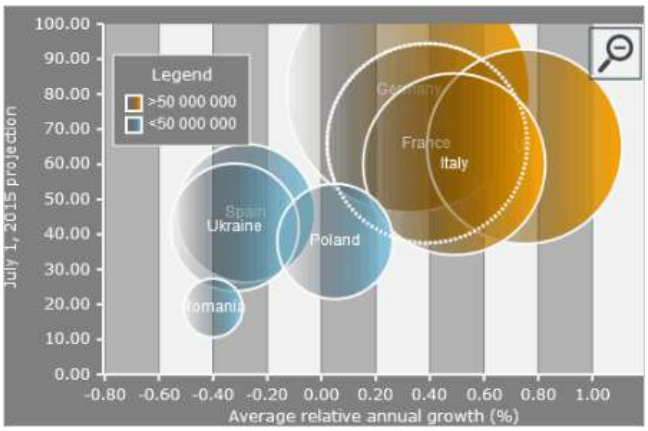
**JavaScript**

```
var plot1 = new Charting.Plot2D();
plot1.gridType = Charting.GridType.Crossed;

plot1.gridColor1 = new Drawing.Color("white");
plot1.gridColor1 = new Drawing.Color("white");
plot1.pinGrid = true;
```

The **Plot2D** chart offers several properties that you can use to customize the grid: gridColor1, gridColor2 and the pinGrid property, which sets how the grid responds when the user scrolls the chart interactively.

A bubble chart with custom grid:




# Tooltips

All chart types can show tooltips. Currently there are two ways to show tooltips.

## I. Using DataBinding

If you read your data from a data source you can use the toolTipsDataFields property to assign fields from your data source that will provide data for the tooltips. Let's assume you have a list of the following objects:

**JavaScript**

```
var StudentData = (function () {
        function StudentData(subject, gpa1, gpa2, gpa3) {
            this.Subject = subject;
            this.GPA1 = gpa1;
            this.GPA2 = gpa2;
            this.GPA3 = gpa3;
        }
        return StudentData;
}());
```

You render a radar chart with three series - each one with data from GPA1, GPA2 and GPA3.

**JavaScript**

```
radarChart.xDataFields = radarChart.yDataFields = new Collections.ObservableCollection(["GPA1",
"GPA2", "GPA3"]);
radarChart.toolTipsDataFields = new Collections.ObservableCollection(["Subject", "Subject",
"Subject"]);
```

This is how you set the data and how you specify that the labels from the Subject field shall appear as tooltips.

Alternatively, you can create three DataBoundSeries objects and set the tooltip data fields.

**JavaScript**

```
DataBoundSeries series1 = new DataBoundSeries(studenDataList);
series1.xDataField = series1.yDataField = "GPA1";
series1.toolTipsDataField = "Subject";
chart.series.add(series1);
//repeat for the next two series with GPA2 and GPA3 data
```

## II. Customize a Series

The second approach is to customize your chosen Series object. First, define the tooltips:

**JavaScript**

```
var tooltips = new Collections.List(["32.7%", "29.5%", "26.2%", "12%", "5%", "2%"]);
funnelChart.series.tooltips = tooltips;
```

The next step is to override the supportedLabels property of your Series and make it return LabelKinds.ToolTip in addition to the type of labels it already supports. Here we use the SimpleSeries class.

**JavaScript**

```
Object.defineProperty(Charting.SimpleSeries.prototype, "supportedLabels", {
            get: function () { return Charting.LabelKinds.InnerLabel |
Charting.LabelKinds.ToolTip; },
            enumerable: true,
            configurable: true
        });
```

The last thing to do is return the correct tooltip when asked for it:

**JavaScript**

```
Charting.SimpleSeries.prototype.getLabel = function (index, kind) {
  if ((kind & Charting.LabelKinds.ToolTip) != 0 && this.tooltips)
   return this.tooltips.items()[index];

  if (this.labels == null)
   return null;
  return this.labels.items()[index];
 };
```

You can follow these steps to add tooltip support to any kind of **Series**.

# Dashboard Layout

The ChartBase class represents a dashboard container of visual components such as plots, line graphics and text labels. Initially the dashboard contains two Panel classes to which its components should be added:

- RootPanel is the root of dashboard components' hierarchy. Its elements are drawn overlapping on top of each other. Here you can add elements whose position should not depend on the size of the dashboard and their siblings. For example, Chart controls display LegendRenderer inside the **RootPanel**.
- LayoutPanel is a vertical stack panel by default. Its components are arranged next to each other and dimensions are calculated automatically from available space. The components that we are going to create in this tutorial will be added here.

Let's start by adding a HTML canvas element to the page and create a **ChartBase** instance from this element. In addition, set the style and dimensions of the **ChartBase** so that it fills its parent **div** element. You can also add aliases to the commonly used namespaces.

**HTML**

```
<div style="position: absolute; left: 0px; top: 60px; bottom: 0px; right: 0px;">
  <canvas id="dashboard" style="display:block;width:100%;height:100%"></canvas>
</div>
```

**JavaScript**

```
var Charting = MindFusion.Charting;
var Controls = MindFusion.Controls;
var Components = MindFusion.Charting.Components;
var Collections = MindFusion.Common.Collections;
var Drawing = MindFusion.Drawing;

var dashboardEl = document.getElementById('dashboard');
dashboardEl.width = dashboardEl.offsetParent.clientWidth;
```

```
dashboardEl.height = dashboardEl.offsetParent.clientHeight;
var dashboard = new Controls.ChartBase(dashboardEl);
```

Next, add a TextComponent that will display the dashboard's title. As **LayoutPanel** arranges its children in a vertical stack, the title will show at the top of the dashboard if it's the first component added. The following code creates and adds the title component, while also setting its appearance properties:

**JavaScript**

```
var tc = new Components.TextComponent();
tc.text = "TITLE TEXT";
tc.fontName = "Times New Roman";
tc.fontStyle = Drawing.FontStyle.Underline;
tc.fontSize = 20.4;
tc.horizontalAlignment = Components.LayoutAlignment.Center;
dashboard.layoutPanel.children.add(tc);
```

We want to divide the remaining space in two rows and two columns, creating four cells in total. For that purpose, add a GridPanel, which can host multiple components in each of its cells. Its default constructor creates one GridRow and one GridColumn, i.e. it defines a single cell. We'll add one more row and column to respective collection properties of the panel:

**JavaScript**

```
var grid = new Components.GridPanel();
grid.columns.add(new Components.GridColumn());
grid.rows.add(new Components.GridRow());
dashboard.layoutPanel.children.add(grid);
```

Now that we have defined our layout, lets create sample data series. MindFusion.Charting offers multiple classes you could use to represent data. All of them implement the Series interface, which you could implement in your own model classes if you prefer. The data class we'll use in this example is BarSeries; it takes a list of values to display as bar height, and automatically assigns bar positions at equal distances. It is convenient for creation of barcharts, but can actually be used with any other type of SeriesRenderer.

**JavaScript**

```
var data1 = new Charting.BarSeries(
    new Collections.List([10, 15, 17, 16, 20, 15, 14, 13, 10, 17, 22, 25, 25, 25, 21, 21]),
    null, null,
    new Collections.List(["nov\n2014", "dec\n2014", "jan\n2015", "feb\n2015", "mar\n2015",
"apr\n2015", "may\n2015", "jun\n2015", "jul\n2015", "aug\n2015", "sep\n2015", "oct\n2015",
"nov\n2015", "dec\n2015", "jan\n2016", "feb\n2016"]));
data1.title = "LP Vinyl";

var data2 = new Charting.BarSeries(
    new Collections.List([13, 13, 14, 16, 23, 25, 20, 19, 16, 17, 15, 13, 13, 17, 18, 21]),
null);
data2.title = "CD";

var data3 = new Charting.BarSeries(
    new Collections.List([29, 28, 29, 31, 33, 35, 33, 27, 28, 28, 27, 29, 29, 31, 33, 29]),
null);
data3.title = "MP3";
```

You can see that **BarSeries** has two overloaded constructors. They let you specify different kinds of labels for the bars. MindFusion.Charting supports six kinds of labels displayed at various locations of rendered graphics. **BarSeries** lets you define inner labels drawn inside bars, top labels drawn above their bars, and X axis labels drawn along the axis. In the example above we'll only show axis labels, passing **null** for the other arguments. The Title values will be drawn by a **LegendRenderer** we are going to add later.

For visualizing the data, you must create an instance of a class derived from **SeriesRenderer**, each drawing a different kind of graphics. In this tutorial we'll see how several of them work, starting with a LineRenderer:

**JavaScript**

```
var lineRenderer = new Charting.LineRenderer(new Collections.ObservableCollection([data1, data2,
data3]));
```

Apart from the kind of graphics drawn, a **SeriesRenderer** lets you specify styling attributes for the graphic primitives, such as stroke and fill colors:

**JavaScript**

```
lineRenderer.seriesStyle = new Charting.PerSeriesStyle(new Collections.List([
new Drawing.Brush(Drawing.Color.fromArgb(0.6, 255, 0, 0)),
new Drawing.Brush(Drawing.Color.fromArgb(0.6, 0, 0, 255)),
new Drawing.Brush(Drawing.Color.fromArgb(0.6, 0, 255, 0))]), new Collections.List([
new Drawing.Brush(new Drawing.Color("Red")),
new Drawing.Brush(new Drawing.Color("Blue")),
new Drawing.Brush(new Drawing.Color("Green"))]));
```

**SeriesRenderer** objects must be placed inside Plot components. In this tutorial we'll use three different kinds of plots: Plot2D and Plot3D for drawing in 2D and 3D Cartesian coordinate systems respectively, and RadarPlot for drawing in polar coordinate system. For the **LineRenderer** we defined above, create a **Plot2D** and set its background:

**JavaScript**

```
var linePlot = new Charting.Plot2D();
linePlot.background = new Drawing.LinearGradientBrush(new Drawing.Color("White"),
Drawing.Color.fromArgb(1, 224, 224, 224), 90);
```

Add the line renderer to plot's SeriesRenderers collection:

**JavaScript**

```
linePlot.seriesRenderers.add(lineRenderer);
```

The data points renderered inside a **Plot2D** are scaled relatively to the Axis ranges defined via its XAxis and YAxis properties. For each **Axis** you can define its first and last visible values by setting MinValue and MaxValue properties. If their values are not specified, the control will calculate them automatically to fit all data provided through plot's **SeriesRenderer** objects. In addition, the Interval property lets you specify the distance between coordinate labels. The **Axis** class also has a Title property that lets you show description for the measure represented by an axis:

**JavaScript**

```
linePlot.xAxis = new Charting.Axis();
linePlot.xAxis.minValue = 0;
linePlot.xAxis.maxValue = 15;
linePlot.xAxis.interval = 1;
linePlot.xAxis.title = "When?";

linePlot.yAxis = new Charting.Axis();
linePlot.yAxis.minValue = 0;
linePlot.yAxis.maxValue = 35;
linePlot.yAxis.interval = 5;
linePlot.yAxis.title = "How many thousands?";
```

If you want to show coordinates for the values drawn inside a plot, you must also add XAxisRenderer and YAxisRenderer components to the dashboard. Each of them can optionally display **Axis** coordinates and **Title**, ticks and data labels. In order to display data labels from a series, you must also set the LabelsSource property to either a **SeriesRenderer** or a **Plot**:

**JavaScript**

```
var xRenderer = new Charting.XAxisRenderer(linePlot.xAxis);
xar.labelsSource = linePlot;
xar.showCoordinates = false;

var yRenderer = new Charting.YAxisRenderer(linePlot.yAxis);
```

In order to align axes to their plot, we'll add an intermediate **GridPanel** inside the top-left cell of the parent grid. We can do this on a couple of lines of code with the help of LayoutBuilder. That class lets us build layout fragments from several components and add them to a parent panel. In this case we'll call the CreatePlotWithBottomAndLeftAxes method to automatically create a **GridPanel** and add specified components to it. Other variants of this method can also automatically add the created panel to dashboard's **LayoutPanel**, but in this tutorial we'll add the result to our custom child panel.

**JavaScript**

```
var builder = new Controls.LayoutBuilder(dashboard);
var panel1 = builder.createPlotWithBottomAndLeftAxes(linePlot, xRenderer, yRenderer);
```

We want to display this fragment inside top-left cell of the **GridPanel** we created earlier. This can be done by setting the GridRow and GridColumn properties defined in base Component class, which specify respectively the row or column index of a cell inside parent grid panel. The index is zero-based, so we set both properties to zero:

**JavaScript**

```
panel1.gridRow = 0;
panel1.gridColumn = 0;
grid.children.add(panel1);
```

Following the same approach, we can display same data using a different kind of graphics and with different appearance attributes. Lets draw bars this time:

**JavaScript**

```
var barDataRenderer = new Charting.BarRenderer(new Collections.ObservableCollection([data1,
data2, data3]));
```

We'll specify a different fill color for each series of bars by providing a list of brushes to PerSeriesStyle class.

**JavaScript**

```
barDataRenderer.seriesStyle = new Charting.PerSeriesStyle(new Collections.List([
    new Drawing.Brush(Drawing.Color.fromArgb(0.6, 128, 0, 128)),
    new Drawing.Brush(Drawing.Color.fromArgb(0.6, 240, 230, 140)),
    new Drawing.Brush(Drawing.Color.fromArgb(0.6, 255, 165, 0))]));
```

You could add the BarRenderer to the same plot that now draws line graphics to create a mixed chart, but lets create a new plot for drawing bars only:

**JavaScript**

```
var barPlot = new Charting.Plot2D();
```

Lets show axes for this plot too. Note that we are assigning same range properties as for the linechart, but through a new Axis instance. If you reused the **Axis** instance (by setting barPlot.XAxis = linePlot.XAxis), the plots would scroll together when the user pans any of them.

**JavaScript**

```
barPlot.xAxis = new Charting.Axis();
barPlot.xAxis.minValue = -1;
barPlot.xAxis.maxValue = 15;
barPlot.xAxis.interval = 1;
barPlot.xAxis.title = "When?";

barPlot.yAxis = new Charting.Axis();
barPlot.yAxis.minValue = 0;
barPlot.yAxis.maxValue = 35;
barPlot.yAxis.interval = 5;
barPlot.yAxis.title = "How many thousands?";
```

Add the bar renderer to its plot:

**JavaScript**

```
barPlot.seriesRenderers.add(barDataRenderer);
```

Use **LayoutBuilder** again to align axes to plot and add them to a grid panel:

**JavaScript**

```
var panel2 = builder.createPlotWithBottomAndRightAxes(barPlot, xRenderer, new
Charting.YAxisRenderer(barPlot.yAxis));

panel2.gridColumn = 1;
panel2.gridRow = 0;
grid.children.add(panel2);
```

Now lets draw a radar chart using RadarRenderer:

**JavaScript**

```
var radarDataRenderer = new Charting.RadarRenderer(new Collections.ObservableCollection([data1,
data2, data3]));
```

Set the AreaOpacity property to fill the radar polygons using a semi-transparent fill:

**JavaScript**

```
radarDataRenderer.areaOpacity = 0.2;
```

Set the stroke and fill colors:

**JavaScript**

```
radarDataRenderer.seriesStyle = new Charting.PerSeriesStyle(new Collections.List([
    new Drawing.Brush(new Drawing.Color("PaleGreen")),
    new Drawing.Brush(new Drawing.Color("PaleTurquoise")),
    new Drawing.Brush(new Drawing.Color("PaleVioletRed"))]), new Collections.List([
    new Drawing.Brush(new Drawing.Color("PaleGreen")),
    new Drawing.Brush(new Drawing.Color("PaleTurquoise")),
    new Drawing.Brush(new Drawing.Color("PaleVioletRed"))]), new Collections.List([5]));
```

Create a **RadarPlot** and set its data range:

**JavaScript**

```
var radarPlot = new Charting.RadarPlot();
radarPlot.defaultAxis.minValue = 0;
radarPlot.defaultAxis.maxValue = 35;
radarPlot.defaultAxis.interval = 5;
```

Set axis appearance attributes:

**JavaScript**

```
radarPlot.axisOptions = new Charting.RadarAxisOptions(radarPlot.defaultAxis);
radarPlot.axisOptions.showSeriesLabels = true;
radarPlot.axisOptions.axisStroke = new Drawing.Brush(new Drawing.Color("LightGray"));
radarPlot.defaultAxis.title = "";
```

Add the radar renderer to its plot:

**JavaScript**

```
radarPlot.seriesRenderers.add(radarDataRenderer);
```

Add the plot to lower-left cell of parent grid:

**JavaScript**

```
radarPlot.gridColumn = 0;
radarPlot.gridRow = 1;
grid.children.add(radarPlot);
```

We now have one last unused cell in our parent grid - lets add four more plots to it using third-level grid panel. In its first cell we'll show horizontal bars. In other ones we'll show 3D bars and also show how to set them up to scroll together. We'll again show the same data as in previous plots:

**JavaScript**

```
var horizontalBar3DDataRenderer = new Charting.BarRenderer(new
Collections.ObservableCollection([data1, data2, data3]));
```

Set their fills:

**JavaScript**

```
horizontalBar3DDataRenderer.seriesStyle = new Charting.PerSeriesStyle(new Collections.List([
    new Drawing.Brush(Drawing.Color.fromArgb(0.6, 128, 0, 128)),
    new Drawing.Brush(Drawing.Color.fromArgb(0.6, 240, 230, 140)),
    new Drawing.Brush(Drawing.Color.fromArgb(0.6, 255, 165, 0))]), null, new
Collections.List([1]));
```

Set the bars orientation:

**JavaScript**

```
horizontalBar3DDataRenderer.horizontalBars = true;
```

Add the horizontal bar renderer to a plot:

**JavaScript**

```
var plot4 = new Charting.Plot2D();
plot4.seriesRenderers.add(horizontalBar3DDataRenderer);
```

Set up plot's axes:

**JavaScript**

```
plot4.yAxis = new Charting.Axis();
plot4.yAxis.minValue = 0;
plot4.yAxis.maxValue = 4;
plot4.yAxis.interval = 1;
plot4.yAxis.title = "Months";

plot4.xAxis = new Charting.Axis();
plot4.xAxis.minValue = 0;
plot4.xAxis.maxValue = 40;
plot4.xAxis.interval = 5;
plot4.xAxis.title = "Thousands";
```

Specify that this plot should scroll vertically:

**JavaScript**

```
plot4.verticalScroll = true;
```

Create axis renderers and set up their appearance:

**JavaScript**

```
```

```
var xAxisRenderer = new Charting.XAxisRenderer(plot4.xAxis);
xAxisRenderer.axisStroke = new Drawing.Brush(new Drawing.Color("Red"));
xAxisRenderer.axisStrokeThickness = 3;
xAxisRenderer.titleBrush = new Drawing.Brush(new Drawing.Color("Orange"));
xAxisRenderer.titleFontName = "Courier New";
xAxisRenderer.titleFontSize = 15;

var yAxisRenderer = new Charting.YAxisRenderer(plot4.yAxis);
yAxisRenderer.axisStroke = new Drawing.Brush(new Drawing.Color("Red"));
yAxisRenderer.axisStrokeThickness = 3;
yAxisRenderer.titleBrush = new Drawing.Brush(new Drawing.Color("Orange"));
yAxisRenderer.titleFontName = "Courier New";
yAxisRenderer.titleFontSize = 15;
```

Create a panel for the plot and axes:

**JavaScript**

```
var panel6 = builder.createPlotWithTopAndLeftAxes(plot4, xAxisRenderer, yAxisRenderer);
panel6.gridColumn = 0;
panel6.gridRow = 0;
```

Now let's create some 3D bars:

**JavaScript**

```
var bar3DDataRenderer = new Charting.BarRenderer3D(new Collections.ObservableCollection([data1,
data2, data3]));
```

Set their style:

**JavaScript**

```
bar3DDataRenderer.seriesStyle = new Charting.PerSeriesStyle(new Collections.List([
    new Drawing.Brush(Drawing.Color.fromArgb(0.6, 128, 0, 128)),
    new Drawing.Brush(Drawing.Color.fromArgb(0.6, 240, 230, 140)),
    new Drawing.Brush(Drawing.Color.fromArgb(0.6, 255, 165, 0))]), null, new
Collections.List([1]));
```

Create a plot for 3D bars:

**JavaScript**

```
var plot3d1 = new Charting.Plot3D();
plot3d1.seriesRenderers.add(bar3DDataRenderer);
```

Set up axes. We'll reuse these **Axis** instances later in order to implement synchronized scrolling in several plots:

**JavaScript**

```
plot3d1.xAxis = new Charting.Axis();
plot3d1.xAxis.minValue = 0;
plot3d1.xAxis.maxValue = 5;
plot3d1.xAxis.interval = 1;
plot3d1.xAxis.title = "Months";

plot3d1.yAxis = new Charting.Axis();
plot3d1.yAxis.minValue = 0;
plot3d1.yAxis.maxValue = 40;
plot3d1.yAxis.interval = 5;
plot3d1.yAxis.title = "Thousands";
```

Add last plot and axes to the layout:

**JavaScript**

```
var xar = new Charting.XAxisRenderer(plot3d1.xAxis);
xar.showCoordinates = false;
xar.titleBrush = new Drawing.Brush(new Drawing.Color("Green"));
xar.titleFontName = "Verdana";
xar.titleFontSize = 10;

var yar = new Charting.YAxisRenderer(plot3d1.yAxis);
yar.titleBrush = new Drawing.Brush(new Drawing.Color("Green"));
yar.titleFontName = "Verdana";
yar.titleFontSize = 10;

var panel3 = builder.createPlotWithTopAndRightAxes(plot3d1, xar, yar);
```

Create a BarOverlayRenderer3D instance, which draws each series as a different row of bars:

**JavaScript**

```
var barOverlay3DRenderer = new Charting.BarOverlayRenderer3D(new
Collections.ObservableCollection([data1, data2, data3]));
```

Set its style:

**JavaScript**

```
barOverlay3DRenderer.seriesStyle = new Charting.PerSeriesStyle(new Collections.List([
    new Drawing.Brush(Drawing.Color.fromArgb(0.6, 128, 0, 128)),
    new Drawing.Brush(Drawing.Color.fromArgb(0.6, 240, 230, 140)),
    new Drawing.Brush(Drawing.Color.fromArgb(0.6, 255, 165, 0))]), null, new
Collections.List([1]));
```

Create a plot and reuse axes of previous one:

**JavaScript**

```
var plot3d3 = new Plot3D();
plot3d3.seriesRenderers.add(barOverlay3DRenderer);
plot3d3.xAxis = plot3d1.xAxis;
plot3d3.yAxis = plot3d1.yAxis;
```

Add this plot and new axis renderers to the layout:

**JavaScript**

```
var xar = new Charting.XAxisRenderer(plot3d3.xAxis);
xar.titleBrush = new Drawing.Brush(new Drawing.Color("Purple"));
xar.titleFontName = "Courier New";
xar.titleFontSize = 12;

var yar = new Charting.YAxisRenderer(plot3d3.yAxis);
yar.titleBrush = new Drawing.Brush(new Drawing.Color("Purple"));
yar.titleFontName = "Courier New";
yar.titleFontSize = 12;

var panel5 = builder.createPlotWithTopAndRightAxes(plot3d3, xar, yar);
```

And the last one will show data series as 3D bar stacks:

**JavaScript**

```
var barStack3DRenderer = new Charting.BarStackRenderer3D(new
Collections.ObservableCollection([data1, data2, data3]));

barStack3DRenderer.seriesStyle = new Charting.PerSeriesStyle(new Collections.List([
    new Drawing.Brush(Drawing.Color.fromArgb(0.6, 128, 0, 128)),
    new Drawing.Brush(Drawing.Color.fromArgb(0.6, 240, 230, 140)),
    new Drawing.Brush(Drawing.Color.fromArgb(0.6, 255, 165, 0))]), null, new
Collections.List([1]));
```

```
var plot3d2 = new Plot3D();
plot3d2.seriesRenderers.add(barStack3DRenderer);
plot3d2.xAxis = plot3d1.xAxis;
```

Here we want a different Y axis range because the stacks will get higher than standalone bars from older plots:

**JavaScript**

```
plot3d2.yAxis = new Charting.Axis();
plot3d2.yAxis.minValue = 0;
plot3d2.yAxis.maxValue = 100;
plot3d2.yAxis.interval = 5;
plot3d2.yAxis.title = "Thousands";
```

Add a panel for the last plot and axes:

**JavaScript**

```
var xar = new Charting.XAxisRenderer(plot3d2.xAxis);
xar.titleBrush = new Drawing.Brush(new Drawing.Color("Blue"));
xar.titleFontName = "Times New Roman";
xar.titleFontSize = 15;

var yar = new Charting.YAxisRenderer(plot3d2.yAxis);
yar.titleBrush = new Drawing.Brush(new Drawing.Color("Blue"));
yar.titleFontName = "Times New Roman";
yar.titleFontSize = 15;
yar.titleFontStyle = Drawing.FontStyle.Bold;

var panel4 = builder.createPlotWithTopAndLeftAxes(plot3d2, xar, yar);
```

Set the grid positions for all child panels:

**JavaScript**

```
panel3.gridColumn = 1;
panel3.gridRow = 0;
panel4.gridColumn = 0;
panel4.gridRow = 1;
panel5.gridColumn = 1;
panel5.gridRow = 1;
```

Finally, arrange them in a grid in the lower-right corner of parent grid:

**JavaScript**

```
var miniGrid = new Components.GridPanel();
miniGrid.columns.add(new Components.GridColumn());
miniGrid.rows.add(new Components.GridRow());
miniGrid.children.add(panel3);
miniGrid.children.add(panel4);
miniGrid.children.add(panel5);
miniGrid.children.add(panel6);

miniGrid.gridColumn = 1;
miniGrid.gridRow = 1;
grid.children.add(miniGrid);
```

Now create a legend and the dashboard is complete. The legend will display the series titles we assigned earlier:

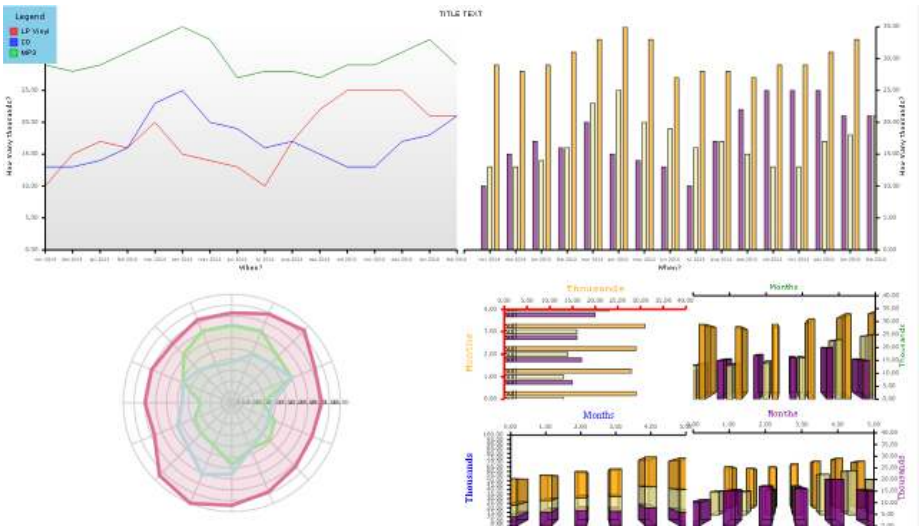
**JavaScript**

```
var legend = new Charting.LegendRenderer();
legend.content.add(lineRenderer);
dashboard.rootPanel.children.add(legend);
```

Finally, call the draw method of the dashboard to render it on the page:

```
JavaScript
dashboard.draw();
```

If you run the application, you should now see this image:



# Reference

# MindFusion.Charting Namespace

## AnnotationRenderer Class

A SeriesRenderer that draws annotations in its containing plot.

### Syntax

```
JavaScript
// class
AnnotationRenderer.prototype = {}
```

### Inheritance Hierarchy

### See Also

AnnotationRenderer Members
MindFusion.Charting Namespace

## AnnotationRenderer Members

A SeriesRenderer that draws annotations in its containing plot.

The following tables list the members exposed by the AnnotationRenderer type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | AnnotationRenderer | Initializes a new instance of the AnnotationRenderer class. |

## Public Methods

| | Name | Description |
|---|---|---|
| | draw | SeriesRenderer.Draw override. Draws the series data in specified RenderContext. |

## See Also

AnnotationRenderer Class
MindFusion.Charting Namespace

# AnnotationRenderer Constructor

Initializes a new instance of the AnnotationRenderer class.

**Namespace:** MindFusion.Charting
**File:** AnnotationRenderer.js

## Syntax

**JavaScript**

```
function AnnotationRenderer (series)
```

### Parameters

*series*
Type: ObservableCollection
A list of Series that should be rendered as annotations.

## See Also

AnnotationRenderer Class
MindFusion.Charting Namespace

# AnnotationRenderer Methods

# AnnotationRenderer.draw Method

SeriesRenderer.Draw override. Draws the series data in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** AnnotationRenderer.js

## Syntax

**JavaScript**

```
function draw (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

AnnotationRenderer Members
AnnotationRenderer Class
MindFusion.Charting Namespace

# AreaRenderer Class

A SeriesRenderer that draws each series as an area in its containing plot.

## Syntax

| JavaScript |
| --- |
| `// class`<br>`AreaRenderer.prototype = {}` |

## Inheritance Hierarchy

## See Also

AreaRenderer Members
MindFusion.Charting Namespace

## AreaRenderer Members

A SeriesRenderer that draws each series as an area in its containing plot.

The following tables list the members exposed by the AreaRenderer type.

## Public Constructors

| | Name | Description |
| --- | --- | --- |
| | AreaRenderer | Initializes a new instance of the AreaRenderer class. |

## Public Properties

| | Name | Description |
| --- | --- | --- |
| | areaOpacity | Gets or sets the opacity of area polygons. |

## Public Methods

| | Name | Description |
| --- | --- | --- |

| | | |
|---|---|---|
| | draw | SeriesRenderer.Draw override. Draws the series data in specified RenderContext. |

## See Also

AreaRenderer Class
MindFusion.Charting Namespace

# AreaRenderer Constructor

Initializes a new instance of the AreaRenderer class.

**Namespace:** MindFusion.Charting
**File:** AreaRenderer.js

## Syntax

**JavaScript**

```
function AreaRenderer (series)
```

### Parameters

*series*
Type: ObservableCollection
A list of Series that should be rendered as areas.

## See Also

AreaRenderer Class
MindFusion.Charting Namespace

# AreaRenderer Methods

# AreaRenderer.draw Method

SeriesRenderer.Draw override. Draws the series data in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** AreaRenderer.js

## Syntax

**JavaScript**

```
function draw (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

AreaRenderer Members
AreaRenderer Class
MindFusion.Charting Namespace

## AreaRenderer Properties

## AreaRenderer.areaOpacity Property

Gets or sets the opacity of area polygons.

**Namespace:** MindFusion.Charting
**File:** AreaRenderer.js

### Syntax

```
JavaScript
get areaOpacity() {}
```

#### Property Value

Type: Number

### See Also

AreaRenderer Members
AreaRenderer Class
MindFusion.Charting Namespace

# AreaStackRenderer Class

A SeriesRenderer that draws stacked areas in its containing plot.

### Syntax

```
JavaScript
// class
AreaStackRenderer.prototype = {}
```

### Inheritance Hierarchy

### See Also

AreaStackRenderer Members
MindFusion.Charting Namespace

## AreaStackRenderer Members

A SeriesRenderer that draws stacked areas in its containing plot.

The following tables list the members exposed by the AreaStackRenderer type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | AreaStackRenderer | Initializes a new instance of the AreaStackRenderer class. |

## Public Methods

| | Name | Description |
|---|---|---|
| | draw | SeriesRenderer.Draw override. Draws the series data in specified RenderContext. |

## See Also

AreaStackRenderer Class
MindFusion.Charting Namespace

# AreaStackRenderer Constructor

Initializes a new instance of the AreaStackRenderer class.

**Namespace:** MindFusion.Charting
**File:** AreaStackRenderer.js

## Syntax

**JavaScript**

```
function AreaStackRenderer (series)
```

### Parameters

*series*
Type: ObservableCollection
A list of Series that should be rendered as stacked areas.

## See Also

AreaStackRenderer Class
MindFusion.Charting Namespace

# AreaStackRenderer Methods

# AreaStackRenderer.draw Method

SeriesRenderer.Draw override. Draws the series data in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** AreaStackRenderer.js

## Syntax

**JavaScript**

```
function draw (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

AreaStackRenderer Members
AreaStackRenderer Class
MindFusion.Charting Namespace

# Axis Class

Represents an Axis range.

## Syntax

| JavaScript |
| --- |
| `// class`<br>`Axis.prototype = {}` |

## Inheritance Hierarchy

## See Also

Axis Members
MindFusion.Charting Namespace

# Axis Members

Represents an Axis range.

The following tables list the members exposed by the Axis type.

## Public Constructors

| | Name | Description |
| --- | --- | --- |
| | Axis | Initializes a new instance of the Axis class. |

## Public Properties

| | Name | Description |
| --- | --- | --- |
| | interval | Gets or sets the size of axis intervals. |
| | maxValue | Gets or sets the largest value displayed on this axis. |
| | minValue | Gets or sets the smallest value displayed on this axis. |

| | | |
|---|---|---|
| | numberFormat | Gets or sets the number format of coordinate labels. |
| | origin | Gets or sets the origin of the **Axis**. |
| | propertyChanged | Occurs when a property value changes. |
| | title | Gets or sets the axis title. |

## Public Methods

| | Name | Description |
|---|---|---|
| | inRange | Determines whether specified value is within the axis range. |
| | mapValueToPixelX | Maps a value from this axis' coordinate system to a pixel position in specified view. |
| | mapValueToPixelY | Maps a value from this axis' coordinate system to a pixel position in specified view. |
| | onPropertyChanged | Raises the PropertyChanged event. |

## See Also

Axis Class
MindFusion.Charting Namespace

# Axis Constructor

Initializes a new instance of the Axis class.

**Namespace:** MindFusion.Charting
**File:** Axis.js

## Syntax

**JavaScript**

```
function Axis ()
```

## See Also

Axis Class
MindFusion.Charting Namespace

# Axis Methods

# Axis.inRange Method

Determines whether specified value is within the axis range.

**Namespace:** MindFusion.Charting
**File:** Axis.js

## Syntax

JavaScript

```
function inRange (value)
```

### Parameters

*value*
Type: Number
The number to check.

### Return Value

Type: Boolean
true if value is inside the current axis range, or false otherwise.

## See Also

Axis Members
Axis Class
MindFusion.Charting Namespace

# Axis.mapValueToPixelX Method

Maps a value from this axis' coordinate system to a pixel position in specified view.

**Namespace:** MindFusion.Charting
**File:** Axis.js

## Syntax

JavaScript

```
function mapValueToPixelX (value, viewSize)
```

### Parameters

*value*
Type: Number
A number value.

*viewSize*
Type: Number
The view size.

### Return Value

Type: Number
X coordinate of pixel.

## See Also

Axis Members
Axis Class
MindFusion.Charting Namespace

# Axis.mapValueToPixelY Method

Maps a value from this axis' coordinate system to a pixel position in specified view.

**Namespace:** MindFusion.Charting
**File:** Axis.js

## Syntax

**JavaScript**

```
function mapValueToPixelY (value, viewSize)
```

### Parameters

*value*
Type: Number
A number value.

*viewSize*
Type: Number
The view size.

### Return Value

Type: Number
Y coordinate of pixel.

## See Also

Axis Members
Axis Class
MindFusion.Charting Namespace

# Axis.onPropertyChanged Method

Raises the PropertyChanged event.

**Namespace:** MindFusion.Charting
**File:** Axis.js

## Syntax

**JavaScript**

```
function onPropertyChanged (propertyName)
```

### Parameters

*propertyName*
Type: String
Specifies the name of changed property.

## See Also

Axis Members
Axis Class
MindFusion.Charting Namespace

# Axis Properties

## Axis.interval Property

Gets or sets the size of axis intervals.

**Namespace:** MindFusion.Charting
**File:** Axis.js

### Syntax

```
JavaScript
get interval() {}
```

#### Property Value

Type: Number

### See Also

Axis Members
Axis Class
MindFusion.Charting Namespace

## Axis.maxValue Property

Gets or sets the largest value displayed on this axis.

**Namespace:** MindFusion.Charting
**File:** Axis.js

### Syntax

```
JavaScript
get maxValue() {}
```

#### Property Value

Type: Number

### See Also

Axis Members
Axis Class
MindFusion.Charting Namespace

## Axis.minValue Property

Gets or sets the smallest value displayed on this axis.

**Namespace:** MindFusion.Charting
**File:** Axis.js

### Syntax

```
JavaScript
get minValue() {}
```

### Property Value

Type: Number

## See Also

Axis Members
Axis Class
MindFusion.Charting Namespace

# Axis.numberFormat Property

Gets or sets the number format of coordinate labels.

**Namespace:** MindFusion.Charting
**File:** Axis.js

## Syntax

```
JavaScript
get numberFormat() {}
```

### Property Value

Type: **Intl.NumberFormat**

## Example

```
JavaScript
axis.numberFormat = new Intl.NumberFormat("en-US", {
    minimumIntegerDigits: 1,
    minimumFractionDigits: 0,
    maximumFractionDigits: 2});
```

## See Also

Axis Members
Axis Class
MindFusion.Charting Namespace

# Axis.origin Property

Gets or sets the origin of the Axis.

**Namespace:** MindFusion.Charting
**File:** Axis.js

## Syntax

```
JavaScript
```

```
get origin() {}
```

### Property Value

Type: Number

## Example

The following code sets some properties of an X-axis, among which also the **origin**:

**JavaScript**

```
chart.xAxis.title = "Temperature";
chart.xAxis.interval = 1;
chart.xAxis.origin = -10;
```

## See Also

Axis Members
Axis Class
MindFusion.Charting Namespace

# Axis.propertyChanged Property

Occurs when a property value changes.

**Namespace:** MindFusion.Charting
**File:** Axis.js

## Syntax

**JavaScript**

```
get propertyChanged() {}
```

### Property Value

Type: PropertyChangedEventDispatcher

## See Also

Axis Members
Axis Class
MindFusion.Charting Namespace

# Axis.title Property

Gets or sets the axis title.

**Namespace:** MindFusion.Charting
**File:** Axis.js

## Syntax

**JavaScript**

```
get title() {}
```

### Property Value

Type: String

## See Also

Axis Members
Axis Class
MindFusion.Charting Namespace

# AxisInfo Class

Stores information about Axis ranges processed by this controller.

## Syntax

| JavaScript |
| --- |
| `// class`<br>`AxisInfo.prototype = {}` |

## Inheritance Hierarchy

## See Also

AxisInfo Members
MindFusion.Charting Namespace

## AxisInfo Members

Stores information about Axis ranges processed by this controller.

The following tables list the members exposed by the AxisInfo type.

## Public Properties

| | Name | Description |
| --- | --- | --- |
|  | origin | The axis origin. |
|  | range | The axis range. |
|  | resolution | The axis resolution. |

## See Also

AxisInfo Class
MindFusion.Charting Namespace

## AxisInfo Properties

# AxisInfo.origin Property

The axis origin.

**Namespace:** MindFusion.Charting
**File:** Plot2DController.js

## Syntax

```
JavaScript
get origin() {}
```

### Property Value

Type: Number

## See Also

AxisInfo Members
AxisInfo Class
MindFusion.Charting Namespace

# AxisInfo.range Property

The axis range.

**Namespace:** MindFusion.Charting
**File:** Plot2DController.js

## Syntax

```
JavaScript
get range() {}
```

### Property Value

Type: Number

## See Also

AxisInfo Members
AxisInfo Class
MindFusion.Charting Namespace

# AxisInfo.resolution Property

The axis resolution.

**Namespace:** MindFusion.Charting
**File:** Plot2DController.js

## Syntax

```
JavaScript
```

```
get resolution() {}
```

### Property Value

Type: Number

## See Also

AxisInfo Members
AxisInfo Class
MindFusion.Charting Namespace

# AxisRenderer Class

A base class for components that render Axis ranges.

## Syntax

**JavaScript**

```
// class
AxisRenderer.prototype = {}
```

## Inheritance Hierarchy

## See Also

AxisRenderer Members
MindFusion.Charting Namespace

# AxisRenderer Members

A base class for components that render Axis ranges.

The following tables list the members exposed by the AxisRenderer type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | AxisRenderer | Initializes a new instance of the AxisRenderer class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | axis | Gets or sets the Axis that will be drawn by this object. |
|  | axisStroke | Gets or sets the Brush that should be used to draw axis lines. |
|  | axisStrokeDashStyle | Gets or sets the dash style axis lines should be stroked with. |
|  | axisStrokeThickness | Gets or sets the thickness axis lines should be stroked with. |

| | | |
|---|---|---|
| | labelBrush | Gets or sets the Brush that should be used to draw axis labels. |
| | labelFontName | Gets or sets the name of font that should be used to draw axis labels. |
| | labelFontSize | Gets or sets the size of font that should be used to draw axis labels. |
| | labelFontStyle | Gets or sets the style of font that should be used to draw axis labels. |
| | labelPadding | Gets or sets the padding space between the axis line and coordinate labels. |
| | labelRotationAngle | Gets or sets the rotation angle of the labels. |
| | labelsSource | Gets or sets the object whose Series labels should be rendered along this axis. |
| | pinLabels | Gets or sets a value indicating whether coordinate labels should be pinned in place or scroll together with the plot when users pan it. |
| | showCoordinates | Gets or sets a value indicating whether this AxisRenderer should draw axis coordinates. |
| | showSeriesLabels | Gets or sets a value indicating whether this AxisRenderer should draw data labels. |
| | showTicks | Gets or sets a value indicating whether this AxisRenderer should draw axis ticks. |
| | titleBrush | Gets or sets the Brush that should be used to draw the axis Title. |
| | titleFontName | Gets or sets the name of font that should be used to draw the axis Title. |
| | titleFontSize | Gets or sets the size of font that should be used to draw the axis Title. |
| | titleFontStyle | Gets or sets the style of font that should be used to draw the axis Title. |

## Public Methods

| | Name | Description |
|---|---|---|
| | effectiveAxisDashStyle | Gets the dash style of axis line strokes. |
| | effectiveAxisStroke | Gets the Brush that should be used to stroke axis lines. |
| | effectiveAxisStrokeThickness | Gets the thickness of axis line strokes. |
| | effectiveLabelBrush | Gets the Brush that should be used to draw axis labels. |
| | effectiveLabelFont | Gets the Font that should be used to draw axis labels. |
| | effectivePen | Gets the Pen that should be used to draw axis lines. |
| | effectiveTitleBrush | Gets the Brush that should be used to draw the axis Title. |

| | | |
|---|---|---|
| | effectiveTitleFont | Gets the Font that should be used to draw the axis Title. |

## See Also

AxisRenderer Class
MindFusion.Charting Namespace

# AxisRenderer Constructor

Initializes a new instance of the AxisRenderer class.

**Namespace:** MindFusion.Charting
**File:** AxisRenderer.js

## Syntax

**JavaScript**

```
function AxisRenderer (axis)
```

### Parameters

*axis*
Type: Axis
The Axis that will be drawn by this object.

## See Also

AxisRenderer Class
MindFusion.Charting Namespace

# AxisRenderer Methods

# AxisRenderer.effectiveAxisDashStyle Method

Gets the dash style of axis line strokes.

**Namespace:** MindFusion.Charting
**File:** AxisRenderer.js

## Syntax

**JavaScript**

```
function effectiveAxisDashStyle (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: DashStyle
A member of the DashStyle enumeration.

## See Also

AxisRenderer Members
AxisRenderer Class
MindFusion.Charting Namespace

# AxisRenderer.effectiveAxisStroke Method

Gets the Brush that should be used to stroke axis lines.

**Namespace:** MindFusion.Charting
**File:** AxisRenderer.js

## Syntax

**JavaScript**

```
function effectiveAxisStroke (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: Brush
A Brush instance.

## See Also

AxisRenderer Members
AxisRenderer Class
MindFusion.Charting Namespace

# AxisRenderer.effectiveAxisStrokeThickness Method

Gets the thickness of axis line strokes.

**Namespace:** MindFusion.Charting
**File:** AxisRenderer.js

## Syntax

**JavaScript**

```
function effectiveAxisStrokeThickness (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: Number
A number value.

## See Also

AxisRenderer Members
AxisRenderer Class
MindFusion.Charting Namespace

# AxisRenderer.effectiveLabelBrush Method

Gets the Brush that should be used to draw axis labels.

**Namespace:** MindFusion.Charting
**File:** AxisRenderer.js

## Syntax

**JavaScript**

```
function effectiveLabelBrush (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: SysBrush
A System.Brush instance.

## See Also

AxisRenderer Members
AxisRenderer Class
MindFusion.Charting Namespace

# AxisRenderer.effectiveLabelFont Method

Gets the Font that should be used to draw axis labels.

**Namespace:** MindFusion.Charting
**File:** AxisRenderer.js

## Syntax

**JavaScript**

```
function effectiveLabelFont (context)
```

### Parameters

*context*

Type: RenderContext
A RenderContext instance.

### Return Value

Type: Font
A System.Drawing.Font instance.

## See Also

AxisRenderer Members
AxisRenderer Class
MindFusion.Charting Namespace

# AxisRenderer.effectivePen Method

Gets the Pen that should be used to draw axis lines.

**Namespace:** MindFusion.Charting
**File:** AxisRenderer.js

## Syntax

**JavaScript**

```
function effectivePen (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: Pen
A System.Pen instance.

## See Also

AxisRenderer Members
AxisRenderer Class
MindFusion.Charting Namespace

# AxisRenderer.effectiveTitleBrush Method

Gets the Brush that should be used to draw the axis Title.

**Namespace:** MindFusion.Charting
**File:** AxisRenderer.js

## Syntax

**JavaScript**

```
function effectiveTitleBrush (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: SysBrush
A System.Brush instance.

## See Also

AxisRenderer Members
AxisRenderer Class
MindFusion.Charting Namespace

# AxisRenderer.effectiveTitleFont Method

Gets the Font that should be used to draw the axis Title.

**Namespace:** MindFusion.Charting
**File:** AxisRenderer.js

## Syntax

**JavaScript**

```
function effectiveTitleFont (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: Font
A System.Drawing.Font instance.

## See Also

AxisRenderer Members
AxisRenderer Class
MindFusion.Charting Namespace

# AxisRenderer Properties

# AxisRenderer.axis Property

Gets or sets the Axis that will be drawn by this object.

**Namespace:** MindFusion.Charting
**File:** AxisRenderer.js

## Syntax

```
JavaScript
get axis() {}
```

### Property Value

Type: Axis

## See Also

AxisRenderer Members
AxisRenderer Class
MindFusion.Charting Namespace

# AxisRenderer.axisStroke Property

Gets or sets the Brush that should be used to draw axis lines.

**Namespace:** MindFusion.Charting
**File:** AxisRenderer.js

## Syntax

```
JavaScript
get axisStroke() {}
```

### Property Value

Type: Brush

## See Also

AxisRenderer Members
AxisRenderer Class
MindFusion.Charting Namespace

# AxisRenderer.axisStrokeDashStyle Property

Gets or sets the dash style axis lines should be stroked with.

**Namespace:** MindFusion.Charting
**File:** AxisRenderer.js

## Syntax

```
JavaScript
get axisStrokeDashStyle() {}
```

### Property Value

Type: DashStyle

## See Also

AxisRenderer Members
AxisRenderer Class
MindFusion.Charting Namespace

# AxisRenderer.axisStrokeThickness Property

Gets or sets the thickness axis lines should be stroked with.

**Namespace:** MindFusion.Charting
**File:** AxisRenderer.js

## Syntax

**JavaScript**

```
get axisStrokeThickness() {}
```

### Property Value

Type: Number

## See Also

AxisRenderer Members
AxisRenderer Class
MindFusion.Charting Namespace

# AxisRenderer.labelBrush Property

Gets or sets the Brush that should be used to draw axis labels.

**Namespace:** MindFusion.Charting
**File:** AxisRenderer.js

## Syntax

**JavaScript**

```
get labelBrush() {}
```

### Property Value

Type: Brush

## See Also

AxisRenderer Members
AxisRenderer Class
MindFusion.Charting Namespace

# AxisRenderer.labelFontName Property

Gets or sets the name of font that should be used to draw axis labels.

**Namespace:** MindFusion.Charting
**File:** AxisRenderer.js

## Syntax

```
JavaScript
get labelFontName() {}
```

### Property Value

Type: String

## See Also

AxisRenderer Members
AxisRenderer Class
MindFusion.Charting Namespace

# AxisRenderer.labelFontSize Property

Gets or sets the size of font that should be used to draw axis labels.

**Namespace:** MindFusion.Charting
**File:** AxisRenderer.js

## Syntax

```
JavaScript
get labelFontSize() {}
```

### Property Value

Type: Number

## See Also

AxisRenderer Members
AxisRenderer Class
MindFusion.Charting Namespace

# AxisRenderer.labelFontStyle Property

Gets or sets the style of font that should be used to draw axis labels.

**Namespace:** MindFusion.Charting
**File:** AxisRenderer.js

## Syntax

```
JavaScript
get labelFontStyle() {}
```

### Property Value

Type: FontStyle

## See Also

AxisRenderer Members
AxisRenderer Class
MindFusion.Charting Namespace

# AxisRenderer.labelPadding Property

Gets or sets the padding space between the axis line and coordinate labels.

**Namespace:** MindFusion.Charting
**File:** AxisRenderer.js

## Syntax

**JavaScript**

```
get labelPadding() {}
```

### Property Value

Type: Number

## See Also

AxisRenderer Members
AxisRenderer Class
MindFusion.Charting Namespace

# AxisRenderer.labelRotationAngle Property

Gets or sets the rotation angle of the labels.

**Namespace:** MindFusion.Charting
**File:** AxisRenderer.js

## Syntax

**JavaScript**

```
get labelRotationAngle() {}
```

### Property Value

Type: Number

## See Also

AxisRenderer Members
AxisRenderer Class
MindFusion.Charting Namespace

## AxisRenderer.labelsSource Property

Gets or sets the object whose Series labels should be rendered along this axis.

**Namespace:** MindFusion.Charting
**File:** AxisRenderer.js

### Syntax

JavaScript

```
get labelsSource() {}
```

#### Property Value

Type: SeriesContainer

### See Also

AxisRenderer Members
AxisRenderer Class
MindFusion.Charting Namespace

## AxisRenderer.pinLabels Property

Gets or sets a value indicating whether coordinate labels should be pinned in place or scroll together with the plot when users pan it.

**Namespace:** MindFusion.Charting
**File:** AxisRenderer.js

### Syntax

JavaScript

```
get pinLabels() {}
```

#### Property Value

Type: Boolean

### See Also

AxisRenderer Members
AxisRenderer Class
MindFusion.Charting Namespace

## AxisRenderer.showCoordinates Property

Gets or sets a value indicating whether this AxisRenderer should draw axis coordinates.

**Namespace:** MindFusion.Charting
**File:** AxisRenderer.js

### Syntax

JavaScript

```
get showCoordinates() {}
```

### Property Value

Type: Boolean

## See Also

AxisRenderer Members
AxisRenderer Class
MindFusion.Charting Namespace

# AxisRenderer.showSeriesLabels Property

Gets or sets a value indicating whether this AxisRenderer should draw data labels.

**Namespace:** MindFusion.Charting
**File:** AxisRenderer.js

## Syntax

**JavaScript**

```
get showSeriesLabels() {}
```

### Property Value

Type: Boolean

## See Also

AxisRenderer Members
AxisRenderer Class
MindFusion.Charting Namespace

# AxisRenderer.showTicks Property

Gets or sets a value indicating whether this AxisRenderer should draw axis ticks.

**Namespace:** MindFusion.Charting
**File:** AxisRenderer.js

## Syntax

**JavaScript**

```
get showTicks() {}
```

### Property Value

Type: Boolean

## See Also

AxisRenderer Members
AxisRenderer Class

## AxisRenderer.titleBrush Property

Gets or sets the Brush that should be used to draw the axis Title.

**Namespace:** MindFusion.Charting
**File:** AxisRenderer.js

### Syntax

**JavaScript**

```
get titleBrush() {}
```

#### Property Value

Type: Brush

### See Also

AxisRenderer Members
AxisRenderer Class
MindFusion.Charting Namespace

## AxisRenderer.titleFontName Property

Gets or sets the name of font that should be used to draw the axis Title.

**Namespace:** MindFusion.Charting
**File:** AxisRenderer.js

### Syntax

**JavaScript**

```
get titleFontName() {}
```

#### Property Value

Type: String

### See Also

AxisRenderer Members
AxisRenderer Class
MindFusion.Charting Namespace

## AxisRenderer.titleFontSize Property

Gets or sets the size of font that should be used to draw the axis Title.

**Namespace:** MindFusion.Charting
**File:** AxisRenderer.js

### Syntax

```
JavaScript
get titleFontSize() {}
```

### Property Value

Type: Number

## See Also

AxisRenderer Members
AxisRenderer Class
MindFusion.Charting Namespace

# AxisRenderer.titleFontStyle Property

Gets or sets the style of font that should be used to draw the axis Title.

**Namespace:** MindFusion.Charting
**File:** AxisRenderer.js

## Syntax

```
JavaScript
get titleFontStyle() {}
```

### Property Value

Type: FontStyle

## See Also

AxisRenderer Members
AxisRenderer Class
MindFusion.Charting Namespace

# BarContainer Interface

Defines properties common to all bar-chart renderers.

## Syntax

```
JavaScript
// interface
BarContainer.prototype = {}
```

## See Also

BarContainer Members
MindFusion.Charting Namespace

# BarContainer Members

Defines properties common to all bar-chart renderers.

The following tables list the members exposed by the BarContainer type.

## Public Properties

| | Name | Description |
|---|---|---|
|  | barSpacingRatio | Gets or sets the ratio of empty space between bars to space occupied by bars. |
|  | horizontalBars | Gets or sets a value identifying whether bars should be horizontal. |

## See Also

BarContainer Interface
MindFusion.Charting Namespace

# BarContainer Properties

## BarContainer.barSpacingRatio Property

Gets or sets the ratio of empty space between bars to space occupied by bars.

**Namespace:** MindFusion.Charting
**File:** BarContainer.js

## Syntax

**JavaScript**

```
get barSpacingRatio() {}
```

### Property Value

Type: Number

## See Also

BarContainer Members
BarContainer Interface
MindFusion.Charting Namespace

## BarContainer.horizontalBars Property

Gets or sets a value identifying whether bars should be horizontal.

**Namespace:** MindFusion.Charting
**File:** BarContainer.js

## Syntax

**JavaScript**

```
get horizontalBars() {}
```

### Property Value

Type: Boolean

## See Also

BarContainer Members
BarContainer Interface
MindFusion.Charting Namespace

# BarLayout Enumeration

Identifies supported bar placements when rendering multiple series.

**Namespace:** MindFusion.Charting
**File:** BarLayout.js

## Syntax

**JavaScript**

```
// enum
BarLayout = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **Overlay** | Render elements at same index from all series as overlaid bars. |
| | **SideBySide** | Render elements at same index from all series as a group of side-by-side bars. |
| | **Stack** | Render elements at same index from all series as stacked bars. |

## See Also

MindFusion.Charting Namespace

# BarModel3D Enumeration

Identifies supported bar shapes when rendering multiple series.

**Namespace:** MindFusion.Charting
**File:** BarModel3D.js

## Syntax

**JavaScript**

```
// enum
BarModel3D = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **Bar** | Draw bars as rectangular cuboids. |
| | **Cylinder** | Draw bars as cylinders. |

## See Also

MindFusion.Charting Namespace

# BarOverlayRenderer Class

A SeriesRenderer that draws overlaying bars in its containing plot.

## Syntax

**JavaScript**

```
// class
BarOverlayRenderer.prototype = {}
```

## Inheritance Hierarchy

## See Also

BarOverlayRenderer Members
MindFusion.Charting Namespace

# BarOverlayRenderer Members

A SeriesRenderer that draws overlaying bars in its containing plot.

The following tables list the members exposed by the BarOverlayRenderer type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | BarOverlayRenderer | Initializes a new instance of the BarOverlayRenderer class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | barSpacingRatio | Gets or sets the ratio of empty space between adjacent bar groups to space occupied by bars. |

| | | |
|---|---|---|
| | horizontalBars | Gets or sets a value identifying whether bars should be horizontal. |

## Public Methods

| | Name | Description |
|---|---|---|
| | adjustDataRange | seriesRenderer.AdjustDataRange override. Adjusts the data range of associated Axis objects after initial measure pass. |
| | backToFrontRatio | Gets the ratio between widths of backmost and frontmost bars. |
| | draw | SeriesRenderer.Draw override. Draws the series data in specified RenderContext. |
| | drawHighlight | Renderer2D.DrawHighlight override. Draws highlighted bar in specified RenderContext. |
| | enumVisibleOverlays | Enumerates the bars visible in current data range. |
| | hitTest | Renderer2D.HitTest override. Hit-tests the bar representations of data items. |
| | measureDataRange | Renderer2D.MeasureDataRange override. Measures the data range of rendered series and assigns it to the associated Axis objects if their MinValue and MaxValue are not set. |

## See Also

BarOverlayRenderer Class
MindFusion.Charting Namespace

# BarOverlayRenderer Constructor

Initializes a new instance of the BarOverlayRenderer class.

**Namespace:** MindFusion.Charting
**File:** BarOverlayRenderer.js

## Syntax

**JavaScript**

```
function BarOverlayRenderer (series)
```

### Parameters

*series*
Type: ObservableCollection
A list of Series that should be rendered as overlaying bars.

## See Also

BarOverlayRenderer Class
MindFusion.Charting Namespace

# BarOverlayRenderer Methods

## BarOverlayRenderer.adjustDataRange Method

seriesRenderer.AdjustDataRange override. Adjusts the data range of associated Axis objects after initial measure pass.

**Namespace:** MindFusion.Charting
**File:** BarOverlayRenderer.js

### Syntax

```
JavaScript
function adjustDataRange (context)
```

#### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### See Also

BarOverlayRenderer Members
BarOverlayRenderer Class
MindFusion.Charting Namespace

## BarOverlayRenderer.backToFrontRatio Method

Gets the ratio between widths of backmost and frontmost bars.

**Namespace:** MindFusion.Charting
**File:** BarOverlayRenderer.js

### Syntax

```
JavaScript
function backToFrontRatio ()
```

#### Return Value

Type: Number
3

### See Also

BarOverlayRenderer Members
BarOverlayRenderer Class
MindFusion.Charting Namespace

## BarOverlayRenderer.draw Method

SeriesRenderer.Draw override. Draws the series data in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** BarOverlayRenderer.js

## Syntax

```
JavaScript
function draw (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

BarOverlayRenderer Members
BarOverlayRenderer Class
MindFusion.Charting Namespace

# BarOverlayRenderer.drawHighlight Method

Renderer2D.DrawHighlight override. Draws highlighted bar in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** BarOverlayRenderer.js

## Syntax

```
JavaScript
function drawHighlight (context, hitResult)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*hitResult*
Type: HitResult
A HitResult instance identifying highlighted data item.

## See Also

BarOverlayRenderer Members
BarOverlayRenderer Class
MindFusion.Charting Namespace

# BarOverlayRenderer.enumVisibleOverlays Method

Enumerates the bars visible in current data range.

**Namespace:** MindFusion.Charting
**File:** BarOverlayRenderer.js

## Syntax

```
JavaScript
function enumVisibleOverlays (context, frontToBack, process)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*frontToBack*
Type: Boolean
true to enumerate from front bar to back bar, or false otherwise.

*process*
Type: ProcessBars
A ProcessBars callback.

## Remarks

The signature of the ProcessBars delegate is as follows:'function(seriesIndex, dataIndex, bounds)'.param {Number} seriesIndex An integer index of series in Series list.param {Number} dataIndex An integer index of data item in specified series.param {MindFusion.Drawing.Rect} bounds A RectangleF representing the boundaries of current bar.The method is not expected to return a value.

## See Also

BarOverlayRenderer Members
BarOverlayRenderer Class
MindFusion.Charting Namespace

# BarOverlayRenderer.hitTest Method

Renderer2D.HitTest override. Hit-tests the bar representations of data items.

**Namespace:** MindFusion.Charting
**File:** BarOverlayRenderer.js

## Syntax

```
JavaScript
function hitTest (context, location)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*location*
Type: Point
A Point specifying where to look for a bar.

### Return Value

Type: HitResult
A HitResult instance identifying the found data item.

## See Also

BarOverlayRenderer Members
BarOverlayRenderer Class
MindFusion.Charting Namespace

# BarOverlayRenderer.measureDataRange Method

Renderer2D.MeasureDataRange override. Measures the data range of rendered series and assigns it to the associated Axis objects if their MinValue and MaxValue are not set.

**Namespace:** MindFusion.Charting
**File:** BarOverlayRenderer.js

## Syntax

**JavaScript**

```
function measureDataRange (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

BarOverlayRenderer Members
BarOverlayRenderer Class
MindFusion.Charting Namespace

# BarOverlayRenderer Properties

# BarOverlayRenderer.barSpacingRatio Property

Gets or sets the ratio of empty space between adjacent bar groups to space occupied by bars.

**Namespace:** MindFusion.Charting
**File:** BarOverlayRenderer.js

## Syntax

**JavaScript**

```
get barSpacingRatio() {}
```

### Property Value

Type: Number

## See Also

BarOverlayRenderer Members
BarOverlayRenderer Class
MindFusion.Charting Namespace

## BarOverlayRenderer.horizontalBars Property

Gets or sets a value identifying whether bars should be horizontal.

**Namespace:** MindFusion.Charting
**File:** BarOverlayRenderer.js

### Syntax

```
JavaScript
get horizontalBars() {}
```

#### Property Value

Type: Boolean

### See Also

BarOverlayRenderer Members
BarOverlayRenderer Class
MindFusion.Charting Namespace

# BarOverlayRenderer3D Class

A SeriesRenderer that draws a row of 3D bars for each series.

### Syntax

```
JavaScript
// class
BarOverlayRenderer3D.prototype = {}
```

### Inheritance Hierarchy

### See Also

BarOverlayRenderer3D Members
MindFusion.Charting Namespace

## BarOverlayRenderer3D Members

A SeriesRenderer that draws a row of 3D bars for each series.

The following tables list the members exposed by the BarOverlayRenderer3D type.

### Public Constructors

| | Name | Description |
|---|---|---|

| | | |
|---|---|---|
| | BarOverlayRenderer3D | Initializes a new instance of the BarOverlayRenderer3D class. |

## Public Methods

| | Name | Description |
|---|---|---|
| | backToFrontRatio | BarOverlayRenderer.BackToFrontRatio override. Gets the ratio between widths of backmost and frontmost bars. |
| | buildModels | Implements Renderer3D.BuildModels. Adds 3D bar models for the series data to specified scene. |

## See Also

BarOverlayRenderer3D Class
MindFusion.Charting Namespace

# BarOverlayRenderer3D Constructor

Initializes a new instance of the BarOverlayRenderer3D class.

**Namespace:** MindFusion.Charting
**File:** BarOverlayRenderer3D.js

## Syntax

**JavaScript**

```
function BarOverlayRenderer3D (series)
```

### Parameters

*series*
Type: ObservableCollection
A list of Series that should be rendered as rows of bars.

## See Also

BarOverlayRenderer3D Class
MindFusion.Charting Namespace

# BarOverlayRenderer3D Methods

# BarOverlayRenderer3D.backToFrontRatio Method

BarOverlayRenderer.BackToFrontRatio override. Gets the ratio between widths of backmost and frontmost bars.

**Namespace:** MindFusion.Charting
**File:** BarOverlayRenderer3D.js

## Syntax

**JavaScript**

```
function backToFrontRatio ()
```

### Return Value

Type: Number
1.

## See Also

BarOverlayRenderer3D Members
BarOverlayRenderer3D Class
MindFusion.Charting Namespace

# BarOverlayRenderer3D.buildModels Method

Implements Renderer3D.BuildModels. Adds 3D bar models for the series data to specified scene.

**Namespace:** MindFusion.Charting
**File:** BarOverlayRenderer3D.js

## Syntax

**JavaScript**

```
function buildModels (scene, context)
```

### Parameters

*scene*

Type: Scene3D
A Scene3D instance.

*context*

Type: RenderContext
A RenderContext instance.

## See Also

BarOverlayRenderer3D Members
BarOverlayRenderer3D Class
MindFusion.Charting Namespace

# BarRenderer Class

A SeriesRenderer that draws bars in its containing plot.

## Syntax

**JavaScript**

```
// class
BarRenderer.prototype = {}
```

## Remarks

Use the class to add a bar chart in a dashboard or create a custom bar chart.

## Example

The following code creates a new BarSeries and a new **BarRenderer** using that series:

**JavaScript**

```
var data = new Charting.BarSeries(
 new Collections.List([13, 13, 14, 16, 23, 25, 20, 19, 16, 17, 15, 13, 13, 17, 18, 21]),
null);
var barRenderer = new Charting.BarRenderer(new Collections.ObservableCollection([data]));
```

## Inheritance Hierarchy

## See Also

BarRenderer Members
MindFusion.Charting Namespace

# BarRenderer Members

A SeriesRenderer that draws bars in its containing plot.

The following tables list the members exposed by the BarRenderer type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | BarRenderer | Initializes a new instance of the BarRenderer class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | barSpacingRatio | Gets or sets the ratio of empty space between adjacent bar groups to space occupied by bars. |
|  | horizontalBars | Gets or sets a value identifying whether bars should be horizontal. |
|  | innerLabelRotation | Gets the rotation angle of inner labels. If not specified, labels are rotated at predefiend angles depending on bars' orientation and layout. |
|  | outerLabelRotation | Gets the rotation angle of outer labels. If not specified, labels are rotated at predefiend angles depending on bars' orientation and layout. |
|  | sameIndexSpacingRatio | Gets or sets the ratio of empty space to occupied space in bar groups drawn for data items as same index in each series. |
|  | stackOuterLabels | Gets whether outer labels are displayed stacked on top of stacked bars, instead of showing them on the bar sides. |

## Public Methods

| | Name | Description |
|---|---|---|
| | adjustDataRange | SeriesRenderer.AdjustDataRange override. Adjusts the data range of associated Axis objects after initial measure pass. |
| | draw | SeriesRenderer.Draw override. Draws the series data in specified RenderContext. |
| | drawHighlight | Renderer2D.DrawHighlight override. Draws highlighted bar in specified RenderContext. |
| | enumVisibleBars | Enumerates the bars visible in current data range. |
| | hitTest | Renderer2D.HitTest override. Hit-tests the bar representations of data items. |
| | measureDataRange | Renderer2D.MeasureDataRange override. Measures the data range of rendered series and assigns it to the associated Axis objects if their MinValue and MaxValue are not set. |

## See Also

BarRenderer Class
MindFusion.Charting Namespace

# BarRenderer Constructor

Initializes a new instance of the BarRenderer class.

**Namespace:** MindFusion.Charting
**File:** BarRenderer.js

## Syntax

**JavaScript**

```
function BarRenderer (series)
```

### Parameters

*series*
Type: ObservableCollection
A list of Series that should be rendered as bars.

## See Also

BarRenderer Class
MindFusion.Charting Namespace

# BarRenderer Methods

# BarRenderer.adjustDataRange Method

SeriesRenderer.AdjustDataRange override. Adjusts the data range of associated Axis objects after initial measure pass.

**Namespace:** MindFusion.Charting
**File:** BarRenderer.js

## Syntax

```
JavaScript
function adjustDataRange (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

BarRenderer Members
BarRenderer Class
MindFusion.Charting Namespace

# BarRenderer.draw Method

SeriesRenderer.Draw override. Draws the series data in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** BarRenderer.js

## Syntax

```
JavaScript
function draw (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

BarRenderer Members
BarRenderer Class
MindFusion.Charting Namespace

# BarRenderer.drawHighlight Method

Renderer2D.DrawHighlight override. Draws highlighted bar in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** BarRenderer.js

## Syntax

```
JavaScript
```

```
function drawHighlight (context, hitResult)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*hitResult*
Type: HitResult
A HitResult instance identifying highlighted data item.

## See Also

BarRenderer Members
BarRenderer Class
MindFusion.Charting Namespace

# BarRenderer.enumVisibleBars Method

Enumerates the bars visible in current data range.

**Namespace:** MindFusion.Charting
**File:** BarRenderer.js

## Syntax

**JavaScript**

```
function enumVisibleBars (context, process)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*process*
Type: ProcessBars
A ProcessBars callback.

## Remarks

The signature of the ProcessBars delegate is as follows:'function(seriesIndex, dataIndex, bounds)'.param {Number} seriesIndex An integer index of series in Series list.param {Number} dataIndex An integer index of data item in specified series.param {MindFusion.Drawing.Rect} bounds A Rect representing the boundaries of current bar.The method is not expected to return a value.

## See Also

BarRenderer Members
BarRenderer Class
MindFusion.Charting Namespace

# BarRenderer.hitTest Method

Renderer2D.HitTest override. Hit-tests the bar representations of data items.

**Namespace:** MindFusion.Charting
**File:** BarRenderer.js

## Syntax

```
JavaScript
function hitTest (context, location)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*location*
Type: Point
A Point specifying where to look for a bar.

### Return Value

Type: HitResult
A HitResult instance identifying the found data item.

## See Also

BarRenderer Members
BarRenderer Class
MindFusion.Charting Namespace

# BarRenderer.measureDataRange Method

Renderer2D.MeasureDataRange override. Measures the data range of rendered series and assigns it to the associated Axis objects if their MinValue and MaxValue are not set.

**Namespace:** MindFusion.Charting
**File:** BarRenderer.js

## Syntax

```
JavaScript
function measureDataRange (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

BarRenderer Members
BarRenderer Class
MindFusion.Charting Namespace

# BarRenderer Properties

## BarRenderer.barSpacingRatio Property

Gets or sets the ratio of empty space between adjacent bar groups to space occupied by bars.

**Namespace:** MindFusion.Charting
**File:** BarRenderer.js

### Syntax

```
JavaScript
get barSpacingRatio() {}
```

#### Property Value

Type: Number

### See Also

BarRenderer Members
BarRenderer Class
MindFusion.Charting Namespace

## BarRenderer.horizontalBars Property

Gets or sets a value identifying whether bars should be horizontal.

**Namespace:** MindFusion.Charting
**File:** BarRenderer.js

### Syntax

```
JavaScript
get horizontalBars() {}
```

#### Property Value

Type: Boolean

### See Also

BarRenderer Members
BarRenderer Class
MindFusion.Charting Namespace

## BarRenderer.innerLabelRotation Property

Gets the rotation angle of inner labels. If not specified, labels are rotated at predefiend angles depending on bars' orientation and layout.

**Namespace:** MindFusion.Charting
**File:** BarRenderer.js

## Syntax

```
JavaScript
get innerLabelRotation() {}
```

### Property Value

A Number.

## Example

The following code creates a new BarSeries with data and inner labels and a new BarRenderer using that series. It sets the rotation of inner labels to 30 degrees:

```
JavaScript
//the second argument specifies the inner labels
var data = new Charting.BarSeries(
 new Collections.List([13, 13, 14, 16, 23, 44]), new Collections.List(["Jan", "Feb", "Mar",
"Apr", "May", "Jun"]));
var barRenderer = new Charting.BarRenderer(new Collections.ObservableCollection([data]));
barRenderer.innerLabelRotation = 30;
```

## See Also

BarRenderer Members
BarRenderer Class
MindFusion.Charting Namespace

# BarRenderer.outerLabelRotation Property

Gets the rotation angle of outer labels. If not specified, labels are rotated at predefiend angles depending on bars' orientation and layout.

**Namespace:** MindFusion.Charting
**File:** BarRenderer.js

## Syntax

```
JavaScript
get outerLabelRotation() {}
```

### Property Value

A Number

## Example

The following code creates a new BarSeries with data and outer labels and a new BarRenderer using that series. It sets the rotation of the outer labels to 30 degrees:

```
JavaScript
//the chart has no inner labels that's why the second argument is null
var data = new Charting.BarSeries(
 new Collections.List([13, 13, 14, 16, 23, 44]), null, new Collections.List(["Jan", "Feb",
"Mar", "Apr", "May", "Jun"]));
```

```
var barRenderer = new Charting.BarRenderer(new Collections.ObservableCollection([data]));
barRenderer.outerLabelRotation = 30;
```

## See Also

BarRenderer Members
BarRenderer Class
MindFusion.Charting Namespace

# BarRenderer.sameIndexSpacingRatio Property

Gets or sets the ratio of empty space to occupied space in bar groups drawn for data items as same index in each series.

**Namespace:** MindFusion.Charting
**File:** BarRenderer.js

## Syntax

**JavaScript**

```
get sameIndexSpacingRatio() {}
```

### Property Value

Type: Number

## See Also

BarRenderer Members
BarRenderer Class
MindFusion.Charting Namespace

# BarRenderer.stackOuterLabels Property

Gets whether outer labels are displayed stacked on top of stacked bars, instead of showing them on the bar sides.

**Namespace:** MindFusion.Charting
**File:** BarRenderer.js

## Syntax

**JavaScript**

```
get stackOuterLabels() {}
```

### Property Value

A Boolean value. **true** if outer labels are drawn at the top of the bar stack; otherwise **false**.

## Example

The following code creates two BarSeries with data and outer labels and a new BarStackRenderer using these series. It specifies that the outer labels must be drawn on top of the bar stacks:

**JavaScript**

```
//the second argument specifies the inner labels, the third argument specifies the top(outer)
labels
var data1 = new Charting.BarSeries(
 new Collections.List([13, 13, 14, 16, 23, 44]), null, new Collections.List(["Jan", "Feb",
"Mar", "Apr",
"May", "Jun"]));
var data2 = new Charting.BarSeries(
 new Collections.List([22, 18, 25, 34, 51, 44, 8]), null, new Collections.List(["Jan", "Feb",
"Mar", "Apr",
"May", "Jun"]));
var barRenderer = new Charting.BarStackRenderer(new Collections.ObservableCollection([data1,
data2]));
barRenderer.stackOuterLabels = true;
```

## See Also

BarRenderer Members
BarRenderer Class
MindFusion.Charting Namespace

# BarRenderer3D Class

A SeriesRenderer that draws 3D bars in its containing plot.

## Syntax

| JavaScript |
| --- |
| `// class`<br>`BarRenderer3D.prototype = {}` |

## Inheritance Hierarchy

## See Also

BarRenderer3D Members
MindFusion.Charting Namespace

# BarRenderer3D Members

A SeriesRenderer that draws 3D bars in its containing plot.

The following tables list the members exposed by the BarRenderer3D type.

## Public Constructors

| | Name | Description |
| --- | --- | --- |
|  | BarRenderer3D | Initializes a new instance of the BarRenderer3D class. |

## Public Methods

| | Name | Description |
| --- | --- | --- |

| | | |
|---|---|---|
| | buildModels | Implements Renderer3D.BuildModels. Adds 3D bar models for the series data to specified scene. |

## See Also

BarRenderer3D Class
MindFusion.Charting Namespace

# BarRenderer3D Constructor

Initializes a new instance of the BarRenderer3D class.

**Namespace:** MindFusion.Charting
**File:** BarRenderer3D.js

## Syntax

**JavaScript**

```
function BarRenderer3D (series)
```

### Parameters

*series*
Type: ObservableCollection
A list of Series that should be rendered as 3D bars.

## See Also

BarRenderer3D Class
MindFusion.Charting Namespace

# BarRenderer3D Methods

# BarRenderer3D.buildModels Method

Implements Renderer3D.BuildModels. Adds 3D bar models for the series data to specified scene.

**Namespace:** MindFusion.Charting
**File:** BarRenderer3D.js

## Syntax

**JavaScript**

```
function buildModels (scene, context)
```

### Parameters

*scene*
Type: Scene3D
A Scene3D instance.

*context*
Type: RenderContext
A RenderContext instance.

## See Also

BarRenderer3D Members
BarRenderer3D Class
MindFusion.Charting Namespace

# BarSeries Class

Represents one-dimensional series defined by e list of data values and several lists of labels.

## Syntax

| JavaScript |
| --- |
| `// class`<br>`BarSeries.prototype = {}` |

## Inheritance Hierarchy

## See Also

BarSeries Members
MindFusion.Charting Namespace

# BarSeries Members

Represents one-dimensional series defined by e list of data values and several lists of labels.

The following tables list the members exposed by the BarSeries type.

## Public Constructors

| | Name | Description |
| --- | --- | --- |
| | BarSeries | Overloaded. |

## Public Properties

| | Name | Description |
| --- | --- | --- |
| | innerLabels | Gets or sets the inner labels. |
| | supportedLabels | SimpleSeries.SupportedLabels override. Returns a combination of InnerLabel, OuterLabel and XAxisLabel. |
| | topLabels | Gets or sets the top labels. |
| | xAxisLabels | Gets or sets the X axis labels. |

## Public Methods

| | Name | Description |
| --- | --- | --- |

| | | |
|---|---|---|
| | getLabel | SimpleSeries.GetLabel override. Returns an element of a labels list. |

## See Also

BarSeries Class
MindFusion.Charting Namespace

# BarSeries Constructor

## Overload List

| Name | Description |
|---|---|
| BarSeries | Initializes a new instance of the BarSeries class. |
| BarSeries | Initializes a new instance of the BarSeries class. |

## See Also

BarSeries Class
MindFusion.Charting Namespace

# BarSeries Constructor

Initializes a new instance of the BarSeries class.

**Namespace:** MindFusion.Charting
**File:** BarSeries.js

## Syntax

**JavaScript**

```
function BarSeries (values, innerLabels, [topLabels, [xAxisLabels]])
```

### Parameters

*values*
Type: List | Array
A list of data values.

*innerLabels*
Type: List | Array
A list of inner labels.

*topLabels*
Optional.
Type: List | Array
A list of top labels.

*xAxisLabels*
Optional.
Type: List | Array
A list of X axis labels.

## Example

The following code creates two BarSeries with data and outer labels and a new BarStackRenderer using these series.

**JavaScript**

```
//the second argument specifies the inner labels, the third argument specifies the top(outer)
labels
var data1 = new Charting.BarSeries(
 [13, 13, 14, 16, 23, 44], null, ["Jan", "Feb", "Mar", "Apr", "May", "Jun"]);
var data2 = new Charting.BarSeries([22, 18, 25, 34, 51, 44, 8], null, ["Jan", "Feb", "Mar",
"Apr", "May", "Jun"]);

var barRenderer = new Charting.BarStackRenderer(new Collections.ObservableCollection([data1,
data2]));
```

## See Also

BarSeries Constructor Overload List
BarSeries Class
MindFusion.Charting Namespace

# BarSeries Constructor

Initializes a new instance of the BarSeries class.

**Namespace:** MindFusion.Charting
**File:** BarSeries.js

## Syntax

**JavaScript**

```
function BarSeries (values, innerLabels, [topLabels, [xAxisLabels]])
```

### Parameters

*values*
Type: List
A list of data values.

*innerLabels*
Type: List
A list of inner labels.

*topLabels*
Optional.
Type: List
A list of top labels.

*xAxisLabels*
Optional.
Type: List
A list of X axis labels.

## See Also

BarSeries Constructor Overload List
BarSeries Class
MindFusion.Charting Namespace

# BarSeries Methods

## BarSeries.getLabel Method

SimpleSeries.GetLabel override. Returns an element of a labels list.

**Namespace:** MindFusion.Charting
**File:** BarSeries.js

### Syntax

```
JavaScript
function getLabel (index, kind)
```

#### Parameters

*index*
Type: Number
An integer value specifying the index of a data item.

*kind*
Type: LabelKinds
A member of the LabelKinds enumeration.

#### Return Value

Type: String
A string containing the item's label.

### See Also

BarSeries Members
BarSeries Class
MindFusion.Charting Namespace

# BarSeries Properties

## BarSeries.innerLabels Property

Gets or sets the inner labels.

**Namespace:** MindFusion.Charting
**File:** BarSeries.js

### Syntax

```
JavaScript
get innerLabels() {}
```

#### Property Value

Type: List

## See Also

BarSeries Members
BarSeries Class
MindFusion.Charting Namespace

# BarSeries.supportedLabels Property

SimpleSeries.SupportedLabels override. Returns a combination of InnerLabel, OuterLabel and XAxisLabel.

**Namespace:** MindFusion.Charting
**File:** BarSeries.js

## Syntax

**JavaScript**

```
get supportedLabels() {}
```

### Property Value

Type: LabelKinds

## See Also

BarSeries Members
BarSeries Class
MindFusion.Charting Namespace

# BarSeries.topLabels Property

Gets or sets the top labels.

**Namespace:** MindFusion.Charting
**File:** BarSeries.js

## Syntax

**JavaScript**

```
get topLabels() {}
```

### Property Value

Type: List

## See Also

BarSeries Members
BarSeries Class
MindFusion.Charting Namespace

## BarSeries.xAxisLabels Property

Gets or sets the X axis labels.

**Namespace:** MindFusion.Charting
**File:** BarSeries.js

### Syntax

| JavaScript |
| --- |
| `get xAxisLabels() {}` |

#### Property Value

Type: List

### See Also

BarSeries Members
BarSeries Class
MindFusion.Charting Namespace

# BarStackRenderer Class

A SeriesRenderer that draws stacked bars in its containing plot.

### Syntax

| JavaScript |
| --- |
| `// class`<br>`BarStackRenderer.prototype = {}` |

### Inheritance Hierarchy

### See Also

BarStackRenderer Members
MindFusion.Charting Namespace

## BarStackRenderer Members

A SeriesRenderer that draws stacked bars in its containing plot.

The following tables list the members exposed by the BarStackRenderer type.

### Public Constructors

| | Name | Description |
| --- | --- | --- |
|  | BarStackRenderer | Initializes a new instance of the BarStackRenderer class. |

### Public Properties

| | Name | Description |
|---|---|---|
|  | barSpacingRatio | Gets or sets the ratio of empty space between adjacent bar groups to space occupied by bars. |
|  | horizontalBars | Gets or sets a value identifying whether bars should be horizontal. |

## Public Methods

| | Name | Description |
|---|---|---|
|  | adjustDataRange | SeriesRenderer.AdjustDataRange override. Adjusts the data range of the associated Axis objects after initial measure pass. |
|  | draw | SeriesRenderer.Draw override. Draws the series data in specified RenderContext. |
|  | drawHighlight | Renderer2D.DrawHighlight override. Draws highlighted bar in specified RenderContext. |
|  | enumVisibleStackBars | Enumerates the bars visible in current data range. |
|  | hitTest | StackRenderer.HitTest override. Hit-tests the bar representations of data items. |
|  | measureDataRange | StackRenderer.MeasureDataRange override. Measures the data range of rendered series and assigns it to the associated Axis objects if their MinValue and MaxValue are not set. |

## See Also

BarStackRenderer Class
MindFusion.Charting Namespace

# BarStackRenderer Constructor

Initializes a new instance of the BarStackRenderer class.

**Namespace:** MindFusion.Charting
**File:** BarStackRenderer.js

## Syntax

**JavaScript**

```
function BarStackRenderer (series)
```

### Parameters

*series*
Type: ObservableCollection
A list of Series that should be rendered as stacked bars.

## See Also

# BarStackRenderer Methods

## BarStackRenderer.adjustDataRange Method

SeriesRenderer.AdjustDataRange override. Adjusts the data range of the associated Axis objects after initial measure pass.

**Namespace:** MindFusion.Charting
**File:** BarStackRenderer.js

### Syntax

```
JavaScript
function adjustDataRange (context)
```

#### Parameters

*context*
A RenderContext instance.

### See Also

BarStackRenderer Members
BarStackRenderer Class
MindFusion.Charting Namespace

## BarStackRenderer.draw Method

SeriesRenderer.Draw override. Draws the series data in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** BarStackRenderer.js

### Syntax

```
JavaScript
function draw (context)
```

#### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### See Also

BarStackRenderer Members
BarStackRenderer Class
MindFusion.Charting Namespace

# BarStackRenderer.drawHighlight Method

Renderer2D.DrawHighlight override. Draws highlighted bar in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** BarStackRenderer.js

## Syntax

**JavaScript**

```
function drawHighlight (context, hitResult)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*hitResult*
Type: HitResult
A HitResult instance identifying highlighted data item.

## See Also

BarStackRenderer Members
BarStackRenderer Class
MindFusion.Charting Namespace

# BarStackRenderer.enumVisibleStackBars Method

Enumerates the bars visible in current data range.

**Namespace:** MindFusion.Charting
**File:** BarStackRenderer.js

## Syntax

**JavaScript**

```
function enumVisibleStackBars (context, process)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*process*
Type: ProcessStackBars
A ProcessStackBars callback.

## Remarks

The signature of the ProcessStackBars delegate is as follows:'function(seriesIndex, dataIndex, bounds)'.param {Number} seriesIndex An integer index of series in Series list.param {Number} dataIndex An integer index of data item in specified series.param {MindFusion.Drawing.Rect} bounds A RectangleF representing the boundaries of current bar.The method is not expected to return a value.

## See Also

BarStackRenderer Members
BarStackRenderer Class
MindFusion.Charting Namespace

# BarStackRenderer.hitTest Method

StackRenderer.HitTest override. Hit-tests the bar representations of data items.

**Namespace:** MindFusion.Charting
**File:** BarStackRenderer.js

## Syntax

**JavaScript**

```
function hitTest (context, location)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*location*
Type: Point
A Point specifying where to look for a bar.

### Return Value

Type: HitResult
A HitResult instance identifying the found data item.

## See Also

BarStackRenderer Members
BarStackRenderer Class
MindFusion.Charting Namespace

# BarStackRenderer.measureDataRange Method

StackRenderer.MeasureDataRange override. Measures the data range of rendered series and assigns it to the associated Axis objects if their MinValue and MaxValue are not set.

**Namespace:** MindFusion.Charting
**File:** BarStackRenderer.js

## Syntax

**JavaScript**

```
function measureDataRange (context)
```

### Parameters

*context*

Type: RenderContext
A RenderContext instance.

## See Also

BarStackRenderer Members
BarStackRenderer Class
MindFusion.Charting Namespace

# BarStackRenderer Properties

## BarStackRenderer.barSpacingRatio Property

Gets or sets the ratio of empty space between adjacent bar groups to space occupied by bars.

**Namespace:** MindFusion.Charting
**File:** BarStackRenderer.js

## Syntax

```
JavaScript
get barSpacingRatio() {}
```

### Property Value

Type: Number

## See Also

BarStackRenderer Members
BarStackRenderer Class
MindFusion.Charting Namespace

## BarStackRenderer.horizontalBars Property

Gets or sets a value identifying whether bars should be horizontal.

**Namespace:** MindFusion.Charting
**File:** BarStackRenderer.js

## Syntax

```
JavaScript
get horizontalBars() {}
```

### Property Value

Type: Boolean

## See Also

BarStackRenderer Members
BarStackRenderer Class
MindFusion.Charting Namespace

# BarStackRenderer3D Class

A SeriesRenderer that draws stacked 3D bars in its containing plot.

## Syntax

**JavaScript**

```
// class
BarStackRenderer3D.prototype = {}
```

## Inheritance Hierarchy

## See Also

BarStackRenderer3D Members
MindFusion.Charting Namespace

# BarStackRenderer3D Members

A SeriesRenderer that draws stacked 3D bars in its containing plot.

The following tables list the members exposed by the BarStackRenderer3D type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | BarStackRenderer3D | Initializes a new instance of the BarStackRenderer3D class. |

## Public Methods

| | Name | Description |
|---|---|---|
|  | buildModels | Implements Renderer3D.BuildModels. Adds 3D bar models for the series data to specified scene. |

## See Also

BarStackRenderer3D Class
MindFusion.Charting Namespace

# BarStackRenderer3D Constructor

Initializes a new instance of the BarStackRenderer3D class.

**Namespace:** MindFusion.Charting
**File:** BarStackRenderer3D.js

## Syntax

```
JavaScript
function BarStackRenderer3D (series)
```

### Parameters

*series*
Type: ObservableCollection
A list of Series that should be rendered as stacked bars.

## See Also

BarStackRenderer3D Class
MindFusion.Charting Namespace

# BarStackRenderer3D Methods

## BarStackRenderer3D.buildModels Method

Implements Renderer3D.BuildModels. Adds 3D bar models for the series data to specified scene.

**Namespace:** MindFusion.Charting
**File:** BarStackRenderer3D.js

## Syntax

```
JavaScript
function buildModels (scene, context)
```

### Parameters

*scene*
Type: Scene3D
A Scene3D instance.

*context*
Type: RenderContext
A RenderContext instance.

## See Also

BarStackRenderer3D Members
BarStackRenderer3D Class
MindFusion.Charting Namespace

# BubbleLabelAlignment Enumeration

Specifies the alignment of a label relative to its associated bubble.

**Namespace:** MindFusion.Charting
**File:** BubbleLabelAlignment.js

## Syntax

| JavaScript |
| --- |
| `// enum`<br>`BubbleLabelAlignment = {}` |

## Members

| | Member name | Description |
| --- | --- | --- |
| | **Above** | Indicates that the label is positioned above the bubble. |
| | **Below** | Indicates that the label is positioned below the bubble. |
| | **Center** | Indicates that the label is centered inside the bubble. |
| | **Left** | Indicates that the label is positioned to the left of the bubble. |
| | **Right** | Indicates that the label is positioned to the right of the bubble. |

## See Also

MindFusion.Charting Namespace

# BubbleRenderer Class

A SeriesRenderer that draws bubbles representing data items in its containing plot.

## Syntax

| JavaScript |
| --- |
| `// class`<br>`BubbleRenderer.prototype = {}` |

## Inheritance Hierarchy

## See Also

BubbleRenderer Members
MindFusion.Charting Namespace

# BubbleRenderer Members

A SeriesRenderer that draws bubbles representing data items in its containing plot.

The following tables list the members exposed by the BubbleRenderer type.

## Public Constructors

| | Name | Description |
| --- | --- | --- |

| | | |
|---|---|---|
| | BubbleRenderer | Initializes a new instance of the BubbleRenderer class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | labelAlignment | Gets or sets the label alignment. |

## Public Methods

| | Name | Description |
|---|---|---|
| | draw | SeriesRenderer.Draw override. Draws bubbles representing data items in specified RenderContext. |
| | drawHighlight | Renderer2D.DrawHighlight override. Draws highlighted bubble in specified RenderContext. |
| | hitTest | Renderer2D.HitTest override. Hit-tests the bubble representations of data items. |

## See Also

BubbleRenderer Class
MindFusion.Charting Namespace

# BubbleRenderer Constructor

Initializes a new instance of the BubbleRenderer class.

**Namespace:** MindFusion.Charting
**File:** BubbleRenderer.js

## Syntax

**JavaScript**

```
function BubbleRenderer (series)
```

### Parameters

*series*
Type: ObservableCollection
A list of Series that should be rendered as bubbles.

## See Also

BubbleRenderer Class
MindFusion.Charting Namespace

# BubbleRenderer Methods

# BubbleRenderer.draw Method

SeriesRenderer.Draw override. Draws bubbles representing data items in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** BubbleRenderer.js

## Syntax

**JavaScript**

```
function draw (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

BubbleRenderer Members
BubbleRenderer Class
MindFusion.Charting Namespace

# BubbleRenderer.drawHighlight Method

Renderer2D.DrawHighlight override. Draws highlighted bubble in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** BubbleRenderer.js

## Syntax

**JavaScript**

```
function drawHighlight (context, hitResult)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*hitResult*
Type: HitResult
A HitResult instance identifying highlighted data item.

## See Also

BubbleRenderer Members
BubbleRenderer Class
MindFusion.Charting Namespace

# BubbleRenderer.hitTest Method

Renderer2D.HitTest override. Hit-tests the bubble representations of data items.

**Namespace:** MindFusion.Charting
**File:** BubbleRenderer.js

## Syntax

```
JavaScript
function hitTest (context, location)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*location*
Type: Point
A Point specifying where to look for a bubble.

### Return Value

Type: HitResult
A HitResult instance identifying the found data item.

## See Also

BubbleRenderer Members
BubbleRenderer Class
MindFusion.Charting Namespace

# BubbleRenderer Properties

# BubbleRenderer.labelAlignment Property

Gets or sets the label alignment.

**Namespace:** MindFusion.Charting
**File:** BubbleRenderer.js

## Syntax

```
JavaScript
get labelAlignment() {}
```

### Property Value

Type: BubbleLabelAlignment

## See Also

BubbleRenderer Members
BubbleRenderer Class
MindFusion.Charting Namespace

# CandlestickRenderer Class

A SeriesRenderer that draws candlesticks in its containing plot.

## Syntax

**JavaScript**

```
// class
CandlestickRenderer.prototype = {}
```

## Inheritance Hierarchy

## See Also

CandlestickRenderer Members
MindFusion.Charting Namespace

# CandlestickRenderer Members

A SeriesRenderer that draws candlesticks in its containing plot.

The following tables list the members exposed by the CandlestickRenderer type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | CandlestickRenderer | Initializes a new instance of the CandlestickRenderer class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | candlestickWidth | Gets or sets the width of the candlesticks. |
| | closeDimension | Gets or sets the index of the dimension, containing close values. |
| | highDimension | Gets or sets the index of the dimension, containing high values. |
| | lowDimension | Gets or sets the index of the dimension, containing low values. |
| | openDimension | Gets or sets the index of the dimension, containing open values. |

## Public Methods

| | Name | Description |
|---|---|---|
| | draw | SeriesRenderer.Draw override. Draws the series data in specified RenderContext. |

| | | |
|---|---|---|
| | drawHighlight | Renderer2D.DrawHighlight override. Draws highlighted candlestick in specified RenderContext. |
| | effectiveFill | Gets the brush that should be used to fill the representation of specified data item. |
| | enumVisibleCandlesticks | Enumerates the candlesticks visible in current data range. |
| | hitTest | Renderer2D.HitTest override. Hit-tests the bar representations of data items. |
| | measureDataRange | Renderer2D.MeasureDataRange override. Measures the data range of rendered series and assigns it to the associated Axis objects if their MinValue and MaxValue are not set. |

## See Also

CandlestickRenderer Class
MindFusion.Charting Namespace

# CandlestickRenderer Constructor

Initializes a new instance of the CandlestickRenderer class.

**Namespace:** MindFusion.Charting
**File:** CandlestickRenderer.js

## Syntax

**JavaScript**

```
function CandlestickRenderer (series)
```

### Parameters

*series*
Type: ObservableCollection
A list of Series that should be rendered by this object.

## See Also

CandlestickRenderer Class
MindFusion.Charting Namespace

# CandlestickRenderer Methods

# CandlestickRenderer.draw Method

SeriesRenderer.Draw override. Draws the series data in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** CandlestickRenderer.js

## Syntax

**JavaScript**

```
function draw (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

CandlestickRenderer Members
CandlestickRenderer Class
MindFusion.Charting Namespace

# CandlestickRenderer.drawHighlight Method

Renderer2D.DrawHighlight override. Draws highlighted candlestick in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** CandlestickRenderer.js

## Syntax

**JavaScript**

```
function drawHighlight (context, hitResult)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*hitResult*
Type: HitResult
A HitResult instance identifying highlighted data item.

## See Also

CandlestickRenderer Members
CandlestickRenderer Class
MindFusion.Charting Namespace

# CandlestickRenderer.effectiveFill Method

Gets the brush that should be used to fill the representation of specified data item.

**Namespace:** MindFusion.Charting
**File:** CandlestickRenderer.js

## Syntax

**JavaScript**

```
function effectiveFill (seriesIndex, dataIndex, context)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series within the list of series rendered by this SeriesRenderer.

*dataIndex*
Type: Number
An integer index of the data item in specified series.

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: Brush
A Brush instance.

## See Also

CandlestickRenderer Members
CandlestickRenderer Class
MindFusion.Charting Namespace

# CandlestickRenderer.enumVisibleCandlesticks Method

Enumerates the candlesticks visible in current data range.

**Namespace:** MindFusion.Charting
**File:** CandlestickRenderer.js

## Syntax

**JavaScript**

```
function enumVisibleCandlesticks (context, process)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*process*
Type: ProcessCandlesticks
A ProcessBars callback.

## Remarks

The signature of the ProcessCandlesticks delegate is as follows:'function(seriesIndex, dataIndex, points, bounds)'.param {Number} seriesIndex An integer index of series in Series list.param {Number} dataIndex An integer index of data item in specified series.param {MindFusion.Drawing.Point[]} points An array of four points containing wick ends' coordinates.param {MindFusion.Drawing.Rect} bounds A RectangleF representing the boundaries of current candlestick.The method is not expected to return a value.

## See Also

CandlestickRenderer Members
CandlestickRenderer Class
MindFusion.Charting Namespace

# CandlestickRenderer.hitTest Method

Renderer2D.HitTest override. Hit-tests the bar representations of data items.

**Namespace:** MindFusion.Charting
**File:** CandlestickRenderer.js

## Syntax

**JavaScript**

```
function hitTest (context, location)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*location*
Type: Point
A Point specifying where to look for a candlestick.

### Return Value

Type: HitResult
A HitResult instance identifying the found data item.

## See Also

CandlestickRenderer Members
CandlestickRenderer Class
MindFusion.Charting Namespace

# CandlestickRenderer.measureDataRange Method

Renderer2D.MeasureDataRange override. Measures the data range of rendered series and assigns it to the associated Axis objects if their MinValue and MaxValue are not set.

**Namespace:** MindFusion.Charting
**File:** CandlestickRenderer.js

## Syntax

**JavaScript**

```
function measureDataRange (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

CandlestickRenderer Members
CandlestickRenderer Class
MindFusion.Charting Namespace

## CandlestickRenderer Properties

## CandlestickRenderer.candlestickWidth Property

Gets or sets the width of the candlesticks.

**Namespace:** MindFusion.Charting
**File:** CandlestickRenderer.js

### Syntax

```
JavaScript
get candlestickWidth() {}
```

#### Property Value

Type: Number

### See Also

CandlestickRenderer Members
CandlestickRenderer Class
MindFusion.Charting Namespace

## CandlestickRenderer.closeDimension Property

Gets or sets the index of the dimension, containing close values.

**Namespace:** MindFusion.Charting
**File:** CandlestickRenderer.js

### Syntax

```
JavaScript
get closeDimension() {}
```

#### Property Value

Type: Number

### See Also

CandlestickRenderer Members
CandlestickRenderer Class
MindFusion.Charting Namespace

## CandlestickRenderer.highDimension Property

Gets or sets the index of the dimension, containing high values.

**Namespace:** MindFusion.Charting
**File:** CandlestickRenderer.js

## Syntax

```
JavaScript
get highDimension() {}
```

### Property Value

Type: Number

## See Also

CandlestickRenderer Members
CandlestickRenderer Class
MindFusion.Charting Namespace

# CandlestickRenderer.lowDimension Property

Gets or sets the index of the dimension, containing low values.

**Namespace:** MindFusion.Charting
**File:** CandlestickRenderer.js

## Syntax

```
JavaScript
get lowDimension() {}
```

### Property Value

Type: Number

## See Also

CandlestickRenderer Members
CandlestickRenderer Class
MindFusion.Charting Namespace

# CandlestickRenderer.openDimension Property

Gets or sets the index of the dimension, containing open values.

**Namespace:** MindFusion.Charting
**File:** CandlestickRenderer.js

## Syntax

```
JavaScript
get openDimension() {}
```

### Property Value

Type: Number

## See Also

CandlestickRenderer Members
CandlestickRenderer Class
MindFusion.Charting Namespace

# CandlestickSeriesStyle Class

Defines appearance attributes for candlestick charts.

## Syntax

**JavaScript**

```
// class
CandlestickSeriesStyle.prototype = {}
```

## Inheritance Hierarchy

## See Also

CandlestickSeriesStyle Members
MindFusion.Charting Namespace

## CandlestickSeriesStyle Members

Defines appearance attributes for candlestick charts.

The following tables list the members exposed by the CandlestickSeriesStyle type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | CandlestickSeriesStyle | Initializes a new instance of CandlestickSeriesStyle. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | fallingBrush | Gets or sets a Brush used to fill candlesticks, which open value is greater than its close value. |
|  | risingBrush | Gets or sets a Brush used to fill candlesticks, which open value is smaller or equal to its close value. |

## Public Methods

| | Name | Description |
|---|---|---|
|  | fill | Implements SeriesStyle.Fill. Returns a Brush instance, depending on the series data. |

| | | |
|---|---|---|
| | stroke | Implements SeriesStyle.Stroke. |
| | strokeDashStyle | Implements SeriesStyle.StrokeDashStyle. |
| | strokeThickness | Implements SeriesStyle.StrokeThickness. |

## See Also

CandlestickSeriesStyle Class
MindFusion.Charting Namespace

# CandlestickSeriesStyle Constructor

Initializes a new instance of CandlestickSeriesStyle.

**Namespace:** MindFusion.Charting
**File:** CandlestickSeriesStyle.js

## Syntax

**JavaScript**

```
function CandlestickSeriesStyle (risingBrush, fallingBrush, stroke, strokeThickness,
strokeDashStyle, renderer)
```

### Parameters

*risingBrush*
Type: void

*fallingBrush*
Type: void

*stroke*
Type: void

*strokeThickness*
Type: void

*strokeDashStyle*
Type: void

*renderer*
Type: void

## See Also

CandlestickSeriesStyle Class
MindFusion.Charting Namespace

# CandlestickSeriesStyle Methods

# CandlestickSeriesStyle.fill Method

Implements SeriesStyle.Fill. Returns a Brush instance, depending on the series data.

**Namespace:** MindFusion.Charting
**File:** CandlestickSeriesStyle.js

## Syntax

```
JavaScript
function fill (seriesIndex, dataIndex)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series.

*dataIndex*
Type: Number
An integer index of a data item in specified series.

### Return Value

Type: Brush
A Brush instance.

## See Also

CandlestickSeriesStyle Members
CandlestickSeriesStyle Class
MindFusion.Charting Namespace

# CandlestickSeriesStyle.stroke Method

Implements SeriesStyle.Stroke.

**Namespace:** MindFusion.Charting
**File:** CandlestickSeriesStyle.js

## Syntax

```
JavaScript
function stroke (seriesIndex, dataIndex)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series.

*dataIndex*
Type: Number
An integer index of a data item in specified series.

### Return Value

Type: Brush
A Brush instance.

## See Also

CandlestickSeriesStyle Members
CandlestickSeriesStyle Class

# CandlestickSeriesStyle.strokeDashStyle Method

Implements SeriesStyle.StrokeDashStyle.

**Namespace:** MindFusion.Charting
**File:** CandlestickSeriesStyle.js

## Syntax

**JavaScript**

```
function strokeDashStyle (seriesIndex, dataIndex)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series.

*dataIndex*
Type: Number
An integer index of a data item in specified series.

### Return Value

Type: DashStyle
A nullable DashStyle value specifying stroke dash style.

## See Also

CandlestickSeriesStyle Members
CandlestickSeriesStyle Class
MindFusion.Charting Namespace

# CandlestickSeriesStyle.strokeThickness Method

Implements SeriesStyle.StrokeThickness.

**Namespace:** MindFusion.Charting
**File:** CandlestickSeriesStyle.js

## Syntax

**JavaScript**

```
function strokeThickness (seriesIndex, dataIndex)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series.

*dataIndex*
Type: Number
An integer index of a data item in specified series.

### Return Value

Type: Number
A nullable number value specifying stroke thickness.

## See Also

CandlestickSeriesStyle Members
CandlestickSeriesStyle Class
MindFusion.Charting Namespace

# CandlestickSeriesStyle Properties

## CandlestickSeriesStyle.fallingBrush Property

Gets or sets a Brush used to fill candlesticks, which open value is greater than its close value.

**Namespace:** MindFusion.Charting
**File:** CandlestickSeriesStyle.js

## Syntax

**JavaScript**

```
get fallingBrush() {}
```

### Property Value

Type: Brush

## See Also

CandlestickSeriesStyle Members
CandlestickSeriesStyle Class
MindFusion.Charting Namespace

## CandlestickSeriesStyle.risingBrush Property

Gets or sets a Brush used to fill candlesticks, which open value is smaller or equal to its close value.

**Namespace:** MindFusion.Charting
**File:** CandlestickSeriesStyle.js

## Syntax

**JavaScript**

```
get risingBrush() {}
```

### Property Value

Type: Brush

## See Also

CandlestickSeriesStyle Members
CandlestickSeriesStyle Class
MindFusion.Charting Namespace

# ChartEvent Class

Represents a timed event, useful for showing event sequences in tower charts.

## Syntax

| JavaScript |
| --- |
| `// class`<br>`ChartEvent.prototype = {}` |

## Inheritance Hierarchy

**MindFusion.Charting.ChartEvent**

## See Also

ChartEvent Members
MindFusion.Charting Namespace

# ChartEvent Members

Represents a timed event, useful for showing event sequences in tower charts.

The following tables list the members exposed by the ChartEvent type.

## Public Constructors

| | Name | Description |
| --- | --- | --- |
|  | ChartEvent | Initializes a new instance of the ChartEvent class. |

## See Also

ChartEvent Class
MindFusion.Charting Namespace

# ChartEvent Constructor

Initializes a new instance of the ChartEvent class.

**Namespace:** MindFusion.Charting
**File:** EventSeries.js

## Syntax

| JavaScript |
| --- |
| `function ChartEvent ()` |

## See Also

ChartEvent Class
MindFusion.Charting Namespace

# CurveAreaRenderer Class

A SeriesRenderer that draws each series as an area with curved boundaries in its containing plot.

## Syntax

**JavaScript**

```
// class
CurveAreaRenderer.prototype = {}
```

## Inheritance Hierarchy

## See Also

CurveAreaRenderer Members
MindFusion.Charting Namespace

# CurveAreaRenderer Members

A SeriesRenderer that draws each series as an area with curved boundaries in its containing plot.

The following tables list the members exposed by the CurveAreaRenderer type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | CurveAreaRenderer | Initializes a new instance of the CurveAreaRenderer class. |

## Public Methods

| | Name | Description |
|---|---|---|
|  | draw | AreaRenderer.Draw override. Draws the series data in specified RenderContext. |

## See Also

CurveAreaRenderer Class
MindFusion.Charting Namespace

# CurveAreaRenderer Constructor

Initializes a new instance of the CurveAreaRenderer class.

**Namespace:** MindFusion.Charting
**File:** CurveAreaRenderer.js

## Syntax

**JavaScript**

```
function CurveAreaRenderer (series)
```

### Parameters

*series*
Type: ObservableCollection
A list of Series that should be rendered as areas.

## See Also

CurveAreaRenderer Class
MindFusion.Charting Namespace

# CurveAreaRenderer Methods

## CurveAreaRenderer.draw Method

AreaRenderer.Draw override. Draws the series data in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** CurveAreaRenderer.js

## Syntax

**JavaScript**

```
function draw (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

CurveAreaRenderer Members
CurveAreaRenderer Class
MindFusion.Charting Namespace

# CurveAreaStackRenderer Class

A SeriesRenderer that draws stacked areas in its containing plot.

## Syntax

**JavaScript**

```
// class
CurveAreaStackRenderer.prototype = {}
```

## Inheritance Hierarchy

## See Also

CurveAreaStackRenderer Members
MindFusion.Charting Namespace

# CurveAreaStackRenderer Members

A SeriesRenderer that draws stacked areas in its containing plot.

The following tables list the members exposed by the CurveAreaStackRenderer type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | CurveAreaStackRenderer | Initializes a new instance of the CurveAreaStackRenderer class. |

## Public Methods

| | Name | Description |
|---|---|---|
|  | draw | SeriesRenderer.Draw override. Draws the series data in specified RenderContext. |

## See Also

CurveAreaStackRenderer Class
MindFusion.Charting Namespace

# CurveAreaStackRenderer Constructor

Initializes a new instance of the CurveAreaStackRenderer class.

**Namespace:** MindFusion.Charting
**File:** CurveAreaStackRenderer.js

## Syntax

**JavaScript**

```
function CurveAreaStackRenderer (series)
```

### Parameters

*series*
Type: ObservableCollection
A list of Series that should be rendered as stacked areas.

## See Also

CurveAreaStackRenderer Class
MindFusion.Charting Namespace

## CurveAreaStackRenderer Methods

## CurveAreaStackRenderer.draw Method

SeriesRenderer.Draw override. Draws the series data in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** CurveAreaStackRenderer.js

### Syntax

```
JavaScript
function draw (context)
```

#### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### See Also

CurveAreaStackRenderer Members
CurveAreaStackRenderer Class
MindFusion.Charting Namespace

# CurveRenderer Class

A SeriesRenderer that draws series as a cardinal spline.

### Syntax

```
JavaScript
// class
CurveRenderer.prototype = {}
```

### Inheritance Hierarchy

### See Also

CurveRenderer Members
MindFusion.Charting Namespace

## CurveRenderer Members

A SeriesRenderer that draws series as a cardinal spline.

The following tables list the members exposed by the CurveRenderer type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | CurveRenderer | Initializes a new instance of the CurveRenderer class. |

## Public Methods

| | Name | Description |
|---|---|---|
| | draw | LineRenderer.Draw override. Draws the series data in specified RenderContext. |

## See Also

CurveRenderer Class
MindFusion.Charting Namespace

# CurveRenderer Constructor

Initializes a new instance of the CurveRenderer class.

**Namespace:** MindFusion.Charting
**File:** CurveRenderer.js

## Syntax

**JavaScript**

```
function CurveRenderer (series)
```

### Parameters

*series*
Type: ObservableCollection
A list of Series that should be rendered as a cardinal spline.

## See Also

CurveRenderer Class
MindFusion.Charting Namespace

# CurveRenderer Methods

# CurveRenderer.draw Method

LineRenderer.Draw override. Draws the series data in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** CurveRenderer.js

## Syntax

**JavaScript**

```
function draw (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

CurveRenderer Members
CurveRenderer Class
MindFusion.Charting Namespace

# DataBoundSeries Class

Represents a series whose data items are retrieved from a data source.

## Syntax

**JavaScript**

```
// class
DataBoundSeries.prototype = {}
```

## Inheritance Hierarchy

## See Also

DataBoundSeries Members
MindFusion.Charting Namespace

## DataBoundSeries Members

Represents a series whose data items are retrieved from a data source.

The following tables list the members exposed by the DataBoundSeries type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | DataBoundSeries | Overloaded. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | dataChanged | Raised when the values in this series change. |
|  | dataSource | Gets or sets the data source this series is bound to. |
|  | dimensions | Implements Series.Dimensions. Returns the number of X/Y/ZDataField properties set. |

| | | |
|---|---|---|
| | innerLabelsDataField | Gets or sets the name of inner labels data field in the data source. |
| | outerLabelsDataField | Gets or sets the name of inner labels data field in the data source. |
| | size | Implements Series.Size. Gets the number of data items in the data source. |
| | supportedLabels | Implements Series.SupportedLabels. Result depends on which *LabelsDataField properties are set. |
| | title | Implements Series.Title. Gets or sets the title of this series. |
| | toolTipsDataField | Gets or sets the name of tooltips data field in the data source. |
| | xAxisLabelsDataField | Gets or sets the name of X axis labels data field in the data source. |
| | xDataField | Gets or sets the name of X data field in the data source. |
| | yAxisLabelsDataField | Gets or sets the name of Y axis labels data field in the data source. |
| | yDataField | Gets or sets the name of Y data field in the data source. |
| | zAxisLabelsDataField | Gets or sets the name of Z axis labels data field in the data source. |
| | zDataField | Gets or sets the name of Z data field in the data source. |

## Public Methods

| | Name | Description |
|---|---|---|
| | getLabel | Implements Series.GetLabel. Returns a label for the specified data item in the data source. |
| | getValue | Implements Series.GetValue. Returns a value for the specified data item in the data source. |
| | isEmphasized | Implements Series.IsEmphasized. DataBoundSeries always returns false. |
| | isSorted | Implements Series.IsSorted. Gets a value indicating whether the series values increase monotonously in specified dimension. |
| | onDataChanged | Raises the DataChanged event. |

## See Also

DataBoundSeries Class
MindFusion.Charting Namespace

# DataBoundSeries Constructor

## Overload List

| Name | Description |
| --- | --- |
| DataBoundSeries | Initializes a new instance of the DataBoundSeries class. |
| DataBoundSeries | Initializes a new instance of the **DataBoundSeries** class. |

## See Also

DataBoundSeries Class
MindFusion.Charting Namespace

# DataBoundSeries Constructor

Initializes a new instance of the DataBoundSeries class.

**Namespace:** MindFusion.Charting
**File:** DataBoundSeries.js

## Syntax

**JavaScript**

```
function DataBoundSeries (dataSource)
```

### Parameters

*dataSource*
Type: any | Array | IEnumerable
A reference to the data source object.

## Remarks

In most cases you would set the dataSource property of a chart control. DataBoundSeries are created automatically according to the contents of the provided **dataSource**.

## Example

The following code creates a **DataBoundSeries** instance from an array:

**JavaScript**

```
var data = [13, 13, 14, 16, 23, 44];
var series = new Charting.DataBoundSeries(data);
```

## See Also

DataBoundSeries Constructor Overload List
DataBoundSeries Class
MindFusion.Charting Namespace

# DataBoundSeries Constructor

Initializes a new instance of the DataBoundSeries class.

**Namespace:** MindFusion.Charting
**File:** DataBoundSeries.js

## Syntax

```
JavaScript
function DataBoundSeries (dataSource)
```

### Parameters

*dataSource*
Type: any, Array, IEnumerable
A reference to the data source object.

## Remarks

In most cases you would set the dataSource property of a chart control. DataBoundSeries are created automatically according to the contents of the provided **dataSource**.

## Example

The following code creates a **DataBoundSeries** instance from a List:

```
JavaScript
var data = new Collections.List([22, 18, 25, 34, 51, 44, 8]);
var series = new Charting.DataBoundSeries(data);
```

## See Also

DataBoundSeries Constructor Overload List
DataBoundSeries Class
MindFusion.Charting Namespace

# DataBoundSeries Methods

# DataBoundSeries.getLabel Method

Implements Series.GetLabel. Returns a label for the specified data item in the data source.

**Namespace:** MindFusion.Charting
**File:** DataBoundSeries.js

## Syntax

```
JavaScript
function getLabel (index, kind)
```

### Parameters

*index*
Type: Number
An integer value specifying the index of a data item.

*kind*
Type: LabelKinds
A member of the LabelKinds enumeration.

### Return Value

Type: String
A string containing the item's label.

## See Also

DataBoundSeries Members
DataBoundSeries Class
MindFusion.Charting Namespace

# DataBoundSeries.getValue Method

Implements Series.GetValue. Returns a value for the specified data item in the data source.

**Namespace:** MindFusion.Charting
**File:** DataBoundSeries.js

## Syntax

**JavaScript**

```
function getValue (index, dimension)
```

### Parameters

*index*
Type: Number
An integer value specifying the index of a data item.

*dimension*
Type: Number
An integer value specifying the dimension whose coordinate to return.

### Return Value

Type: Number
A double-precision number representing the value of a data item in the data source.

## See Also

DataBoundSeries Members
DataBoundSeries Class
MindFusion.Charting Namespace

# DataBoundSeries.isEmphasized Method

Implements Series.IsEmphasized. DataBoundSeries always returns false.

**Namespace:** MindFusion.Charting
**File:** DataBoundSeries.js

## Syntax

**JavaScript**

```
function isEmphasized (index)
```

### Parameters

*index*
Type: Number
An integer value specifying the index of a data item.

### Return Value

Type: Boolean
true to emphasize the specified data item, or false otherwise.

## See Also

DataBoundSeries Members
DataBoundSeries Class
MindFusion.Charting Namespace

# DataBoundSeries.isSorted Method

Implements Series.IsSorted. Gets a value indicating whether the series values increase monotonously in specified dimension.

**Namespace:** MindFusion.Charting
**File:** DataBoundSeries.js

## Syntax

**JavaScript**

```
function isSorted (dimension)
```

### Parameters

*dimension*
Type: Number
An integer value identifying the dimension.

### Return Value

Type: Boolean
true if the values are sorted in specified dimension, or false otherwise.

## See Also

DataBoundSeries Members
DataBoundSeries Class
MindFusion.Charting Namespace

# DataBoundSeries.onDataChanged Method

Raises the DataChanged event.

**Namespace:** MindFusion.Charting
**File:** DataBoundSeries.js

## Syntax

```
JavaScript
function onDataChanged ()
```

## See Also

DataBoundSeries Members
DataBoundSeries Class
MindFusion.Charting Namespace

# DataBoundSeries Properties

# DataBoundSeries.dataChanged Property

Raised when the values in this series change.

**Namespace:** MindFusion.Charting
**File:** DataBoundSeries.js

## Syntax

```
JavaScript
get dataChanged() {}
```

### Property Value

Type: EventDispatcher

## See Also

DataBoundSeries Members
DataBoundSeries Class
MindFusion.Charting Namespace

# DataBoundSeries.dataSource Property

Gets or sets the data source this series is bound to.

**Namespace:** MindFusion.Charting
**File:** DataBoundSeries.js

## Syntax

```
JavaScript
get dataSource() {}
```

### Property Value

Type: any

## See Also

DataBoundSeries Members
DataBoundSeries Class

## DataBoundSeries.dimensions Property

Implements Series.Dimensions. Returns the number of X/Y/ZDataField properties set.

**Namespace:** MindFusion.Charting
**File:** DataBoundSeries.js

### Syntax

```
JavaScript
get dimensions() {}
```

#### Property Value

Type: Number

### See Also

DataBoundSeries Members
DataBoundSeries Class
MindFusion.Charting Namespace

## DataBoundSeries.innerLabelsDataField Property

Gets or sets the name of inner labels data field in the data source.

**Namespace:** MindFusion.Charting
**File:** DataBoundSeries.js

### Syntax

```
JavaScript
get innerLabelsDataField() {}
```

#### Property Value

Type: String

### See Also

DataBoundSeries Members
DataBoundSeries Class
MindFusion.Charting Namespace

## DataBoundSeries.outerLabelsDataField Property

Gets or sets the name of inner labels data field in the data source.

**Namespace:** MindFusion.Charting
**File:** DataBoundSeries.js

### Syntax

```
JavaScript
get outerLabelsDataField() {}
```

### Property Value

Type: String

## See Also

DataBoundSeries Members
DataBoundSeries Class
MindFusion.Charting Namespace

# DataBoundSeries.size Property

Implements Series.Size. Gets the number of data items in the data source.

**Namespace:** MindFusion.Charting
**File:** DataBoundSeries.js

## Syntax

```
JavaScript
get size() {}
```

### Property Value

Type: Number

## See Also

DataBoundSeries Members
DataBoundSeries Class
MindFusion.Charting Namespace

# DataBoundSeries.supportedLabels Property

Implements Series.SupportedLabels. Result depends on which *LabelsDataField properties are set.

**Namespace:** MindFusion.Charting
**File:** DataBoundSeries.js

## Syntax

```
JavaScript
get supportedLabels() {}
```

### Property Value

Type: LabelKinds

## See Also

DataBoundSeries Members
DataBoundSeries Class
MindFusion.Charting Namespace

## DataBoundSeries.title Property

Implements Series.Title. Gets or sets the title of this series.

**Namespace:** MindFusion.Charting
**File:** DataBoundSeries.js

### Syntax

```
JavaScript
get title() {}
```

#### Property Value

Type: String

### See Also

DataBoundSeries Members
DataBoundSeries Class
MindFusion.Charting Namespace

## DataBoundSeries.toolTipsDataField Property

Gets or sets the name of tooltips data field in the data source.

**Namespace:** MindFusion.Charting
**File:** DataBoundSeries.js

### Syntax

```
JavaScript
get toolTipsDataField() {}
```

#### Property Value

Type: String

### See Also

DataBoundSeries Members
DataBoundSeries Class
MindFusion.Charting Namespace

## DataBoundSeries.xAxisLabelsDataField Property

Gets or sets the name of X axis labels data field in the data source.

**Namespace:** MindFusion.Charting
**File:** DataBoundSeries.js

## Syntax

```
JavaScript
get xAxisLabelsDataField() {}
```

### Property Value

Type: String

## See Also

DataBoundSeries Members
DataBoundSeries Class
MindFusion.Charting Namespace

# DataBoundSeries.xDataField Property

Gets or sets the name of X data field in the data source.

**Namespace:** MindFusion.Charting
**File:** DataBoundSeries.js

## Syntax

```
JavaScript
get xDataField() {}
```

### Property Value

Type: String

## See Also

DataBoundSeries Members
DataBoundSeries Class
MindFusion.Charting Namespace

# DataBoundSeries.yAxisLabelsDataField Property

Gets or sets the name of Y axis labels data field in the data source.

**Namespace:** MindFusion.Charting
**File:** DataBoundSeries.js

## Syntax

```
JavaScript
get yAxisLabelsDataField() {}
```

### Property Value

Type: String

## See Also

DataBoundSeries Members
DataBoundSeries Class
MindFusion.Charting Namespace

# DataBoundSeries.yDataField Property

Gets or sets the name of Y data field in the data source.

**Namespace:** MindFusion.Charting
**File:** DataBoundSeries.js

## Syntax

**JavaScript**

```
get yDataField() {}
```

### Property Value

Type: String

## See Also

DataBoundSeries Members
DataBoundSeries Class
MindFusion.Charting Namespace

# DataBoundSeries.zAxisLabelsDataField Property

Gets or sets the name of Z axis labels data field in the data source.

**Namespace:** MindFusion.Charting
**File:** DataBoundSeries.js

## Syntax

**JavaScript**

```
get zAxisLabelsDataField() {}
```

### Property Value

Type: String

## See Also

DataBoundSeries Members
DataBoundSeries Class
MindFusion.Charting Namespace

# DataBoundSeries.zDataField Property

Gets or sets the name of Z data field in the data source.

**Namespace:** MindFusion.Charting
**File:** DataBoundSeries.js

## Syntax

```
JavaScript
get zDataField() {}
```

### Property Value

Type: String

## See Also

DataBoundSeries Members
DataBoundSeries Class
MindFusion.Charting Namespace

# DateTimeFormat Enumeration

Specifies formats for DateTime values.

**Namespace:** MindFusion.Charting
**File:** DateTimeFormat.js

## Syntax

```
JavaScript
// enum
DateTimeFormat = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **CustomDateTime** | Indicates that the number should be formatted as a DateTime value according to a custom set string. |
| | **FullDateTime** | Indicates that the number should be formatted as a DateTime value with a full date pattern. |
| | **LongDate** | Indicates that the number should be formatted as a DateTime value with a long date pattern. |
| | **LongTime** | Indicates that the number should be formatted as a DateTime value with a long time pattern. |
| | **MonthDateTime** | Indicates that the number should be formatted as a DateTime value with a month day pattern. |
| | **None** | The DateTime value is not formatted |
| | **ShortDate** | Indicates that the number should be formatted as a DateTime value with a short date pattern. |

| | | |
|---|---|---|
| | **ShortTime** | Indicates that the number should be formatted as a DateTime value with a short time pattern. |
| | **YearDateTime** | Indicates that the number should be formatted as a DateTime value with a year month date pattern. |

## See Also

MindFusion.Charting Namespace

# DateTimeSeries Class

Represents a series that contains Date values as X coordinates and number values as Y coordinates.

## Syntax

**JavaScript**

```
// class
DateTimeSeries.prototype = {}
```

## Inheritance Hierarchy

## See Also

DateTimeSeries Members
MindFusion.Charting Namespace

# DateTimeSeries Members

Represents a series that contains Date values as X coordinates and number values as Y coordinates.

The following tables list the members exposed by the DateTimeSeries type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | DateTimeSeries | Overloaded. |

## Public Properties

| | Name | Description |
|---|---|---|
| | customDateTimeFormat | Gets or sets a custom format string for Date labels. |
| | dataChanged | Raised when the values in this series change. |
| | dates | Gets or sets a list of Date values used as X coordinates of this series. |
| | dateTimeFormat | Gets or sets a value indicating how to format Date values as labels. |
| | dimensions | Implements Series.Dimensions. Returns 2. |

| | | |
|---|---|---|
| | labelPrefix | Gets or sets a prefix added in front of formatted Date labels. |
| | labelSuffix | Gets or sets a suffix appended to formatted Date labels. |
| | maxDate | Gets or sets the end of the time range. |
| | maxValue | Gets or sets the coordinate corresponding to MaxDate. |
| | minDate | Gets or sets the start of the time range. |
| | minValue | Gets or sets the coordinate corresponding to MinDate. |
| | size | Implements Series.Size. Returns the number of elements provided in Date list. |
| | supportedLabels | Implements Series.SupportedLabels. Returns LabelKinds.XAxisLabel. |
| | title | Implements Series.Title. Gets or sets the title of this series. |

## Public Methods

| | Name | Description |
|---|---|---|
| | getLabel | Implements Series.GetLabel. Returns Date value at specified index as an XAxisLabel. |
| | getValue | Implements Series.GetValue. Returns a value for the specified data item in this series. |
| | isEmphasized | Implements Series.IsEmphasized. DateTimeSeries always returns false. |
| | isSorted | Implements Series.IsSorted. DateTimeSeries always returns false. |
| | onDataChanged | Raises the DataChanged event. |

## See Also

DateTimeSeries Class
MindFusion.Charting Namespace

# DateTimeSeries Constructor

## Overload List

| Name | Description |
|---|---|
| DateTimeSeries | Initializes a new instance of the DateTimeSeries class. |
| DateTimeSeries | Initializes a new instance of the **DateTimeSeries** class. |

## See Also

DateTimeSeries Class
MindFusion.Charting Namespace

# DateTimeSeries Constructor

Initializes a new instance of the DateTimeSeries class.

**Namespace:** MindFusion.Charting
**File:** DateTimeSeries.js

## Syntax

**JavaScript**

```
function DateTimeSeries (dates, values, minDate, maxDate)
```

### Parameters

*dates*
Type: List | Array
A list of Date values.

*values*
Type: List | Array
A list of number values.

*minDate*
Type: Date
Identifies the start of the time range.

*maxDate*
Type: Date
Identifies the end of the time range.

## Example

The following code creates a new DateTimeSeries that uses the last 10 years as Date values and an array with income data. The first and last Date in the date array are used as min and max values.

**JavaScript**

```
// create a sample series
var years = [];
var dt = new Date(new Date().getFullYear() - 10, 11, 31);
for (; ;)
{
 if (dt.getFullYear() > new Date().getFullYear())
  break;
 years.push(new Date(dt.setFullYear(dt.getFullYear() + 1)));
}

var income = [12, 13.2, 15.6, 17.8, 39, 20, 29, 79, 101, 120, 122];

var series = new DateTimeSeries(
 years, income, years.item(0), years.item(years.count() - 1));
```

## See Also

DateTimeSeries Constructor Overload List
DateTimeSeries Class
MindFusion.Charting Namespace

# DateTimeSeries Constructor

Initializes a new instance of the DateTimeSeries class.

**Namespace:** MindFusion.Charting
**File:** DateTimeSeries.js

## Syntax

**JavaScript**

```
function DateTimeSeries (dates, values, minDate, maxDate)
```

### Parameters

*dates*
Type: List
A list of Date values.

*values*
Type: **List**
A list of number values.

*minDate*
Type: Date
Identifies the start of the time range.

*maxDate*
Type: Date
Identifies the end of the time range.

## Example

The following code creates a new DateTimeSeries that uses the last 10 years as Date values and a List with income data. The first and last Date in the list are used as min and max values.

**JavaScript**

```
// create sample series
var years = new Collections.List();
var dt = new Date(new Date().getFullYear() - 10, 11, 31);
for (; ;)
{
 if (dt.getFullYear() > new Date().getFullYear())
  break;
 years.add(new Date(dt.setFullYear(dt.getFullYear() + 1)));
}

var income = new Collections.List([12, 13.2, 15.6, 17.8, 39, 20, 29, 79, 101, 120, 122]);

var series = new DateTimeSeries(
 years, income, years.item(0), years.item(years.count() - 1));
```

## See Also

DateTimeSeries Constructor Overload List
DateTimeSeries Class
MindFusion.Charting Namespace

# DateTimeSeries Methods

# DateTimeSeries.getLabel Method

Implements Series.GetLabel. Returns Date value at specified index as an XAxisLabel.

**Namespace:** MindFusion.Charting
**File:** DateTimeSeries.js

## Syntax

```
JavaScript
function getLabel (index, kind)
```

### Parameters

*index*
Type: Number
An integer value specifying the index of a data item.

*kind*
Type: LabelKinds
A member of the LabelKinds enumeration.

### Return Value

Type: String
A string containing the item's label.

## See Also

DateTimeSeries Members
DateTimeSeries Class
MindFusion.Charting Namespace

# DateTimeSeries.getValue Method

Implements Series.GetValue. Returns a value for the specified data item in this series.

**Namespace:** MindFusion.Charting
**File:** DateTimeSeries.js

## Syntax

```
JavaScript
function getValue (index, dimension)
```

### Parameters

*index*
Type: Number
An integer value specifying the index of a data item.

*dimension*
Type: Number
An integer value specifying the dimension whose coordinate to return.

### Return Value

Type: Number
A double-precision number representing the value of a data item in the series.

## See Also

DateTimeSeries Members
DateTimeSeries Class
MindFusion.Charting Namespace

# DateTimeSeries.isEmphasized Method

Implements Series.IsEmphasized. DateTimeSeries always returns false.

**Namespace:** MindFusion.Charting
**File:** DateTimeSeries.js

## Syntax

**JavaScript**

```
function isEmphasized (index)
```

### Parameters

*index*
Type: Number
An integer value specifying the index of a data item.

### Return Value

Type: Boolean
true to emphasize the specified data item, or false otherwise.

## See Also

DateTimeSeries Members
DateTimeSeries Class
MindFusion.Charting Namespace

# DateTimeSeries.isSorted Method

Implements Series.IsSorted. DateTimeSeries always returns false.

**Namespace:** MindFusion.Charting
**File:** DateTimeSeries.js

## Syntax

**JavaScript**

```
function isSorted (dimension)
```

### Parameters

*dimension*
Type: Number
An integer value identifying the dimension.

### Return Value

Type: Boolean
true if the values are sorted in specified dimension, or false otherwise.

## See Also

DateTimeSeries Members
DateTimeSeries Class
MindFusion.Charting Namespace

# DateTimeSeries.onDataChanged Method

Raises the DataChanged event.

**Namespace:** MindFusion.Charting
**File:** DateTimeSeries.js

## Syntax

**JavaScript**

```
function onDataChanged ()
```

## See Also

DateTimeSeries Members
DateTimeSeries Class
MindFusion.Charting Namespace

# DateTimeSeries Properties

# DateTimeSeries.customDateTimeFormat Property

Gets or sets a custom format string for Date labels.

**Namespace:** MindFusion.Charting
**File:** DateTimeSeries.js

## Syntax

**JavaScript**

```
get customDateTimeFormat() {}
```

### Property Value

Type: String

## See Also

DateTimeSeries Members
DateTimeSeries Class
MindFusion.Charting Namespace

# DateTimeSeries.dataChanged Property

Raised when the values in this series change.

**Namespace:** MindFusion.Charting
**File:** DateTimeSeries.js

## Syntax

```
JavaScript
get dataChanged() {}
```

### Property Value

Type: EventDispatcher

## See Also

DateTimeSeries Members
DateTimeSeries Class
MindFusion.Charting Namespace

# DateTimeSeries.dates Property

Gets or sets a list of Date values used as X coordinates of this series.

**Namespace:** MindFusion.Charting
**File:** DateTimeSeries.js

## Syntax

```
JavaScript
get dates() {}
```

### Property Value

Type: List

## See Also

DateTimeSeries Members
DateTimeSeries Class
MindFusion.Charting Namespace

# DateTimeSeries.dateTimeFormat Property

Gets or sets a value indicating how to format Date values as labels.

**Namespace:** MindFusion.Charting
**File:** DateTimeSeries.js

## Syntax

```
JavaScript
get dateTimeFormat() {}
```

### Property Value

Type: String

## See Also

DateTimeSeries Members
DateTimeSeries Class
MindFusion.Charting Namespace

# DateTimeSeries.dimensions Property

Implements Series.Dimensions. Returns 2.

**Namespace:** MindFusion.Charting
**File:** DateTimeSeries.js

## Syntax

**JavaScript**

```
get dimensions() {}
```

### Property Value

Type: Number

## See Also

DateTimeSeries Members
DateTimeSeries Class
MindFusion.Charting Namespace

# DateTimeSeries.labelPrefix Property

Gets or sets a prefix added in front of formatted Date labels.

**Namespace:** MindFusion.Charting
**File:** DateTimeSeries.js

## Syntax

**JavaScript**

```
get labelPrefix() {}
```

### Property Value

Type: String

## See Also

DateTimeSeries Members
DateTimeSeries Class
MindFusion.Charting Namespace

# DateTimeSeries.labelSuffix Property

Gets or sets a suffix appended to formatted Date labels.

**Namespace:** MindFusion.Charting
**File:** DateTimeSeries.js

## Syntax

```
JavaScript
get labelSuffix() {}
```

### Property Value

Type: String

## See Also

DateTimeSeries Members
DateTimeSeries Class
MindFusion.Charting Namespace

# DateTimeSeries.maxDate Property

Gets or sets the end of the time range.

**Namespace:** MindFusion.Charting
**File:** DateTimeSeries.js

## Syntax

```
JavaScript
get maxDate() {}
```

### Property Value

Type: Date

## See Also

DateTimeSeries Members
DateTimeSeries Class
MindFusion.Charting Namespace

# DateTimeSeries.maxValue Property

Gets or sets the coordinate corresponding to MaxDate.

**Namespace:** MindFusion.Charting
**File:** DateTimeSeries.js

## Syntax

```
JavaScript
```

```
get maxValue() {}
```

### Property Value

Type: Number

## See Also

DateTimeSeries Members
DateTimeSeries Class
MindFusion.Charting Namespace

# DateTimeSeries.minDate Property

Gets or sets the start of the time range.

**Namespace:** MindFusion.Charting
**File:** DateTimeSeries.js

## Syntax

**JavaScript**

```
get minDate() {}
```

### Property Value

Type: Date

## See Also

DateTimeSeries Members
DateTimeSeries Class
MindFusion.Charting Namespace

# DateTimeSeries.minValue Property

Gets or sets the coordinate corresponding to MinDate.

**Namespace:** MindFusion.Charting
**File:** DateTimeSeries.js

## Syntax

**JavaScript**

```
get minValue() {}
```

### Property Value

Type: Number

## See Also

DateTimeSeries Members
DateTimeSeries Class

## DateTimeSeries.size Property

Implements Series.Size. Returns the number of elements provided in Date list.

**Namespace:** MindFusion.Charting
**File:** DateTimeSeries.js

### Syntax

```
JavaScript
get size() {}
```

#### Property Value

Type: Number

### See Also

DateTimeSeries Members
DateTimeSeries Class
MindFusion.Charting Namespace

## DateTimeSeries.supportedLabels Property

Implements Series.SupportedLabels. Returns LabelKinds.XAxisLabel.

**Namespace:** MindFusion.Charting
**File:** DateTimeSeries.js

### Syntax

```
JavaScript
get supportedLabels() {}
```

#### Property Value

Type: LabelKinds

### See Also

DateTimeSeries Members
DateTimeSeries Class
MindFusion.Charting Namespace

## DateTimeSeries.title Property

Implements Series.Title. Gets or sets the title of this series.

**Namespace:** MindFusion.Charting
**File:** DateTimeSeries.js

### Syntax

| JavaScript |
| --- |
| `get title() {}` |

### Property Value

Type: String

## See Also

DateTimeSeries Members
DateTimeSeries Class
MindFusion.Charting Namespace

# EventSeries Class

Represents a series containing order or timing data, useful forshowing event sequences in tower charts.

## Syntax

| JavaScript |
| --- |
| `// class`<br>`EventSeries.prototype = {}` |

## Inheritance Hierarchy

**MindFusion.Charting.EventSeries**

## See Also

EventSeries Members
MindFusion.Charting Namespace

# EventSeries Members

Represents a series containing order or timing data, useful forshowing event sequences in tower charts.

The following tables list the members exposed by the EventSeries type.

## Public Constructors

| | Name | Description |
| --- | --- | --- |
| | EventSeries | Initializes a new instance of the EventSeries class. |

## Public Methods

| | Name | Description |
| --- | --- | --- |
| | getLabel | Implements Series.GetLabel. |
| | getValue | Implements Series.GetValue. Returns a coordinate of specified point in this series. |

| | | |
|---|---|---|
| | isEmphasized | Implements Series.IsEmphasized. EventSeries always returns false. |
| | isSorted | Implements Series.IsSorted. |
| | onDataChanged | Raises the DataChanged event. |

## Public Events

| | Name | Description |
|---|---|---|
| | dataChanged | Raised when the values in this series change. |

## See Also

EventSeries Class
MindFusion.Charting Namespace

# EventSeries Constructor

Initializes a new instance of the EventSeries class.

**Namespace:** MindFusion.Charting
**File:** EventSeries.js

## Syntax

**JavaScript**

```
function EventSeries (values)
```

### Parameters

*values*
Type:
List.<ChartEvent> | Array.<ChartEvent>. A list of ChartEvent objects.

## See Also

EventSeries Class
MindFusion.Charting Namespace

# EventSeries Methods

# EventSeries.getLabel Method

Implements Series.GetLabel.

**Namespace:** MindFusion.Charting
**File:** EventSeries.js

## Syntax

**JavaScript**

```
function getLabel (index, kind)
```

### Parameters

*index*
Type: Number
Number. An integer value specifying the index of a data item.

*kind*
Type: LabelKinds
LabelKinds. A member of the LabelKinds enumeration.

### Return Value

Type: String
String. A string containing the item's label.

## See Also

EventSeries Members
EventSeries Class
MindFusion.Charting Namespace

# EventSeries.getValue Method

Implements Series.GetValue. Returns a coordinate of specified point in this series.

**Namespace:** MindFusion.Charting
**File:** EventSeries.js

## Syntax

**JavaScript**

```
function getValue (index, dimension)
```

### Parameters

*index*
Type: Number
Number. An integer value specifying the index of a data item.

*dimension*
Type: Number
Number. An integer value specifying the dimension whose coordinate to return.

### Return Value

Type: Number
Number. A double-precision number representing the value of a data item in the series.

## See Also

EventSeries Members
EventSeries Class
MindFusion.Charting Namespace

# EventSeries.isEmphasized Method

Implements Series.IsEmphasized. EventSeries always returns false.

**Namespace:** MindFusion.Charting
**File:** EventSeries.js

## Syntax

**JavaScript**

```
function isEmphasized (index)
```

### Parameters

*index*
Type: Number
Number. An integer value specifying the index of a data item.

### Return Value

Type: Boolean
Boolean. true to emphasize the specified data item, or false otherwise.

## See Also

EventSeries Members
EventSeries Class
MindFusion.Charting Namespace

# EventSeries.isSorted Method

Implements Series.IsSorted.

**Namespace:** MindFusion.Charting
**File:** EventSeries.js

## Syntax

**JavaScript**

```
function isSorted (dimension)
```

### Parameters

*dimension*
Type: Number
Number. An integer value identifying the dimension.

### Return Value

Type: Boolean
Boolean. true if the values are sorted in specified dimension, or false otherwise.

## See Also

EventSeries Members
EventSeries Class

MindFusion.Charting Namespace

## EventSeries.onDataChanged Method

Raises the DataChanged event.

**Namespace:** MindFusion.Charting
**File:** EventSeries.js

### Syntax

**JavaScript**

```
function onDataChanged ()
```

### See Also

EventSeries Members
EventSeries Class
MindFusion.Charting Namespace

## EventSeries Events

## EventSeries.dataChanged Event

Raised when the values in this series change.

**Namespace:** MindFusion.Charting
**File:** EventSeries.js

### Syntax

### See Also

EventSeries Members
EventSeries Class
MindFusion.Charting Namespace

# FunctionSeries Class

Represents a series that calculates its values from provided formula.

### Syntax

**JavaScript**

```
// class
FunctionSeries.prototype = {}
```

### Inheritance Hierarchy

**MindFusion.Charting.FunctionSeries**

## See Also

FunctionSeries Members
MindFusion.Charting Namespace

# FunctionSeries Members

Represents a series that calculates its values from provided formula.

The following tables list the members exposed by the FunctionSeries type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | FunctionSeries | Initializes a new instance of the **FunctionSeries** class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | dimensions | Gets the number of dimensions. |
| | expression | Gets the function used to evaluate the series at a specific X. |
| | size | Gets the number of elements in Data list. |
| | supportedLabels | Gets the type of supported labels. |
| | title | Gets or sets the title of this series. |

## Public Methods

| | Name | Description |
|---|---|---|
| | getLabel | Implements Series.GetLabel. Returns an element of the labels list. |
| | getValue | Implements Series.GetValue. Returns a value for the specified data item in this series. |
| | isEmphasized | Implements Series.IsEmphasized. Gets a value indicating whetherthe specified data item should be emphasized by the SeriesRenderer. |
| | isSorted | Implements Series.IsSorted. |
| | onDataChanged | Raises the DataChanged event. |

## Public Events

| | Name | Description |
|---|---|---|
| | dataChanged | Raised when the values in this series change. |

## See Also

FunctionSeries Class
MindFusion.Charting Namespace

# FunctionSeries Constructor

Initializes a new instance of the FunctionSeries class.

**Namespace:** MindFusion.Charting
**File:** FunctionSeries.js

## Syntax

```
JavaScript
function FunctionSeries (expression, size, minX, maxX)
```

### Parameters

*expression*
Type: function
function. The function used to evaluate the series at a specific X.

*size*
Type: number
number. An integer value specifying the number of values to calculate.

*minX*
Type: number
number. A double-precision number specifying the minimum value of function range.

*maxX*
Type: number
number. A double-precision number specifying the maximum value of function range.

## See Also

FunctionSeries Class
MindFusion.Charting Namespace

# FunctionSeries Methods

# FunctionSeries.getLabel Method

Implements Series.GetLabel. Returns an element of the labels list.

**Namespace:** MindFusion.Charting
**File:** FunctionSeries.js

## Syntax

```
JavaScript
function getLabel (index, kind)
```

### Parameters

*index*

Type: Number
Number. An integer value specifying the index of a data item.

*kind*
Type: LabelKinds
MindFusion.Charting.LabelKinds. A member of the LabelKinds enumeration.

### Return Value

Type: String
String. A string containing the item's label.

## See Also

FunctionSeries Members
FunctionSeries Class
MindFusion.Charting Namespace

# FunctionSeries.getValue Method

Implements Series.GetValue. Returns a value for the specified data item in this series.

**Namespace:** MindFusion.Charting
**File:** FunctionSeries.js

## Syntax

**JavaScript**

```
function getValue (index, dimension)
```

### Parameters

*index*
Type: Number
Number. An integer value specifying the index of a data item.

*dimension*
Type: Number
Number. An integer value specifying the dimension whose coordinate to return.

### Return Value

Type: Number
Number. A double-precision number representing the value of a data item in the series.

## See Also

FunctionSeries Members
FunctionSeries Class
MindFusion.Charting Namespace

# FunctionSeries.isEmphasized Method

Implements Series.IsEmphasized. Gets a value indicating whetherthe specified data item should be emphasized by the SeriesRenderer.

**Namespace:** MindFusion.Charting
**File:** FunctionSeries.js

## Syntax

**JavaScript**

```
function isEmphasized (index)
```

### Parameters

*index*
Type: Number
Number. An integer value specifying the index of a data item.

### Return Value

Type: Boolean
Boolean. true to emphasize the specified data item, or false otherwise.

## See Also

FunctionSeries Members
FunctionSeries Class
MindFusion.Charting Namespace

# FunctionSeries.isSorted Method

Implements Series.IsSorted.

**Namespace:** MindFusion.Charting
**File:** FunctionSeries.js

## Syntax

**JavaScript**

```
function isSorted (dimension)
```

### Parameters

*dimension*
Type: Number
Number. An integer value identifying the dimension.

### Return Value

Type: Boolean
Boolean. true if the values are sorted in specified dimension, or false otherwise.

## See Also

FunctionSeries Members
FunctionSeries Class
MindFusion.Charting Namespace

# FunctionSeries.onDataChanged Method

Raises the DataChanged event.

**Namespace:** MindFusion.Charting
**File:** FunctionSeries.js

## Syntax

```
JavaScript
function onDataChanged ()
```

## See Also

FunctionSeries Members
FunctionSeries Class
MindFusion.Charting Namespace

# FunctionSeries Properties

# FunctionSeries.dimensions Property

Gets the number of dimensions.

**Namespace:** MindFusion.Charting
**File:** FunctionSeries.js

## Syntax

```
JavaScript
get dimensions() {}
set dimensions(value) {}
```

### Property Value

Type: Number
*Number*

The number of dimensions. Returns 2.

## See Also

FunctionSeries Members
FunctionSeries Class
MindFusion.Charting Namespace

# FunctionSeries.expression Property

Gets the function used to evaluate the series at a specific X.

**Namespace:** MindFusion.Charting
**File:** FunctionSeries.js

## Syntax

```
JavaScript
```

```
get expression() {}
set expression(value) {}
```

### Property Value

Type: String
*String*

The function used to evaluate the series at a specific X.

## See Also

FunctionSeries Members
FunctionSeries Class
MindFusion.Charting Namespace

# FunctionSeries.size Property

Gets the number of elements in Data list.

**Namespace:** MindFusion.Charting
**File:** FunctionSeries.js

## Syntax

**JavaScript**

```
get size() {}
set size(value) {}
```

### Property Value

Type: Number
*Number*

The number of elements in Data list.

## See Also

FunctionSeries Members
FunctionSeries Class
MindFusion.Charting Namespace

# FunctionSeries.supportedLabels Property

Gets the type of supported labels.

**Namespace:** MindFusion.Charting
**File:** FunctionSeries.js

## Syntax

**JavaScript**

```
get supportedLabels() {}
set supportedLabels(value) {}
```

### Property Value

Type: String
*String*

The type of supported labels. Returns None.

## See Also

FunctionSeries Members
FunctionSeries Class
MindFusion.Charting Namespace

# FunctionSeries.title Property

Gets or sets the title of this series.

**Namespace:** MindFusion.Charting
**File:** FunctionSeries.js

## Syntax

**JavaScript**

```
get title() {}
set title(value) {}
```

### Property Value

Type: String
*String*

The title of this series.

## See Also

FunctionSeries Members
FunctionSeries Class
MindFusion.Charting Namespace

# FunctionSeries Events

# FunctionSeries.dataChanged Event

Raised when the values in this series change.

**Namespace:** MindFusion.Charting
**File:** FunctionSeries.js

## Syntax

## See Also

FunctionSeries Members
FunctionSeries Class
MindFusion.Charting Namespace

# FunnelRenderer Class

Implements a SeriesRenderer used to draw funnel charts. Funnel charts represent the stages of a process and show relative values or measurements associated with each stage.

## Syntax

**JavaScript**

```
// class
FunnelRenderer.prototype = {}
```

## Inheritance Hierarchy

## See Also

FunnelRenderer Members
MindFusion.Charting Namespace

# FunnelRenderer Members

Implements a SeriesRenderer used to draw funnel charts. Funnel charts represent the stages of a process and show relative values or measurements associated with each stage.

The following tables list the members exposed by the FunnelRenderer type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | FunnelRenderer | Initializes a new instance of the FunnelRenderer class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | dimension | Gets or sets the dimension index that should be used to read data from the Series. |
|  | segmentSpacing | Gets or sets the spacing between segments. |
|  | series | Gets or sets the Series drawn by this FunnelRenderer. |
|  | stemWidth | Gets or sets the width of the funnel stem. |

## Public Methods

| | Name | Description |
|---|---|---|
|  | draw | SeriesRenderer.Draw override. Draws the series data in specified RenderContext. |
|  | drawHighlight | SeriesRenderer.DrawHighlight override. Draws highlighted slice in specified RenderContext. |
|  | enumSegments | Enumerates the segments of the funnel. |

| | | |
|---|---|---|
| | enumSeries | Implements the SeriesContainer interface. |
| | hitTest | SeriesRenderer.HitTest override. Hit-tests the slices representing data items. |

## See Also

FunnelRenderer Class
MindFusion.Charting Namespace

# FunnelRenderer Constructor

Initializes a new instance of the FunnelRenderer class.

**Namespace:** MindFusion.Charting
**File:** FunnelRenderer.js

## Syntax

**JavaScript**

```
function FunnelRenderer (series, [dimension])
```

### Parameters

*series*
Type: Series
The Series that should be rendered as a funnel.

*dimension*
Optional.
Type: Number
Specifies the Series dimension index that should be used to access data.

## See Also

FunnelRenderer Class
MindFusion.Charting Namespace

# FunnelRenderer Methods

# FunnelRenderer.draw Method

SeriesRenderer.Draw override. Draws the series data in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** FunnelRenderer.js

## Syntax

**JavaScript**

```
function draw (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

FunnelRenderer Members
FunnelRenderer Class
MindFusion.Charting Namespace

# FunnelRenderer.drawHighlight Method

SeriesRenderer.DrawHighlight override. Draws highlighted slice in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** FunnelRenderer.js

## Syntax

**JavaScript**

```
function drawHighlight (context, hitResult)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*hitResult*
Type: HitResult
A HitResult instance identifying highlighted data item.

## See Also

FunnelRenderer Members
FunnelRenderer Class
MindFusion.Charting Namespace

# FunnelRenderer.enumSegments Method

Enumerates the segments of the funnel.

**Namespace:** MindFusion.Charting
**File:** FunnelRenderer.js

## Syntax

**JavaScript**

```
function enumSegments (context, process)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*process*
Type: ProcessSegment
A ProcessSegment callback.

## Remarks

The signature of the ProcessSegment delegate is as follows:'function(dataIndex, points)'.param {Number} dataIndex An integer index of data item in Series.param {MindFusion.Drawing.Point[]} points The defining points of the segment's bounding polygon.

## See Also

FunnelRenderer Members
FunnelRenderer Class
MindFusion.Charting Namespace

# FunnelRenderer.enumSeries Method

Implements the SeriesContainer interface.

**Namespace:** MindFusion.Charting
**File:** FunnelRenderer.js

## Syntax

**JavaScript**

```
function enumSeries ()
```

### Return Value

Type: IEnumerable
An instance of the IEnumerable<Series> class.

## See Also

FunnelRenderer Members
FunnelRenderer Class
MindFusion.Charting Namespace

# FunnelRenderer.hitTest Method

SeriesRenderer.HitTest override. Hit-tests the slices representing data items.

**Namespace:** MindFusion.Charting
**File:** FunnelRenderer.js

## Syntax

**JavaScript**

```
function hitTest (context, location)
```

### Parameters

*context*

Type: RenderContext
A RenderContext instance.

*location*
Type: Point
A Point specifying where to look for a pie slice.

### Return Value

Type: HitResult
A HitResult instance identifying the found data item.

## See Also

FunnelRenderer Members
FunnelRenderer Class
MindFusion.Charting Namespace

# FunnelRenderer Properties

## FunnelRenderer.dimension Property

Gets or sets the dimension index that should be used to read data from the Series.

**Namespace:** MindFusion.Charting
**File:** FunnelRenderer.js

## Syntax

```
JavaScript
get dimension() {}
```

### Property Value

Type: Number

## See Also

FunnelRenderer Members
FunnelRenderer Class
MindFusion.Charting Namespace

## FunnelRenderer.segmentSpacing Property

Gets or sets the spacing between segments.

**Namespace:** MindFusion.Charting
**File:** FunnelRenderer.js

## Syntax

```
JavaScript
get segmentSpacing() {}
```

### Property Value

Type: Number

## See Also

FunnelRenderer Members
FunnelRenderer Class
MindFusion.Charting Namespace

# FunnelRenderer.series Property

Gets or sets the Series drawn by this FunnelRenderer.

**Namespace:** MindFusion.Charting
**File:** FunnelRenderer.js

## Syntax

**JavaScript**

```
get series() {}
```

### Property Value

Type: Series

## See Also

FunnelRenderer Members
FunnelRenderer Class
MindFusion.Charting Namespace

# FunnelRenderer.stemWidth Property

Gets or sets the width of the funnel stem.

**Namespace:** MindFusion.Charting
**File:** FunnelRenderer.js

## Syntax

**JavaScript**

```
get stemWidth() {}
```

### Property Value

Type: Number

## See Also

FunnelRenderer Members
FunnelRenderer Class
MindFusion.Charting Namespace

# GridType Enumeration

Indicates Plot2D grid type.

**Namespace:** MindFusion.Charting
**File:** GridType.js

## Syntax

**JavaScript**

```
// enum
GridType = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **Crossed** | Draw crossed stripes. |
| | **Horizontal** | Draw horizontal stripes. |
| | **None** | Do not draw grid. |
| | **Vertical** | Draw vertical stripes. |

## See Also

MindFusion.Charting Namespace

# HitResult Class

Represents the result of a hit-test operation.

## Syntax

**JavaScript**

```
// class
HitResult.prototype = {}
```

## Inheritance Hierarchy

## See Also

HitResult Members
MindFusion.Charting Namespace

### HitResult Members

Represents the result of a hit-test operation.

The following tables list the members exposed by the HitResult type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | HitResult | Initializes a new instance of the HitResult class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | index | Gets or sets the data item index within its Series. |
| | plot | Gets or sets the Plot that has been hit. |
| | renderer | Gets or sets the SeriesRenderer that has been hit. |
| | series | Gets or sets the Series that has been hit. |
| | value | Gets or sets the data item value. |

## See Also

HitResult Class
MindFusion.Charting Namespace

# HitResult Constructor

Initializes a new instance of the HitResult class.

**Namespace:** MindFusion.Charting
**File:** HitResult.js

## Syntax

**JavaScript**

```
function HitResult (renderer, index, value, series)
```

### Parameters

*renderer*
Type: SeriesRenderer
A SeriesRenderer whose data item has been hit.

*index*
Type: Number
An integer index of the data item.

*value*
The item's value.

*series*
Type: Series
The Series that contains the found item.

## See Also

HitResult Class
MindFusion.Charting Namespace

# HitResult Properties

## HitResult.index Property

Gets or sets the data item index within its Series.

**Namespace:** MindFusion.Charting
**File:** HitResult.js

### Syntax

```
JavaScript
get index() {}
```

#### Property Value

Type: Number

### See Also

HitResult Members
HitResult Class
MindFusion.Charting Namespace

## HitResult.plot Property

Gets or sets the Plot that has been hit.

**Namespace:** MindFusion.Charting
**File:** HitResult.js

### Syntax

```
JavaScript
get plot() {}
```

#### Property Value

Type: Plot

### See Also

HitResult Members
HitResult Class
MindFusion.Charting Namespace

## HitResult.renderer Property

Gets or sets the SeriesRenderer that has been hit.

**Namespace:** MindFusion.Charting
**File:** HitResult.js

### Syntax

```
JavaScript
get renderer() {}
```

### Property Value

Type: SeriesRenderer

## See Also

HitResult Members
HitResult Class
MindFusion.Charting Namespace

# HitResult.series Property

Gets or sets the Series that has been hit.

**Namespace:** MindFusion.Charting
**File:** HitResult.js

## Syntax

```
JavaScript
get series() {}
```

### Property Value

Type: Series

## See Also

HitResult Members
HitResult Class
MindFusion.Charting Namespace

# HitResult.value Property

Gets or sets the data item value.

**Namespace:** MindFusion.Charting
**File:** HitResult.js

## Syntax

```
JavaScript
get value() {}
```

### Property Value

Type: Number

## See Also

HitResult Members
HitResult Class
MindFusion.Charting Namespace

# LabelKinds Enumeration

Identifies the various kinds of labels that can be drawn for data items.

**Namespace:** MindFusion.Charting
**File:** LabelKinds.js

## Syntax

**JavaScript**

```
// enum
LabelKinds = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **All** | Draw all labels supported by a series. |
| | **InnerLabel** | Draw inner label in graphic representation of data item. |
| | **None** | Do not draw labels. |
| | **OuterLabel** | Draw outer label nearby graphic representation of data item. |
| | **ToolTip** | Show label as a tooltip. |
| | **XAxisLabel** | Draw data label at X axis. |
| | **YAxisLabel** | Draw data label at Y axis. |
| | **ZAxisLabel** | Draw data label at Z axis. |

## See Also

MindFusion.Charting Namespace

# LegendController Class

Lets users move a legend within the boundaries of its parent Panel.

## Syntax

**JavaScript**

```
// class
LegendController.prototype = {}
```

## Inheritance Hierarchy

## See Also

LegendController Members
MindFusion.Charting Namespace

# LegendController Members

Lets users move a legend within the boundaries of its parent Panel.

The following tables list the members exposed by the LegendController type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | LegendController | Initializes a new instance of the LegendController class. |

## Public Methods

| | Name | Description |
|---|---|---|
| | onMouseDown | Implements ComponentController.OnMouseDown. Called when the user presses a mouse button. |
| | onMouseMove | Implements ComponentController.OnMouseMove. Called when the user moves the mouse. |

## See Also

LegendController Class
MindFusion.Charting Namespace

# LegendController Constructor

Initializes a new instance of the LegendController class.

**Namespace:** MindFusion.Charting
**File:** LegendController.js

## Syntax

| JavaScript |
|---|
| `function LegendController (renderContext)` |

### Parameters

*renderContext*
Type: RenderContext
A RenderContext instance.

## See Also

LegendController Class
MindFusion.Charting Namespace

# LegendController Methods

## LegendController.onMouseDown Method

Implements ComponentController.OnMouseDown. Called when the user presses a mouse button.

**Namespace:** MindFusion.Charting
**File:** LegendController.js

### Syntax

```
JavaScript
function onMouseDown (x, y)
```

#### Parameters

*x*
Type: Number
A number value specifying the horizontal position of mouse pointer.

*y*
Type: Number
A number value specifying the vertical position of mouse pointer.

### See Also

LegendController Members
LegendController Class
MindFusion.Charting Namespace

## LegendController.onMouseMove Method

Implements ComponentController.OnMouseMove. Called when the user moves the mouse.

**Namespace:** MindFusion.Charting
**File:** LegendController.js

### Syntax

```
JavaScript
function onMouseMove (x, y)
```

#### Parameters

*x*
Type: Number
A number value specifying the horizontal position of mouse pointer.

*y*
Type: Number
A number value specifying the vertical position of mouse pointer.

### See Also

LegendController Members
LegendController Class

# LegendRenderer Class

A component that renders chart legend.

## Syntax

```
// class
LegendRenderer.prototype = {}
```

## Inheritance Hierarchy

## See Also

LegendRenderer Members
MindFusion.Charting Namespace

# LegendRenderer Members

A component that renders chart legend.

The following tables list the members exposed by the LegendRenderer type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | LegendRenderer | Initializes a new instance of the LegendRenderer class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | allowMove | Gets or sets a value indicating whether users are allowed to move this legend. |
| | background | Gets or sets the Brush that should be used to fill the background of this legend. |
| | borderStroke | Gets or sets the Brush that should be used to stroke the borders of this legend. |
| | borderStrokeDashStyle | Gets or sets the stroke dash style of legend borders. |
| | borderStrokeThickness | Gets or sets the stroke thickness of legend borders. |
| | content | Gets or sets a list of SeriesRenderer objects whose Series should be displayed in this legend. |
| | elementLabelKind | Gets or sets the type of label to show for series elements. This property is used only when showSeriesElements is set to true. |

| | | |
|---|---|---|
| | maxItemsPerColumn | Gets or sets the maximum number ot items per display column. |
| | padding | Gets or sets the padding space between content and borders of this legend. |
| | showSeriesElements | Gets or sets a value, indicatating whether this legend will display individual series element labels instead of the series titles. |
| | showTitle | Gets or sets a value indicating whether to display the legend title. |
| | spacing | Gets or sets the distance between adjacent entries in the legend. |
| | title | Gets or sets the legend's title. |
| | titleBrush | Gets or sets the Brush that should be used to draw the legend's Title. |
| | titleFontName | Gets or sets the name of font that should be used to draw the legend's Title. |
| | titleFontSize | Gets or sets the size of font that should be used to draw the legend's Title. |
| | titleFontStyle | Gets or sets the style of font that should be used to draw the legend's Title. |

## Public Methods

| | Name | Description |
|---|---|---|
| | createController | Component.CreateController override. Returns a ComponentController used to interact with this component. |
| | draw | Component.Draw override. Draws legend elements in specified RenderContext. |
| | effectiveBorderPen | Gets the Pen that should be used to draw the legend borders. |
| | effectiveBorderStroke | Gets the Brush that should be used to stroke the legend borders. |
| | effectiveBorderStrokeThickness | Gets the thickness of border strokes. |
| | effectiveFill | Gets the Brush that should be used to fill the legend background. |
| | effectiveStrokeDashStyle | Gets the dash style of border strokes. |
| | effectiveTitleBrush | Gets the Brush that should be used to draw the legend's Title. |
| | effectiveTitleFont | Gets the Font that should be used to draw the legend's Title. |
| | measure | Component.Measure override. Measures the desired size of this component. |

## See Also

LegendRenderer Class
MindFusion.Charting Namespace

## LegendRenderer Constructor

Initializes a new instance of the LegendRenderer class.

**Namespace:** MindFusion.Charting
**File:** LegendRenderer.js

### Syntax

```
JavaScript
function LegendRenderer ()
```

### See Also

LegendRenderer Class
MindFusion.Charting Namespace

## LegendRenderer Methods

## LegendRenderer.createController Method

Component.CreateController override. Returns a ComponentController used to interact with this component.

**Namespace:** MindFusion.Charting
**File:** LegendRenderer.js

### Syntax

```
JavaScript
function createController (context)
```

#### Parameters

*context*
Type: RenderContext
A RenderContext instance.

#### Return Value

Type: ComponentController
Instance of the LegendController class.

### See Also

LegendRenderer Members
LegendRenderer Class
MindFusion.Charting Namespace

# LegendRenderer.draw Method

Component.Draw override. Draws legend elements in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** LegendRenderer.js

## Syntax

**JavaScript**

```
function draw (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

LegendRenderer Members
LegendRenderer Class
MindFusion.Charting Namespace

# LegendRenderer.effectiveBorderPen Method

Gets the Pen that should be used to draw the legend borders.

**Namespace:** MindFusion.Charting
**File:** LegendRenderer.js

## Syntax

**JavaScript**

```
function effectiveBorderPen (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: Pen
A Pen instance.

## See Also

LegendRenderer Members
LegendRenderer Class
MindFusion.Charting Namespace

# LegendRenderer.effectiveBorderStroke Method

Gets the Brush that should be used to stroke the legend borders.

**Namespace:** MindFusion.Charting
**File:** LegendRenderer.js

## Syntax

**JavaScript**

```
function effectiveBorderStroke (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: Brush
A Brush instance.

## See Also

LegendRenderer Members
LegendRenderer Class
MindFusion.Charting Namespace

# LegendRenderer.effectiveBorderStrokeThickness Method

Gets the thickness of border strokes.

**Namespace:** MindFusion.Charting
**File:** LegendRenderer.js

## Syntax

**JavaScript**

```
function effectiveBorderStrokeThickness (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: Number
A number value.

## See Also

LegendRenderer Members
LegendRenderer Class

# LegendRenderer.effectiveFill Method

Gets the Brush that should be used to fill the legend background.

**Namespace:** MindFusion.Charting
**File:** LegendRenderer.js

## Syntax

**JavaScript**

```
function effectiveFill (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: Brush
A Brush instance.

## See Also

LegendRenderer Members
LegendRenderer Class
MindFusion.Charting Namespace

# LegendRenderer.effectiveStrokeDashStyle Method

Gets the dash style of border strokes.

**Namespace:** MindFusion.Charting
**File:** LegendRenderer.js

## Syntax

**JavaScript**

```
function effectiveStrokeDashStyle (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: DashStyle
A member of the DashStyle enumeration.

## See Also

LegendRenderer Members
LegendRenderer Class
MindFusion.Charting Namespace

# LegendRenderer.effectiveTitleBrush Method

Gets the Brush that should be used to draw the legend's Title.

**Namespace:** MindFusion.Charting
**File:** LegendRenderer.js

## Syntax

**JavaScript**

```
function effectiveTitleBrush (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: SysBrush
A System.Brush instance.

## See Also

LegendRenderer Members
LegendRenderer Class
MindFusion.Charting Namespace

# LegendRenderer.effectiveTitleFont Method

Gets the Font that should be used to draw the legend's Title.

**Namespace:** MindFusion.Charting
**File:** LegendRenderer.js

## Syntax

**JavaScript**

```
function effectiveTitleFont (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: Font
A System.Drawing.Font instance.

## See Also

LegendRenderer Members
LegendRenderer Class
MindFusion.Charting Namespace

# LegendRenderer.measure Method

Component.Measure override. Measures the desired size of this component.

**Namespace:** MindFusion.Charting
**File:** LegendRenderer.js

## Syntax

**JavaScript**

```
function measure (maxWidth, maxHeight, context)
```

### Parameters

*maxWidth*
Type: Number
The maximum width provided by parent component.

*maxHeight*
Type: Number
The maximum height provided by parent component.

*context*
Type: RenderContext
A RenderContext instance.

## See Also

LegendRenderer Members
LegendRenderer Class
MindFusion.Charting Namespace

# LegendRenderer Properties

# LegendRenderer.allowMove Property

Gets or sets a value indicating whether users are allowed to move this legend.

**Namespace:** MindFusion.Charting
**File:** LegendRenderer.js

## Syntax

**JavaScript**

```
get allowMove() {}
```

### Property Value

Type: Boolean

## See Also

LegendRenderer Members
LegendRenderer Class
MindFusion.Charting Namespace

# LegendRenderer.background Property

Gets or sets the Brush that should be used to fill the background of this legend.

**Namespace:** MindFusion.Charting
**File:** LegendRenderer.js

## Syntax

**JavaScript**

```
get background() {}
```

### Property Value

Type: Brush

## See Also

LegendRenderer Members
LegendRenderer Class
MindFusion.Charting Namespace

# LegendRenderer.borderStroke Property

Gets or sets the Brush that should be used to stroke the borders of this legend.

**Namespace:** MindFusion.Charting
**File:** LegendRenderer.js

## Syntax

**JavaScript**

```
get borderStroke() {}
```

### Property Value

Type: Brush

## See Also

LegendRenderer Members
LegendRenderer Class
MindFusion.Charting Namespace

# LegendRenderer.borderStrokeDashStyle Property

Gets or sets the stroke dash style of legend borders.

**Namespace:** MindFusion.Charting
**File:** LegendRenderer.js

## Syntax

```
JavaScript
get borderStrokeDashStyle() {}
```

### Property Value

Type: DashStyle

## See Also

LegendRenderer Members
LegendRenderer Class
MindFusion.Charting Namespace

# LegendRenderer.borderStrokeThickness Property

Gets or sets the stroke thickness of legend borders.

**Namespace:** MindFusion.Charting
**File:** LegendRenderer.js

## Syntax

```
JavaScript
get borderStrokeThickness() {}
```

### Property Value

Type: Number

## See Also

LegendRenderer Members
LegendRenderer Class
MindFusion.Charting Namespace

# LegendRenderer.content Property

Gets or sets a list of SeriesRenderer objects whose Series should be displayed in this legend.

**Namespace:** MindFusion.Charting
**File:** LegendRenderer.js

## Syntax

```
JavaScript
get content() {}
```

### Property Value

Type: ObservableCollection

## See Also

LegendRenderer Members
LegendRenderer Class
MindFusion.Charting Namespace

# LegendRenderer.elementLabelKind Property

Gets or sets the type of label to show for series elements. This property is used only when showSeriesElements is set to true.

**Namespace:** MindFusion.Charting
**File:** LegendRenderer.js

## Syntax

```
JavaScript
get elementLabelKind() {}
```

### Property Value

Type: LabelKinds

## See Also

LegendRenderer Members
LegendRenderer Class
MindFusion.Charting Namespace

# LegendRenderer.maxItemsPerColumn Property

Gets or sets the maximum number ot items per display column.

**Namespace:** MindFusion.Charting
**File:** LegendRenderer.js

## Syntax

```
JavaScript
get maxItemsPerColumn() {}
```

### Property Value

Type: Number

## See Also

LegendRenderer Members
LegendRenderer Class
MindFusion.Charting Namespace

## LegendRenderer.padding Property

Gets or sets the padding space between content and borders of this legend.

**Namespace:** MindFusion.Charting
**File:** LegendRenderer.js

### Syntax

```
JavaScript
get padding() {}
```

#### Property Value

Type: Number

### See Also

LegendRenderer Members
LegendRenderer Class
MindFusion.Charting Namespace

## LegendRenderer.showSeriesElements Property

Gets or sets a value, indicatating whether this legend will display individual series element labels instead of the series titles.

**Namespace:** MindFusion.Charting
**File:** LegendRenderer.js

### Syntax

```
JavaScript
get showSeriesElements() {}
```

#### Property Value

Type: Boolean

### See Also

LegendRenderer Members
LegendRenderer Class
MindFusion.Charting Namespace

## LegendRenderer.showTitle Property

Gets or sets a value indicating whether to display the legend title.

**Namespace:** MindFusion.Charting
**File:** LegendRenderer.js

### Syntax

```
JavaScript
```

```
get showTitle() {}
```

### Property Value

Type: Boolean

## See Also

LegendRenderer Members
LegendRenderer Class
MindFusion.Charting Namespace

# LegendRenderer.spacing Property

Gets or sets the distance between adjacent entries in the legend.

**Namespace:** MindFusion.Charting
**File:** LegendRenderer.js

## Syntax

**JavaScript**

```
get spacing() {}
```

### Property Value

Type: Number

## See Also

LegendRenderer Members
LegendRenderer Class
MindFusion.Charting Namespace

# LegendRenderer.title Property

Gets or sets the legend's title.

**Namespace:** MindFusion.Charting
**File:** LegendRenderer.js

## Syntax

**JavaScript**

```
get title() {}
```

### Property Value

Type: String

## See Also

LegendRenderer Members
LegendRenderer Class

## LegendRenderer.titleBrush Property

Gets or sets the Brush that should be used to draw the legend's Title.

**Namespace:** MindFusion.Charting
**File:** LegendRenderer.js

### Syntax

**JavaScript**

```
get titleBrush() {}
```

#### Property Value

Type: Brush

### See Also

LegendRenderer Members
LegendRenderer Class
MindFusion.Charting Namespace

## LegendRenderer.titleFontName Property

Gets or sets the name of font that should be used to draw the legend's Title.

**Namespace:** MindFusion.Charting
**File:** LegendRenderer.js

### Syntax

**JavaScript**

```
get titleFontName() {}
```

#### Property Value

Type: String

### See Also

LegendRenderer Members
LegendRenderer Class
MindFusion.Charting Namespace

## LegendRenderer.titleFontSize Property

Gets or sets the size of font that should be used to draw the legend's Title.

**Namespace:** MindFusion.Charting
**File:** LegendRenderer.js

### Syntax

```
JavaScript
get titleFontSize() {}
```

### Property Value

Type: Number

## See Also

LegendRenderer Members
LegendRenderer Class
MindFusion.Charting Namespace

# LegendRenderer.titleFontStyle Property

Gets or sets the style of font that should be used to draw the legend's Title.

**Namespace:** MindFusion.Charting
**File:** LegendRenderer.js

## Syntax

```
JavaScript
get titleFontStyle() {}
```

### Property Value

Type: FontStyle

## See Also

LegendRenderer Members
LegendRenderer Class
MindFusion.Charting Namespace

# LineRenderer Class

A SeriesRenderer that draws series as polylines.

## Syntax

```
JavaScript
// class
LineRenderer.prototype = {}
```

## Inheritance Hierarchy

## See Also

LineRenderer Members
MindFusion.Charting Namespace

# LineRenderer Members

A SeriesRenderer that draws series as polylines.

The following tables list the members exposed by the LineRenderer type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | LineRenderer | Initializes a new instance of the LineRenderer class. |

## Public Methods

| | Name | Description |
|---|---|---|
| | draw | SeriesRenderer.Draw override. Draws the series data in specified RenderContext. |

## See Also

LineRenderer Class
MindFusion.Charting Namespace

# LineRenderer Constructor

Initializes a new instance of the LineRenderer class.

**Namespace:** MindFusion.Charting
**File:** LineRenderer.js

## Syntax

| JavaScript |
|---|
| `function LineRenderer (series)` |

### Parameters

*series*
Type: ObservableCollection
A list of Series that should be rendered as polylines.

## See Also

LineRenderer Class
MindFusion.Charting Namespace

# LineRenderer Methods

# LineRenderer.draw Method

SeriesRenderer.Draw override. Draws the series data in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** LineRenderer.js

## Syntax

```
JavaScript
function draw (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

LineRenderer Members
LineRenderer Class
MindFusion.Charting Namespace

# LineType Enumeration

Identifies how to connect adjacent data points in line and area charts.

**Namespace:** MindFusion.Charting
**File:** LineType.js

## Syntax

```
JavaScript
// enum
LineType = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **Curve** | Connect points using a cardinal spline. |
| | **Polyline** | Connect points using straight line segments. |
| | **Step** | Connect points using steps. |

## See Also

MindFusion.Charting Namespace

# Margins Class

Describes rectangular margins.

## Syntax

| JavaScript |
| --- |
| `// class`<br>`Margins.prototype = {}` |

## Inheritance Hierarchy

## See Also

Margins Members
MindFusion.Charting Namespace

# Margins Members

Describes rectangular margins.

The following tables list the members exposed by the Margins type.

## Public Constructors

| | Name | Description |
| --- | --- | --- |
|  | Margins | Initializes a new instance of the Margins structure. |

## Public Properties

| | Name | Description |
| --- | --- | --- |
|  | bottom | Gets or sets the width of the bottom side of the frame. |
|  | height | Returns the sum of Top and Bottom margins. |
|  | left | Gets or sets the width of the left side of the frame. |
|  | right | Gets or sets the width of the right side of the frame. |
|  | top | Gets or sets the width of the top side of the frame. |
|  | width | Returns the sum of Left and Right margins. |

## See Also

Margins Class
MindFusion.Charting Namespace

# Margins Constructor

Initializes a new instance of the Margins structure.

**Namespace:** MindFusion.Charting
**File:** Margins.js

## Syntax

| JavaScript |
| --- |
| |

```
function Margins (left, top, right, bottom)
```

### Parameters

*left*
Type: void

*top*
Type: void

*right*
Type: void

*bottom*
Type: void

## See Also

Margins Class
MindFusion.Charting Namespace

# Margins Properties

## Margins.bottom Property

Gets or sets the width of the bottom side of the frame.

**Namespace:** MindFusion.Charting
**File:** Margins.js

## Syntax

**JavaScript**

```
get bottom() {}
```

### Property Value

Type: Number

## See Also

Margins Members
Margins Class
MindFusion.Charting Namespace

## Margins.height Property

Returns the sum of Top and Bottom margins.

**Namespace:** MindFusion.Charting
**File:** Margins.js

## Syntax

**JavaScript**

```
get height() {}
```

### Property Value

Type: Number

## See Also

Margins Members
Margins Class
MindFusion.Charting Namespace

# Margins.left Property

Gets or sets the width of the left side of the frame.

**Namespace:** MindFusion.Charting
**File:** Margins.js

## Syntax

**JavaScript**

```
get left() {}
```

### Property Value

Type: Number

## See Also

Margins Members
Margins Class
MindFusion.Charting Namespace

# Margins.right Property

Gets or sets the width of the right side of the frame.

**Namespace:** MindFusion.Charting
**File:** Margins.js

## Syntax

**JavaScript**

```
get right() {}
```

### Property Value

Type: Number

## See Also

Margins Members
Margins Class

## Margins.top Property

Gets or sets the width of the top side of the frame.

**Namespace:** MindFusion.Charting
**File:** Margins.js

### Syntax

```
JavaScript
get top() {}
```

#### Property Value

Type: Number

### See Also

Margins Members
Margins Class
MindFusion.Charting Namespace

## Margins.width Property

Returns the sum of Left and Right margins.

**Namespace:** MindFusion.Charting
**File:** Margins.js

### Syntax

```
JavaScript
get width() {}
```

#### Property Value

Type: Number

### See Also

Margins Members
Margins Class
MindFusion.Charting Namespace

# MixedSeriesStyle Class

Implements SeriesStyle using mixed per-element and uniform values for attributes.

### Syntax

JavaScript

```
// class
MixedSeriesStyle.prototype = {}
```

## Inheritance Hierarchy

## See Also

MixedSeriesStyle Members
MindFusion.Charting Namespace

# MixedSeriesStyle Members

Implements SeriesStyle using mixed per-element and uniform values for attributes.

The following tables list the members exposed by the MixedSeriesStyle type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | MixedSeriesStyle | Initializes a new instance of the MixedSeriesStyle class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | commonFills | Gets or sets a list of brushes, each Brush used to fill all elements of a series. |
| | commonStrokeDashStyles | Gets or sets a list of dash styles, each style applied to all elements of a series. |
| | commonStrokes | Gets or sets a list of brushes, each Brush used to stroke all elements of a series. |
| | commonStrokeThicknesses | Gets or sets a list of stroke thicknesses, each thickness applied to all elements of a series. |
| | fills | Gets or sets a list of Brush lists, each list used to draw a different series, and each Brush used to fill individual elements of a series. |
| | strokeDashStyles | Gets or sets a list of DashStyle lists, each list used to stroke a different series, and each DashStyle value specifying style of individual elements of a series. |
| | strokes | Gets or sets a list of Brush lists, each list used to stroke a different series, and each Brush used to stroke individual elements of a series. |
| | strokeThicknesses | Gets or sets a list of number lists, each list used to stroke a different series, and each number value specifying thickness of individual elements of a series. |
| | uniformFill | Gets or sets a Brush used to fill all elements of all series uniformly. |

| | | |
|---|---|---|
| | uniformStroke | Gets or sets a Brush used to stroke all elements of all series uniformly. |
| | uniformStrokeDashStyle | Gets or sets uniform stroke dash style for all elements of all series. |
| | uniformStrokeThickness | Gets or sets uniform stroke thickness for all elements of all series. |

## Public Methods

| | Name | Description |
|---|---|---|
| | fill | Implements SeriesStyle.Stroke. Returns a Brush instance fromFills, CommonFills or UniformFill properties. |
| | stroke | Implements SeriesStyle.Stroke. Returns a Brush instance fromStrokes, CommonStrokes or UniformStroke properties. |
| | strokeDashStyle | Implements SeriesStyle.StrokeDashStyle. Returns a value from StrokeDashStyles, CommonStrokeDashStyles or UniformStrokeDashStyle properties. |
| | strokeThickness | Implements SeriesStyle.StrokeThickness. Returns a value from StrokeThicknesses, CommonStrokeThicknesses or UniformStrokeThickness properties. |

## See Also

MixedSeriesStyle Class
MindFusion.Charting Namespace

# MixedSeriesStyle Constructor

Initializes a new instance of the MixedSeriesStyle class.

**Namespace:** MindFusion.Charting
**File:** MixedSeriesStyle.js

## Syntax

**JavaScript**

```
function MixedSeriesStyle ([fill, [stroke, [strokeThickness, [strokeDashStyle]]]])
```

### Parameters

*fill*
Optional.
Type: Brush
A Brush used to fill all series elements.

*stroke*
Optional.
Type: Brush
A Brush used to stroke all series elements.

*strokeThickness*
Optional.

Type: Number
Uniform thickness of series elements' strokes.

*strokeDashStyle*
Optional.
Type: DashStyle
Uniform dash style of series elements' strokes.

## See Also

MixedSeriesStyle Class
MindFusion.Charting Namespace

# MixedSeriesStyle Methods

## MixedSeriesStyle.fill Method

Implements SeriesStyle.Stroke. Returns a Brush instance fromFills, CommonFills or UniformFill properties.

**Namespace:** MindFusion.Charting
**File:** MixedSeriesStyle.js

## Syntax

**JavaScript**

```
function fill (seriesIndex, dataIndex)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series.

*dataIndex*
Type: Number
An integer index of a data item in specified series.

### Return Value

Type: Brush
A Brush instance.

## See Also

MixedSeriesStyle Members
MixedSeriesStyle Class
MindFusion.Charting Namespace

## MixedSeriesStyle.stroke Method

Implements SeriesStyle.Stroke. Returns a Brush instance fromStrokes, CommonStrokes or UniformStroke properties.

**Namespace:** MindFusion.Charting
**File:** MixedSeriesStyle.js

## Syntax

```
JavaScript
function stroke (seriesIndex, dataIndex)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series.

*dataIndex*
Type: Number
An integer index of a data item in specified series.

### Return Value

Type: Brush
A Brush instance.

## See Also

MixedSeriesStyle Members
MixedSeriesStyle Class
MindFusion.Charting Namespace

# MixedSeriesStyle.strokeDashStyle Method

Implements SeriesStyle.StrokeDashStyle. Returns a value from StrokeDashStyles, CommonStrokeDashStyles or UniformStrokeDashStyle properties.

**Namespace:** MindFusion.Charting
**File:** MixedSeriesStyle.js

## Syntax

```
JavaScript
function strokeDashStyle (seriesIndex, dataIndex)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series.

*dataIndex*
Type: Number
An integer index of a data item in specified series.

### Return Value

Type: DashStyle
A nullable DashStyle value specifying stroke dash style.

## See Also

MixedSeriesStyle Members
MixedSeriesStyle Class
MindFusion.Charting Namespace

## MixedSeriesStyle.strokeThickness Method

Implements SeriesStyle.StrokeThickness. Returns a value from StrokeThicknesses, CommonStrokeThicknesses or UniformStrokeThickness properties.

**Namespace:** MindFusion.Charting
**File:** MixedSeriesStyle.js

### Syntax

**JavaScript**

```
function strokeThickness (seriesIndex, dataIndex)
```

#### Parameters

*seriesIndex*
Type: Number
An integer index of a Series.

*dataIndex*
Type: Number
An integer index of a data item in specified series.

#### Return Value

Type: Number
A nullable number value specifying stroke thickness.

### See Also

MixedSeriesStyle Members
MixedSeriesStyle Class
MindFusion.Charting Namespace

## MixedSeriesStyle Properties

## MixedSeriesStyle.commonFills Property

Gets or sets a list of brushes, each Brush used to fill all elements of a series.

**Namespace:** MindFusion.Charting
**File:** MixedSeriesStyle.js

### Syntax

**JavaScript**

```
get commonFills() {}
```

#### Property Value

Type: List

## See Also

MixedSeriesStyle Members
MixedSeriesStyle Class
MindFusion.Charting Namespace

# MixedSeriesStyle.commonStrokeDashStyles Property

Gets or sets a list of dash styles, each style applied to all elements of a series.

**Namespace:** MindFusion.Charting
**File:** MixedSeriesStyle.js

## Syntax

**JavaScript**

```
get commonStrokeDashStyles() {}
```

### Property Value

Type: DashStyle

## See Also

MixedSeriesStyle Members
MixedSeriesStyle Class
MindFusion.Charting Namespace

# MixedSeriesStyle.commonStrokes Property

Gets or sets a list of brushes, each Brush used to stroke all elements of a series.

**Namespace:** MindFusion.Charting
**File:** MixedSeriesStyle.js

## Syntax

**JavaScript**

```
get commonStrokes() {}
```

### Property Value

Type: List

## See Also

MixedSeriesStyle Members
MixedSeriesStyle Class
MindFusion.Charting Namespace

# MixedSeriesStyle.commonStrokeThicknesses Property

Gets or sets a list of stroke thicknesses, each thickness applied to all elements of a series.

**Namespace:** MindFusion.Charting
**File:** MixedSeriesStyle.js

## Syntax

```
JavaScript
get commonStrokeThicknesses() {}
```

### Property Value

Type: Number

## See Also

MixedSeriesStyle Members
MixedSeriesStyle Class
MindFusion.Charting Namespace

# MixedSeriesStyle.fills Property

Gets or sets a list of Brush lists, each list used to draw a different series, and each Brush used to fill individual elements of a series.

**Namespace:** MindFusion.Charting
**File:** MixedSeriesStyle.js

## Syntax

```
JavaScript
get fills() {}
```

### Property Value

Type: List>

## See Also

MixedSeriesStyle Members
MixedSeriesStyle Class
MindFusion.Charting Namespace

# MixedSeriesStyle.strokeDashStyles Property

Gets or sets a list of DashStyle lists, each list used to stroke a different series, and each DashStyle value specifying style of individual elements of a series.

**Namespace:** MindFusion.Charting
**File:** MixedSeriesStyle.js

## Syntax

```
JavaScript
get strokeDashStyles() {}
```

### Property Value

Type: List>

## See Also

MixedSeriesStyle Members
MixedSeriesStyle Class
MindFusion.Charting Namespace

# MixedSeriesStyle.strokes Property

Gets or sets a list of Brush lists, each list used to stroke a different series, and each Brush used to stroke individual elements of a series.

**Namespace:** MindFusion.Charting
**File:** MixedSeriesStyle.js

## Syntax

```
JavaScript
get strokes() {}
```

### Property Value

Type: List>

## See Also

MixedSeriesStyle Members
MixedSeriesStyle Class
MindFusion.Charting Namespace

# MixedSeriesStyle.strokeThicknesses Property

Gets or sets a list of number lists, each list used to stroke a different series, and each number value specifying thickness of individual elements of a series.

**Namespace:** MindFusion.Charting
**File:** MixedSeriesStyle.js

## Syntax

```
JavaScript
get strokeThicknesses() {}
```

### Property Value

Type: List>

## See Also

MixedSeriesStyle Members
MixedSeriesStyle Class
MindFusion.Charting Namespace

## MixedSeriesStyle.uniformFill Property

Gets or sets a Brush used to fill all elements of all series uniformly.

**Namespace:** MindFusion.Charting
**File:** MixedSeriesStyle.js

### Syntax

```
JavaScript
get uniformFill() {}
```

#### Property Value

Type: Brush

### See Also

MixedSeriesStyle Members
MixedSeriesStyle Class
MindFusion.Charting Namespace

## MixedSeriesStyle.uniformStroke Property

Gets or sets a Brush used to stroke all elements of all series uniformly.

**Namespace:** MindFusion.Charting
**File:** MixedSeriesStyle.js

### Syntax

```
JavaScript
get uniformStroke() {}
```

#### Property Value

Type: Brush

### See Also

MixedSeriesStyle Members
MixedSeriesStyle Class
MindFusion.Charting Namespace

## MixedSeriesStyle.uniformStrokeDashStyle Property

Gets or sets uniform stroke dash style for all elements of all series.

**Namespace:** MindFusion.Charting
**File:** MixedSeriesStyle.js

## Syntax

```
JavaScript
get uniformStrokeDashStyle() {}
```

### Property Value

Type: DashStyle

## See Also

MixedSeriesStyle Members
MixedSeriesStyle Class
MindFusion.Charting Namespace

# MixedSeriesStyle.uniformStrokeThickness Property

Gets or sets uniform stroke thickness for all elements of all series.

**Namespace:** MindFusion.Charting
**File:** MixedSeriesStyle.js

## Syntax

```
JavaScript
get uniformStrokeThickness() {}
```

### Property Value

Type: Number

## See Also

MixedSeriesStyle Members
MixedSeriesStyle Class
MindFusion.Charting Namespace

# Pair Class

Contains two values.

## Syntax

```
JavaScript
// class
Pair.prototype = {}
```

## Inheritance Hierarchy

## See Also

# Pair Members

Contains two values.

The following tables list the members exposed by the Pair type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | Pair | Initializes a new instance of the Pair class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | first | Gets or sets the first value. |
|  | second | Gets or sets the second value. |

## Public Methods

| | Name | Description |
|---|---|---|
|  | toString | Object.ToString override. |

## See Also

Pair Class
MindFusion.Charting Namespace

# Pair Constructor

Initializes a new instance of the Pair class.

**Namespace:** MindFusion.Charting
**File:** Pair.js

## Syntax

| JavaScript |
|---|
| `function Pair (first, second)` |

### Parameters

*first*
Type: void
The first value.

*second*
Type: void
The second value.

## See Also

Pair Class
MindFusion.Charting Namespace

# Pair Methods

## Pair.toString Method

Object.ToString override.

**Namespace:** MindFusion.Charting
**File:** Pair.js

## Syntax

**JavaScript**

```
function toString ()
```

## See Also

Pair Members
Pair Class
MindFusion.Charting Namespace

# Pair Properties

## Pair.first Property

Gets or sets the first value.

**Namespace:** MindFusion.Charting
**File:** Pair.js

## Syntax

**JavaScript**

```
get first() {}
```

### Property Value

Type: T1

## See Also

Pair Members
Pair Class
MindFusion.Charting Namespace

## Pair.second Property

Gets or sets the second value.

**Namespace:** MindFusion.Charting
**File:** Pair.js

### Syntax

```
JavaScript
get second() {}
```

#### Property Value

Type: T1

### See Also

Pair Members
Pair Class
MindFusion.Charting Namespace

# PanController Class

Represents a controller that pans its plot's data range.

### Syntax

```
JavaScript
// class
PanController.prototype = {}
```

### Inheritance Hierarchy

### See Also

PanController Members
MindFusion.Charting Namespace

## PanController Members

Represents a controller that pans its plot's data range.

The following tables list the members exposed by the PanController type.

### Public Constructors

| | Name | Description |
|---|---|---|
| | PanController | Initializes a new instance of the PanController class. |

### Public Properties

| | Name | Description |
|---|---|---|
| | enableAnimation | Gets or sets whether to enable pan inertia. |

## Public Methods

| | Name | Description |
|---|---|---|
| | getRunningAnimation | For internal use. |
| | onMouseDown | Plot2DController.OnMouseDown override. Called when the user presses a mouse button. |
| | onMouseMove | PlotController.OnMouseMove override. Called when the user moves the mouse. |
| | onMouseUp | PlotController.OnMouseUp override. Called when the user releases a mouse button. |

## See Also

PanController Class
MindFusion.Charting Namespace

# PanController Constructor

Initializes a new instance of the PanController class.

**Namespace:** MindFusion.Charting
**File:** PanController.js

## Syntax

**JavaScript**

```
function PanController (renderContext, vertical)
```

### Parameters

*renderContext*
Type: RenderContext
A RenderContext instance.

*vertical*
Type: Boolean
true to implement interaction along the Y axis, or false for the X axis.

## See Also

PanController Class
MindFusion.Charting Namespace

# PanController Methods

# PanController.getRunningAnimation Method

For internal use.

**Namespace:** MindFusion.Charting
**File:** PanController.js

## Syntax

**JavaScript**

```
function getRunningAnimation ()
```

### Return Value

Type: ComponentAnimation
An instance of a ComponentAnimation -derived class.

## See Also

PanController Members
PanController Class
MindFusion.Charting Namespace

# PanController.onMouseDown Method

Plot2DController.OnMouseDown override. Called when the user presses a mouse button.

**Namespace:** MindFusion.Charting
**File:** PanController.js

## Syntax

**JavaScript**

```
function onMouseDown (x, y)
```

### Parameters

*x*
Type: Number
A number value specifying the horizontal position of mouse pointer.

*y*
Type: Number
A number value specifying the vertical position of mouse pointer.

## See Also

PanController Members
PanController Class
MindFusion.Charting Namespace

# PanController.onMouseMove Method

PlotController.OnMouseMove override. Called when the user moves the mouse.

**Namespace:** MindFusion.Charting
**File:** PanController.js

## Syntax

```
JavaScript
function onMouseMove (x, y)
```

### Parameters

*x*
Type: Number
A number value specifying the horizontal position of mouse pointer.

*y*
Type: Number
A number value specifying the vertical position of mouse pointer.

## See Also

PanController Members
PanController Class
MindFusion.Charting Namespace

# PanController.onMouseUp Method

PlotController.OnMouseUp override. Called when the user releases a mouse button.

**Namespace:** MindFusion.Charting
**File:** PanController.js

## Syntax

```
JavaScript
function onMouseUp (x, y)
```

### Parameters

*x*
Type: Number
A number value specifying the horizontal position of mouse pointer.

*y*
Type: Number
A number value specifying the vertical position of mouse pointer.

## See Also

PanController Members
PanController Class
MindFusion.Charting Namespace

# PanController Properties

## PanController.enableAnimation Property

Gets or sets whether to enable pan inertia.

**Namespace:** MindFusion.Charting
**File:** PanController.js

### Syntax

```
JavaScript
get enableAnimation() {}
```

#### Property Value

Type: Boolean

### See Also

PanController Members
PanController Class
MindFusion.Charting Namespace

# PanControllerAnimation Class

Represents a controller that pans its plot's data range.

### Syntax

```
JavaScript
// class
PanControllerAnimation.prototype = {}
```

### Inheritance Hierarchy

### See Also

PanControllerAnimation Members
MindFusion.Charting Namespace

## PanControllerAnimation Members

Represents a controller that pans its plot's data range.

The following tables list the members exposed by the PanControllerAnimation type.

### See Also

PanControllerAnimation Class
MindFusion.Charting Namespace

# PerElementSeriesStyle Class

Implements SeriesStyle using different attributes for each data item.

## Syntax

**JavaScript**

```
// class
PerElementSeriesStyle.prototype = {}
```

## Inheritance Hierarchy

## See Also

PerElementSeriesStyle Members
MindFusion.Charting Namespace

# PerElementSeriesStyle Members

Implements SeriesStyle using different attributes for each data item.

The following tables list the members exposed by the PerElementSeriesStyle type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | PerElementSeriesStyle | Initializes a new instance of PerElementSeriesStyle. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | fills | Gets or sets a list of Brush lists, each list used to draw a different series, and each Brush used to fill individual elements of a series. |
|  | strokeDashStyles | Gets or sets a list of DashStyle lists, each list used to stroke a different series, and each DashStyle value specifying style of individual elements of a series. |
|  | strokes | Gets or sets a list of Brush lists, each list used to stroke a different series, and each Brush used to stroke individual elements of a series. |
|  | strokeThicknesses | Gets or sets a list of number lists, each list used to stroke a different series, and each number value specifying thickness of individual elements of a series. |

## Public Methods

| | Name | Description |
|---|---|---|
|  | fill | Implements SeriesStyle.Fill. Returns a Brush from the Fills list that matches specified series and data indices. |

| | | |
|---|---|---|
| | stroke | Implements SeriesStyle.Stroke. Returns a Brush from the Strokes list that matches specified series and data indices. |
| | strokeDashStyle | Implements SeriesStyle.StrokeDashStyle. Returns a value from the StrokeDashStyles list that matches specified series and data indices. |
| | strokeThickness | Implements SeriesStyle.StrokeThickness. Returns a value from the StrokeThicknesses list that matches specified series and data indices. |

## See Also

PerElementSeriesStyle Class
MindFusion.Charting Namespace

# PerElementSeriesStyle Constructor

Initializes a new instance of PerElementSeriesStyle.

**Namespace:** MindFusion.Charting
**File:** PerElementSeriesStyle.js

## Syntax

**JavaScript**

```
function PerElementSeriesStyle ([fills, [strokes, [strokeThicknesses, [strokeDashStyles]]]])
```

### Parameters

*fills*
Optional.
Type: List>
A list of Brush lists, each list used to draw a different series,and each Brush used to fill individual elements of a series.

*strokes*
Optional.
Type: List>
A list of Brush lists, each list used to stroke a different series,and each Brush used to stroke individual elements of a series.

*strokeThicknesses*
Optional.
Type: List>
A list of number lists, each list used to stroke a different series,and each number value specifying thickness of individual elements of a series.

*strokeDashStyles*
Optional.
Type: List>
A list of DashStyle lists, each list used to stroke a different series,and each DashStyle value specifying style of individual elements of a series.

## See Also

PerElementSeriesStyle Class
MindFusion.Charting Namespace

# PerElementSeriesStyle Methods

## PerElementSeriesStyle.fill Method

Implements SeriesStyle.Fill. Returns a Brush from the Fills list that matches specified series and data indices.

**Namespace:** MindFusion.Charting
**File:** PerElementSeriesStyle.js

## Syntax

```
JavaScript
function fill (seriesIndex, dataIndex)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series.

*dataIndex*
Type: Number
An integer index of a data item in specified series.

### Return Value

Type: Brush
A Brush instance.

## See Also

PerElementSeriesStyle Members
PerElementSeriesStyle Class
MindFusion.Charting Namespace

## PerElementSeriesStyle.stroke Method

Implements SeriesStyle.Stroke. Returns a Brush from the Strokes list that matches specified series and data indices.

**Namespace:** MindFusion.Charting
**File:** PerElementSeriesStyle.js

## Syntax

```
JavaScript
function stroke (seriesIndex, dataIndex)
```

### Parameters

*seriesIndex*

Type: Number
An integer index of a Series.

*dataIndex*
Type: Number
An integer index of a data item in specified series.

### Return Value

Type: Brush
A Brush instance.

## See Also

PerElementSeriesStyle Members
PerElementSeriesStyle Class
MindFusion.Charting Namespace

# PerElementSeriesStyle.strokeDashStyle Method

Implements SeriesStyle.StrokeDashStyle. Returns a value from the StrokeDashStyles list that matches specified series and data indices.

**Namespace:** MindFusion.Charting
**File:** PerElementSeriesStyle.js

## Syntax

**JavaScript**

```
function strokeDashStyle (seriesIndex, dataIndex)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series.

*dataIndex*
Type: Number
An integer index of a data item in specified series.

### Return Value

Type: DashStyle
A nullable DashStyle value specifying stroke dash style.

## See Also

PerElementSeriesStyle Members
PerElementSeriesStyle Class
MindFusion.Charting Namespace

# PerElementSeriesStyle.strokeThickness Method

Implements SeriesStyle.StrokeThickness. Returns a value from the StrokeThicknesses list that matches specified series and data indices.

**Namespace:** MindFusion.Charting
**File:** PerElementSeriesStyle.js

## Syntax

```
JavaScript
function strokeThickness (seriesIndex, dataIndex)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series.

*dataIndex*
Type: Number
An integer index of a data item in specified series.

### Return Value

Type: Number
A nullable number value specifying stroke thickness.

## See Also

PerElementSeriesStyle Members
PerElementSeriesStyle Class
MindFusion.Charting Namespace

# PerElementSeriesStyle Properties

# PerElementSeriesStyle.fills Property

Gets or sets a list of Brush lists, each list used to draw a different series, and each Brush used to fill individual elements of a series.

**Namespace:** MindFusion.Charting
**File:** PerElementSeriesStyle.js

## Syntax

```
JavaScript
get fills() {}
```

### Property Value

Type: List>

## See Also

PerElementSeriesStyle Members
PerElementSeriesStyle Class
MindFusion.Charting Namespace

## PerElementSeriesStyle.strokeDashStyles Property

Gets or sets a list of DashStyle lists, each list used to stroke a different series, and each DashStyle value specifying style of individual elements of a series.

**Namespace:** MindFusion.Charting
**File:** PerElementSeriesStyle.js

### Syntax

```
JavaScript
get strokeDashStyles() {}
```

#### Property Value

Type: List>

### See Also

PerElementSeriesStyle Members
PerElementSeriesStyle Class
MindFusion.Charting Namespace

## PerElementSeriesStyle.strokes Property

Gets or sets a list of Brush lists, each list used to stroke a different series, and each Brush used to stroke individual elements of a series.

**Namespace:** MindFusion.Charting
**File:** PerElementSeriesStyle.js

### Syntax

```
JavaScript
get strokes() {}
```

#### Property Value

Type: List>

### See Also

PerElementSeriesStyle Members
PerElementSeriesStyle Class
MindFusion.Charting Namespace

## PerElementSeriesStyle.strokeThicknesses Property

Gets or sets a list of number lists, each list used to stroke a different series, and each number value specifying thickness of individual elements of a series.

**Namespace:** MindFusion.Charting
**File:** PerElementSeriesStyle.js

### Syntax

| JavaScript |
| --- |
| `get strokeThicknesses() {}` |

### Property Value

Type: List>

## See Also

PerElementSeriesStyle Members
PerElementSeriesStyle Class
MindFusion.Charting Namespace

# PerSeriesStyle Class

Implements SeriesStyle using a different set of attributes for each series,where all elements in a single series have common appearance.

## Syntax

| JavaScript |
| --- |
| `// class`<br>`PerSeriesStyle.prototype = {}` |

## Inheritance Hierarchy

## See Also

PerSeriesStyle Members
MindFusion.Charting Namespace

# PerSeriesStyle Members

Implements SeriesStyle using a different set of attributes for each series,where all elements in a single series have common appearance.

The following tables list the members exposed by the PerSeriesStyle type.

## Public Constructors

| | Name | Description |
| --- | --- | --- |
|  | PerSeriesStyle | Initializes a new instance of the PerSeriesStyle class. |

## Public Properties

| | Name | Description |
| --- | --- | --- |
|  | fills | Gets or sets a list of Brushes, each Brush used to fill all elements of a series. |

| | | |
|---|---|---|
| | strokeDashStyles | Gets or sets a list of dash styles, each style applied to all elements of a series. |
| | strokes | Gets or sets a list of Brushes, each Brush used to stroke all elements of a series. |
| | strokeThicknesses | Gets or sets a list of stroke thicknesses, each thickness applied to all elements of a series. |

## Public Methods

| | Name | Description |
|---|---|---|
| | fill | Implements SeriesStyle.Fill. Returns a Brush from the Fills list that matches specified series index. |
| | stroke | Implements SeriesStyle.Stroke. Returns a Brush from the Strokes list that matches specified series index. |
| | strokeDashStyle | Implements SeriesStyle.StrokeDashStyle. Returns a value from the StrokeDashStyles list that matches specified series index. |
| | strokeThickness | Implements SeriesStyle.StrokeThickness. Returns a value from the StrokeThicknesses list that matches specified series index. |

## See Also

PerSeriesStyle Class
MindFusion.Charting Namespace

# PerSeriesStyle Constructor

Initializes a new instance of the PerSeriesStyle class.

**Namespace:** MindFusion.Charting
**File:** PerSeriesStyle.js

## Syntax

**JavaScript**

```
function PerSeriesStyle ([fills, [strokes, [strokeThicknesses, [strokeDashStyles]]]])
```

### Parameters

*fills*
Optional.
Type: List
A list of Brushes, each Brush used to fill all elements of a series.

*strokes*
Optional.
Type: List
A list of Brushes, each Brush used to stroke all elements of a series.

*strokeThicknesses*
Optional.

Type: List
A list of stroke thicknesses, each thickness applied to all elements of a series.

*strokeDashStyles*
Optional.
Type: List
A list of dash styles, each style applied to all elements of a series.

## See Also

PerSeriesStyle Class
MindFusion.Charting Namespace

# PerSeriesStyle Methods

# PerSeriesStyle.fill Method

Implements SeriesStyle.Fill. Returns a Brush from the Fills list that matches specified series index.

**Namespace:** MindFusion.Charting
**File:** PerSeriesStyle.js

## Syntax

**JavaScript**

```
function fill (seriesIndex, dataIndex)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series.

*dataIndex*
Type: Number
An integer index of a data item in specified series.

### Return Value

Type: Brush
A Brush instance.

## See Also

PerSeriesStyle Members
PerSeriesStyle Class
MindFusion.Charting Namespace

# PerSeriesStyle.stroke Method

Implements SeriesStyle.Stroke. Returns a Brush from the Strokes list that matches specified series index.

**Namespace:** MindFusion.Charting
**File:** PerSeriesStyle.js

## Syntax

```
JavaScript
function stroke (seriesIndex, dataIndex)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series.

*dataIndex*
Type: Number
An integer index of a data item in specified series.

### Return Value

Type: Brush
A Brush instance.

## See Also

PerSeriesStyle Members
PerSeriesStyle Class
MindFusion.Charting Namespace

# PerSeriesStyle.strokeDashStyle Method

Implements SeriesStyle.StrokeDashStyle. Returns a value from the StrokeDashStyles list that matches specified series index.

**Namespace:** MindFusion.Charting
**File:** PerSeriesStyle.js

## Syntax

```
JavaScript
function strokeDashStyle (seriesIndex, dataIndex)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series.

*dataIndex*
Type: Number
An integer index of a data item in specified series.

### Return Value

Type: DashStyle
A nullable DashStyle value specifying stroke dash style.

## See Also

PerSeriesStyle Members
PerSeriesStyle Class
MindFusion.Charting Namespace

# PerSeriesStyle.strokeThickness Method

Implements SeriesStyle.StrokeThickness. Returns a value from the StrokeThicknesses list that matches specified series index.

**Namespace:** MindFusion.Charting
**File:** PerSeriesStyle.js

## Syntax

**JavaScript**

```
function strokeThickness (seriesIndex, dataIndex)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series.

*dataIndex*
Type: Number
An integer index of a data item in specified series.

### Return Value

Type: Number
A nullable number value specifying stroke thickness.

## See Also

PerSeriesStyle Members
PerSeriesStyle Class
MindFusion.Charting Namespace

# PerSeriesStyle Properties

# PerSeriesStyle.fills Property

Gets or sets a list of Brushes, each Brush used to fill all elements of a series.

**Namespace:** MindFusion.Charting
**File:** PerSeriesStyle.js

## Syntax

**JavaScript**

```
get fills() {}
```

### Property Value

Type: List

## See Also

PerSeriesStyle Members
PerSeriesStyle Class
MindFusion.Charting Namespace

# PerSeriesStyle.strokeDashStyles Property

Gets or sets a list of dash styles, each style applied to all elements of a series.

**Namespace:** MindFusion.Charting
**File:** PerSeriesStyle.js

## Syntax

**JavaScript**

```
get strokeDashStyles() {}
```

### Property Value

Type: List

## See Also

PerSeriesStyle Members
PerSeriesStyle Class
MindFusion.Charting Namespace

# PerSeriesStyle.strokes Property

Gets or sets a list of Brushes, each Brush used to stroke all elements of a series.

**Namespace:** MindFusion.Charting
**File:** PerSeriesStyle.js

## Syntax

**JavaScript**

```
get strokes() {}
```

### Property Value

Type: List

## See Also

PerSeriesStyle Members
PerSeriesStyle Class
MindFusion.Charting Namespace

# PerSeriesStyle.strokeThicknesses Property

Gets or sets a list of stroke thicknesses, each thickness applied to all elements of a series.

**Namespace:** MindFusion.Charting
**File:** PerSeriesStyle.js

## Syntax

```
JavaScript
get strokeThicknesses() {}
```

### Property Value

Type: List

## See Also

PerSeriesStyle Members
PerSeriesStyle Class
MindFusion.Charting Namespace

# PieRadarRenderer Class

A SeriesRenderer that draws pie-radars, where data items are represented circular sectors of equal central angles, and the sector corresponding to a data item has radius proportional to its value.

## Syntax

```
JavaScript
// class
PieRadarRenderer.prototype = {}
```

## Inheritance Hierarchy

## See Also

PieRadarRenderer Members
MindFusion.Charting Namespace

# PieRadarRenderer Members

A SeriesRenderer that draws pie-radars, where data items are represented circular sectors of equal central angles, and the sector corresponding to a data item has radius proportional to its value.

The following tables list the members exposed by the PieRadarRenderer type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | PieRadarRenderer | Initializes a new instance of the PieRadarRenderer class. |

## Public Properties

| | Name | Description |
|---|---|---|

| | | |
|---|---|---|
| | alignToAxis | Gets or sets a value indicating whether sectors should be aligned to axes in the radar grid or centered around them. |

## Public Methods

| | Name | Description |
|---|---|---|
| | draw | RadarRenderer.Draw override. Draws the series data in specified RenderContext. |
| | drawHighlight | RadarRenderer.DrawHighlight override. Draws highlighted data item in specified RenderContext. |
| | enumSectors | Enumerates the pie-radar sectors corresponding to each series. |

## See Also

PieRadarRenderer Class
MindFusion.Charting Namespace

# PieRadarRenderer Constructor

Initializes a new instance of the PieRadarRenderer class.

**Namespace:** MindFusion.Charting
**File:** PieRadarRenderer.js

## Syntax

**JavaScript**

```
function PieRadarRenderer (series)
```

### Parameters

*series*
Type: ObservableCollection
A list of Series that should be rendered by this object.

## See Also

PieRadarRenderer Class
MindFusion.Charting Namespace

# PieRadarRenderer Methods

# PieRadarRenderer.draw Method

RadarRenderer.Draw override. Draws the series data in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** PieRadarRenderer.js

## Syntax

```
JavaScript
function draw (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

PieRadarRenderer Members
PieRadarRenderer Class
MindFusion.Charting Namespace

# PieRadarRenderer.drawHighlight Method

RadarRenderer.DrawHighlight override. Draws highlighted data item in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** PieRadarRenderer.js

## Syntax

```
JavaScript
function drawHighlight (context, hitResult)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*hitResult*
Type: HitResult
A HitResult instance identifying highlighted data item.

## See Also

PieRadarRenderer Members
PieRadarRenderer Class
MindFusion.Charting Namespace

# PieRadarRenderer.enumSectors Method

Enumerates the pie-radar sectors corresponding to each series.

**Namespace:** MindFusion.Charting
**File:** PieRadarRenderer.js

## Syntax

```
JavaScript
function enumSectors (context, process)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*process*
Type: ProcessSectors
A ProcessSectors callback.

## See Also

PieRadarRenderer Members
PieRadarRenderer Class
MindFusion.Charting Namespace

# PieRadarRenderer Properties

## PieRadarRenderer.alignToAxis Property

Gets or sets a value indicating whether sectors should be aligned to axes in the radar grid or centered around them.

**Namespace:** MindFusion.Charting
**File:** PieRadarRenderer.js

## Syntax

**JavaScript**

```
get alignToAxis() {}
```

### Property Value

Type: Boolean

## See Also

PieRadarRenderer Members
PieRadarRenderer Class
MindFusion.Charting Namespace

# PieRenderer Class

A SeriesRenderer that draws pies in its containing plot.

## Syntax

**JavaScript**

```
// class
PieRenderer.prototype = {}
```

## Inheritance Hierarchy

## See Also

PieRenderer Members
MindFusion.Charting Namespace

# PieRenderer Members

A SeriesRenderer that draws pies in its containing plot.

The following tables list the members exposed by the PieRenderer type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | PieRenderer | Initializes a new instance of the PieRenderer class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | dimension | Gets or sets the dimension index that should be used to read data from the Series. |
| | doughnut | Gets or sets a value indicating whether the pie should be rendered as a doughnut. |
| | series | Gets or sets the Series drawn by this PieRenderer. |

## Public Methods

| | Name | Description |
|---|---|---|
| | draw | SeriesRenderer.Draw override. Draws the series data in specified RenderContext. |
| | drawHighlight | SeriesRenderer.DrawHighlight override. Draws highlighted slice in specified RenderContext. |
| | enumSeries | Implements the SeriesContainer interface. |
| | enumSlices | Enumerates the slices of the pie. |
| | hitTest | SeriesRenderer.HitTest override. Hit-tests the slices representing data items. |

## See Also

PieRenderer Class
MindFusion.Charting Namespace

# PieRenderer Constructor

Initializes a new instance of the PieRenderer class.

**Namespace:** MindFusion.Charting
**File:** PieRenderer.js

## Syntax

```
JavaScript
function PieRenderer (series, [dimension])
```

### Parameters

*series*
Type: Series
The Series that should be rendered as a pie.

*dimension*
Optional.
Type: Number
Specifies the Series dimension index that should be used to access data.

## See Also

PieRenderer Class
MindFusion.Charting Namespace

# PieRenderer Methods

## PieRenderer.draw Method

SeriesRenderer.Draw override. Draws the series data in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** PieRenderer.js

## Syntax

```
JavaScript
function draw (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

PieRenderer Members
PieRenderer Class
MindFusion.Charting Namespace

## PieRenderer.drawHighlight Method

SeriesRenderer.DrawHighlight override. Draws highlighted slice in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** PieRenderer.js

## Syntax

```
JavaScript
function drawHighlight (context, hitResult)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*hitResult*
Type: HitResult
A HitResult instance identifying highlighted data item.

## See Also

PieRenderer Members
PieRenderer Class
MindFusion.Charting Namespace

# PieRenderer.enumSeries Method

Implements the SeriesContainer interface.

**Namespace:** MindFusion.Charting
**File:** PieRenderer.js

## Syntax

```
JavaScript
function enumSeries ()
```

### Return Value

Type: IEnumerable
An instance of the IEnumerable<Series> class.

## See Also

PieRenderer Members
PieRenderer Class
MindFusion.Charting Namespace

# PieRenderer.enumSlices Method

Enumerates the slices of the pie.

**Namespace:** MindFusion.Charting
**File:** PieRenderer.js

## Syntax

```
JavaScript
```

```
function enumSlices (context, process)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*process*
Type: ProcessSlice
A ProcessBars callback.

## Remarks

The signature of the ProcessSlice delegate is as follows:'function(dataIndex, rect, diameter, startAngle, sweepAngle)'.param {Number} dataIndex An integer index of data item in Series.param {MindFusion.Drawing.Rect} rect The bounding rectangle of the slice' circle.param {Number} diameter The diameter of the pie.param {Number} startAngle The start angle of the slice' sector.param {Number} sweepAngle The sweep angle of the slice' sector.The method is not expected to return a value.

## See Also

PieRenderer Members
PieRenderer Class
MindFusion.Charting Namespace

# PieRenderer.hitTest Method

SeriesRenderer.HitTest override. Hit-tests the slices representing data items.

**Namespace:** MindFusion.Charting
**File:** PieRenderer.js

## Syntax

**JavaScript**

```
function hitTest (context, location)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*location*
Type: Point
A Point specifying where to look for a pie slice.

### Return Value

Type: HitResult
A HitResult instance identifying the found data item.

## See Also

PieRenderer Members
PieRenderer Class
MindFusion.Charting Namespace

# PieRenderer Properties

## PieRenderer.dimension Property

Gets or sets the dimension index that should be used to read data from the Series.

**Namespace:** MindFusion.Charting
**File:** PieRenderer.js

### Syntax

```
JavaScript
get dimension() {}
```

#### Property Value

Type: Number

### See Also

PieRenderer Members
PieRenderer Class
MindFusion.Charting Namespace

## PieRenderer.doughnut Property

Gets or sets a value indicating whether the pie should be rendered as a doughnut.

**Namespace:** MindFusion.Charting
**File:** PieRenderer.js

### Syntax

```
JavaScript
get doughnut() {}
```

#### Property Value

Type: Boolean

### See Also

PieRenderer Members
PieRenderer Class
MindFusion.Charting Namespace

## PieRenderer.series Property

Gets or sets the Series drawn by this PieRenderer.

**Namespace:** MindFusion.Charting
**File:** PieRenderer.js

## Syntax

```
JavaScript
get series() {}
```

### Property Value

Type: Series

## See Also

PieRenderer Members
PieRenderer Class
MindFusion.Charting Namespace

# PieSeries Class

Represents one-dimensional series defined by e list of data values and lists of inner and outer labels, convenient for providing data to PieRenderer.

## Syntax

```
JavaScript
// class
PieSeries.prototype = {}
```

## Inheritance Hierarchy

## See Also

PieSeries Members
MindFusion.Charting Namespace

# PieSeries Members

Represents one-dimensional series defined by e list of data values and lists of inner and outer labels, convenient for providing data to PieRenderer.

The following tables list the members exposed by the PieSeries type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | PieSeries | Overloaded. |

## Public Properties

| | Name | Description |
|---|---|---|

| | | |
|---|---|---|
| | detachedSlices | Gets or sets indices of detached slices. |
| | innerLabels | Gets or sets the inner labels. |
| | outerLabels | Gets or sets the outer labels. |
| | supportedLabels | Implements Series.SupportedLabels. Returns a combination of InnerLabel and OuterLabel. |

## Public Methods

| | Name | Description |
|---|---|---|
| | getLabel | SimpleSeries.GetLabel override. Returns an element of the inner or outer label lists. |

## See Also

PieSeries Class
MindFusion.Charting Namespace

# PieSeries Constructor

## Overload List

| Name | Description |
|---|---|
| PieSeries | Initializes a new instance of the PieSeries class. |
| PieSeries | Initializes a new instance of the **PieSeries** class. |

## See Also

PieSeries Class
MindFusion.Charting Namespace

# PieSeries Constructor

Initializes a new instance of the PieSeries class.

**Namespace:** MindFusion.Charting
**File:** PieSeries.js

## Syntax

**JavaScript**

```
function PieSeries (values, innerLabels, outerLabels)
```

### Parameters

*values*
Type: List | Array
A list of data values.

*innerLabels*

Type: List | Array
A list of inner labels.

*outerLabels*

Type: List | Array
A list of outer labels.

## Example

The following code creates a new PieSeries and sets it as the series to be rendered by an instance of the PieChart control:

**JavaScript**

```
pieChart.series = new Charting.PieSeries([20, 60, 40, 55], ["January", "February", "March",
"April"], ["January", "February", "March", "April"]);
pieChart.series.title = "Pie Series";
```

## See Also

PieSeries Constructor Overload List
PieSeries Class
MindFusion.Charting Namespace

# PieSeries Constructor

Initializes a new instance of the PieSeries class.

**Namespace:** MindFusion.Charting
**File:** PieSeries.js

## Syntax

**JavaScript**

```
function PieSeries (values, innerLabels, outerLabels)
```

### Parameters

*values*

Type: List
A list of data values.

*innerLabels*

Type: List
A list of inner labels.

*outerLabels*

Type: List
A list of outer labels.

## Example

The following code creates a new PieSeries and sets it as the series to be rendered by an instance of the PieChart control:

**JavaScript**

```
pieChart.series = new Charting.PieSeries(
    new Collections.List([20, 60, 40, 55]),
```

```
    new Collections.List(["January", "February", "March", "April"]),
    new Collections.List(["January", "February", "March", "April"]));
pieChart.series.title = "Pie Series";
```

## See Also

PieSeries Constructor Overload List
PieSeries Class
MindFusion.Charting Namespace

# PieSeries Methods

## PieSeries.getLabel Method

SimpleSeries.GetLabel override. Returns an element of the inner or outer label lists.

**Namespace:** MindFusion.Charting
**File:** PieSeries.js

## Syntax

**JavaScript**

```
function getLabel (index, kind)
```

### Parameters

*index*
Type: Number
An integer value specifying the index of a data item.

*kind*
Type: LabelKinds
A member of the LabelKinds enumeration.

### Return Value

Type: String
A string containing the item's label.

## See Also

PieSeries Members
PieSeries Class
MindFusion.Charting Namespace

# PieSeries Properties

## PieSeries.detachedSlices Property

Gets or sets indices of detached slices.

**Namespace:** MindFusion.Charting
**File:** PieSeries.js

## Syntax

```
JavaScript
get detachedSlices() {}
```

### Property Value

Type: List

## See Also

PieSeries Members
PieSeries Class
MindFusion.Charting Namespace

# PieSeries.innerLabels Property

Gets or sets the inner labels.

**Namespace:** MindFusion.Charting
**File:** PieSeries.js

## Syntax

```
JavaScript
get innerLabels() {}
```

### Property Value

Type: List

## See Also

PieSeries Members
PieSeries Class
MindFusion.Charting Namespace

# PieSeries.outerLabels Property

Gets or sets the outer labels.

**Namespace:** MindFusion.Charting
**File:** PieSeries.js

## Syntax

```
JavaScript
get outerLabels() {}
```

### Property Value

Type: List

## See Also

PieSeries Members
PieSeries Class
MindFusion.Charting Namespace

## PieSeries.supportedLabels Property

Implements Series.SupportedLabels. Returns a combination of InnerLabel and OuterLabel.

**Namespace:** MindFusion.Charting
**File:** PieSeries.js

### Syntax

**JavaScript**

```
get supportedLabels() {}
```

#### Property Value

Type: LabelKinds

### See Also

PieSeries Members
PieSeries Class
MindFusion.Charting Namespace

# Plot Class

The base class for components used to draw chart graphics.

### Syntax

**JavaScript**

```
// class
Plot.prototype = {}
```

### Inheritance Hierarchy

### See Also

Plot Members
MindFusion.Charting Namespace

## Plot Members

The base class for components used to draw chart graphics.

The following tables list the members exposed by the Plot type.

### Public Properties

| | Name | Description |
|---|---|---|

| | | |
|---|---|---|
| | background | Gets or sets the Brush that should be used to fill the background of this plot. |
| | borderStroke | Gets or sets the Brush that should be used to stroke the borders of this plot. |
| | borderStrokeDashStyle | Gets or sets the stroke dash style of plot borders. |
| | borderStrokeThickness | Gets or sets the stroke thickness of plot borders. |
| | highlightedItem | Gets or sets the data item that should be drawn highlighted. |
| | highlightStroke | Gets or sets the Brush used to stroke highlighted item. |
| | highlightStrokeDashStyle | Gets or sets the stroke dash style of highlighted item. |
| | highlightStrokeThickness | Gets or sets the stroke thickness of highlighted item. |
| | seriesRenderers | Gets or sets the SeriesRenderer objects that should draw inside this plot. |
| | seriesStyle | Gets or sets a SeriesStyle whose attributes should be used to draw series inside this plot, unless their own renderers have a local SeriesStyle value set for respective attribute. |

## Public Methods

| | Name | Description |
|---|---|---|
| | adjustDataRanges | Called at the end of data measurement pass. |
| | domainDimension | Implements the SeriesContainer interface. Gets the index of the dimension whose set of values should be treated as domain of plotted function. |
| | draw | Component.Draw override. Draws a grid and all chart graphics represented by contained SeriesRenderer objects. |
| | drawGrid | Draws the plot's grid. |
| | drawOrigins | Draws the **plot**'s axis origin lines. |
| | drawPlot | Draws the plot's background and borders. |
| | effectiveBorder | Gets the Pen that should be used to draw the plot's borders. |
| | effectiveBorderStroke | Gets the Brush that should be used to stroke the plot's borders. |
| | effectiveBorderStrokeThickness | Gets the thickness of border strokes. |
| | effectiveFill | Gets the Brush that should be used to fill the plot's background. |
| | effectiveStrokeDashStyle | Gets the dash style of border strokes. |
| | enumSeries | Implements SeriesContainer. |
| | measure | Component.Measure override. Measures the desired size of this component. |
| | measureDataRanges | Measures data of all SeriesRenderer objects drawn inside this plot. |

| | | |
|---|---|---|
| | onRendererDataChanged | Called when a Series raises its DataChanged event. |
| | startMeasureData | Called in the beginning of data measurement pass. |
| | unsubscribe | Unsubscribes from DataChanged and PropertyChanged events of all SeriesRenderer objects in specified list. |

## See Also

Plot Class
MindFusion.Charting Namespace

# Plot Methods

## Plot.adjustDataRanges Method

Called at the end of data measurement pass.

**Namespace:** MindFusion.Charting
**File:** Plot.js

## Syntax

**JavaScript**

```
function adjustDataRanges (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

Plot Members
Plot Class
MindFusion.Charting Namespace

## Plot.domainDimension Method

Implements the SeriesContainer interface. Gets the index of the dimension whose set of values should be treated as domain of plotted function.

**Namespace:** MindFusion.Charting
**File:** Plot.js

## Syntax

**JavaScript**

```
function domainDimension (series)
```

### Parameters

*series*

Type: Series
A Series in this container.

## See Also

Plot Members
Plot Class
MindFusion.Charting Namespace

# Plot.draw Method

Component.Draw override. Draws a grid and all chart graphics represented by contained SeriesRenderer objects.

**Namespace:** MindFusion.Charting
**File:** Plot.js

## Syntax

**JavaScript**

```
function draw (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

Plot Members
Plot Class
MindFusion.Charting Namespace

# Plot.drawGrid Method

Draws the plot's grid.

**Namespace:** MindFusion.Charting
**File:** Plot.js

## Syntax

**JavaScript**

```
function drawGrid (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

Plot Members
Plot Class
MindFusion.Charting Namespace

# Plot.drawOrigins Method

Draws the plot's axis origin lines.

**Namespace:** MindFusion.Charting
**File:** Plot.js

## Syntax

```
JavaScript
function drawOrigins (context)
```

### Parameters

*context*
An instance of the RenderContext class.

## Remarks

Use the origin property of Axis to specify the location of the axis origin, if different from 0.

## See Also

Plot Members
Plot Class
MindFusion.Charting Namespace

# Plot.drawPlot Method

Draws the plot's background and borders.

**Namespace:** MindFusion.Charting
**File:** Plot.js

## Syntax

```
JavaScript
function drawPlot (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

Plot Members
Plot Class
MindFusion.Charting Namespace

# Plot.effectiveBorder Method

Gets the Pen that should be used to draw the plot's borders.

**Namespace:** MindFusion.Charting
**File:** Plot.js

## Syntax

**JavaScript**

```
function effectiveBorder (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: Pen
A Pen instance.

## See Also

Plot Members
Plot Class
MindFusion.Charting Namespace

# Plot.effectiveBorderStroke Method

Gets the Brush that should be used to stroke the plot's borders.

**Namespace:** MindFusion.Charting
**File:** Plot.js

## Syntax

**JavaScript**

```
function effectiveBorderStroke (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: Brush
A Brush instance.

## See Also

Plot Members
Plot Class

# Plot.effectiveBorderStrokeThickness Method

Gets the thickness of border strokes.

**Namespace:** MindFusion.Charting
**File:** Plot.js

## Syntax

**JavaScript**

```
function effectiveBorderStrokeThickness (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: Number
A number value.

## See Also

Plot Members
Plot Class
MindFusion.Charting Namespace

# Plot.effectiveFill Method

Gets the Brush that should be used to fill the plot's background.

**Namespace:** MindFusion.Charting
**File:** Plot.js

## Syntax

**JavaScript**

```
function effectiveFill (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: Brush
A Brush instance.

## See Also

Plot Members
Plot Class
MindFusion.Charting Namespace

# Plot.effectiveStrokeDashStyle Method

Gets the dash style of border strokes.

**Namespace:** MindFusion.Charting
**File:** Plot.js

## Syntax

```
JavaScript
function effectiveStrokeDashStyle (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: DashStyle
A member of the DashStyle enumeration.

## See Also

Plot Members
Plot Class
MindFusion.Charting Namespace

# Plot.enumSeries Method

Implements SeriesContainer.

**Namespace:** MindFusion.Charting
**File:** Plot.js

## Syntax

```
JavaScript
function enumSeries ()
```

### Return Value

Type: IEnumerable
An instance of the IEnumerable<Series> class.

## See Also

Plot Members
Plot Class
MindFusion.Charting Namespace

# Plot.measure Method

Component.Measure override. Measures the desired size of this component.

**Namespace:** MindFusion.Charting
**File:** Plot.js

## Syntax

```
JavaScript
function measure (maxWidth, maxHeight, context)
```

### Parameters

*maxWidth*
Type: Number
The maximum width provided by parent component.

*maxHeight*
Type: Number
The maximum height provided by parent component.

*context*
Type: RenderContext
A RenderContext instance.

## See Also

Plot Members
Plot Class
MindFusion.Charting Namespace

# Plot.measureDataRanges Method

Measures data of all SeriesRenderer objects drawn inside this plot.

**Namespace:** MindFusion.Charting
**File:** Plot.js

## Syntax

```
JavaScript
function measureDataRanges (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

Plot Members
Plot Class
MindFusion.Charting Namespace

# Plot.onRendererDataChanged Method

Called when a Series raises its DataChanged event.

**Namespace:** MindFusion.Charting
**File:** Plot.js

## Syntax

**JavaScript**

```
function onRendererDataChanged (sender, e)
```

### Parameters

*sender*
Type: Object
The event sender.

*e*
Type: EventArgs
An EventArgs instance.

## See Also

Plot Members
Plot Class
MindFusion.Charting Namespace

# Plot.startMeasureData Method

Called in the beginning of data measurement pass.

**Namespace:** MindFusion.Charting
**File:** Plot.js

## Syntax

**JavaScript**

```
function startMeasureData (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

Plot Members
Plot Class
MindFusion.Charting Namespace

## Plot.unsubscribe Method

Unsubscribes from DataChanged and PropertyChanged events of all SeriesRenderer objects in specified list.

**Namespace:** MindFusion.Charting
**File:** Plot.js

### Syntax

```
JavaScript
function unsubscribe (seriesRenderers)
```

#### Parameters

*seriesRenderers*
Type: ObservableCollection
A list of SeriesRenderer objects.

### See Also

Plot Members
Plot Class
MindFusion.Charting Namespace

## Plot Properties

## Plot.background Property

Gets or sets the Brush that should be used to fill the background of this plot.

**Namespace:** MindFusion.Charting
**File:** Plot.js

### Syntax

```
JavaScript
get background() {}
```

#### Property Value

Type: Brush

### See Also

Plot Members
Plot Class
MindFusion.Charting Namespace

# Plot.borderStroke Property

Gets or sets the Brush that should be used to stroke the borders of this plot.

**Namespace:** MindFusion.Charting
**File:** Plot.js

## Syntax

```
JavaScript
get borderStroke() {}
```

### Property Value

Type: Brush

## See Also

Plot Members
Plot Class
MindFusion.Charting Namespace

# Plot.borderStrokeDashStyle Property

Gets or sets the stroke dash style of plot borders.

**Namespace:** MindFusion.Charting
**File:** Plot.js

## Syntax

```
JavaScript
get borderStrokeDashStyle() {}
```

### Property Value

Type: DashStyle

## See Also

Plot Members
Plot Class
MindFusion.Charting Namespace

# Plot.borderStrokeThickness Property

Gets or sets the stroke thickness of plot borders.

**Namespace:** MindFusion.Charting
**File:** Plot.js

## Syntax

```
JavaScript
```

```
get borderStrokeThickness() {}
```

### Property Value

Type: Number

## See Also

Plot Members
Plot Class
MindFusion.Charting Namespace

# Plot.highlightedItem Property

Gets or sets the data item that should be drawn highlighted.

**Namespace:** MindFusion.Charting
**File:** Plot.js

## Syntax

**JavaScript**

```
get highlightedItem() {}
```

### Property Value

Type: HitResult

## See Also

Plot Members
Plot Class
MindFusion.Charting Namespace

# Plot.highlightStroke Property

Gets or sets the Brush used to stroke highlighted item.

**Namespace:** MindFusion.Charting
**File:** Plot.js

## Syntax

**JavaScript**

```
get highlightStroke() {}
```

### Property Value

Type: Brush

## See Also

Plot Members
Plot Class

MindFusion.Charting Namespace

## Plot.highlightStrokeDashStyle Property

Gets or sets the stroke dash style of highlighted item.

**Namespace:** MindFusion.Charting
**File:** Plot.js

### Syntax

**JavaScript**

```
get highlightStrokeDashStyle() {}
```

#### Property Value

Type: DashStyle

### See Also

Plot Members
Plot Class
MindFusion.Charting Namespace

## Plot.highlightStrokeThickness Property

Gets or sets the stroke thickness of highlighted item.

**Namespace:** MindFusion.Charting
**File:** Plot.js

### Syntax

**JavaScript**

```
get highlightStrokeThickness() {}
```

#### Property Value

Type: Number

### See Also

Plot Members
Plot Class
MindFusion.Charting Namespace

## Plot.seriesRenderers Property

Gets or sets the SeriesRenderer objects that should draw inside this plot.

**Namespace:** MindFusion.Charting
**File:** Plot.js

### Syntax

```
JavaScript
get seriesRenderers() {}
```

### Property Value

Type: ObservableCollection

## See Also

Plot Members
Plot Class
MindFusion.Charting Namespace

# Plot.seriesStyle Property

Gets or sets a SeriesStyle whose attributes should be used to draw series inside this plot, unless their own renderers have a local SeriesStyle value set for respective attribute.

**Namespace:** MindFusion.Charting
**File:** Plot.js

## Syntax

```
JavaScript
get seriesStyle() {}
```

### Property Value

Type: SeriesStyle

## See Also

Plot Members
Plot Class
MindFusion.Charting Namespace

# Plot2D Class

A plot whose series are rendered relatively to a two-dimensional Cartesian coordinate system.

## Syntax

```
JavaScript
// class
Plot2D.prototype = {}
```

## Inheritance Hierarchy

## See Also

# Plot2D Members

A plot whose series are rendered relatively to a two-dimensional Cartesian coordinate system.

The following tables list the members exposed by the Plot2D type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | Plot2D | Initializes a new instance of the Plot2D class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | allowPan | Gets or sets a value indicating whether users are allowed to pan this plot. |
| | gridColor1 | Gets or sets the main color of the grid. |
| | gridColor2 | Gets or sets the alternating color of the grid. |
| | gridType | Gets or sets the type of grid to draw in this plot. |
| | isZoomed | Gets or sets a value indicating whether the user has zoomed into this plot. |
| | pinGrid | Gets or sets a value indicating whether grid stripes should be pinned in place or scroll together with the plot when users pan it. |
| | verticalScroll | Specifies whether the plot should scroll vertically when panned. |
| | xAxis | Gets or sets default Axis instance used to map X data coordinates of series rendered inside this plot to the plot's pixels. |
| | yAxis | Gets or sets default Axis instance used to map Y data coordinates of series rendered inside this plot to the plot's pixels. |
| | zoomChanged | Raised when the user zooms into this plot's data range. |

## Public Methods

| | Name | Description |
|---|---|---|
| | createController | Component.CreateController override. Returns a ComponentController used to interact with this component. |
| | drawGrid | Plot.DrawGrid override. Draws the grid specified by GridType property. |
| | drawOrigins | Plot.DrawOrigins override. Draws the plot's axis origin lines. |

| | | |
|---|---|---|
| | resetZoom | Resets the zoom level to original axis ranges. |
| | visit | Component.Visit override. Calls visitor's VisitPlot method. |
| | zoomOut | Zooms out from current data range. |

## See Also

Plot2D Class
MindFusion.Charting Namespace

# Plot2D Constructor

Initializes a new instance of the Plot2D class.

**Namespace:** MindFusion.Charting
**File:** Plot2D.js

## Syntax

**JavaScript**

```
function Plot2D ()
```

## See Also

Plot2D Class
MindFusion.Charting Namespace

# Plot2D Methods

# Plot2D.createController Method

Component.CreateController override. Returns a ComponentController used to interact with this component.

**Namespace:** MindFusion.Charting
**File:** Plot2D.js

## Syntax

**JavaScript**

```
function createController (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: ComponentController
An instance of the PanController class.

## See Also

Plot2D Members
Plot2D Class
MindFusion.Charting Namespace

# Plot2D.drawGrid Method

Plot.DrawGrid override. Draws the grid specified by GridType property.

**Namespace:** MindFusion.Charting
**File:** Plot2D.js

## Syntax

**JavaScript**

```
function drawGrid (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

Plot2D Members
Plot2D Class
MindFusion.Charting Namespace

# Plot2D.drawOrigins Method

Plot.DrawOrigins override. Draws the plot's axis origin lines.

**Namespace:** MindFusion.Charting
**File:** Plot2D.js

## Syntax

**JavaScript**

```
function drawOrigins (context)
```

### Parameters

*context*
An instance of the RenderContext class.

## Remarks

Use the origin property of the Axis class to specify the origin of an axis, if different from 0.

## See Also

Plot2D Members
Plot2D Class
MindFusion.Charting Namespace

## Plot2D.resetZoom Method

Resets the zoom level to original axis ranges.

**Namespace:** MindFusion.Charting
**File:** Plot2D.js

### Syntax

**JavaScript**

```
function resetZoom ()
```

### See Also

Plot2D Members
Plot2D Class
MindFusion.Charting Namespace

## Plot2D.visit Method

Component.Visit override. Calls visitor's VisitPlot method.

**Namespace:** MindFusion.Charting
**File:** Plot2D.js

### Syntax

**JavaScript**

```
function visit (visitor)
```

#### Parameters

*visitor*
Type: ComponentVisitor
An instance of a ComponentVisitor -derived class.

### See Also

Plot2D Members
Plot2D Class
MindFusion.Charting Namespace

## Plot2D.zoomOut Method

Zooms out from current data range.

**Namespace:** MindFusion.Charting
**File:** Plot2D.js

### Syntax

JavaScript

```
function zoomOut (axes)
```

### Parameters

*axes*
Type: IEnumerable
The axes whose ranges should be scaled.

## See Also

Plot2D Members
Plot2D Class
MindFusion.Charting Namespace

# Plot2D Properties

# Plot2D.allowPan Property

Gets or sets a value indicating whether users are allowed to pan this plot.

**Namespace:** MindFusion.Charting
**File:** Plot2D.js

## Syntax

JavaScript

```
get allowPan() {}
```

### Property Value

Type: Boolean

## See Also

Plot2D Members
Plot2D Class
MindFusion.Charting Namespace

# Plot2D.gridColor1 Property

Gets or sets the main color of the grid.

**Namespace:** MindFusion.Charting
**File:** Plot2D.js

## Syntax

JavaScript

```
get gridColor1() {}
```

### Property Value

Type: Color

## See Also

Plot2D Members
Plot2D Class
MindFusion.Charting Namespace

# Plot2D.gridColor2 Property

Gets or sets the alternating color of the grid.

**Namespace:** MindFusion.Charting
**File:** Plot2D.js

## Syntax

**JavaScript**

```
get gridColor2() {}
```

### Property Value

Type: Color

## See Also

Plot2D Members
Plot2D Class
MindFusion.Charting Namespace

# Plot2D.gridType Property

Gets or sets the type of grid to draw in this plot.

**Namespace:** MindFusion.Charting
**File:** Plot2D.js

## Syntax

**JavaScript**

```
get gridType() {}
```

### Property Value

Type: GridType

## See Also

Plot2D Members
Plot2D Class
MindFusion.Charting Namespace

## Plot2D.isZoomed Property

Gets or sets a value indicating whether the user has zoomed into this plot.

**Namespace:** MindFusion.Charting
**File:** Plot2D.js

### Syntax

```
JavaScript
get isZoomed() {}
```

#### Property Value

Type: Boolean

### See Also

Plot2D Members
Plot2D Class
MindFusion.Charting Namespace

## Plot2D.pinGrid Property

Gets or sets a value indicating whether grid stripes should be pinned in place or scroll together with the plot when users pan it.

**Namespace:** MindFusion.Charting
**File:** Plot2D.js

### Syntax

```
JavaScript
get pinGrid() {}
```

#### Property Value

Type: Boolean

### See Also

Plot2D Members
Plot2D Class
MindFusion.Charting Namespace

## Plot2D.verticalScroll Property

Specifies whether the plot should scroll vertically when panned.

**Namespace:** MindFusion.Charting
**File:** Plot2D.js

### Syntax

```
JavaScript
```

```
get verticalScroll() {}
```

### Property Value

Type: Boolean

## See Also

Plot2D Members
Plot2D Class
MindFusion.Charting Namespace

# Plot2D.xAxis Property

Gets or sets default Axis instance used to map X data coordinates of series rendered inside this plot to the plot's pixels.

**Namespace:** MindFusion.Charting
**File:** Plot2D.js

## Syntax

**JavaScript**

```
get xAxis() {}
```

### Property Value

Type: Axis

## See Also

Plot2D Members
Plot2D Class
MindFusion.Charting Namespace

# Plot2D.yAxis Property

Gets or sets default Axis instance used to map Y data coordinates of series rendered inside this plot to the plot's pixels.

**Namespace:** MindFusion.Charting
**File:** Plot2D.js

## Syntax

**JavaScript**

```
get yAxis() {}
```

### Property Value

Type: Axis

## See Also

Plot2D Members
Plot2D Class
MindFusion.Charting Namespace

## Plot2D.zoomChanged Property

Raised when the user zooms into this plot's data range.

**Namespace:** MindFusion.Charting
**File:** Plot2D.js

### Syntax

**JavaScript**

```
get zoomChanged() {}
```

#### Property Value

Type: EventDispatcher

### See Also

Plot2D Members
Plot2D Class
MindFusion.Charting Namespace

# Plot2DController Class

A base class for controllers that let users interact with Plot2D objects.

### Syntax

**JavaScript**

```
// class
Plot2DController.prototype = {}
```

### Inheritance Hierarchy

### See Also

Plot2DController Members
MindFusion.Charting Namespace

## Plot2DController Members

A base class for controllers that let users interact with Plot2D objects.

The following tables list the members exposed by the Plot2DController type.

### Public Constructors

| | Name | Description |
|---|---|---|

| | | |
|---|---|---|
| | Plot2DController | Initializes a new instance of the Plot2DController class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | axisRanges | Contains information about Axis ranges processed by this controller. |
| | vertical | Gets or sets a value indicating whether interaction is done along the X or Y axis. |

## Public Methods

| | Name | Description |
|---|---|---|
| | onMouseDown | PlotController.OnMouseDown override. Called when the user presses a mouse button. |
| | plotToAxisValue | Converts a plot pixel position to a logical value from specified axis' coordinate system. |

## See Also

Plot2DController Class
MindFusion.Charting Namespace

# Plot2DController Constructor

Initializes a new instance of the Plot2DController class.

**Namespace:** MindFusion.Charting
**File:** Plot2DController.js

## Syntax

**JavaScript**

```
function Plot2DController (renderContext, vertical)
```

### Parameters

*renderContext*
Type: RenderContext
A RenderContext instance.

*vertical*
Type: Boolean
true to implement interaction along the Y axis, or false for the X axis.

## See Also

Plot2DController Class
MindFusion.Charting Namespace

# Plot2DController Methods

## Plot2DController.onMouseDown Method

PlotController.OnMouseDown override. Called when the user presses a mouse button.

**Namespace:** MindFusion.Charting
**File:** Plot2DController.js

## Syntax

**JavaScript**

```
function onMouseDown (x, y)
```

### Parameters

*x*
Type: Number
A number value specifying the horizontal position of mouse pointer.

*y*
Type: Number
A number value specifying the vertical position of mouse pointer.

## See Also

Plot2DController Members
Plot2DController Class
MindFusion.Charting Namespace

## Plot2DController.plotToAxisValue Method

Converts a plot pixel position to a logical value from specified axis' coordinate system.

**Namespace:** MindFusion.Charting
**File:** Plot2DController.js

## Syntax

**JavaScript**

```
function plotToAxisValue (axis, value)
```

### Parameters

*axis*
Type: Axis
The target Axis.

*value*
Type: Number
The value to convert.

### Return Value

Type: Number
The converted value.

## See Also

Plot2DController Members
Plot2DController Class
MindFusion.Charting Namespace

# Plot2DController Properties

## Plot2DController.axisRanges Property

Contains information about Axis ranges processed by this controller.

**Namespace:** MindFusion.Charting
**File:** Plot2DController.js

## Syntax

```
JavaScript
get axisRanges() {}
```

### Property Value

Type: Dictionary

## See Also

Plot2DController Members
Plot2DController Class
MindFusion.Charting Namespace

## Plot2DController.vertical Property

Gets or sets a value indicating whether interaction is done along the X or Y axis.

**Namespace:** MindFusion.Charting
**File:** Plot2DController.js

## Syntax

```
JavaScript
get vertical() {}
```

### Property Value

Type: Boolean

## See Also

Plot2DController Members
Plot2DController Class
MindFusion.Charting Namespace

# Plot3D Class

A plot whose series are rendered relatively to a three-dimensional Cartesian coordinate system.

## Syntax

**JavaScript**

```
// class
Plot3D.prototype = {}
```

## Inheritance Hierarchy

## See Also

Plot3D Members
MindFusion.Charting Namespace

# Plot3D Members

A plot whose series are rendered relatively to a three-dimensional Cartesian coordinate system.

The following tables list the members exposed by the Plot3D type.

## Public Properties

| | Name | Description |
|---|---|---|
|  | zAxis | Gets or sets default Axis instance used to map Z data coordinates of series rendered inside this plot to the plot's pixels. |

## Public Methods

| | Name | Description |
|---|---|---|
|  | draw | Plot.Draw override. Draws a projection of 3D models generated by contained Renderer3D objects. |

## See Also

Plot3D Class
MindFusion.Charting Namespace

# Plot3D Methods

## Plot3D.draw Method

Plot.Draw override. Draws a projection of 3D models generated by contained Renderer3D objects.

**Namespace:** MindFusion.Charting
**File:** Plot3D.js

## Syntax

JavaScript

```
function draw (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

Plot3D Members
Plot3D Class
MindFusion.Charting Namespace

# Plot3D Properties

## Plot3D.zAxis Property

Gets or sets default Axis instance used to map Z data coordinates of series rendered inside this plot to the plot's pixels.

**Namespace:** MindFusion.Charting
**File:** Plot3D.js

## Syntax

JavaScript

```
get zAxis() {}
```

### Property Value

Type: Axis

## See Also

Plot3D Members
Plot3D Class
MindFusion.Charting Namespace

# PlotController Class

A base class for controllers implementing user interaction with Plot components.

## Syntax

JavaScript

```
// class
PlotController.prototype = {}
```

## Inheritance Hierarchy

## See Also

PlotController Members
MindFusion.Charting Namespace

# PlotController Members

A base class for controllers implementing user interaction with Plot components.

The following tables list the members exposed by the PlotController type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | PlotController | Initializes a new instance of the PlotController class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | component | Gets the component controlled by this PlotController. |

## Public Methods

| | Name | Description |
|---|---|---|
| | drawInteraction | Implements ComponentController.DrawInteraction. Draws a representation of the current state of user interaction on specified Graphics surface. |
| | getCursorHint | For internal use. |
| | getRunningAnimation | For internal use. |
| | onMouseDown | Implements ComponentController.OnMouseDown. Called when the user presses a mouse button. |
| | onMouseMove | Implements ComponentController.OnMouseMove. Called when the user moves the mouse. |
| | onMouseUp | Implements ComponentController.OnMouseUp. Called when the user releases a mouse button. |
| | onMouseWheel | Implements ComponentController.onMouseWheel. Called when the user scrolls with the middle mouse button. |

## See Also

PlotController Class
MindFusion.Charting Namespace

# PlotController Constructor

Initializes a new instance of the PlotController class.

**Namespace:** MindFusion.Charting
**File:** PlotController.js

## Syntax

```
JavaScript
function PlotController (renderContext)
```

### Parameters

*renderContext*
Type: RenderContext
A RenderContext instance.

## See Also

PlotController Class
MindFusion.Charting Namespace

# PlotController Methods

# PlotController.drawInteraction Method

Implements ComponentController.DrawInteraction. Draws a representation of the current state of user interaction on specified Graphics surface.

**Namespace:** MindFusion.Charting
**File:** PlotController.js

## Syntax

```
JavaScript
function drawInteraction (graphics)
```

### Parameters

*graphics*
A Graphics instance.

## See Also

PlotController Members
PlotController Class
MindFusion.Charting Namespace

# PlotController.getCursorHint Method

For internal use.

**Namespace:** MindFusion.Charting
**File:** PlotController.js

## Syntax

```
JavaScript
function getCursorHint (x, y)
```

### Parameters

*x*
Type: void

*y*
Type: void

### Return Value

Type: CursorHint
A member of the CursorHint enumeration.

## See Also

PlotController Members
PlotController Class
MindFusion.Charting Namespace

# PlotController.getRunningAnimation Method

For internal use.

**Namespace:** MindFusion.Charting
**File:** PlotController.js

## Syntax

```
JavaScript
function getRunningAnimation ()
```

### Return Value

Type: ComponentAnimation
An instance of a ComponentAnimation -derived class.

## See Also

PlotController Members
PlotController Class
MindFusion.Charting Namespace

# PlotController.onMouseDown Method

Implements ComponentController.OnMouseDown. Called when the user presses a mouse button.

**Namespace:** MindFusion.Charting
**File:** PlotController.js

## Syntax

```
JavaScript
```

```
function onMouseDown (x, y)
```

### Parameters

*x*
Type: Number
A number value specifying the horizontal position of mouse pointer.

*y*
Type: Number
A number value specifying the vertical position of mouse pointer.

## See Also

PlotController Members
PlotController Class
MindFusion.Charting Namespace

# PlotController.onMouseMove Method

Implements ComponentController.OnMouseMove. Called when the user moves the mouse.

**Namespace:** MindFusion.Charting
**File:** PlotController.js

## Syntax

**JavaScript**

```
function onMouseMove (x, y)
```

### Parameters

*x*
Type: Number
A number value specifying the horizontal position of mouse pointer.

*y*
Type: Number
A number value specifying the vertical position of mouse pointer.

## See Also

PlotController Members
PlotController Class
MindFusion.Charting Namespace

# PlotController.onMouseUp Method

Implements ComponentController.OnMouseUp. Called when the user releases a mouse button.

**Namespace:** MindFusion.Charting
**File:** PlotController.js

## Syntax

**JavaScript**

```
function onMouseUp (x, y)
```

### Parameters

*x*
Type: Number
A number value specifying the horizontal position of mouse pointer.

*y*
Type: Number
A number value specifying the vertical position of mouse pointer.

## See Also

PlotController Members
PlotController Class
MindFusion.Charting Namespace

# PlotController.onMouseWheel Method

Implements ComponentController.onMouseWheel. Called when the user scrolls with the middle mouse button.

**Namespace:** MindFusion.Charting
**File:** PlotController.js

## Syntax

**JavaScript**

```
function onMouseWheel (x, y, delta)
```

### Parameters

*x*
Type: Number
A number value specifying the horizontal position of mouse pointer.

*y*
Type: Number
A number value specifying the vertical position of mouse pointer.

*delta*
Type: Number
A number value specifying the amount and the direction for the mouse scroll.

## See Also

PlotController Members
PlotController Class
MindFusion.Charting Namespace

# PlotController Properties

# PlotController.component Property

Gets the component controlled by this PlotController.

**Namespace:** MindFusion.Charting
**File:** PlotController.js

## Syntax

```
JavaScript
get component() {}
```

### Property Value

Type: Component

## See Also

PlotController Members
PlotController Class
MindFusion.Charting Namespace

# Point3D Class

Represents a point in 3D space.

## Syntax

```
JavaScript
// class
Point3D.prototype = {}
```

## Inheritance Hierarchy

## See Also

Point3D Members
MindFusion.Charting Namespace

# Point3D Members

Represents a point in 3D space.

The following tables list the members exposed by the Point3D type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | Point3D | Initializes a new instance of the Point3D structure. |

## Public Properties

| | Name | Description |
|---|---|---|
| | x | Gets or sets the X coordinate of this point. |

| | | |
|---|---|---|
| | y | Gets or sets the Y coordinate of this point. |
| | z | Gets or sets the Z coordinate of this point. |

## Public Methods

| | Name | Description |
|---|---|---|
| | add | Sums two points' coordinates. |
| | distance | Gets the distance to specified point. |
| | scale | Scales the point's coordinates by specified scale factor. |
| | toString | Returns a string representation of this point. |

## See Also

Point3D Class
MindFusion.Charting Namespace

# Point3D Constructor

Initializes a new instance of the Point3D structure.

**Namespace:** MindFusion.Charting
**File:** Point3D.js

## Syntax

**JavaScript**

```
function Point3D (x, y, z)
```

### Parameters

*x*
Type: Number
X coordinate of the point.

*y*
Type: Number
Y coordinate of the point.

*z*
Type: Number
Z coordinate of the point.

## See Also

Point3D Class
MindFusion.Charting Namespace

# Point3D Methods

# Point3D.add Method

Sums two points' coordinates.

**Namespace:** MindFusion.Charting
**File:** Point3D.js

## Syntax

```
JavaScript
function add (p1, p2)
```

### Parameters

*p1*
Type: Point3D
The first point.

*p2*
Type: Point3D
The second point.

### Return Value

Type: Point3D
A Point3D containing sum of coordinates.

## See Also

Point3D Members
Point3D Class
MindFusion.Charting Namespace

# Point3D.distance Method

Gets the distance to specified point.

**Namespace:** MindFusion.Charting
**File:** Point3D.js

## Syntax

```
JavaScript
function distance (p)
```

### Parameters

*p*
Type: Point3D
A Point3D instance.

### Return Value

Type: Number
A number value representing the distance.

## See Also

Point3D Members
Point3D Class
MindFusion.Charting Namespace

# Point3D.scale Method

Scales the point's coordinates by specified scale factor.

**Namespace:** MindFusion.Charting
**File:** Point3D.js

## Syntax

```
JavaScript
function scale (scale)
```

### Parameters

*scale*
Type: Number
A number value by which to multiply point's coordinates.

### Return Value

Type: Point3D
A Point3D containing scaled coordinates.

## See Also

Point3D Members
Point3D Class
MindFusion.Charting Namespace

# Point3D.toString Method

Returns a string representation of this point.

**Namespace:** MindFusion.Charting
**File:** Point3D.js

## Syntax

```
JavaScript
function toString ()
```

### Return Value

Type: String
A string representation of this point.

## See Also

Point3D Members
Point3D Class
MindFusion.Charting Namespace

# Point3D Properties

## Point3D.x Property

Gets or sets the X coordinate of this point.

**Namespace:** MindFusion.Charting
**File:** Point3D.js

### Syntax

```
JavaScript
get x() {}
```

#### Property Value

Type: Number

### See Also

Point3D Members
Point3D Class
MindFusion.Charting Namespace

## Point3D.y Property

Gets or sets the Y coordinate of this point.

**Namespace:** MindFusion.Charting
**File:** Point3D.js

### Syntax

```
JavaScript
get y() {}
```

#### Property Value

Type: Number

### See Also

Point3D Members
Point3D Class
MindFusion.Charting Namespace

## Point3D.z Property

Gets or sets the Z coordinate of this point.

**Namespace:** MindFusion.Charting
**File:** Point3D.js

## Syntax

```
JavaScript
get z() {}
```

### Property Value

Type: Number

## See Also

Point3D Members
Point3D Class
MindFusion.Charting Namespace

# PointSeries Class

Represents a data series defined by a list of Point objects.

## Syntax

```
JavaScript
// class
PointSeries.prototype = {}
```

## Inheritance Hierarchy

## See Also

PointSeries Members
MindFusion.Charting Namespace

## PointSeries Members

Represents a data series defined by a list of Point objects.

The following tables list the members exposed by the PointSeries type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | PointSeries | Overloaded. |

## Public Properties

| | Name | Description |
|---|---|---|
| | dataChanged | Raised when the values in this series change. |

| | | |
|---|---|---|
| | dimensions | Implements Series.Dimensions. Returns 2. |
| | labels | Gets or sets the data labels. |
| | points | Gets or sets the series' points. |
| | size | Implements Series.Size. Returns the number of elements in Points list. |
| | supportedLabels | Implements Series.SupportedLabels. Returns InnerLabel. |
| | title | Implements Series.Title. Gets or sets the title of this series. |

## Public Methods

| | Name | Description |
|---|---|---|
| | getLabel | Implements Series.GetLabel. Returns an element of the Labels list. |
| | getValue | Implements Series.GetValue. Returns a coordinate of specified point in this series. |
| | isEmphasized | Implements Series.IsEmphasized. PointSeries always returns false. |
| | isSorted | Implements Series.IsSorted. PointSeries always returns false. |
| | onDataChanged | Raises the DataChanged event. |

## See Also

PointSeries Class
MindFusion.Charting Namespace

# PointSeries Constructor

## Overload List

| Name | Description |
|---|---|
| PointSeries | Initializes a new instance of the PointSeries class. |
| PointSeries | Initializes a new instance of the **PointSeries** class. |

## See Also

PointSeries Class
MindFusion.Charting Namespace

# PointSeries Constructor

Initializes a new instance of the PointSeries class.

**Namespace:** MindFusion.Charting
**File:** PointSeries.js

## Syntax

JavaScript

```
function PointSeries (points, labels)
```

### Parameters

*points*
Type: List | Array
A list of points.

*labels*
Type: **List** | Array
A list of labels.

## Example

The following code creates a new PointSeries using two arrays with values:

JavaScript

```
var series = new Charting.PointSeries(
 [
  new Drawing.Point(0.32, 81),
  new Drawing.Point(0.39, 66),
  new Drawing.Point(0.75, 65),
  new Drawing.Point(0.49, 60)
  ],
 ["Germany", "France", "UK", "Italy"]);
series.title = "Sales > $100K";
```

## See Also

PointSeries Constructor Overload List
PointSeries Class
MindFusion.Charting Namespace

# PointSeries Constructor

Initializes a new instance of the PointSeries class.

**Namespace:** MindFusion.Charting
**File:** PointSeries.js

## Syntax

JavaScript

```
function PointSeries (points, labels)
```

### Parameters

*points*
Type: List
A list of points.

*labels*
Type: **List**
A list of labels.

## Example

The following code creates a new PointSeries using two List-s with values:

**JavaScript**

```
var series = new Charting.PointSeries(
 new Collections.List
  ([
   new Drawing.Point(0.32, 81),
   new Drawing.Point(0.39, 66),
   new Drawing.Point(0.75, 65),
   new Drawing.Point(0.49, 60)
  ]),
 new Collections.List(["Germany", "France", "UK", "Italy"]));
series.title = "Sales > $100K";
```

## See Also

PointSeries Constructor Overload List
PointSeries Class
MindFusion.Charting Namespace

# PointSeries Methods

# PointSeries.getLabel Method

Implements Series.GetLabel. Returns an element of the Labels list.

**Namespace:** MindFusion.Charting
**File:** PointSeries.js

## Syntax

**JavaScript**

```
function getLabel (index, kind)
```

### Parameters

*index*
Type: Number
An integer value specifying the index of a data item.

*kind*
Type: LabelKinds
A member of the LabelKinds enumeration.

### Return Value

Type: String
A string containing the item's label.

## See Also

PointSeries Members
PointSeries Class
MindFusion.Charting Namespace

# PointSeries.getValue Method

Implements Series.GetValue. Returns a coordinate of specified point in this series.

**Namespace:** MindFusion.Charting
**File:** PointSeries.js

## Syntax

**JavaScript**

```
function getValue (index, dimension)
```

### Parameters

*index*
Type: Number
An integer value specifying the index of a data item.

*dimension*
Type: Number
An integer value specifying the dimension whose coordinate to return.

### Return Value

Type: Number
A double-precision number representing the value of a data item in the series.

## See Also

PointSeries Members
PointSeries Class
MindFusion.Charting Namespace

# PointSeries.isEmphasized Method

Implements Series.IsEmphasized. PointSeries always returns false.

**Namespace:** MindFusion.Charting
**File:** PointSeries.js

## Syntax

**JavaScript**

```
function isEmphasized (index)
```

### Parameters

*index*
Type: Number
An integer value specifying the index of a data item.

### Return Value

Type: Boolean
true to emphasize the specified data item, or false otherwise.

## See Also

PointSeries Members
PointSeries Class
MindFusion.Charting Namespace

# PointSeries.isSorted Method

Implements Series.IsSorted. PointSeries always returns false.

**Namespace:** MindFusion.Charting
**File:** PointSeries.js

## Syntax

```
JavaScript
function isSorted (dimension)
```

### Parameters

*dimension*
Type: Number
An integer value identifying the dimension.

### Return Value

Type: Boolean
true if the values are sorted in specified dimension, or false otherwise.

## See Also

PointSeries Members
PointSeries Class
MindFusion.Charting Namespace

# PointSeries.onDataChanged Method

Raises the DataChanged event.

**Namespace:** MindFusion.Charting
**File:** PointSeries.js

## Syntax

```
JavaScript
function onDataChanged ()
```

## See Also

PointSeries Members
PointSeries Class
MindFusion.Charting Namespace

# PointSeries Properties

## PointSeries.dataChanged Property

Raised when the values in this series change.

**Namespace:** MindFusion.Charting
**File:** PointSeries.js

### Syntax

```
JavaScript
get dataChanged() {}
```

#### Property Value

Type: EventDispatcher

### See Also

PointSeries Members
PointSeries Class
MindFusion.Charting Namespace

## PointSeries.dimensions Property

Implements Series.Dimensions. Returns 2.

**Namespace:** MindFusion.Charting
**File:** PointSeries.js

### Syntax

```
JavaScript
get dimensions() {}
```

#### Property Value

Type: Number

### See Also

PointSeries Members
PointSeries Class
MindFusion.Charting Namespace

## PointSeries.labels Property

Gets or sets the data labels.

**Namespace:** MindFusion.Charting
**File:** PointSeries.js

### Syntax

```
JavaScript
```

```
get labels() {}
```

### Property Value

Type: List

## See Also

PointSeries Members
PointSeries Class
MindFusion.Charting Namespace

# PointSeries.points Property

Gets or sets the series' points.

**Namespace:** MindFusion.Charting
**File:** PointSeries.js

## Syntax

**JavaScript**

```
get points() {}
```

### Property Value

Type: List

## See Also

PointSeries Members
PointSeries Class
MindFusion.Charting Namespace

# PointSeries.size Property

Implements Series.Size. Returns the number of elements in Points list.

**Namespace:** MindFusion.Charting
**File:** PointSeries.js

## Syntax

**JavaScript**

```
get size() {}
```

### Property Value

Type: Number

## See Also

PointSeries Members
PointSeries Class

## PointSeries.supportedLabels Property

Implements Series.SupportedLabels. Returns InnerLabel.

**Namespace:** MindFusion.Charting
**File:** PointSeries.js

### Syntax

```
JavaScript
get supportedLabels() {}
```

#### Property Value

Type: LabelKinds

### See Also

PointSeries Members
PointSeries Class
MindFusion.Charting Namespace

## PointSeries.title Property

Implements Series.Title. Gets or sets the title of this series.

**Namespace:** MindFusion.Charting
**File:** PointSeries.js

### Syntax

```
JavaScript
get title() {}
```

#### Property Value

Type: String

### See Also

PointSeries Members
PointSeries Class
MindFusion.Charting Namespace

# PointSeries3D Class

Represents a data series defined by a list of Point3D objects.

### Syntax

**JavaScript**

```
// class
PointSeries3D.prototype = {}
```

## Inheritance Hierarchy

## See Also

PointSeries3D Members
MindFusion.Charting Namespace

# PointSeries3D Members

Represents a data series defined by a list of Point3D objects.

The following tables list the members exposed by the PointSeries3D type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | PointSeries3D | Overloaded. |

## Public Properties

| | Name | Description |
|---|---|---|
| | dataChanged | Raised when the values in this series change. |
| | dimensions | Implements Series.Dimensions. Returns 3. |
| | labels | Gets or sets the data labels. |
| | points | Gets or sets the series' points. |
| | size | Implements Series.Size. Returns the number of elements in Points list. |
| | supportedLabels | Implements Series.SupportedLabels. Returns InnerLabel. |
| | title | Implements Series.Title. Gets or sets the title of this series. |

## Public Methods

| | Name | Description |
|---|---|---|
| | getLabel | Implements Series.GetLabel. Returns an element of the Labels list. |
| | getValue | Implements Series.GetValue. Returns a coordinate of specified point in this series. |
| | isEmphasized | Implements Series.IsEmphasized. PointSeries3D always returns false. |
| | isSorted | Implements Series.IsSorted. PointSeries3D always returns false. |

| | | |
|---|---|---|
| | onDataChanged | Raises the DataChanged event. |

## See Also

PointSeries3D Class
MindFusion.Charting Namespace

# PointSeries3D Constructor

## Overload List

| Name | Description |
|---|---|
| PointSeries3D | Initializes a new instance of the PointSeries3D class. |
| PointSeries3D | Initializes a new instance of the PointSeries3D class. |

## See Also

PointSeries3D Class
MindFusion.Charting Namespace

# PointSeries3D Constructor

Initializes a new instance of the PointSeries3D class.

**Namespace:** MindFusion.Charting
**File:** PointSeries3D.js

## Syntax

**JavaScript**

```
function PointSeries3D (points, labels)
```

### Parameters

*points*
Type: List | Array
A list of 3D points.

*labels*
Type: List | Array
A list of labels.

## See Also

PointSeries3D Constructor Overload List
PointSeries3D Class
MindFusion.Charting Namespace

# PointSeries3D Constructor

Initializes a new instance of the PointSeries3D class.

**Namespace:** MindFusion.Charting
**File:** PointSeries3D.js

## Syntax

**JavaScript**

```
function PointSeries3D (points, labels)
```

### Parameters

*points*
Type: List
A list of 3D points.

*labels*
Type: List
A list of labels.

## See Also

PointSeries3D Constructor Overload List
PointSeries3D Class
MindFusion.Charting Namespace

# PointSeries3D Methods

# PointSeries3D.getLabel Method

Implements Series.GetLabel. Returns an element of the Labels list.

**Namespace:** MindFusion.Charting
**File:** PointSeries3D.js

## Syntax

**JavaScript**

```
function getLabel (index, kind)
```

### Parameters

*index*
Type: Number
An integer value specifying the index of a data item.

*kind*
Type: LabelKinds
A member of the LabelKinds enumeration.

### Return Value

Type: String
A string containing the item's label.

## See Also

PointSeries3D Members
PointSeries3D Class
MindFusion.Charting Namespace

# PointSeries3D.getValue Method

Implements Series.GetValue. Returns a coordinate of specified point in this series.

**Namespace:** MindFusion.Charting
**File:** PointSeries3D.js

## Syntax

**JavaScript**

```
function getValue (index, dimension)
```

### Parameters

*index*
Type: Number
An integer value specifying the index of a data item.

*dimension*
Type: Number
An integer value specifying the dimension whose coordinate to return.

### Return Value

Type: Number
A double-precision number representing the value of a data item in the series.

## See Also

PointSeries3D Members
PointSeries3D Class
MindFusion.Charting Namespace

# PointSeries3D.isEmphasized Method

Implements Series.IsEmphasized. PointSeries3D always returns false.

**Namespace:** MindFusion.Charting
**File:** PointSeries3D.js

## Syntax

**JavaScript**

```
function isEmphasized (index)
```

### Parameters

*index*

Type: Number
An integer value specifying the index of a data item.

### Return Value

Type: Boolean
true to emphasize the specified data item, or false otherwise.

## See Also

PointSeries3D Members
PointSeries3D Class
MindFusion.Charting Namespace

# PointSeries3D.isSorted Method

Implements Series.IsSorted. PointSeries3D always returns false.

**Namespace:** MindFusion.Charting
**File:** PointSeries3D.js

## Syntax

**JavaScript**

```
function isSorted (dimension)
```

### Parameters

*dimension*
Type: Number
An integer value identifying the dimension.

### Return Value

Type: Boolean
true if the values are sorted in specified dimension, or false otherwise.

## See Also

PointSeries3D Members
PointSeries3D Class
MindFusion.Charting Namespace

# PointSeries3D.onDataChanged Method

Raises the DataChanged event.

**Namespace:** MindFusion.Charting
**File:** PointSeries3D.js

## Syntax

**JavaScript**

```
function onDataChanged ()
```

## See Also

PointSeries3D Members
PointSeries3D Class
MindFusion.Charting Namespace

# PointSeries3D Properties

## PointSeries3D.dataChanged Property

Raised when the values in this series change.

**Namespace:** MindFusion.Charting
**File:** PointSeries3D.js

## Syntax

```
JavaScript
get dataChanged() {}
```

### Property Value

Type: EventDispatcher

## See Also

PointSeries3D Members
PointSeries3D Class
MindFusion.Charting Namespace

## PointSeries3D.dimensions Property

Implements Series.Dimensions. Returns 3.

**Namespace:** MindFusion.Charting
**File:** PointSeries3D.js

## Syntax

```
JavaScript
get dimensions() {}
```

### Property Value

Type: Number

## See Also

PointSeries3D Members
PointSeries3D Class
MindFusion.Charting Namespace

# PointSeries3D.labels Property

Gets or sets the data labels.

**Namespace:** MindFusion.Charting
**File:** PointSeries3D.js

## Syntax

```
JavaScript
get labels() {}
```

### Property Value

Type: List

## See Also

PointSeries3D Members
PointSeries3D Class
MindFusion.Charting Namespace

# PointSeries3D.points Property

Gets or sets the series' points.

**Namespace:** MindFusion.Charting
**File:** PointSeries3D.js

## Syntax

```
JavaScript
get points() {}
```

### Property Value

Type: List

## See Also

PointSeries3D Members
PointSeries3D Class
MindFusion.Charting Namespace

# PointSeries3D.size Property

Implements Series.Size. Returns the number of elements in Points list.

**Namespace:** MindFusion.Charting
**File:** PointSeries3D.js

## Syntax

```
JavaScript
```

```
get size() {}
```

### Property Value

Type: Number

## See Also

PointSeries3D Members
PointSeries3D Class
MindFusion.Charting Namespace

# PointSeries3D.supportedLabels Property

Implements Series.SupportedLabels. Returns InnerLabel.

**Namespace:** MindFusion.Charting
**File:** PointSeries3D.js

## Syntax

**JavaScript**

```
get supportedLabels() {}
```

### Property Value

Type: LabelKinds

## See Also

PointSeries3D Members
PointSeries3D Class
MindFusion.Charting Namespace

# PointSeries3D.title Property

Implements Series.Title. Gets or sets the title of this series.

**Namespace:** MindFusion.Charting
**File:** PointSeries3D.js

## Syntax

**JavaScript**

```
get title() {}
```

### Property Value

Type: String

## See Also

PointSeries3D Members
PointSeries3D Class

# PolarPlot Class

A plot used to draw graphics in polar coordinate system.

## Syntax

```
JavaScript
// class
PolarPlot.prototype = {}
```

## Inheritance Hierarchy

## See Also

PolarPlot Members
MindFusion.Charting Namespace

# PolarPlot Members

A plot used to draw graphics in polar coordinate system.

The following tables list the members exposed by the PolarPlot type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | PolarPlot | Initializes a new instance of the PolarPlot class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | allowRotate | Gets or sets a value indicating whether users are allowed to rotate this plot. |
|  | padding | Gets or set padding space between the plot's border and series graphics. |
|  | startAngle | Gets or set the angle where first data item of series should be drawn. |

## Public Methods

| | Name | Description |
|---|---|---|
|  | createController | Component.CreateController override. Returns a ComponentController used to interact with this component. |
|  | rotatePoint | Rotates point around specified rotation center at specified angle. |

| | | |
|---|---|---|
| | visit | Component.Visit override. Calls visitor's VisitPlot method. |

## See Also

PolarPlot Class
MindFusion.Charting Namespace

# PolarPlot Constructor

Initializes a new instance of the PolarPlot class.

**Namespace:** MindFusion.Charting
**File:** PolarPlot.js

## Syntax

**JavaScript**

```
function PolarPlot ()
```

## See Also

PolarPlot Class
MindFusion.Charting Namespace

# PolarPlot Methods

# PolarPlot.createController Method

Component.CreateController override. Returns a ComponentController used to interact with this component.

**Namespace:** MindFusion.Charting
**File:** PolarPlot.js

## Syntax

**JavaScript**

```
function createController (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: ComponentController
An instance of the RotationController class.

## See Also

PolarPlot Members
PolarPlot Class
MindFusion.Charting Namespace

# PolarPlot.rotatePoint Method

Rotates point around specified rotation center at specified angle.

**Namespace:** MindFusion.Charting
**File:** PolarPlot.js

## Syntax

```
JavaScript
function rotatePoint (pointToRotate, centerPoint, angleInDegrees)
```

### Parameters

*pointToRotate*
Type: Point
The point to rotate.

*centerPoint*
Type: Point
The rotation center.

*angleInDegrees*
Type: Number
The rotation angle.

### Return Value

Type: Point

## See Also

PolarPlot Members
PolarPlot Class
MindFusion.Charting Namespace

# PolarPlot.visit Method

Component.Visit override. Calls visitor's VisitPlot method.

**Namespace:** MindFusion.Charting
**File:** PolarPlot.js

## Syntax

```
JavaScript
function visit (visitor)
```

### Parameters

*visitor*

Type: ComponentVisitor
An instance of a ComponentVisitor -derived class.

## See Also

PolarPlot Members
PolarPlot Class
MindFusion.Charting Namespace

# PolarPlot Properties

## PolarPlot.allowRotate Property

Gets or sets a value indicating whether users are allowed to rotate this plot.

**Namespace:** MindFusion.Charting
**File:** PolarPlot.js

## Syntax

```
JavaScript
get allowRotate() {}
```

### Property Value

Type: Boolean

## See Also

PolarPlot Members
PolarPlot Class
MindFusion.Charting Namespace

## PolarPlot.padding Property

Gets or set padding space between the plot's border and series graphics.

**Namespace:** MindFusion.Charting
**File:** PolarPlot.js

## Syntax

```
JavaScript
get padding() {}
```

### Property Value

Type: Number

## See Also

PolarPlot Members
PolarPlot Class
MindFusion.Charting Namespace

## PolarPlot.startAngle Property

Gets or set the angle where first data item of series should be drawn.

**Namespace:** MindFusion.Charting
**File:** PolarPlot.js

### Syntax

```
JavaScript
get startAngle() {}
```

#### Property Value

Type: Number

### See Also

PolarPlot Members
PolarPlot Class
MindFusion.Charting Namespace

# RadarAxisOptions Class

Provides properties for customizing axis rendering in radar charts.

### Syntax

```
JavaScript
// class
RadarAxisOptions.prototype = {}
```

### Inheritance Hierarchy

### See Also

RadarAxisOptions Members
MindFusion.Charting Namespace

## RadarAxisOptions Members

Provides properties for customizing axis rendering in radar charts.

The following tables list the members exposed by the RadarAxisOptions type.

### Public Constructors

| | Name | Description |
|---|---|---|

| | | |
|---|---|---|
| | RadarAxisOptions | Initializes a new instance of the RadarAxisOptions class. |

## Public Methods

| | Name | Description |
|---|---|---|
| | draw | AxisRenderer.Draw override. |
| | effectiveAxis | AxisRenderer.EffectiveAxis override. |
| | measure | AxisRenderer.Measure override. |

## See Also

RadarAxisOptions Class
MindFusion.Charting Namespace

# RadarAxisOptions Constructor

Initializes a new instance of the RadarAxisOptions class.

**Namespace:** MindFusion.Charting
**File:** RadarAxisOptions.js

## Syntax

**JavaScript**

```
function RadarAxisOptions (axis)
```

### Parameters

*axis*
Type: void

## See Also

RadarAxisOptions Class
MindFusion.Charting Namespace

# RadarAxisOptions Methods

# RadarAxisOptions.draw Method

AxisRenderer.Draw override.

**Namespace:** MindFusion.Charting
**File:** RadarAxisOptions.js

## Syntax

**JavaScript**

```
function draw (context)
```

### Parameters

*context*
Type: void

## See Also

RadarAxisOptions Members
RadarAxisOptions Class
MindFusion.Charting Namespace

# RadarAxisOptions.effectiveAxis Method

AxisRenderer.EffectiveAxis override.

**Namespace:** MindFusion.Charting
**File:** RadarAxisOptions.js

## Syntax

**JavaScript**

```
function effectiveAxis (context)
```

### Parameters

*context*
Type: void

## See Also

RadarAxisOptions Members
RadarAxisOptions Class
MindFusion.Charting Namespace

# RadarAxisOptions.measure Method

AxisRenderer.Measure override.

**Namespace:** MindFusion.Charting
**File:** RadarAxisOptions.js

## Syntax

**JavaScript**

```
function measure (maxWidth, maxHeight, context)
```

### Parameters

*maxWidth*
Type: void

*maxHeight*
Type: void

*context*
Type: void

## See Also

RadarAxisOptions Members
RadarAxisOptions Class
MindFusion.Charting Namespace

# RadarGridType Enumeration

Identifies the grid types supported by RadarPlot.

**Namespace:** MindFusion.Charting
**File:** RadarGridType.js

## Syntax

**JavaScript**

```
// enum
RadarGridType = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **Radar** | Identifies radar grid consisting as concentric circles. |
| | **Spiderweb** | Identifies spider-web grid consisting as concentric regular polygons. |

## See Also

MindFusion.Charting Namespace

# RadarPlot Class

A plot containing radar-chart graphics.

## Syntax

**JavaScript**

```
// class
RadarPlot.prototype = {}
```

## Inheritance Hierarchy

## See Also

RadarPlot Members
MindFusion.Charting Namespace

# RadarPlot Members

A plot containing radar-chart graphics.

The following tables list the members exposed by the RadarPlot type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | RadarPlot | Initializes a new instance of the RadarPlot class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | axes | Gets the list of Axis objects representing ranges of variables represented in the radar chart. |
| | axisOptions | Gets a RadarAxisOptions object providing properties for customizing axis rendering in radar charts. |
| | defaultAxis | Sets a default Axis object used when data item index does not have corresponding element in the Axes collection. |
| | gridColor1 | Gets or sets the main color of the grid. |
| | gridColor2 | Gets or sets the alternating color of the grid. |
| | gridDivisions | Gets the number of concentric shapes to draw in the grid. |
| | gridType | Gets or sets the type of grid to draw in this plot. |
| | showCoordinates | Gets or sets a value indicating whether the grid should display axis coordinates. |
| | uniformScale | Gets or sets a value indicating whether all axes should display same data range. |

## Public Methods

| | Name | Description |
|---|---|---|
| | drawGrid | Plot.DrawGrid override. Draws spiderweb of radial grid. |
| | getEffectiveAxis | Gets the Axis representing the range for specified data variable index. |
| | startMeasureData | Plot.StartMeasureData override. Called in the beginning of data measurement pass. |

## See Also

RadarPlot Class
MindFusion.Charting Namespace

# RadarPlot Constructor

Initializes a new instance of the RadarPlot class.

**Namespace:** MindFusion.Charting
**File:** RadarPlot.js

## Syntax

```
JavaScript
function RadarPlot ()
```

## See Also

RadarPlot Class
MindFusion.Charting Namespace

# RadarPlot Methods

# RadarPlot.drawGrid Method

Plot.DrawGrid override. Draws spiderweb of radial grid.

**Namespace:** MindFusion.Charting
**File:** RadarPlot.js

## Syntax

```
JavaScript
function drawGrid (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

RadarPlot Members
RadarPlot Class
MindFusion.Charting Namespace

# RadarPlot.getEffectiveAxis Method

Gets the Axis representing the range for specified data variable index.

**Namespace:** MindFusion.Charting
**File:** RadarPlot.js

## Syntax

```
JavaScript
function getEffectiveAxis (index)
```

### Parameters

*index*
Type: Number
An integer index of data items in series.

### Return Value

Type: Axis
The associated Axis.

## See Also

RadarPlot Members
RadarPlot Class
MindFusion.Charting Namespace

# RadarPlot.startMeasureData Method

Plot.StartMeasureData override. Called in the beginning of data measurement pass.

**Namespace:** MindFusion.Charting
**File:** RadarPlot.js

## Syntax

**JavaScript**

```
function startMeasureData (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

RadarPlot Members
RadarPlot Class
MindFusion.Charting Namespace

# RadarPlot Properties

# RadarPlot.axes Property

Gets the list of Axis objects representing ranges of variables represented in the radar chart.

**Namespace:** MindFusion.Charting
**File:** RadarPlot.js

## Syntax

**JavaScript**

```
get axes() {}
```

### Property Value

Type: ObservableCollection

## See Also

RadarPlot Members
RadarPlot Class
MindFusion.Charting Namespace

# RadarPlot.axisOptions Property

Gets a RadarAxisOptions object providing properties for customizing axis rendering in radar charts.

**Namespace:** MindFusion.Charting
**File:** RadarPlot.js

## Syntax

**JavaScript**

```
get axisOptions() {}
```

### Property Value

Type: RadarAxisOptions

## See Also

RadarPlot Members
RadarPlot Class
MindFusion.Charting Namespace

# RadarPlot.defaultAxis Property

Sets a default Axis object used when data item index does not have corresponding element in the Axes collection.

**Namespace:** MindFusion.Charting
**File:** RadarPlot.js

## Syntax

**JavaScript**

```
get defaultAxis() {}
```

### Property Value

Type: Axis

## See Also

RadarPlot Members
RadarPlot Class
MindFusion.Charting Namespace

# RadarPlot.gridColor1 Property

Gets or sets the main color of the grid.

**Namespace:** MindFusion.Charting
**File:** RadarPlot.js

## Syntax

```
JavaScript
get gridColor1() {}
```

### Property Value

Type: Color

## See Also

RadarPlot Members
RadarPlot Class
MindFusion.Charting Namespace

# RadarPlot.gridColor2 Property

Gets or sets the alternating color of the grid.

**Namespace:** MindFusion.Charting
**File:** RadarPlot.js

## Syntax

```
JavaScript
get gridColor2() {}
```

### Property Value

Type: Color

## See Also

RadarPlot Members
RadarPlot Class
MindFusion.Charting Namespace

# RadarPlot.gridDivisions Property

Gets the number of concentric shapes to draw in the grid.

**Namespace:** MindFusion.Charting
**File:** RadarPlot.js

## Syntax

```
JavaScript
```

```
get gridDivisions() {}
```

### Property Value

Type: Number

## See Also

RadarPlot Members
RadarPlot Class
MindFusion.Charting Namespace

# RadarPlot.gridType Property

Gets or sets the type of grid to draw in this plot.

**Namespace:** MindFusion.Charting
**File:** RadarPlot.js

## Syntax

**JavaScript**

```
get gridType() {}
```

### Property Value

Type: RadarGridType

## See Also

RadarPlot Members
RadarPlot Class
MindFusion.Charting Namespace

# RadarPlot.showCoordinates Property

Gets or sets a value indicating whether the grid should display axis coordinates.

**Namespace:** MindFusion.Charting
**File:** RadarPlot.js

## Syntax

**JavaScript**

```
get showCoordinates() {}
```

### Property Value

Type: Boolean

## See Also

RadarPlot Members
RadarPlot Class

MindFusion.Charting Namespace

## RadarPlot.uniformScale Property

Gets or sets a value indicating whether all axes should display same data range.

**Namespace:** MindFusion.Charting
**File:** RadarPlot.js

### Syntax

**JavaScript**

```
get uniformScale() {}
```

#### Property Value

Type: Boolean

### See Also

RadarPlot Members
RadarPlot Class
MindFusion.Charting Namespace

# RadarRenderer Class

A SeriesRenderer that draws series as polygons in polar coordinate system,where adjacent data points are at equal angular distances and radial coordinates correspond to magnitude of data item values.

### Syntax

**JavaScript**

```
// class
RadarRenderer.prototype = {}
```

### Inheritance Hierarchy

### See Also

RadarRenderer Members
MindFusion.Charting Namespace

## RadarRenderer Members

A SeriesRenderer that draws series as polygons in polar coordinate system,where adjacent data points are at equal angular distances and radial coordinates correspond to magnitude of data item values.

The following tables list the members exposed by the RadarRenderer type.

### Public Constructors

| | Name | Description |
|---|---|---|

| | | |
|---|---|---|
| | RadarRenderer | Initializes a new instance of the RadarRenderer class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | areaOpacity | Gets or sets the opacity of radar polygons. |
| | series | Gets or sets a list of Series drawn by this RadarRenderer. |

## Public Methods

| | Name | Description |
|---|---|---|
| | data | Returns the data value of specified series item. |
| | draw | SeriesRenderer.Draw override. Draws the series data in specified RenderContext. |
| | drawHighlight | SeriesRenderer.DrawHighlight override. Draws highlighted data item in specified RenderContext. |
| | enumSeries | Implement SeriesContainer. |
| | enumVisibleRanges | Enumerates each series once providing a list of all visible points at each iteration. |
| | getMaxValue | Gets the maximum value from all series located at specified index. |
| | getMinValue | Gets the minimum value from all series located at specified index. |
| | hitTest | SeriesRenderer.HitTest override. Hit-tests the visual representation of the series for a data item. |
| | measureDataRange | SeriesRenderer.MeasureDataRange override. Measures the data range of rendered series and assigns it to the associated Axis objects if their MinValue and MaxValue are not set. |

## See Also

RadarRenderer Class
MindFusion.Charting Namespace

# RadarRenderer Constructor

Initializes a new instance of the RadarRenderer class.

**Namespace:** MindFusion.Charting
**File:** RadarRenderer.js

## Syntax

| JavaScript |
|---|
| `function RadarRenderer (series)` |

### Parameters

*series*
Type: ObservableCollection
A list of Series that should be rendered by this object.

## See Also

RadarRenderer Class
MindFusion.Charting Namespace

# RadarRenderer Methods

# RadarRenderer.data Method

Returns the data value of specified series item.

**Namespace:** MindFusion.Charting
**File:** RadarRenderer.js

## Syntax

**JavaScript**

```
function data (seriesIndex, dataIndex)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series within the list of series rendered by this Renderer2D.

*dataIndex*
Type: Number
An integer index of the data item in specified series.

### Return Value

Type: Number
A number value.

## See Also

RadarRenderer Members
RadarRenderer Class
MindFusion.Charting Namespace

# RadarRenderer.draw Method

SeriesRenderer.Draw override. Draws the series data in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** RadarRenderer.js

## Syntax

```
JavaScript
function draw (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

RadarRenderer Members
RadarRenderer Class
MindFusion.Charting Namespace

# RadarRenderer.drawHighlight Method

SeriesRenderer.DrawHighlight override. Draws highlighted data item in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** RadarRenderer.js

## Syntax

```
JavaScript
function drawHighlight (context, hitResult)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*hitResult*
Type: HitResult
A HitResult instance identifying highlighted data item.

## See Also

RadarRenderer Members
RadarRenderer Class
MindFusion.Charting Namespace

# RadarRenderer.enumSeries Method

Implement SeriesContainer.

**Namespace:** MindFusion.Charting
**File:** RadarRenderer.js

## Syntax

```
JavaScript
function enumSeries ()
```

### Return Value

Type: IEnumerable
An instance of the IEnumerable<Series> class.

## See Also

RadarRenderer Members
RadarRenderer Class
MindFusion.Charting Namespace

# RadarRenderer.enumVisibleRanges Method

Enumerates each series once providing a list of all visible points at each iteration.

**Namespace:** MindFusion.Charting
**File:** RadarRenderer.js

## Syntax

**JavaScript**

```
function enumVisibleRanges (context, process)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*process*
Type: ProcessRange
A ProcessRange callback.

## Remarks

The signature of the ProcessRange delegate is as follows:'function(seriesIndex, points)'.param {Number} seriesIndexAn integer index of series in Series list.param {MindFusion.Charting.Collections.List<Point>} points A list of Point values containing the plot coordinates corresponding to data items in currently visible range.The method is not expected to return a value.

## See Also

RadarRenderer Members
RadarRenderer Class
MindFusion.Charting Namespace

# RadarRenderer.getMaxValue Method

Gets the maximum value from all series located at specified index.

**Namespace:** MindFusion.Charting
**File:** RadarRenderer.js

## Syntax

**JavaScript**

```
function getMaxValue (index)
```

### Parameters

*index*
Type: Number
An integer index of data element in the series.

### Return Value

Type: Number
A number value representing the largest data element.

## See Also

RadarRenderer Members
RadarRenderer Class
MindFusion.Charting Namespace

# RadarRenderer.getMinValue Method

Gets the minimum value from all series located at specified index.

**Namespace:** MindFusion.Charting
**File:** RadarRenderer.js

## Syntax

**JavaScript**

```
function getMinValue (index)
```

### Parameters

*index*
Type: Number
An integer index of data element in the series.

### Return Value

Type: Number
A number value representing the smallest data element.

## See Also

RadarRenderer Members
RadarRenderer Class
MindFusion.Charting Namespace

# RadarRenderer.hitTest Method

SeriesRenderer.HitTest override. Hit-tests the visual representation of the series for a data item.

**Namespace:** MindFusion.Charting
**File:** RadarRenderer.js

## Syntax

```
JavaScript
function hitTest (context, location)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*location*
Type: Point
A Point specifying where to look for data items.

### Return Value

Type: HitResult
A HitResult instance identifying the found data item.

## See Also

RadarRenderer Members
RadarRenderer Class
MindFusion.Charting Namespace

# RadarRenderer.measureDataRange Method

SeriesRenderer.MeasureDataRange override. Measures the data range of rendered series and assigns it to the associated Axis objects if their MinValue and MaxValue are not set.

**Namespace:** MindFusion.Charting
**File:** RadarRenderer.js

## Syntax

```
JavaScript
function measureDataRange (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

RadarRenderer Members
RadarRenderer Class
MindFusion.Charting Namespace

# RadarRenderer Properties

# RadarRenderer.areaOpacity Property

Gets or sets the opacity of radar polygons.

**Namespace:** MindFusion.Charting
**File:** RadarRenderer.js

## Syntax

```
JavaScript
get areaOpacity() {}
```

### Property Value

Type: Number

## See Also

RadarRenderer Members
RadarRenderer Class
MindFusion.Charting Namespace

# RadarRenderer.series Property

Gets or sets a list of Series drawn by this RadarRenderer.

**Namespace:** MindFusion.Charting
**File:** RadarRenderer.js

## Syntax

```
JavaScript
get series() {}
```

### Property Value

Type: ObservableCollection

## See Also

RadarRenderer Members
RadarRenderer Class
MindFusion.Charting Namespace

# RadarScatterRenderer Class

A SeriesRenderer that draws scatter in radar plots.

## Syntax

```
JavaScript
// class
RadarScatterRenderer.prototype = {}
```

## Inheritance Hierarchy

## See Also

RadarScatterRenderer Members
MindFusion.Charting Namespace

# RadarScatterRenderer Members

A SeriesRenderer that draws scatter in radar plots.

The following tables list the members exposed by the RadarScatterRenderer type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | RadarScatterRenderer | Initializes a new instance of the RadarScatterRenderer class. |

## Public Methods

| | Name | Description |
|---|---|---|
|  | enumPoints | ScatterRenderer.EnumPoints override. Enumerates the data values of rendered series mapped to plot's 2D coordinates. |
|  | measureDataRange | Renderer2D.MeasureDataRange override. Measures the data range of rendered series and assigns it to the associated Axis objects if their MinValue and MaxValue are not set. |

## See Also

RadarScatterRenderer Class
MindFusion.Charting Namespace

# RadarScatterRenderer Constructor

Initializes a new instance of the RadarScatterRenderer class.

**Namespace:** MindFusion.Charting
**File:** RadarScatterRenderer.js

## Syntax

**JavaScript**

```
function RadarScatterRenderer (series)
```

### Parameters

*series*

Type: ObservableCollection
A list of Series that should be rendered as scatter.

## See Also

RadarScatterRenderer Class
MindFusion.Charting Namespace

# RadarScatterRenderer Methods

## RadarScatterRenderer.enumPoints Method

ScatterRenderer.EnumPoints override. Enumerates the data values of rendered series mapped to plot's 2D coordinates.

**Namespace:** MindFusion.Charting
**File:** RadarScatterRenderer.js

### Syntax

```
JavaScript
function enumPoints (context, process)
```

#### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*process*
Type: ProcessPoint
A ProcessPoint callback.

### Remarks

The signature of the ProcessPoint delegate is as follows:'function(seriesIndex, dataIndex, point)'.param {Number} seriesIndex An integer index of series in Series list.param {Number} dataIndex An integer index of data item in specified series.param {Point} point A Point instance containing the Plot2D coordinates corresponding to current data item.The method is not expected to return a value.

### See Also

RadarScatterRenderer Members
RadarScatterRenderer Class
MindFusion.Charting Namespace

## RadarScatterRenderer.measureDataRange Method

Renderer2D.MeasureDataRange override. Measures the data range of rendered series and assigns it to the associated Axis objects if their MinValue and MaxValue are not set.

**Namespace:** MindFusion.Charting
**File:** RadarScatterRenderer.js

### Syntax

```
JavaScript
function measureDataRange (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

RadarScatterRenderer Members
RadarScatterRenderer Class
MindFusion.Charting Namespace

# RadarType Enumeration

Identifies the type of radar-chart.

**Namespace:** MindFusion.Charting
**File:** RadarType.js

## Syntax

**JavaScript**

```
// enum
RadarType = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **Pie** | Draw data items as circular sectors. |
| | **Polygon** | Draw each series as a polygon. |

## See Also

MindFusion.Charting Namespace

# RenderContext Class

Provides contextual information about the dashboard to its child components when calling their draw, layout and user input methods.

## Syntax

**JavaScript**

```
// class
RenderContext.prototype = {}
```

## Inheritance Hierarchy

## See Also

RenderContext Members
MindFusion.Charting Namespace

# RenderContext Members

Provides contextual information about the dashboard to its child components when calling their draw, layout and user input methods.

The following tables list the members exposed by the RenderContext type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | RenderContext | Initializes a new instance of the RenderContext clas. |

## Public Properties

| | Name | Description |
|---|---|---|
| | component | Gets or sets the component currently being painted. |
| | graphics | Gets or sets the current drawing surface. |
| | theme | Gets or sets the Theme used to resolve appearance attributes. |
| | xAxis | Gets the chart's XAxis. |
| | yAxis | Gets the chart's YAxis. |

## Public Methods

| | Name | Description |
|---|---|---|
| | getClipRect | Gets the clip rectangle relative to specified Component. |
| | getXAxis | Gets the X axis assigned to a Renderer2D or returns one from parent plot or chart. |
| | getYAxis | Gets the Y axis assigned to a Renderer2D or returns one from parent plot or chart. |
| | textStyle | Gets a TextStyle from current Theme. |

## See Also

RenderContext Class
MindFusion.Charting Namespace

# RenderContext Constructor

Initializes a new instance of the RenderContext clas.

**Namespace:** MindFusion.Charting
**File:** RenderContext.js

## Syntax

```
JavaScript
function RenderContext (clipRect, theme)
```

### Parameters

*clipRect*
Type: Rect
A Rect specifying the clip rectangle.

*theme*
Type: Theme
A Theme instance where appearance attributes should be derived from.

## See Also

RenderContext Class
MindFusion.Charting Namespace

# RenderContext Methods

# RenderContext.getClipRect Method

Gets the clip rectangle relative to specified Component.

**Namespace:** MindFusion.Charting
**File:** RenderContext.js

## Syntax

```
JavaScript
function getClipRect (relativeTo)
```

### Parameters

*relativeTo*
Type: Component
A Component instance.

### Return Value

Type: Rect
A Rect instance.

## See Also

RenderContext Members
RenderContext Class
MindFusion.Charting Namespace

# RenderContext.getXAxis Method

Gets the X axis assigned to a Renderer2D or returns one from parent plot or chart.

**Namespace:** MindFusion.Charting
**File:** RenderContext.js

## Syntax

```
JavaScript
function getXAxis ([series])
```

### Parameters

*series*
Optional.
Type: Renderer2D
A Renderer2D instance.

### Return Value

Type: Axis
An Axis instance.

## See Also

RenderContext Members
RenderContext Class
MindFusion.Charting Namespace

# RenderContext.getYAxis Method

Gets the Y axis assigned to a Renderer2D or returns one from parent plot or chart.

**Namespace:** MindFusion.Charting
**File:** RenderContext.js

## Syntax

```
JavaScript
function getYAxis ([series])
```

### Parameters

*series*
Optional.
Type: Renderer2D
A Renderer2D instance.

### Return Value

Type: Axis
An Axis instance.

## See Also

RenderContext Members
RenderContext Class
MindFusion.Charting Namespace

# RenderContext.textStyle Method

Gets a TextStyle from current Theme.

**Namespace:** MindFusion.Charting
**File:** RenderContext.js

## Syntax

```
JavaScript
function textStyle (hint)
```

### Parameters

*hint*
Type: TextStyleHint
A member of the TextStyleHint enumeration.

### Return Value

Type: TextStyle
A TextStyle instance.

## See Also

RenderContext Members
RenderContext Class
MindFusion.Charting Namespace

# RenderContext Properties

# RenderContext.component Property

Gets or sets the component currently being painted.

**Namespace:** MindFusion.Charting
**File:** RenderContext.js

## Syntax

```
JavaScript
get component() {}
```

### Property Value

Type: Component

## See Also

RenderContext Members
RenderContext Class
MindFusion.Charting Namespace

# RenderContext.graphics Property

Gets or sets the current drawing surface.

**Namespace:** MindFusion.Charting
**File:** RenderContext.js

## Syntax

```
JavaScript
get graphics() {}
```

### Property Value

Type: Graphics

## See Also

RenderContext Members
RenderContext Class
MindFusion.Charting Namespace

# RenderContext.theme Property

Gets or sets the Theme used to resolve appearance attributes.

**Namespace:** MindFusion.Charting
**File:** RenderContext.js

## Syntax

```
JavaScript
get theme() {}
```

### Property Value

Type: Theme

## See Also

RenderContext Members
RenderContext Class
MindFusion.Charting Namespace

# RenderContext.xAxis Property

Gets the chart's XAxis.

**Namespace:** MindFusion.Charting
**File:** RenderContext.js

## Syntax

```
JavaScript
```

```
get xAxis() {}
```

### Property Value

Type: Axis

## See Also

RenderContext Members
RenderContext Class
MindFusion.Charting Namespace

# RenderContext.yAxis Property

Gets the chart's YAxis.

**Namespace:** MindFusion.Charting
**File:** RenderContext.js

## Syntax

**JavaScript**

```
get yAxis() {}
```

### Property Value

Type: Axis

## See Also

RenderContext Members
RenderContext Class
MindFusion.Charting Namespace

# RenderContext2D Class

Provides contextual information about the dashboard to its child components when calling their draw, layout and user input methods.

## Syntax

**JavaScript**

```
// class
RenderContext2D.prototype = {}
```

## Inheritance Hierarchy

## See Also

RenderContext2D Members
MindFusion.Charting Namespace

# RenderContext2D Members

Provides contextual information about the dashboard to its child components when calling their draw, layout and user input methods.

The following tables list the members exposed by the RenderContext2D type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | RenderContext2D | Initializes a new instance of the RenderContext class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | chart | Gets the chart. |
|  | xAxis | Gets the chart's XAxis. |
|  | yAxis | Gets the chart's YAxis. |

## See Also

RenderContext2D Class
MindFusion.Charting Namespace

# RenderContext2D Constructor

Initializes a new instance of the RenderContext class.

**Namespace:** MindFusion.Charting
**File:** RenderContext2D.js

## Syntax

**JavaScript**

```
function RenderContext2D (clipRect, theme)
```

### Parameters

*clipRect*

Type: void
A RectD specifying the clip rectangle.

*theme*

Type: void
A Theme instance where appearance attributes should be derived from.

## See Also

RenderContext2D Class
MindFusion.Charting Namespace

# RenderContext2D Properties

## RenderContext2D.chart Property

Gets the chart.

**Namespace:** MindFusion.Charting
**File:** RenderContext2D.js

### Syntax

```
JavaScript
get chart() {}
```

#### Property Value

Type: BiaxialChart

### See Also

RenderContext2D Members
RenderContext2D Class
MindFusion.Charting Namespace

## RenderContext2D.xAxis Property

Gets the chart's XAxis.

**Namespace:** MindFusion.Charting
**File:** RenderContext2D.js

### Syntax

```
JavaScript
get xAxis() {}
```

#### Property Value

Type: Axis

### See Also

RenderContext2D Members
RenderContext2D Class
MindFusion.Charting Namespace

## RenderContext2D.yAxis Property

Gets the chart's YAxis.

**Namespace:** MindFusion.Charting
**File:** RenderContext2D.js

### Syntax

| JavaScript |
| --- |
| `get yAxis() {}` |

### Property Value

Type: Axis

## See Also

RenderContext2D Members
RenderContext2D Class
MindFusion.Charting Namespace

# Renderer2D Class

A base class for series renderers that draw in two dimensional Cartesian coordinate system.

## Syntax

| JavaScript |
| --- |
| `// class`<br>`Renderer2D.prototype = {}` |

## Inheritance Hierarchy

## See Also

Renderer2D Members
MindFusion.Charting Namespace

# Renderer2D Members

A base class for series renderers that draw in two dimensional Cartesian coordinate system.

The following tables list the members exposed by the Renderer2D type.

## Public Constructors

| | Name | Description |
| --- | --- | --- |
|  | Renderer2D | Initializes a new instance of the Renderer2D class. |

## Public Properties

| | Name | Description |
| --- | --- | --- |
|  | series | Gets or sets a list of Series drawn by this Renderer2D. |
|  | xAxis | Gets the X axis associated with this Renderer2D. |
|  | yAxis | Gets the Y axis associated with this Renderer2D. |

## Public Methods

| | Name | Description |
|---|---|---|
| | domainData | Gets the domain axis coordinate (input value) of specified data item. |
| | drawHighlight | SeriesRenderer.DrawHighlight override. Draws highlighted data item in specified RenderContext. |
| | enumSeries | Implements the SeriesContainer interface. |
| | enumVisiblePointPairs | Enumerates the data values of rendered series mapped to plot 2D coordinates. |
| | enumVisiblePoints | Enumerates the data values of rendered series mapped to plot 2D coordinates. |
| | enumVisiblePoints1 | Enumerates the data values of rendered series mapped to plot 2D coordinates. |
| | enumVisibleRanges | Enumerates each series once providing a list of all visible points at each iteration. |
| | getMaxXSum | Gets the maximum sum of X data values locates at same index in all series. |
| | getMaxYSum | Gets the maximum sum of Y data values locates at same index in all series. |
| | getPixel | Gets the coordinates of specified data item when mapped to a Plot2D pixel. |
| | getPixel1 | Gets the coordinates of specified data item when mapped to a Plot2D pixel. |
| | hitTest | SeriesRenderer.HitTest override. Hit-tests the visual representation of the series for a data item. |
| | imageData | Gets the image axis coordinate (output value) of specified data item. |
| | measureDataRange | SeriesRenderer.MeasureDataRange override. Measures the data range of rendered series and assigns it to the associated Axis objects if their MinValue and MaxValue are not set. |
| | startMeasureData | SeriesRenderer.StartMeasureData override. Called to reset measure accumulators at the beginning of a measure pass. |
| | xData | Gets the X value of specified data item. |
| | xSorted | Gets a value indicating whether values of specified series increase monotonously in X dimension. |
| | yData | Gets the Y value of specified data item. |
| | ySorted | Gets a value indicating whether values of specified series increase monotonously in Y dimension. |

## See Also

Renderer2D Class
MindFusion.Charting Namespace

# Renderer2D Constructor

Initializes a new instance of the Renderer2D class.

**Namespace:** MindFusion.Charting
**File:** Renderer2D.js

## Syntax

**JavaScript**

```
function Renderer2D (series)
```

### Parameters

*series*
Type: ObservableCollection
A list of Series that should be rendered by this object.

## See Also

Renderer2D Class
MindFusion.Charting Namespace

# Renderer2D Methods

# Renderer2D.domainData Method

Gets the domain axis coordinate (input value) of specified data item.

**Namespace:** MindFusion.Charting
**File:** Renderer2D.js

## Syntax

**JavaScript**

```
function domainData (seriesIndex, dataIndex)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series within the list of series rendered by this Renderer2D.

*dataIndex*
Type: Number
An integer index of the data item in specified series.

### Return Value

Type: Number
A number value.

## See Also

Renderer2D Members
Renderer2D Class
MindFusion.Charting Namespace

# Renderer2D.drawHighlight Method

SeriesRenderer.DrawHighlight override. Draws highlighted data item in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** Renderer2D.js

## Syntax

**JavaScript**

```
function drawHighlight (context, hitResult)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*hitResult*
Type: HitResult
A HitResult instance identifying highlighted data item.

## See Also

Renderer2D Members
Renderer2D Class
MindFusion.Charting Namespace

# Renderer2D.enumSeries Method

Implements the SeriesContainer interface.

**Namespace:** MindFusion.Charting
**File:** Renderer2D.js

## Syntax

**JavaScript**

```
function enumSeries ()
```

### Return Value

Type: IEnumerable
An instance of the IEnumerable<Series> class.

## See Also

Renderer2D Members
Renderer2D Class

# Renderer2D.enumVisiblePointPairs Method

Enumerates the data values of rendered series mapped to plot 2D coordinates.

**Namespace:** MindFusion.Charting
**File:** Renderer2D.js

## Syntax

**JavaScript**

```
function enumVisiblePointPairs (context, process)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*process*
Type: ProcessPoints
A ProcessPoints callback.

## Remarks

The signature of the ProcessPoints delegate is as follows:'function(seriesIndex, dataIndex, point)'.param {Number} seriesIndex An integer index of series in Series list.param {Number} dataIndex An integer index of data item in specified series.param {Point} point1 A Point instance containing the Plot2D coordinates corresponding to previous data item.param {Point} point2 A Point instance containing the Plot2D coordinates corresponding to current data item.The method is not expected to return a value.

## See Also

Renderer2D Members
Renderer2D Class
MindFusion.Charting Namespace

# Renderer2D.enumVisiblePoints Method

Enumerates the data values of rendered series mapped to plot 2D coordinates.

**Namespace:** MindFusion.Charting
**File:** Renderer2D.js

## Syntax

**JavaScript**

```
function enumVisiblePoints (context, frontToBack, process)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*frontToBack*
Type: Boolean
true if the series list should be enumerated from front to back, or false otherwise.

*process*
Type: ProcessPoint
A ProcessPoint callback.

## Remarks

The signature of the ProcessPoint delegate is as follows:'function(seriesIndex, dataIndex, point)'.param {Number} seriesIndex An integer index of series in Series list.param {Number} dataIndex An integer index of data item in specified series.param {Point} point A Point instance containing the Plot2D coordinates corresponding to current data item.The method is not expected to return a value.

## See Also

Renderer2D Members
Renderer2D Class
MindFusion.Charting Namespace

# Renderer2D.enumVisiblePoints1 Method

Enumerates the data values of rendered series mapped to plot 2D coordinates.

**Namespace:** MindFusion.Charting
**File:** Renderer2D.js

## Syntax

**JavaScript**

```
function enumVisiblePoints1 (context, process)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*process*
Type: ProcessPoint
A ProcessPoint callback.

## Remarks

The signature of the ProcessPoint delegate is as follows:'function(seriesIndex, dataIndex, point)'.param {Number} seriesIndex An integer index of series in Series list.param {Number} dataIndex An integer index of data item in specified series.param {Point} point A Point instance containing the Plot2D coordinates corresponding to current data item.The method is not expected to return a value.

## See Also

Renderer2D Members
Renderer2D Class
MindFusion.Charting Namespace

# Renderer2D.enumVisibleRanges Method

Enumerates each series once providing a list of all visible points at each iteration.

**Namespace:** MindFusion.Charting
**File:** Renderer2D.js

## Syntax

**JavaScript**

```
function enumVisibleRanges (context, process)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*process*
Type: ProcessRange
A ProcessRange callback.

## Remarks

The signature of the ProcessRange delegate is as follows:'function(seriesIndex, points)'.param {Number} seriesIndex An integer index of series in Series list.param {List<Point>} points A list of Point values containing the Plot2D coordinates corresponding to data items in currently visible range.The method is not expected to return a value.

## See Also

Renderer2D Members
Renderer2D Class
MindFusion.Charting Namespace

# Renderer2D.getMaxXSum Method

Gets the maximum sum of X data values locates at same index in all series.

**Namespace:** MindFusion.Charting
**File:** Renderer2D.js

## Syntax

**JavaScript**

```
function getMaxXSum ()
```

### Return Value

Type: Number
A number value containing the maximum sum.

## See Also

Renderer2D Members
Renderer2D Class

# Renderer2D.getMaxYSum Method

Gets the maximum sum of Y data values locates at same index in all series.

**Namespace:** MindFusion.Charting
**File:** Renderer2D.js

## Syntax

**JavaScript**

```
function getMaxYSum ()
```

### Return Value

Type: Number
A number value containing the maximum sum.

## See Also

Renderer2D Members
Renderer2D Class
MindFusion.Charting Namespace

# Renderer2D.getPixel Method

Gets the coordinates of specified data item when mapped to a Plot2D pixel.

**Namespace:** MindFusion.Charting
**File:** Renderer2D.js

## Syntax

**JavaScript**

```
function getPixel (seriesIndex, dataIndex, xAxis, yAxis, component)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series within the list of series rendered by this Renderer2D.

*dataIndex*
Type: Number
An integer index of the data item in specified series.

*xAxis*
Type: Axis
An Axis reference specifying the X axis.

*yAxis*
Type: Axis
An Axis reference specifying the Y axis.

*component*

Type: Component
An instance of Plot2D or derived class.

### Return Value

Type: Point
A Point containing Plot2D coordinates corresponding to specified data item.

## See Also

Renderer2D Members
Renderer2D Class
MindFusion.Charting Namespace

# Renderer2D.getPixel1 Method

Gets the coordinates of specified data item when mapped to a Plot2D pixel.

**Namespace:** MindFusion.Charting
**File:** Renderer2D.js

## Syntax

**JavaScript**

```
function getPixel1 (valueX, xAxis, valueY, yAxis, component)
```

### Parameters

*valueX*
Type: Number
The X value of data item in logical coordinate system.

*xAxis*
Type: Axis
The X Axis from which logical value is mapped to plot's actual width.

*valueY*
Type: Number
The Y value of data item in logical coordinate system.

*yAxis*
Type: Axis
The Y Axis from which logical value is mapped to plot's actual height.

*component*
Type: Component
An instance of Plot2D or derived class.

### Return Value

Type: Point

## See Also

Renderer2D Members
Renderer2D Class
MindFusion.Charting Namespace

# Renderer2D.hitTest Method

SeriesRenderer.HitTest override. Hit-tests the visual representation of the series for a data item.

**Namespace:** MindFusion.Charting
**File:** Renderer2D.js

## Syntax

**JavaScript**

```
function hitTest (context, location)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*location*
Type: Point
A Point specifying where to look for data items.

### Return Value

Type: HitResult
A HitResult instance identifying the found data item.

## See Also

Renderer2D Members
Renderer2D Class
MindFusion.Charting Namespace

# Renderer2D.imageData Method

Gets the image axis coordinate (output value) of specified data item.

**Namespace:** MindFusion.Charting
**File:** Renderer2D.js

## Syntax

**JavaScript**

```
function imageData (seriesIndex, dataIndex)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series within the list of series rendered by this Renderer2D.

*dataIndex*
Type: Number
An integer index of the data item in specified series.

### Return Value

Type: Number
A number value.

## See Also

Renderer2D Members
Renderer2D Class
MindFusion.Charting Namespace

# Renderer2D.measureDataRange Method

SeriesRenderer.MeasureDataRange override. Measures the data range of rendered series and assigns it to the associated Axis objects if their MinValue and MaxValue are not set.

**Namespace:** MindFusion.Charting
**File:** Renderer2D.js

## Syntax

**JavaScript**

```
function measureDataRange (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

Renderer2D Members
Renderer2D Class
MindFusion.Charting Namespace

# Renderer2D.startMeasureData Method

SeriesRenderer.StartMeasureData override. Called to reset measure accumulators at the beginning of a measure pass.

**Namespace:** MindFusion.Charting
**File:** Renderer2D.js

## Syntax

**JavaScript**

```
function startMeasureData (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

Renderer2D Members
Renderer2D Class
MindFusion.Charting Namespace

# Renderer2D.xData Method

Gets the X value of specified data item.

**Namespace:** MindFusion.Charting
**File:** Renderer2D.js

## Syntax

**JavaScript**

```
function xData (seriesIndex, dataIndex)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series within the list of series rendered by this Renderer2D.

*dataIndex*
Type: Number
An integer index of the data item in specified series.

### Return Value

Type: Number
A number value.

## See Also

Renderer2D Members
Renderer2D Class
MindFusion.Charting Namespace

# Renderer2D.xSorted Method

Gets a value indicating whether values of specified series increase monotonously in X dimension.

**Namespace:** MindFusion.Charting
**File:** Renderer2D.js

## Syntax

**JavaScript**

```
function xSorted (seriesIndex)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series within the list of series rendered by this Renderer2D.

### Return Value

Type: Boolean
true if specified series is sorted, or false otherwise.

## See Also

Renderer2D Members
Renderer2D Class
MindFusion.Charting Namespace

# Renderer2D.yData Method

Gets the Y value of specified data item.

**Namespace:** MindFusion.Charting
**File:** Renderer2D.js

## Syntax

```
JavaScript
function yData (seriesIndex, dataIndex)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series within the list of series rendered by this Renderer2D.

*dataIndex*
Type: Number
An integer index of the data item in specified series.

### Return Value

Type: Number
A number value.

## See Also

Renderer2D Members
Renderer2D Class
MindFusion.Charting Namespace

# Renderer2D.ySorted Method

Gets a value indicating whether values of specified series increase monotonously in Y dimension.

**Namespace:** MindFusion.Charting
**File:** Renderer2D.js

## Syntax

```
JavaScript
function ySorted (seriesIndex)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series within the list of series rendered by this Renderer2D.

### Return Value

Type: Boolean
true if specified series is sorted, or false otherwise.

## See Also

Renderer2D Members
Renderer2D Class
MindFusion.Charting Namespace

# Renderer2D Properties

## Renderer2D.series Property

Gets or sets a list of Series drawn by this Renderer2D.

**Namespace:** MindFusion.Charting
**File:** Renderer2D.js

## Syntax

**JavaScript**

```
get series() {}
```

### Property Value

Type: ObservableCollection

## See Also

Renderer2D Members
Renderer2D Class
MindFusion.Charting Namespace

## Renderer2D.xAxis Property

Gets the X axis associated with this Renderer2D.

**Namespace:** MindFusion.Charting
**File:** Renderer2D.js

## Syntax

**JavaScript**

```
get xAxis() {}
```

### Property Value

Type: Axis

## See Also

Renderer2D Members
Renderer2D Class
MindFusion.Charting Namespace

# Renderer2D.yAxis Property

Gets the Y axis associated with this Renderer2D.

**Namespace:** MindFusion.Charting
**File:** Renderer2D.js

## Syntax

**JavaScript**

```
get yAxis() {}
```

### Property Value

Type: Axis

## See Also

Renderer2D Members
Renderer2D Class
MindFusion.Charting Namespace

# Renderer3D Interface

Defines the interface called by Plot3D to build 3D models.

## Syntax

**JavaScript**

```
// interface
Renderer3D.prototype = {}
```

## See Also

Renderer3D Members
MindFusion.Charting Namespace

# Renderer3D Members

Defines the interface called by Plot3D to build 3D models.

The following tables list the members exposed by the Renderer3D type.

## Public Methods

| | Name | Description |
|---|---|---|

| | | |
|---|---|---|
| | buildModels | Adds models for the series data to specified scene. |

## See Also

Renderer3D Interface
MindFusion.Charting Namespace

# Renderer3D Methods

## Renderer3D.buildModels Method

Adds models for the series data to specified scene.

**Namespace:** MindFusion.Charting
**File:** Renderer3D.js

## Syntax

**JavaScript**

```
function buildModels (scene, context)
```

### Parameters

*scene*
Type: Scene3D
A Scene3D instance.

*context*
Type: RenderContext
A RenderContext instance.

## See Also

Renderer3D Members
Renderer3D Interface
MindFusion.Charting Namespace

# RotationController Class

Represents a controller that rotates a polar plot by changing its StartAngle.

## Syntax

**JavaScript**

```
// class
RotationController.prototype = {}
```

## Inheritance Hierarchy

## See Also

# RotationController Members

Represents a controller that rotates a polar plot by changing its StartAngle.

The following tables list the members exposed by the RotationController type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | RotationController | Initializes a new instance of the RotationController class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | enableAnimation | Gets or sets whether to enable rotation inertia. |

## Public Methods

| | Name | Description |
|---|---|---|
|  | getRunningAnimation | For internal use. |
|  | onMouseDown | PlotController.OnMouseDown override. Called when the user presses a mouse button. |
|  | onMouseMove | PlotController.OnMouseMove override. Called when the user moves the mouse. |
|  | onMouseUp | PlotController.OnMouseUp override. Called when the user releases a mouse button. |

## See Also

RotationController Class
MindFusion.Charting Namespace

# RotationController Constructor

Initializes a new instance of the RotationController class.

**Namespace:** MindFusion.Charting
**File:** RotationController.js

## Syntax

**JavaScript**

```
function RotationController (renderContext)
```

### Parameters

*renderContext*
Type: RenderContext
A RenderContext instance.

## See Also

RotationController Class
MindFusion.Charting Namespace

# RotationController Methods

## RotationController.getRunningAnimation Method

For internal use.

**Namespace:** MindFusion.Charting
**File:** RotationController.js

## Syntax

```
JavaScript
function getRunningAnimation ()
```

### Return Value

Type: ComponentAnimation
An instance of a ComponentAnimation -derived class.

## See Also

RotationController Members
RotationController Class
MindFusion.Charting Namespace

## RotationController.onMouseDown Method

PlotController.OnMouseDown override. Called when the user presses a mouse button.

**Namespace:** MindFusion.Charting
**File:** RotationController.js

## Syntax

```
JavaScript
function onMouseDown (x, y)
```

### Parameters

*x*
Type: Number
A number value specifying the horizontal position of mouse pointer.

*y*

Type: Number
A number value specifying the vertical position of mouse pointer.

## See Also

RotationController Members
RotationController Class
MindFusion.Charting Namespace

# RotationController.onMouseMove Method

PlotController.OnMouseMove override. Called when the user moves the mouse.

**Namespace:** MindFusion.Charting
**File:** RotationController.js

## Syntax

**JavaScript**

```
function onMouseMove (x, y)
```

### Parameters

*x*
Type: Number
A number value specifying the horizontal position of mouse pointer.

*y*
Type: Number
A number value specifying the vertical position of mouse pointer.

## See Also

RotationController Members
RotationController Class
MindFusion.Charting Namespace

# RotationController.onMouseUp Method

PlotController.OnMouseUp override. Called when the user releases a mouse button.

**Namespace:** MindFusion.Charting
**File:** RotationController.js

## Syntax

**JavaScript**

```
function onMouseUp (x, y)
```

### Parameters

*x*
Type: Number
A number value specifying the horizontal position of mouse pointer.

*y*

Type: Number
A number value specifying the vertical position of mouse pointer.

## See Also

RotationController Members
RotationController Class
MindFusion.Charting Namespace

# RotationController Properties

## RotationController.enableAnimation Property

Gets or sets whether to enable rotation inertia.

**Namespace:** MindFusion.Charting
**File:** RotationController.js

## Syntax

```
JavaScript
get enableAnimation() {}
```

### Property Value

Type: Boolean

## See Also

RotationController Members
RotationController Class
MindFusion.Charting Namespace

# RotationControllerAnimation Class

Represents a controller that rotates a polar plot by changing its StartAngle.

## Syntax

```
JavaScript
// class
RotationControllerAnimation.prototype = {}
```

## Inheritance Hierarchy

## See Also

RotationControllerAnimation Members
MindFusion.Charting Namespace

## RotationControllerAnimation Members

Represents a controller that rotates a polar plot by changing its StartAngle.

The following tables list the members exposed by the RotationControllerAnimation type.

### See Also

RotationControllerAnimation Class
MindFusion.Charting Namespace

# ScatterRenderer Class

A SeriesRenderer that draws scatter in its containing plot.

### Syntax

**JavaScript**

```
// class
ScatterRenderer.prototype = {}
```

### Inheritance Hierarchy

### See Also

ScatterRenderer Members
MindFusion.Charting Namespace

## ScatterRenderer Members

A SeriesRenderer that draws scatter in its containing plot.

The following tables list the members exposed by the ScatterRenderer type.

### Public Constructors

| | Name | Description |
|---|---|---|
|  | ScatterRenderer | Initializes a new instance of the ScatterRenderer class. |

### Public Properties

| | Name | Description |
|---|---|---|
|  | shape | Gets or sets the type of scatter shapes. |
|  | shapeSize | Gets or sets the size of scatter shapes. |

### Public Methods

| | Name | Description |
|---|---|---|

| | | |
|---|---|---|
| | draw | SeriesRenderer.Draw override. Draws the series data in specified RenderContext. |
| | drawHighlight | Renderer2D.DrawHighlight override. Draws highlighted shape in specified RenderContext. |
| | enumPoints | Enumerates the data values of rendered series mapped to plot 2D coordinates. |
| | hitTest | Renderer2D.HitTest override. Hit-tests the scatter representations of data items. |

## See Also

ScatterRenderer Class
MindFusion.Charting Namespace

# ScatterRenderer Constructor

Initializes a new instance of the ScatterRenderer class.

**Namespace:** MindFusion.Charting
**File:** ScatterRenderer.js

## Syntax

**JavaScript**

```
function ScatterRenderer (series)
```

### Parameters

*series*
Type: ObservableCollection
A list of Series that should be rendered as scatter.

## See Also

ScatterRenderer Class
MindFusion.Charting Namespace

# ScatterRenderer Methods

# ScatterRenderer.draw Method

SeriesRenderer.Draw override. Draws the series data in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** ScatterRenderer.js

## Syntax

**JavaScript**

```
function draw (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

ScatterRenderer Members
ScatterRenderer Class
MindFusion.Charting Namespace

# ScatterRenderer.drawHighlight Method

Renderer2D.DrawHighlight override. Draws highlighted shape in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** ScatterRenderer.js

## Syntax

**JavaScript**

```
function drawHighlight (context, hitResult)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*hitResult*
Type: HitResult
A HitResult instance identifying highlighted data item.

## See Also

ScatterRenderer Members
ScatterRenderer Class
MindFusion.Charting Namespace

# ScatterRenderer.enumPoints Method

Enumerates the data values of rendered series mapped to plot 2D coordinates.

**Namespace:** MindFusion.Charting
**File:** ScatterRenderer.js

## Syntax

**JavaScript**

```
function enumPoints (context, process)
```

### Parameters

*context*

Type: RenderContext
A RenderContext instance.

*process*
Type: ProcessPoint
A ProcessPoint callback.

## Remarks

The signature of the ProcessPoint delegate is as follows:'function(seriesIndex, dataIndex, point)'.param {Number} seriesIndex An integer index of series in Series list.param {Number} dataIndex An integer index of data item in specified series.param {Point} point A Point instance containing the Plot2D coordinates corresponding to current data item.The method is not expected to return a value.

## See Also

ScatterRenderer Members
ScatterRenderer Class
MindFusion.Charting Namespace

# ScatterRenderer.hitTest Method

Renderer2D.HitTest override. Hit-tests the scatter representations of data items.

**Namespace:** MindFusion.Charting
**File:** ScatterRenderer.js

## Syntax

**JavaScript**

```
function hitTest (context, location)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*location*
Type: Point
A Point specifying where to look for data items.

### Return Value

Type: HitResult
A HitResult instance identifying the found data item.

## See Also

ScatterRenderer Members
ScatterRenderer Class
MindFusion.Charting Namespace

# ScatterRenderer Properties

## ScatterRenderer.shape Property

Gets or sets the type of scatter shapes.

**Namespace:** MindFusion.Charting
**File:** ScatterRenderer.js

### Syntax

```
JavaScript
get shape() {}
```

#### Property Value

Type: ScatterType

### See Also

ScatterRenderer Members
ScatterRenderer Class
MindFusion.Charting Namespace

## ScatterRenderer.shapeSize Property

Gets or sets the size of scatter shapes.

**Namespace:** MindFusion.Charting
**File:** ScatterRenderer.js

### Syntax

```
JavaScript
get shapeSize() {}
```

#### Property Value

Type: Number

### See Also

ScatterRenderer Members
ScatterRenderer Class
MindFusion.Charting Namespace

# ScatterType Enumeration

Identifies the shapes rendered by a ScatterRenderer.

**Namespace:** MindFusion.Charting
**File:** ScatterType.js

### Syntax

```
JavaScript
```

```
// enum
ScatterType = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **Circle** | Draw circles. |
| | **Diamond** | Draw diamonds. |
| | **None** | Do not draw scatter. |
| | **Square** | Draw squares. |
| | **Triangle** | Draw triangles. |

## See Also

MindFusion.Charting Namespace

# Series Interface

Specifies the interface that data providers should implementing order to feed data to SeriesRenderer classes.

## Syntax

| JavaScript |
|---|
| `// interface`<br>`Series.prototype = {}` |

## See Also

Series Members
MindFusion.Charting Namespace

# Series Members

Specifies the interface that data providers should implementing order to feed data to SeriesRenderer classes.

The following tables list the members exposed by the Series type.

## Public Properties

| | Name | Description |
|---|---|---|
| | dataChanged | Raised when the values in this series change. |
| | dimensions | Gets the number of values (coordinates) that can be returned by each data item. |
| | size | Gets the number of data items in this series. |

| | | |
|---|---|---|
| | supportedLabels | Gets the kind of labels supported by this series. |
| | title | Gets the title of this series. |

## Public Methods

| | Name | Description |
|---|---|---|
| | getLabel | Returns a label for the specified data item in this series. |
| | getValue | Returns a value for the specified data item in this series. |
| | isEmphasized | Gets a value indicating whether the specified data item should be emphasized by the SeriesRenderer. |
| | isSortedfunction | Gets a value indicating whether the series values increase monotonously in specified dimension. |

## See Also

Series Interface
MindFusion.Charting Namespace

# Series Methods

# Series.getLabel Method

Returns a label for the specified data item in this series.

**Namespace:** MindFusion.Charting
**File:** Series.js

## Syntax

**JavaScript**

```
function getLabel (index, kind)
```

### Parameters

*index*
Type: Number
An integer value specifying the index of a data item.

*kind*
Type: LabelKinds
A member of the LabelKinds enumeration.

### Return Value

Type: String
A string containing the item's label.

## See Also

Series Members
Series Interface

# Series.getValue Method

Returns a value for the specified data item in this series.

**Namespace:** MindFusion.Charting
**File:** Series.js

## Syntax

**JavaScript**

```
function getValue (index, dimension)
```

### Parameters

*index*
Type: Number
An integer value specifying the index of a data item.

*dimension*
Type: Number
An integer value specifying the dimension whose coordinate to return.

### Return Value

Type: Number
A double-precision number representing the value of a data item in the series.

## See Also

Series Members
Series Interface
MindFusion.Charting Namespace

# Series.isEmphasized Method

Gets a value indicating whether the specified data item should be emphasized by the SeriesRenderer.

**Namespace:** MindFusion.Charting
**File:** Series.js

## Syntax

**JavaScript**

```
function isEmphasized (index)
```

### Parameters

*index*
Type: Number
An integer value specifying the index of a data item.

### Return Value

Type: Boolean
true to emphasize the specified data item, or false otherwise.

## See Also

Series Members
Series Interface
MindFusion.Charting Namespace

# Series.isSortedfunction Method

Gets a value indicating whether the series values increase monotonously in specified dimension.

**Namespace:** MindFusion.Charting
**File:** Series.js

## Syntax

**JavaScript**

```
function isSortedfunction (dimension)
```

### Parameters

*dimension*
Type: Number
An integer value identifying the dimension.

### Return Value

Type: Boolean
true if the values are sorted in specified dimension, or false otherwise.

## See Also

Series Members
Series Interface
MindFusion.Charting Namespace

# Series Properties

# Series.dataChanged Property

Raised when the values in this series change.

**Namespace:** MindFusion.Charting
**File:** Series.js

## Syntax

**JavaScript**

```
get dataChanged() {}
```

### Property Value

Type: EventArgs>

## See Also

Series Members
Series Interface
MindFusion.Charting Namespace

# Series.dimensions Property

Gets the number of values (coordinates) that can be returned by each data item.

**Namespace:** MindFusion.Charting
**File:** Series.js

## Syntax

```
JavaScript
get dimensions() {}
```

### Property Value

Type: Number

## See Also

Series Members
Series Interface
MindFusion.Charting Namespace

# Series.size Property

Gets the number of data items in this series.

**Namespace:** MindFusion.Charting
**File:** Series.js

## Syntax

```
JavaScript
get size() {}
```

### Property Value

Type: Number

## See Also

Series Members
Series Interface
MindFusion.Charting Namespace

## Series.supportedLabels Property

Gets the kind of labels supported by this series.

**Namespace:** MindFusion.Charting
**File:** Series.js

### Syntax

```
JavaScript
get supportedLabels() {}
```

#### Property Value

Type: LabelKinds

### See Also

Series Members
Series Interface
MindFusion.Charting Namespace

## Series.title Property

Gets the title of this series.

**Namespace:** MindFusion.Charting
**File:** Series.js

### Syntax

```
JavaScript
get title() {}
```

#### Property Value

Type: String

### See Also

Series Members
Series Interface
MindFusion.Charting Namespace

# Series2D Class

Represents a data series defined by two lists of values containing respectively X and Y coordinates.

### Syntax

```
JavaScript
// class
Series2D.prototype = {}
```

## Inheritance Hierarchy

## See Also

Series2D Members
MindFusion.Charting Namespace

# Series2D Members

Represents a data series defined by two lists of values containing respectively X and Y coordinates.

The following tables list the members exposed by the Series2D type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | Series2D | Overloaded. |

## Public Properties

| | Name | Description |
|---|---|---|
| | dataChanged | Raised when the values in this series change. |
| | dimensions | Implements Series.Dimensions. Returns 2. |
| | labels | Gets or sets the data labels. |
| | size | Implements Series.Size. Returns the number of elements in XData list. |
| | supportedLabels | Implements Series.SupportedLabels. Gets or sets the kind of labels reported as supported by this series. |
| | title | Implements Series.Title. Gets or sets the title of this series. |
| | xData | Gets or sets the X coordinates of data items. |
| | yData | Gets or sets the Y coordinates of data items. |

## Public Methods

| | Name | Description |
|---|---|---|
| | getLabel | Implements Series.GetLabel. Returns an element of the Labels list. |
| | getValue | Implements Series.GetValue. Returns a value for the specified data item in this series. |
| | isEmphasized | Implements Series.IsEmphasized. Series2D always returns false. |
| | isSorted | Implements Series.IsSorted. Series2D always returns false. |
| | onDataChanged | Raises the DataChanged event. |

## See Also

Series2D Class
MindFusion.Charting Namespace

# Series2D Constructor

## Overload List

| Name | Description |
|---|---|
| Series2D | Initializes a new instance of the **Series2D** class. |
| Series2D | Initializes a new instance of the Series2D class. |

## See Also

Series2D Class
MindFusion.Charting Namespace

# Series2D Constructor

Initializes a new instance of the Series2D class.

**Namespace:** MindFusion.Charting
**File:** Series2D.js

## Syntax

**JavaScript**

```
function Series2D (xValues, yValues, labels)
```

### Parameters

*xValues*
Type: List | Array
A list of X coordinates.

*yValues*
Type: **List** | Array
A list of Y coordinates.

*labels*
Type: **List** | Array
A list of labels.

## Example

The following code creates a new Series2D using arrays with values:

**JavaScript**

```
var series = new Charting.Series2D([10, 20, 30, 40], [50, 40, 50, 5], ["Tomatoes",
"Cucumbers", "Peppers", "Lettuce"]);
series.title = "Traditional";
```

## See Also

Series2D Constructor Overload List
Series2D Class
MindFusion.Charting Namespace

# Series2D Constructor

Initializes a new instance of the Series2D class.

**Namespace:** MindFusion.Charting
**File:** Series2D.js

## Syntax

**JavaScript**

```
function Series2D (xValues, yValues, labels)
```

### Parameters

*xValues*
Type: List
A list of X coordinates.

*yValues*
Type: **List**
A list of Y coordinates.

*labels*
Type: **List**
A list of labels.

## Example

The following code creates a new Series2D instance using three List-s with values:

**JavaScript**

```
// create a sample series
var labels = new Collections.List(["Tomatoes", "Cucumbers", "Peppers", "Lettuce"]);


var series = new Charting.Series2D(new Collections.List([10, 20, 30, 40]), new
Collections.List([50, 40, 50, 5]), labels);
series.title = "Traditional";
```

## See Also

Series2D Constructor Overload List
Series2D Class
MindFusion.Charting Namespace

# Series2D Methods

# Series2D.getLabel Method

Implements Series.GetLabel. Returns an element of the Labels list.

**Namespace:** MindFusion.Charting
**File:** Series2D.js

## Syntax

```
JavaScript
function getLabel (index, kind)
```

### Parameters

*index*
Type: Number
An integer value specifying the index of a data item.

*kind*
Type: LabelKinds
A member of the LabelKinds enumeration.

### Return Value

Type: String
A string containing the item's label.

## See Also

Series2D Members
Series2D Class
MindFusion.Charting Namespace

# Series2D.getValue Method

Implements Series.GetValue. Returns a value for the specified data item in this series.

**Namespace:** MindFusion.Charting
**File:** Series2D.js

## Syntax

```
JavaScript
function getValue (index, dimension)
```

### Parameters

*index*
Type: Number
An integer value specifying the index of a data item.

*dimension*
Type: Number
An integer value specifying the dimension whose coordinate to return.

### Return Value

Type: Number
A double-precision number representing the value of a data item in the series.

## See Also

Series2D Members
Series2D Class
MindFusion.Charting Namespace

# Series2D.isEmphasized Method

Implements Series.IsEmphasized. Series2D always returns false.

**Namespace:** MindFusion.Charting
**File:** Series2D.js

## Syntax

**JavaScript**

```
function isEmphasized (index)
```

### Parameters

*index*
Type: Number
An integer value specifying the index of a data item.

### Return Value

Type: Boolean
true to emphasize the specified data item, or false otherwise.

## See Also

Series2D Members
Series2D Class
MindFusion.Charting Namespace

# Series2D.isSorted Method

Implements Series.IsSorted. Series2D always returns false.

**Namespace:** MindFusion.Charting
**File:** Series2D.js

## Syntax

**JavaScript**

```
function isSorted (dimension)
```

### Parameters

*dimension*
Type: Number
An integer value identifying the dimension.

### Return Value

Type: Boolean
true if the values are sorted in specified dimension, or false otherwise.

## See Also

Series2D Members
Series2D Class
MindFusion.Charting Namespace

# Series2D.onDataChanged Method

Raises the DataChanged event.

**Namespace:** MindFusion.Charting
**File:** Series2D.js

## Syntax

**JavaScript**

```
function onDataChanged ()
```

## See Also

Series2D Members
Series2D Class
MindFusion.Charting Namespace

# Series2D Properties

# Series2D.dataChanged Property

Raised when the values in this series change.

**Namespace:** MindFusion.Charting
**File:** Series2D.js

## Syntax

**JavaScript**

```
get dataChanged() {}
```

### Property Value

Type: EventDispatcher

## See Also

Series2D Members
Series2D Class
MindFusion.Charting Namespace

# Series2D.dimensions Property

Implements Series.Dimensions. Returns 2.

**Namespace:** MindFusion.Charting
**File:** Series2D.js

## Syntax

```
JavaScript
get dimensions() {}
```

### Property Value

Type: Number

## See Also

Series2D Members
Series2D Class
MindFusion.Charting Namespace

# Series2D.labels Property

Gets or sets the data labels.

**Namespace:** MindFusion.Charting
**File:** Series2D.js

## Syntax

```
JavaScript
get labels() {}
```

### Property Value

Type: List

## See Also

Series2D Members
Series2D Class
MindFusion.Charting Namespace

# Series2D.size Property

Implements Series.Size. Returns the number of elements in XData list.

**Namespace:** MindFusion.Charting
**File:** Series2D.js

## Syntax

```
JavaScript
get size() {}
```

### Property Value

Type: Number

## See Also

Series2D Members
Series2D Class
MindFusion.Charting Namespace

# Series2D.supportedLabels Property

Implements Series.SupportedLabels. Gets or sets the kind of labels reported as supported by this series.

**Namespace:** MindFusion.Charting
**File:** Series2D.js

## Syntax

```
JavaScript
get supportedLabels() {}
```

### Property Value

Type: LabelKinds

## See Also

Series2D Members
Series2D Class
MindFusion.Charting Namespace

# Series2D.title Property

Implements Series.Title. Gets or sets the title of this series.

**Namespace:** MindFusion.Charting
**File:** Series2D.js

## Syntax

```
JavaScript
get title() {}
```

### Property Value

Type: String

## See Also

Series2D Members
Series2D Class
MindFusion.Charting Namespace

## Series2D.xData Property

Gets or sets the X coordinates of data items.

**Namespace:** MindFusion.Charting
**File:** Series2D.js

### Syntax

```
JavaScript
get xData() {}
```

#### Property Value

Type: List

### See Also

Series2D Members
Series2D Class
MindFusion.Charting Namespace

## Series2D.yData Property

Gets or sets the Y coordinates of data items.

**Namespace:** MindFusion.Charting
**File:** Series2D.js

### Syntax

```
JavaScript
get yData() {}
```

#### Property Value

Type: List

### See Also

Series2D Members
Series2D Class
MindFusion.Charting Namespace

# Series3D Class

Represents a data series defined by three lists of values containing respectively X, Y and Z coordinates.

### Syntax

```
JavaScript
// class
Series3D.prototype = {}
```

## Inheritance Hierarchy

## See Also

Series3D Members
MindFusion.Charting Namespace

# Series3D Members

Represents a data series defined by three lists of values containing respectively X, Y and Z coordinates.

The following tables list the members exposed by the Series3D type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | Series3D | Overloaded. |

## Public Properties

| | Name | Description |
|---|---|---|
| | dataChanged | Raised when the values in this series change. |
| | dimensions | Implements Series.Dimensions. Returns 3. |
| | labels | Gets or sets the data labels. |
| | size | Implements Series.Size. Returns the number of elements in XData list. |
| | supportedLabels | Implements Series.SupportedLabels. Returns InnerLabel. |
| | title | Implements Series.Title. Gets or sets the title of this series. |
| | xData | Gets or sets the X coordinates of data items. |
| | yData | Gets or sets the Y coordinates of data items. |
| | zData | Gets or sets the Z coordinates of data items. |

## Public Methods

| | Name | Description |
|---|---|---|
| | getLabel | Implements Series.GetLabel. Returns an element of the Labels list. |
| | getValue | Implements Series.GetValue. Returns a value for the specified data item in this series. |
| | isEmphasized | Implements Series.IsEmphasized. Series3D always returns false. |
| | isSorted | Implements Series.IsSorted. Series3D always returns false. |
| | onDataChanged | Raises the DataChanged event. |

## See Also

Series3D Class
MindFusion.Charting Namespace

# Series3D Constructor

## Overload List

| Name | Description |
|---|---|
| Series3D | Initializes a new instance of the Series3D class. |
| Series3D | Initializes a new instance of the **Series3D** class. |

## See Also

Series3D Class
MindFusion.Charting Namespace

# Series3D Constructor

Initializes a new instance of the Series3D class.

**Namespace:** MindFusion.Charting
**File:** Series3D.js

## Syntax

**JavaScript**

```
function Series3D (xValues, yValues, zValues, labels)
```

### Parameters

*xValues*
Type: List | Array
A list of X coordinates.

*yValues*
Type: **List** | Array
A list of Y coordinates.

*zValues*
Type: **List** | Array
A list of Z coordinates.

*labels*
Type: **List** | Array
A list of labels.

## Example

The following code creates a new Series3D instance using four arrays with values:

**JavaScript**

```
var series = new Charting.Series3D(10, 20, 30, 40], [50, 40, 50, 5], [19, 17, 6, 12],
["Tomatoes", "Cucumbers", "Peppers", "Lettuce"]);
series.title = "Vegetables";
```

## See Also

Series3D Constructor Overload List
Series3D Class
MindFusion.Charting Namespace

# Series3D Constructor

Initializes a new instance of the Series3D class.

**Namespace:** MindFusion.Charting
**File:** Series3D.js

## Syntax

**JavaScript**

```
function Series3D (xValues, yValues, zValues, labels)
```

### Parameters

*xValues*
Type: List
A list of X coordinates.

*yValues*
Type: **List**
A list of Y coordinates.

*zValues*
Type: **List**
A list of Z coordinates.

*labels*
Type: **List**
A list of labels.

## Example

The following code creates a new Series3D instance using four List-s with values:

**JavaScript**

```
// create a sample series
var labels = new Collections.List(["Tomatoes", "Cucumbers", "Peppers", "Lettuce"]);


var series = new Charting.Series3D(new Collections.List([10, 20, 30, 40]), new
Collections.List([50, 40, 50, 5]), new Collections.List([19, 17, 6, 12]),  labels);
series.title = "Vegetables";
```

## See Also

Series3D Constructor Overload List
Series3D Class
MindFusion.Charting Namespace

# Series3D Methods

## Series3D.getLabel Method

Implements Series.GetLabel. Returns an element of the Labels list.

**Namespace:** MindFusion.Charting
**File:** Series3D.js

### Syntax

**JavaScript**

```
function getLabel (index, kind)
```

#### Parameters

*index*
Type: Number
An integer value specifying the index of a data item.

*kind*
Type: LabelKinds
A member of the LabelKinds enumeration.

#### Return Value

Type: String
A string containing the item's label.

### See Also

Series3D Members
Series3D Class
MindFusion.Charting Namespace

## Series3D.getValue Method

Implements Series.GetValue. Returns a value for the specified data item in this series.

**Namespace:** MindFusion.Charting
**File:** Series3D.js

### Syntax

**JavaScript**

```
function getValue (index, dimension)
```

#### Parameters

*index*
Type: Number
An integer value specifying the index of a data item.

*dimension*
Type: Number
An integer value specifying the dimension whose coordinate to return.

### Return Value

Type: Number
A double-precision number representing the value of a data item in the series.

## See Also

Series3D Members
Series3D Class
MindFusion.Charting Namespace

# Series3D.isEmphasized Method

Implements Series.IsEmphasized. Series3D always returns false.

**Namespace:** MindFusion.Charting
**File:** Series3D.js

## Syntax

**JavaScript**

```
function isEmphasized (index)
```

### Parameters

*index*
Type: Number
An integer value specifying the index of a data item.

### Return Value

Type: Boolean
true to emphasize the specified data item, or false otherwise.

## See Also

Series3D Members
Series3D Class
MindFusion.Charting Namespace

# Series3D.isSorted Method

Implements Series.IsSorted. Series3D always returns false.

**Namespace:** MindFusion.Charting
**File:** Series3D.js

## Syntax

**JavaScript**

```
function isSorted (dimension)
```

### Parameters

*dimension*

Type: Number
An integer value identifying the dimension.

### Return Value

Type: Boolean
true if the values are sorted in specified dimension, or false otherwise.

## See Also

Series3D Members
Series3D Class
MindFusion.Charting Namespace

# Series3D.onDataChanged Method

Raises the DataChanged event.

**Namespace:** MindFusion.Charting
**File:** Series3D.js

## Syntax

**JavaScript**

```
function onDataChanged ()
```

## See Also

Series3D Members
Series3D Class
MindFusion.Charting Namespace

# Series3D Properties

# Series3D.dataChanged Property

Raised when the values in this series change.

**Namespace:** MindFusion.Charting
**File:** Series3D.js

## Syntax

**JavaScript**

```
get dataChanged() {}
```

### Property Value

Type: EventDispatcher

## See Also

Series3D Members
Series3D Class

MindFusion.Charting Namespace

## Series3D.dimensions Property

Implements Series.Dimensions. Returns 3.

**Namespace:** MindFusion.Charting
**File:** Series3D.js

### Syntax

```
JavaScript
get dimensions() {}
```

#### Property Value

Type: Number

### See Also

Series3D Members
Series3D Class
MindFusion.Charting Namespace

## Series3D.labels Property

Gets or sets the data labels.

**Namespace:** MindFusion.Charting
**File:** Series3D.js

### Syntax

```
JavaScript
get labels() {}
```

#### Property Value

Type: List

### See Also

Series3D Members
Series3D Class
MindFusion.Charting Namespace

## Series3D.size Property

Implements Series.Size. Returns the number of elements in XData list.

**Namespace:** MindFusion.Charting
**File:** Series3D.js

### Syntax

```
JavaScript
get size() {}
```

### Property Value

Type: Number

## See Also

Series3D Members
Series3D Class
MindFusion.Charting Namespace

# Series3D.supportedLabels Property

Implements Series.SupportedLabels. Returns InnerLabel.

**Namespace:** MindFusion.Charting
**File:** Series3D.js

## Syntax

```
JavaScript
get supportedLabels() {}
```

### Property Value

Type: LabelKinds

## See Also

Series3D Members
Series3D Class
MindFusion.Charting Namespace

# Series3D.title Property

Implements Series.Title. Gets or sets the title of this series.

**Namespace:** MindFusion.Charting
**File:** Series3D.js

## Syntax

```
JavaScript
get title() {}
```

### Property Value

Type: String

## See Also

Series3D Members
Series3D Class
MindFusion.Charting Namespace

## Series3D.xData Property

Gets or sets the X coordinates of data items.

**Namespace:** MindFusion.Charting
**File:** Series3D.js

### Syntax

```
JavaScript
get xData() {}
```

#### Property Value

Type: List

### See Also

Series3D Members
Series3D Class
MindFusion.Charting Namespace

## Series3D.yData Property

Gets or sets the Y coordinates of data items.

**Namespace:** MindFusion.Charting
**File:** Series3D.js

### Syntax

```
JavaScript
get yData() {}
```

#### Property Value

Type: List

### See Also

Series3D Members
Series3D Class
MindFusion.Charting Namespace

## Series3D.zData Property

Gets or sets the Z coordinates of data items.

**Namespace:** MindFusion.Charting
**File:** Series3D.js

## Syntax

```
JavaScript
get zData() {}
```

### Property Value

Type: List

## See Also

Series3D Members
Series3D Class
MindFusion.Charting Namespace

# SeriesContainer Interface

Defines an interface for enumerating all series associated with a component of the dashboard.

## Syntax

```
JavaScript
// interface
SeriesContainer.prototype = {}
```

## See Also

SeriesContainer Members
MindFusion.Charting Namespace

## SeriesContainer Members

Defines an interface for enumerating all series associated with a component of the dashboard.

The following tables list the members exposed by the SeriesContainer type.

## Public Methods

| | Name | Description |
|---|---|---|
| | domainDimension | Gets the index of the dimension whose set of values should be treated as domain of plotted function. |
| | enumSeries | Enumerates the series associated with this component. |

## See Also

SeriesContainer Interface
MindFusion.Charting Namespace

# SeriesContainer Methods

## SeriesContainer.domainDimension Method

Gets the index of the dimension whose set of values should be treated as domain of plotted function.

**Namespace:** MindFusion.Charting
**File:** SeriesContainer.js

### Syntax

```
JavaScript
function domainDimension (series)
```

#### Parameters

*series*
Type: Series
A Series in this container.

### See Also

SeriesContainer Members
SeriesContainer Interface
MindFusion.Charting Namespace

## SeriesContainer.enumSeries Method

Enumerates the series associated with this component.

**Namespace:** MindFusion.Charting
**File:** SeriesContainer.js

### Syntax

```
JavaScript
function enumSeries ()
```

### See Also

SeriesContainer Members
SeriesContainer Interface
MindFusion.Charting Namespace

# SeriesRenderer Class

Base class for renderer objects that draw data series inside a Plot.

### Syntax

```
JavaScript
```

```
// class
SeriesRenderer.prototype = {}
```

## Inheritance Hierarchy

## See Also

SeriesRenderer Members
MindFusion.Charting Namespace

# SeriesRenderer Members

Base class for renderer objects that draw data series inside a Plot.

The following tables list the members exposed by the SeriesRenderer type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | SeriesRenderer | Initializes a new instance of the SeriesRenderer class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | labelBrush | Gets or sets the Brush that should be used to draw labels of data items. |
|  | labelFontName | Gets or sets the name of font that should be used to draw labels of data items. |
|  | labelFontSize | Gets or sets the size of font that should be used to draw labels of data items. |
|  | labelFontStyle | Gets or sets the style of font that should be used to draw labels of data items. |
|  | propertyChanged | Occurs when a property value changes. |
|  | seriesStyle | A SeriesStyle instance specifying the appearance of series associated with this renderer. |
|  | showDataLabels | Gets or sets the kind of data labels to draw. |
|  | showHighlight | Gets or sets a value indicating whether to show highlights on data items. |
|  | showToolTips | Gets or sets a value indicating whether to show tooltips. |

## Public Methods

| | Name | Description |
|---|---|---|
|  | adjustDataRange | Adjusts the data range of associated Axis objects after initial measure pass. |

| | | |
|---|---|---|
| | domainDimension | Implements the SeriesContainer interface. Gets the index of the dimension whose set of values should be treated as domain of plotted function. |
| | draw | Draws the series data in specified RenderContext. |
| | drawHighlight | Draws highlighted data item in specified RenderContext. |
| | effectiveFill | Gets the brush that should be used to fill the representation of specified data item. |
| | effectiveLabelBrush | Gets the System.Drawing.Brush that should be used to draw labels of data items. |
| | effectiveLabelFont | Gets the Font that should be used to draw labels of data items. |
| | effectiveStroke | Gets the brush that should be used to stroke the representation of specified data item. |
| | effectiveStrokeDashStyle | Gets the dash style of the pen that should be used to stroke the representation of specified data item. |
| | effectiveStrokePen | Gets the Pen that should be used to stroke the representation of specified data item. |
| | effectiveStrokeThickness | Gets the thickness of the pen that should be used to stroke the representation of specified data item. |
| | hitTest | Hit-tests the visual representation of the series for a data item. |
| | measureDataRange | Measures the data range of rendered series and assigns it to the associated Axis objects if their MinValue and MaxValue are not set. |
| | onPropertyChanged | Raises the PropertyChanged event. |
| | onSeriesDataChanged | Called when Series raise their DataChanged event. |
| | searchStyle | Returns a SeriesStyle object that meets specified criteria. |
| | setToolTip | Sets tooltip text and location. |
| | startMeasureData | Called to reset measure accumulators at the beginning of a measure pass. |
| | subscribe | Subscribes to the DataChanged event of specified Series. |
| | unsubscribe | Unsubscribes from the DataChanged event of specified Series. |

## See Also

SeriesRenderer Class
MindFusion.Charting Namespace

## SeriesRenderer Constructor

Initializes a new instance of the SeriesRenderer class.

**Namespace:** MindFusion.Charting
**File:** SeriesRenderer.js

## Syntax

```
JavaScript
function SeriesRenderer ()
```

## See Also

SeriesRenderer Class
MindFusion.Charting Namespace

# SeriesRenderer Methods

# SeriesRenderer.adjustDataRange Method

Adjusts the data range of associated Axis objects after initial measure pass.

**Namespace:** MindFusion.Charting
**File:** SeriesRenderer.js

## Syntax

```
JavaScript
function adjustDataRange (context)
```

### Parameters

*context*
Type: void
A RenderContext instance.

## See Also

SeriesRenderer Members
SeriesRenderer Class
MindFusion.Charting Namespace

# SeriesRenderer.domainDimension Method

Implements the SeriesContainer interface. Gets the index of the dimension whose set of values should be treated as domain of plotted function.

**Namespace:** MindFusion.Charting
**File:** SeriesRenderer.js

## Syntax

```
JavaScript
function domainDimension (series)
```

### Parameters

*series*
Type: Series
A Series in this container.

## See Also

SeriesRenderer Members
SeriesRenderer Class
MindFusion.Charting Namespace

# SeriesRenderer.draw Method

Draws the series data in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** SeriesRenderer.js

## Syntax

**JavaScript**

```
function draw (context)
```

### Parameters

*context*
Type: void
A RenderContext instance.

## See Also

SeriesRenderer Members
SeriesRenderer Class
MindFusion.Charting Namespace

# SeriesRenderer.drawHighlight Method

Draws highlighted data item in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** SeriesRenderer.js

## Syntax

**JavaScript**

```
function drawHighlight (context, hitResult)
```

### Parameters

*context*
Type: void
A RenderContext instance.

*hitResult*

Type: void
A HitResult instance identifying highlighted data item.

## See Also

SeriesRenderer Members
SeriesRenderer Class
MindFusion.Charting Namespace

# SeriesRenderer.effectiveFill Method

Gets the brush that should be used to fill the representation of specified data item.

**Namespace:** MindFusion.Charting
**File:** SeriesRenderer.js

## Syntax

**JavaScript**

```
function effectiveFill (seriesIndex, dataIndex, context)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series within the list of series rendered by this SeriesRenderer.

*dataIndex*
Type: Number
An integer index of the data item in specified series.

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: Brush
A Brush instance.

## See Also

SeriesRenderer Members
SeriesRenderer Class
MindFusion.Charting Namespace

# SeriesRenderer.effectiveLabelBrush Method

Gets the System.Drawing.Brush that should be used to draw labels of data items.

**Namespace:** MindFusion.Charting
**File:** SeriesRenderer.js

## Syntax

**JavaScript**

```
function effectiveLabelBrush (context)
```

### Parameters

*context*
Type: void
A RenderContext instance used to find styles and theme from dashboard hierarchy.

## See Also

SeriesRenderer Members
SeriesRenderer Class
MindFusion.Charting Namespace

# SeriesRenderer.effectiveLabelFont Method

Gets the Font that should be used to draw labels of data items.

**Namespace:** MindFusion.Charting
**File:** SeriesRenderer.js

## Syntax

**JavaScript**

```
function effectiveLabelFont (context)
```

### Parameters

*context*
Type: void
A RenderContext instance used to find styles and theme from dashboard hierarchy.

## See Also

SeriesRenderer Members
SeriesRenderer Class
MindFusion.Charting Namespace

# SeriesRenderer.effectiveStroke Method

Gets the brush that should be used to stroke the representation of specified data item.

**Namespace:** MindFusion.Charting
**File:** SeriesRenderer.js

## Syntax

**JavaScript**

```
function effectiveStroke (seriesIndex, dataIndex, context)
```

### Parameters

*seriesIndex*

Type: Number
An integer index of a Series within the list of series rendered by this SeriesRenderer.

*dataIndex*
Type: Number
An integer index of the data item in specified series.

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: Brush
A Brush instance.

## See Also

SeriesRenderer Members
SeriesRenderer Class
MindFusion.Charting Namespace

# SeriesRenderer.effectiveStrokeDashStyle Method

Gets the dash style of the pen that should be used to stroke the representation of specified data item.

**Namespace:** MindFusion.Charting
**File:** SeriesRenderer.js

## Syntax

**JavaScript**

```
function effectiveStrokeDashStyle (seriesIndex, dataIndex, context)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series within the list of series rendered by this SeriesRenderer.

*dataIndex*
Type: Number
An integer index of the data item in specified series.

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: DashStyle
A member of the DashStyle enumeration.

## See Also

SeriesRenderer Members
SeriesRenderer Class
MindFusion.Charting Namespace

# SeriesRenderer.effectiveStrokePen Method

Gets the Pen that should be used to stroke the representation of specified data item.

**Namespace:** MindFusion.Charting
**File:** SeriesRenderer.js

## Syntax

```
JavaScript
function effectiveStrokePen (seriesIndex, dataIndex, context)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series within the list of series rendered by this SeriesRenderer.

*dataIndex*
Type: Number
An integer index of the data item in specified series.

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: Pen
A Pen instance.

## See Also

SeriesRenderer Members
SeriesRenderer Class
MindFusion.Charting Namespace

# SeriesRenderer.effectiveStrokeThickness Method

Gets the thickness of the pen that should be used to stroke the representation of specified data item.

**Namespace:** MindFusion.Charting
**File:** SeriesRenderer.js

## Syntax

```
JavaScript
function effectiveStrokeThickness (seriesIndex, dataIndex, context)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series within the list of series rendered by this SeriesRenderer.

*dataIndex*
Type: Number
An integer index of the data item in specified series.

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: Number
A number value.

## See Also

SeriesRenderer Members
SeriesRenderer Class
MindFusion.Charting Namespace

# SeriesRenderer.hitTest Method

Hit-tests the visual representation of the series for a data item.

**Namespace:** MindFusion.Charting
**File:** SeriesRenderer.js

## Syntax

**JavaScript**

```
function hitTest (context, location)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*location*
Type: Point
A Point specifying where to look for data items.

### Return Value

Type: HitResult
A HitResult instance identifying the found data item.

## See Also

SeriesRenderer Members
SeriesRenderer Class
MindFusion.Charting Namespace

# SeriesRenderer.measureDataRange Method

Measures the data range of rendered series and assigns it to the associated Axis objects if their MinValue and MaxValue are not set.

**Namespace:** MindFusion.Charting
**File:** SeriesRenderer.js

## Syntax

```
JavaScript
function measureDataRange (context)
```

### Parameters

*context*
Type: void
A RenderContext instance.

## See Also

SeriesRenderer Members
SeriesRenderer Class
MindFusion.Charting Namespace

# SeriesRenderer.onPropertyChanged Method

Raises the PropertyChanged event.

**Namespace:** MindFusion.Charting
**File:** SeriesRenderer.js

## Syntax

```
JavaScript
function onPropertyChanged (propertyName)
```

### Parameters

*propertyName*
Type: void
Specifies the name of changed property.

## See Also

SeriesRenderer Members
SeriesRenderer Class
MindFusion.Charting Namespace

# SeriesRenderer.onSeriesDataChanged Method

Called when Series raise their DataChanged event.

**Namespace:** MindFusion.Charting
**File:** SeriesRenderer.js

## Syntax

```
JavaScript
function onSeriesDataChanged (sender, e)
```

### Parameters

*sender*

Type: Object
The Series raising the event.

*e*
Type: EventArgs
An EventArgs instance.

## See Also

SeriesRenderer Members
SeriesRenderer Class
MindFusion.Charting Namespace

# SeriesRenderer.searchStyle Method

Returns a SeriesStyle object that meets specified criteria.

**Namespace:** MindFusion.Charting
**File:** SeriesRenderer.js

## Syntax

**JavaScript**

```
function searchStyle (check, context)
```

### Parameters

*check*
Type: void
A StyleCheck delegate that tests SeriesStyle for some criteria.

*context*
Type: void
A RenderContext instance used to access styles and theme from the dashboard hierarchy.

## See Also

SeriesRenderer Members
SeriesRenderer Class
MindFusion.Charting Namespace

# SeriesRenderer.setToolTip Method

Sets tooltip text and location.

**Namespace:** MindFusion.Charting
**File:** SeriesRenderer.js

## Syntax

**JavaScript**

```
function setToolTip (text, location)
```

### Parameters

*text*

Type: void
A string specifying tooltip text.

*location*
Type: void
A Point specifying tooltip location.

## See Also

SeriesRenderer Members
SeriesRenderer Class
MindFusion.Charting Namespace

# SeriesRenderer.startMeasureData Method

Called to reset measure accumulators at the beginning of a measure pass.

**Namespace:** MindFusion.Charting
**File:** SeriesRenderer.js

## Syntax

**JavaScript**

```
function startMeasureData (context)
```

### Parameters

*context*
Type: void
A RenderContext instance.

## See Also

SeriesRenderer Members
SeriesRenderer Class
MindFusion.Charting Namespace

# SeriesRenderer.subscribe Method

Subscribes to the DataChanged event of specified Series.

**Namespace:** MindFusion.Charting
**File:** SeriesRenderer.js

## Syntax

**JavaScript**

```
function subscribe (series)
```

### Parameters

*series*
Type: Series
A Series instance.

## See Also

SeriesRenderer Members
SeriesRenderer Class
MindFusion.Charting Namespace

# SeriesRenderer.unsubscribe Method

Unsubscribes from the DataChanged event of specified Series.

**Namespace:** MindFusion.Charting
**File:** SeriesRenderer.js

## Syntax

**JavaScript**

```
function unsubscribe (series)
```

### Parameters

*series*
Type: Series
A Series instance.

## See Also

SeriesRenderer Members
SeriesRenderer Class
MindFusion.Charting Namespace

# SeriesRenderer Properties

# SeriesRenderer.labelBrush Property

Gets or sets the Brush that should be used to draw labels of data items.

**Namespace:** MindFusion.Charting
**File:** SeriesRenderer.js

## Syntax

**JavaScript**

```
get labelBrush() {}
```

### Property Value

Type: Brush

## See Also

SeriesRenderer Members
SeriesRenderer Class
MindFusion.Charting Namespace

## SeriesRenderer.labelFontName Property

Gets or sets the name of font that should be used to draw labels of data items.

**Namespace:** MindFusion.Charting
**File:** SeriesRenderer.js

### Syntax

```
JavaScript
get labelFontName() {}
```

#### Property Value

Type: String

### See Also

SeriesRenderer Members
SeriesRenderer Class
MindFusion.Charting Namespace

## SeriesRenderer.labelFontSize Property

Gets or sets the size of font that should be used to draw labels of data items.

**Namespace:** MindFusion.Charting
**File:** SeriesRenderer.js

### Syntax

```
JavaScript
get labelFontSize() {}
```

#### Property Value

Type: Number

### See Also

SeriesRenderer Members
SeriesRenderer Class
MindFusion.Charting Namespace

## SeriesRenderer.labelFontStyle Property

Gets or sets the style of font that should be used to draw labels of data items.

**Namespace:** MindFusion.Charting
**File:** SeriesRenderer.js

### Syntax

```
JavaScript
```

```
get labelFontStyle() {}
```

### Property Value

Type: FontStyle

## See Also

SeriesRenderer Members
SeriesRenderer Class
MindFusion.Charting Namespace

# SeriesRenderer.propertyChanged Property

Occurs when a property value changes.

**Namespace:** MindFusion.Charting
**File:** SeriesRenderer.js

## Syntax

**JavaScript**

```
get propertyChanged() {}
```

### Property Value

Type: PropertyChangedEventDispatcher

## See Also

SeriesRenderer Members
SeriesRenderer Class
MindFusion.Charting Namespace

# SeriesRenderer.seriesStyle Property

A SeriesStyle instance specifying the appearance of series associated with this renderer.

**Namespace:** MindFusion.Charting
**File:** SeriesRenderer.js

## Syntax

**JavaScript**

```
get seriesStyle() {}
```

### Property Value

Type: SeriesStyle

## See Also

SeriesRenderer Members
SeriesRenderer Class

MindFusion.Charting Namespace

## SeriesRenderer.showDataLabels Property

Gets or sets the kind of data labels to draw.

**Namespace:** MindFusion.Charting
**File:** SeriesRenderer.js

### Syntax

JavaScript

```
get showDataLabels() {}
```

#### Property Value

Type: Boolean

### See Also

SeriesRenderer Members
SeriesRenderer Class
MindFusion.Charting Namespace

## SeriesRenderer.showHighlight Property

Gets or sets a value indicating whether to show highlights on data items.

**Namespace:** MindFusion.Charting
**File:** SeriesRenderer.js

### Syntax

JavaScript

```
get showHighlight() {}
```

#### Property Value

Type: Boolean

### See Also

SeriesRenderer Members
SeriesRenderer Class
MindFusion.Charting Namespace

## SeriesRenderer.showToolTips Property

Gets or sets a value indicating whether to show tooltips.

**Namespace:** MindFusion.Charting
**File:** SeriesRenderer.js

### Syntax

| JavaScript |
| --- |
| `get showToolTips() {}` |

### Property Value

Type: Boolean

## See Also

SeriesRenderer Members
SeriesRenderer Class
MindFusion.Charting Namespace

# SeriesStyle Interface

Defines appearance attributes of series elements.

## Syntax

| JavaScript |
| --- |
| `// interface`<br>`SeriesStyle.prototype = {}` |

## See Also

SeriesStyle Members
MindFusion.Charting Namespace

### SeriesStyle Members

Defines appearance attributes of series elements.

The following tables list the members exposed by the SeriesStyle type.

## Public Methods

| | Name | Description |
| --- | --- | --- |
| | fill | Returns a Brush that should be used to fill specified element of specified series. |
| | stroke | Returns a Brush that should be used to stroke specified element of specified series. |
| | strokeDashStyle | Returns the dash style of the stroke used to draw specified element of specified series. |
| | strokeThickness | Returns the thickness of the stroke used to draw specified element of specified series. |

## See Also

SeriesStyle Interface
MindFusion.Charting Namespace

## SeriesStyle Methods

## SeriesStyle.fill Method

Returns a Brush that should be used to fill specified element of specified series.

**Namespace:** MindFusion.Charting
**File:** SeriesStyle.js

## Syntax

**JavaScript**

```
function fill (seriesIndex, dataIndex)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series.

*dataIndex*
Type: Number
An integer index of a data item in specified series.

### Return Value

Type: Brush
A Brush instance.

## See Also

SeriesStyle Members
SeriesStyle Interface
MindFusion.Charting Namespace

## SeriesStyle.stroke Method

Returns a Brush that should be used to stroke specified element of specified series.

**Namespace:** MindFusion.Charting
**File:** SeriesStyle.js

## Syntax

**JavaScript**

```
function stroke (seriesIndex, dataIndex)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series.

*dataIndex*
Type: Number
An integer index of a data item in specified series.

### Return Value

Type: Brush
A Brush instance.

## See Also

SeriesStyle Members
SeriesStyle Interface
MindFusion.Charting Namespace

# SeriesStyle.strokeDashStyle Method

Returns the dash style of the stroke used to draw specified element of specified series.

**Namespace:** MindFusion.Charting
**File:** SeriesStyle.js

## Syntax

**JavaScript**

```
function strokeDashStyle (seriesIndex, dataIndex)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series.

*dataIndex*
Type: Number
An integer index of a data item in specified series.

### Return Value

Type: DashStyle
A nullable DashStyle value specifying stroke dash style.

## See Also

SeriesStyle Members
SeriesStyle Interface
MindFusion.Charting Namespace

# SeriesStyle.strokeThickness Method

Returns the thickness of the stroke used to draw specified element of specified series.

**Namespace:** MindFusion.Charting
**File:** SeriesStyle.js

## Syntax

JavaScript

```
function strokeThickness (seriesIndex, dataIndex)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series.

*dataIndex*
Type: Number
An integer index of a data item in specified series.

### Return Value

Type: Number
A nullable number value specifying stroke thickness.

## See Also

SeriesStyle Members
SeriesStyle Interface
MindFusion.Charting Namespace

# SimpleSeries Class

Represents one-dimensional series defined by e list of data values and a list of labels.

## Syntax

JavaScript

```
// class
SimpleSeries.prototype = {}
```

## Inheritance Hierarchy

## See Also

SimpleSeries Members
MindFusion.Charting Namespace

# SimpleSeries Members

Represents one-dimensional series defined by e list of data values and a list of labels.

The following tables list the members exposed by the SimpleSeries type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | SimpleSeries | Overloaded. |

## Public Properties

| | Name | Description |
|---|---|---|
| | data | Gets or sets the data values contained in this series. |
| | dataChanged | Raised when the values in this series change. |
| | dimensions | Implements Series.Dimensions. Returns 1. |
| | size | Implements Series.Size. Returns the number of elements in Data list. |
| | supportedLabels | Implements Series.SupportedLabels. Returns InnerLabel. |
| | title | Implements Series.Title. Gets or sets the title of this series. |

## Public Methods

| | Name | Description |
|---|---|---|
| | getLabel | Implements Series.GetLabel. Returns an element of the labels list. |
| | getValue | Implements Series.GetValue. Returns a value for the specified data item in this series. |
| | isEmphasized | Implements Series.IsEmphasized. Gets a value indicating whether the specified data item should be emphasized by the SeriesRenderer. |
| | isSorted | Implements Series.IsSorted. SimpleSeries always returns false. |
| | onDataChanged | Raises the DataChanged event. |

## See Also

SimpleSeries Class
MindFusion.Charting Namespace

# SimpleSeries Constructor

## Overload List

| Name | Description |
|---|---|
| SimpleSeries | Initializes a new instance of the SimpleSeries class. |
| SimpleSeries | Initializes a new instance of the **SimpleSeries** class. |
| SimpleSeries | Initializes a new instance of the **SimpleSeries** class. |

## See Also

SimpleSeries Class
MindFusion.Charting Namespace

# SimpleSeries Constructor

Initializes a new instance of the SimpleSeries class.

**Namespace:** MindFusion.Charting
**File:** SimpleSeries.js

## Syntax

**JavaScript**

```
function SimpleSeries (values, labels)
```

### Parameters

*values*

Type: List | Array
A list of data values.

*labels*

Type: **List** | Array
A list of labels.

## Example

The following code creates a SimpleSeries and assigns it to the series property of a FunnelChart. The **SimpleSeries** uses two arrays:

**JavaScript**

```
// create sample data

funnelChart.series = new Charting.SimpleSeries([70, 60, 50, 30, 20, 15, 10, 4], ["Unqualified
prospects", "Leads", "Initial communication",
 "Customer evaluation", "Negotiation", "Purchase order received", "Delivery", "Payment"]);
```

## See Also

SimpleSeries Constructor Overload List
SimpleSeries Class
MindFusion.Charting Namespace

# SimpleSeries Constructor

Initializes a new instance of the SimpleSeries class.

**Namespace:** MindFusion.Charting
**File:** SimpleSeries.js

## Syntax

**JavaScript**

```
function SimpleSeries (values, labels, [tooltips])
```

### Parameters

*values*
Type: List | Array
A list of data values.

*labels*
Type: List | Array
A list of labels.

*tooltips*
Optional.
Type: List | Array
A list of tooltips.

## Example

The following code creates a new SimpleSeries with 12 values and labels. It also sets the data values as tooltips:

**JavaScript**

```
var data = new Collections.List([2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24]);
var labels = new Collections.List([
 "Jan", "Feb", "Mar", "Apr", "May", "Jun",
 "Jul", "Aug", "Sep", "Oct", "Nov", "Dec"
]);
var series = new Charting.SimpleSeries(data, labels, data);
```

## See Also

SimpleSeries Constructor Overload List
SimpleSeries Class
MindFusion.Charting Namespace

# SimpleSeries Constructor

Initializes a new instance of the SimpleSeries class.

**Namespace:** MindFusion.Charting
**File:** SimpleSeries.js

## Syntax

**JavaScript**

```
function SimpleSeries (values, labels)
```

### Parameters

*values*
Type: List
A list of data values.

*labels*
Type: **List**
A list of labels.

## Example

The following code creates a SimpleSeries and assigns it to the series property of a FunnelChart. The **SimpleSeries** uses two Lists:

**JavaScript**

```
// create sample data
var values = new Collections.List([70, 60, 50, 30, 20, 15, 10, 4]);
var labels = new Collections.List(["Unqualified prospects", "Leads", "Initial communication",
 "Customer evaluation", "Negotiation", "Purchase order received", "Delivery", "Payment"]);
funnelChart.series = new Charting.SimpleSeries(values, labels);
```

## See Also

SimpleSeries Constructor Overload List
SimpleSeries Class
MindFusion.Charting Namespace

# SimpleSeries Methods

# SimpleSeries.getLabel Method

Implements Series.GetLabel. Returns an element of the labels list.

**Namespace:** MindFusion.Charting
**File:** SimpleSeries.js

## Syntax

**JavaScript**

```
function getLabel (index, kind)
```

### Parameters

*index*
Type: Number
An integer value specifying the index of a data item.

*kind*
Type: LabelKinds
A member of the LabelKinds enumeration.

### Return Value

Type: String
A string containing the item's label.

## See Also

SimpleSeries Members
SimpleSeries Class
MindFusion.Charting Namespace

# SimpleSeries.getValue Method

Implements Series.GetValue. Returns a value for the specified data item in this series.

**Namespace:** MindFusion.Charting
**File:** SimpleSeries.js

## Syntax

```
JavaScript
function getValue (index, dimension)
```

### Parameters

*index*
Type: Number
An integer value specifying the index of a data item.

*dimension*
Type: Number
An integer value specifying the dimension whose coordinate to return.

### Return Value

Type: Number
A double-precision number representing the value of a data item in the series.

## See Also

SimpleSeries Members
SimpleSeries Class
MindFusion.Charting Namespace

# SimpleSeries.isEmphasized Method

Implements Series.IsEmphasized. Gets a value indicating whether the specified data item should be emphasized by the SeriesRenderer.

**Namespace:** MindFusion.Charting
**File:** SimpleSeries.js

## Syntax

```
JavaScript
function isEmphasized (index)
```

### Parameters

*index*
Type: Number
An integer value specifying the index of a data item.

### Return Value

Type: Boolean
true to emphasize the specified data item, or false otherwise.

## See Also

SimpleSeries Members
SimpleSeries Class
MindFusion.Charting Namespace

## SimpleSeries.isSorted Method

Implements Series.IsSorted. SimpleSeries always returns false.

**Namespace:** MindFusion.Charting
**File:** SimpleSeries.js

### Syntax

```
JavaScript
function isSorted (dimension)
```

#### Parameters

*dimension*
Type: Number
An integer value identifying the dimension.

#### Return Value

Type: Boolean
true if the values are sorted in specified dimension, or false otherwise.

### See Also

SimpleSeries Members
SimpleSeries Class
MindFusion.Charting Namespace

## SimpleSeries.onDataChanged Method

Raises the DataChanged event.

**Namespace:** MindFusion.Charting
**File:** SimpleSeries.js

### Syntax

```
JavaScript
function onDataChanged ()
```

### See Also

SimpleSeries Members
SimpleSeries Class
MindFusion.Charting Namespace

## SimpleSeries Properties

# SimpleSeries.data Property

Gets or sets the data values contained in this series.

**Namespace:** MindFusion.Charting
**File:** SimpleSeries.js

## Syntax

```
JavaScript
get data() {}
```

### Property Value

Type: List

## See Also

SimpleSeries Members
SimpleSeries Class
MindFusion.Charting Namespace

# SimpleSeries.dataChanged Property

Raised when the values in this series change.

**Namespace:** MindFusion.Charting
**File:** SimpleSeries.js

## Syntax

```
JavaScript
get dataChanged() {}
```

### Property Value

Type: EventDispatcher

## See Also

SimpleSeries Members
SimpleSeries Class
MindFusion.Charting Namespace

# SimpleSeries.dimensions Property

Implements Series.Dimensions. Returns 1.

**Namespace:** MindFusion.Charting
**File:** SimpleSeries.js

## Syntax

```
JavaScript
```

```
get dimensions() {}
```

### Property Value

Type: Number

## See Also

SimpleSeries Members
SimpleSeries Class
MindFusion.Charting Namespace

# SimpleSeries.size Property

Implements Series.Size. Returns the number of elements in Data list.

**Namespace:** MindFusion.Charting
**File:** SimpleSeries.js

## Syntax

**JavaScript**

```
get size() {}
```

### Property Value

Type: Number

## See Also

SimpleSeries Members
SimpleSeries Class
MindFusion.Charting Namespace

# SimpleSeries.supportedLabels Property

Implements Series.SupportedLabels. Returns InnerLabel.

**Namespace:** MindFusion.Charting
**File:** SimpleSeries.js

## Syntax

**JavaScript**

```
get supportedLabels() {}
```

### Property Value

Type: LabelKinds

## See Also

SimpleSeries Members
SimpleSeries Class

## SimpleSeries.title Property

Implements Series.Title. Gets or sets the title of this series.

**Namespace:** MindFusion.Charting
**File:** SimpleSeries.js

### Syntax

**JavaScript**

```
get title() {}
```

#### Property Value

Type: String

### See Also

SimpleSeries Members
SimpleSeries Class
MindFusion.Charting Namespace

# StackRenderer Class

A base SeriesRenderer for stacked graphics.

### Syntax

**JavaScript**

```
// class
StackRenderer.prototype = {}
```

### Inheritance Hierarchy

### See Also

StackRenderer Members
MindFusion.Charting Namespace

## StackRenderer Members

A base SeriesRenderer for stacked graphics.

The following tables list the members exposed by the StackRenderer type.

### Public Constructors

| | Name | Description |
|---|---|---|
|  | StackRenderer | Initializes a new instance of the StackRenderer class. |

## Public Methods

| | Name | Description |
|---|---|---|
| | enumVisibleStackPoints | Enumerates the data values of rendered series mapped to plot 2D coordinates. |
| | enumVisibleStackRanges | Enumerates each series once providing a list of all visible points at each iteration. |
| | getPixel | Gets the coordinates of specified data item when mapped to a Plot2D pixel. |
| | hitTest | Renderer2D.HitTest override. Hit-tests the stack representations of data items. |
| | measureDataRange | Renderer2D.MeasureDataRange override. Measures the data range of rendered series and assigns it to the associated Axis objects if their MinValue and MaxValue are not set. |

## See Also

StackRenderer Class
MindFusion.Charting Namespace

# StackRenderer Constructor

Initializes a new instance of the StackRenderer class.

**Namespace:** MindFusion.Charting
**File:** StackRenderer.js

## Syntax

**JavaScript**

```
function StackRenderer (series)
```

### Parameters

*series*
Type: ObservableCollection

## See Also

StackRenderer Class
MindFusion.Charting Namespace

# StackRenderer Methods

# StackRenderer.enumVisibleStackPoints Method

Enumerates the data values of rendered series mapped to plot 2D coordinates.

**Namespace:** MindFusion.Charting
**File:** StackRenderer.js

## Syntax

**JavaScript**

```
function enumVisibleStackPoints (context, process)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*process*
Type: ProcessStackPoint
A ProcessStackPoint callback.

## Remarks

The signature of the ProcessStackPoint delegate is as follows:'function(seriesIndex, dataIndex, stackPoint, prevPoint)'.param {Number} seriesIndexAn integer index of series in Series list.param {Number} dataIndex An integer index of data item in specified series.param {MindFusion.Drawing.Point} stackPoint A Point instance containing the Plot2D coordinates corresponding to current data item.param {MindFusion.Drawing.Point} prevPoint A Point instance containing the Plot2D coordinates corresponding to previous data item.The method is not expected to return a value.

## See Also

StackRenderer Members
StackRenderer Class
MindFusion.Charting Namespace

# StackRenderer.enumVisibleStackRanges Method

Enumerates each series once providing a list of all visible points at each iteration.

**Namespace:** MindFusion.Charting
**File:** StackRenderer.js

## Syntax

**JavaScript**

```
function enumVisibleStackRanges (context, process)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*process*
Type: ProcessStackRange
A ProcessStackRange callback.

## Remarks

The signature of the ProcessStackRange delegate is as follows:'function(seriesIndex, points, prevPoints)'.param {Number} seriesIndex An integer index of series in Series list.param

{MindFusion.Charting.Collections.List<Point>} points A list of Point values containing the Plot2D coordinatescorresponding to data items in currently visible range.param {MindFusion.Charting.Collections.List<Point>} prevPoints A list of Point values containing the Plot2D coordinates corresponding to previous series in currently visible range.The method is not expected to return a value.

## See Also

StackRenderer Members
StackRenderer Class
MindFusion.Charting Namespace

# StackRenderer.getPixel Method

Gets the coordinates of specified data item when mapped to a Plot2D pixel.

**Namespace:** MindFusion.Charting
**File:** StackRenderer.js

## Syntax

**JavaScript**

```
function getPixel (seriesIndex, dataIndex, xAxis, yAxis, component)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series within the list of series rendered by this Renderer2D.

*dataIndex*
Type: Number
An integer index of the data item in specified series.

*xAxis*
Type: Axis
An Axis reference specifying the X axis.

*yAxis*
Type: Axis
An Axis reference specifying the Y axis.

*component*
Type: Component
An instance of Plot2D or derived class.

### Return Value

Type: Point
A Point containing Plot2D coordinates corresponding to specified data item.

## See Also

StackRenderer Members
StackRenderer Class
MindFusion.Charting Namespace

# StackRenderer.hitTest Method

Renderer2D.HitTest override. Hit-tests the stack representations of data items.

**Namespace:** MindFusion.Charting
**File:** StackRenderer.js

## Syntax

**JavaScript**

```
function hitTest (context, location)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*location*
Type: Point
A Point specifying where to look for a data point.

### Return Value

Type: HitResult
A HitResult instance identifying the found data item.

## See Also

StackRenderer Members
StackRenderer Class
MindFusion.Charting Namespace

# StackRenderer.measureDataRange Method

Renderer2D.MeasureDataRange override. Measures the data range of rendered series and assigns it to the associated Axis objects if their MinValue and MaxValue are not set.

**Namespace:** MindFusion.Charting
**File:** StackRenderer.js

## Syntax

**JavaScript**

```
function measureDataRange (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

StackRenderer Members
StackRenderer Class
MindFusion.Charting Namespace

# StepAreaRenderer Class

A SeriesRenderer that draws series as filled steps between data points.

## Syntax

| JavaScript |
| --- |
| `// class`<br>`StepAreaRenderer.prototype = {}` |

## Inheritance Hierarchy

## See Also

StepAreaRenderer Members
MindFusion.Charting Namespace

# StepAreaRenderer Members

A SeriesRenderer that draws series as filled steps between data points.

The following tables list the members exposed by the StepAreaRenderer type.

## Public Constructors

| | Name | Description |
| --- | --- | --- |
|  | StepAreaRenderer | Initializes a new instance of the StepAreaRenderer class. |

## Public Properties

| | Name | Description |
| --- | --- | --- |
|  | areaOpacity | Sets or sets the opacity of area polygons. |

## Public Methods

| | Name | Description |
| --- | --- | --- |
|  | draw | SeriesRenderer.Draw override. Draws the series data in specified RenderContext. |

## See Also

StepAreaRenderer Class
MindFusion.Charting Namespace

# StepAreaRenderer Constructor

Initializes a new instance of the StepAreaRenderer class.

**Namespace:** MindFusion.Charting
**File:** StepAreaRenderer.js

## Syntax

**JavaScript**

```
function StepAreaRenderer (series)
```

### Parameters

*series*
Type: ObservableCollection
A list of Series that should be rendered by this object.

## See Also

StepAreaRenderer Class
MindFusion.Charting Namespace

# StepAreaRenderer Methods

# StepAreaRenderer.draw Method

SeriesRenderer.Draw override. Draws the series data in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** StepAreaRenderer.js

## Syntax

**JavaScript**

```
function draw (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

StepAreaRenderer Members
StepAreaRenderer Class
MindFusion.Charting Namespace

# StepAreaRenderer Properties

## StepAreaRenderer.areaOpacity Property

Sets or sets the opacity of area polygons.

**Namespace:** MindFusion.Charting
**File:** StepAreaRenderer.js

### Syntax

```
JavaScript
get areaOpacity() {}
```

#### Property Value

Type: Number

### See Also

StepAreaRenderer Members
StepAreaRenderer Class
MindFusion.Charting Namespace

# StepRenderer Class

A SeriesRenderer that draws series as steps between data points.

### Syntax

```
JavaScript
// class
StepRenderer.prototype = {}
```

### Inheritance Hierarchy

### See Also

StepRenderer Members
MindFusion.Charting Namespace

## StepRenderer Members

A SeriesRenderer that draws series as steps between data points.

The following tables list the members exposed by the StepRenderer type.

### Public Constructors

| | Name | Description |
|---|---|---|
|  | StepRenderer | Initializes a new instance of the StepRenderer class. |

### Public Methods

| | Name | Description |
|---|---|---|
|  | draw | SeriesRenderer.Draw override. Draws the series data in specified RenderContext. |

## See Also

StepRenderer Class
MindFusion.Charting Namespace

# StepRenderer Constructor

Initializes a new instance of the StepRenderer class.

**Namespace:** MindFusion.Charting
**File:** StepRenderer.js

## Syntax

**JavaScript**

```
function StepRenderer (series)
```

### Parameters

*series*
Type: ObservableCollection
A list of Series that should be rendered by this object.

## See Also

StepRenderer Class
MindFusion.Charting Namespace

# StepRenderer Methods

# StepRenderer.draw Method

SeriesRenderer.Draw override. Draws the series data in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** StepRenderer.js

## Syntax

**JavaScript**

```
function draw (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

StepRenderer Members
StepRenderer Class
MindFusion.Charting Namespace

# StockPrice Class

Represents a data object, containing open, close, low and high values for a certain date.

## Syntax

**JavaScript**

```
// class
StockPrice.prototype = {}
```

## Inheritance Hierarchy

## See Also

StockPrice Members
MindFusion.Charting Namespace

### StockPrice Members

Represents a data object, containing open, close, low and high values for a certain date.

The following tables list the members exposed by the StockPrice type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | StockPrice | Initializes a new instance of the StockPrice class. |

## See Also

StockPrice Class
MindFusion.Charting Namespace

### StockPrice Constructor

Initializes a new instance of the StockPrice class.

**Namespace:** MindFusion.Charting
**File:** StockPriceSeries.js

## Syntax

**JavaScript**

```
function StockPrice (open, close, low, high, [date])
```

### Parameters

*open*
Type: Number
The open value.

*close*
Type: Number
The close value.

*low*
Type: Number
The low value.

*high*
Type: Number
The high value.

*date*
Optional.
Type: Date
The corresponding date.

## See Also

StockPrice Class
MindFusion.Charting Namespace

# StockPriceSeries Class

Represents a data series defined by a list of StockPrice objects.

## Syntax

**JavaScript**

```
// class
StockPriceSeries.prototype = {}
```

## Inheritance Hierarchy

## See Also

StockPriceSeries Members
MindFusion.Charting Namespace

### StockPriceSeries Members

Represents a data series defined by a list of StockPrice objects.

The following tables list the members exposed by the StockPriceSeries type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | StockPriceSeries | Overloaded. |

## Public Properties

| | Name | Description |
|---|---|---|
| | customDateTimeFormat | Gets or sets a custom format string for DateTime labels. |
| | dataChanged | Raised when the values in this series change. |
| | dateTimeFormat | Gets or sets a value indicating how to format DateTime values as labels. |
| | dimensions | Implements Series.Dimensions. Returns 5. |
| | labelPrefix | Gets or sets a prefix added in front of formatted DateTime labels. |
| | labelSuffix | Gets or sets a suffix appended to formatted DateTime labels. |
| | size | Implements Series.Size. Returns the number of elements in Values list. |
| | supportedLabels | Implements Series.SupportedLabels. Returns InnerLabel. |
| | title | Implements Series.Title. Gets or sets the title of this series. |
| | values | Gets or sets the series' points. |

## Public Methods

| | Name | Description |
|---|---|---|
| | getLabel | Implements Series.GetLabel. Returns DateTime value at specified index as an XAxisLabel. |
| | getValue | Implements Series.GetValue. Returns a coordinate of specified point in this series. |
| | isEmphasized | Implements Series.IsEmphasized. StockPriceSeries always returns false. |
| | isSorted | Implements Series.IsSorted. StockPriceSeries always returns false. |
| | onDataChanged | Raises the DataChanged event. |

## See Also

StockPriceSeries Class
MindFusion.Charting Namespace

# StockPriceSeries Constructor

## Overload List

| Name | Description |
|---|---|
| StockPriceSeries | Initializes a new instance of the StockPriceSeries class. |

| | |
|---|---|
| StockPriceSeries | Initializes a new instance of the **StockPriceSeries** class. |

## See Also

StockPriceSeries Class
MindFusion.Charting Namespace

# StockPriceSeries Constructor

Initializes a new instance of the StockPriceSeries class.

**Namespace:** MindFusion.Charting
**File:** StockPriceSeries.js

## Syntax

**JavaScript**

```
function StockPriceSeries (values)
```

### Parameters

*values*
Type: List | Array
A list of StockPrice objects.

## Example

The following code creates a new StockPriceSeries with an array of 5 StockPrice objects:

**JavaScript**

```
var stockData = [
    [15.99, 16.07, 15.80, 15.94],
    [15.93, 16.03, 15.89, 15.97],
    [15.97, 16.43, 15.94, 16.40],
    [16.36, 16.52, 16.34, 16.46],
    [16.44, 16.47, 16.23, 16.38]


];

// create sample data
var dataList = [];
for (var i = 0; i < 5; i++)
{
 var date = new Date(new Date().getFullYear(), new Date().getMonth());
        var dataItem = new Charting.StockPrice(stockData[i][0], stockData[i][3], stockData[i]
[2], stockData[i][1], new Date(date.setDate(1 + i)));
 dataList.push(dataItem);
}

var series = new StockPriceSeries(dataList);
```

## See Also

StockPriceSeries Constructor Overload List
StockPriceSeries Class
MindFusion.Charting Namespace

# StockPriceSeries Constructor

Initializes a new instance of the StockPriceSeries class.

**Namespace:** MindFusion.Charting
**File:** StockPriceSeries.js

## Syntax

**JavaScript**

```
function StockPriceSeries (values)
```

### Parameters

*values*

Type: List
A list of StockPrice objects.

## Example

The following code creates a new StockPriceSeries with a List of 5 StockPrice objects:

**JavaScript**

```
var stockData = [
    [15.99, 16.07, 15.80, 15.94],
    [15.93, 16.03, 15.89, 15.97],
    [15.97, 16.43, 15.94, 16.40],
    [16.36, 16.52, 16.34, 16.46],
    [16.44, 16.47, 16.23, 16.38]


];

// create sample data
var dataList = new Collections.List();
for (var i = 0; i < 5; i++)
{
 var date = new Date(new Date().getFullYear(), new Date().getMonth());
        var dataItem = new Charting.StockPrice(stockData[i][0], stockData[i][3], stockData[i]
[2], stockData[i][1], new Date(date.setDate(1 + i)));
 dataList.add(dataItem);
}

var series = new StockPriceSeries(dataList);
```

## See Also

StockPriceSeries Constructor Overload List
StockPriceSeries Class
MindFusion.Charting Namespace

# StockPriceSeries Methods

# StockPriceSeries.getLabel Method

Implements Series.GetLabel. Returns DateTime value at specified index as an XAxisLabel.

**Namespace:** MindFusion.Charting
**File:** StockPriceSeries.js

## Syntax

```
JavaScript
function getLabel (index, kind)
```

### Parameters

*index*
Type: Number
An integer value specifying the index of a data item.

*kind*
Type: LabelKinds
A member of the LabelKinds enumeration.

### Return Value

Type: String
A string containing the item's label.

## See Also

StockPriceSeries Members
StockPriceSeries Class
MindFusion.Charting Namespace

# StockPriceSeries.getValue Method

Implements Series.GetValue. Returns a coordinate of specified point in this series.

**Namespace:** MindFusion.Charting
**File:** StockPriceSeries.js

## Syntax

```
JavaScript
function getValue (index, dimension)
```

### Parameters

*index*
Type: Number
An integer value specifying the index of a data item.

*dimension*
Type: Number
An integer value specifying the dimension whose coordinate to return.

### Return Value

Type: Number
A double-precision number representing the value of a data item in the series.

## See Also

StockPriceSeries Members
StockPriceSeries Class

# StockPriceSeries.isEmphasized Method

Implements Series.IsEmphasized. StockPriceSeries always returns false.

**Namespace:** MindFusion.Charting
**File:** StockPriceSeries.js

## Syntax

**JavaScript**

```
function isEmphasized (index)
```

### Parameters

*index*
Type: Number
An integer value specifying the index of a data item.

### Return Value

Type: Boolean
true to emphasize the specified data item, or false otherwise.

## See Also

StockPriceSeries Members
StockPriceSeries Class
MindFusion.Charting Namespace

# StockPriceSeries.isSorted Method

Implements Series.IsSorted. StockPriceSeries always returns false.

**Namespace:** MindFusion.Charting
**File:** StockPriceSeries.js

## Syntax

**JavaScript**

```
function isSorted (dimension)
```

### Parameters

*dimension*
Type: Number
An integer value identifying the dimension.

### Return Value

Type: Boolean
true if the values are sorted in specified dimension, or false otherwise.

## See Also

StockPriceSeries Members
StockPriceSeries Class
MindFusion.Charting Namespace

## StockPriceSeries.onDataChanged Method

Raises the DataChanged event.

**Namespace:** MindFusion.Charting
**File:** StockPriceSeries.js

### Syntax

```
JavaScript
function onDataChanged ()
```

### See Also

StockPriceSeries Members
StockPriceSeries Class
MindFusion.Charting Namespace

## StockPriceSeries Properties

## StockPriceSeries.customDateTimeFormat Property

Gets or sets a custom format string for DateTime labels.

**Namespace:** MindFusion.Charting
**File:** StockPriceSeries.js

### Syntax

```
JavaScript
get customDateTimeFormat() {}
```

#### Property Value

Type: String

### See Also

StockPriceSeries Members
StockPriceSeries Class
MindFusion.Charting Namespace

## StockPriceSeries.dataChanged Property

Raised when the values in this series change.

**Namespace:** MindFusion.Charting
**File:** StockPriceSeries.js

## Syntax

```
JavaScript
get dataChanged() {}
```

### Property Value

Type: EventDispatcher

## See Also

StockPriceSeries Members
StockPriceSeries Class
MindFusion.Charting Namespace

# StockPriceSeries.dateTimeFormat Property

Gets or sets a value indicating how to format DateTime values as labels.

**Namespace:** MindFusion.Charting
**File:** StockPriceSeries.js

## Syntax

```
JavaScript
get dateTimeFormat() {}
```

### Property Value

Type: DateTimeFormat

## See Also

StockPriceSeries Members
StockPriceSeries Class
MindFusion.Charting Namespace

# StockPriceSeries.dimensions Property

Implements Series.Dimensions. Returns 5.

**Namespace:** MindFusion.Charting
**File:** StockPriceSeries.js

## Syntax

```
JavaScript
get dimensions() {}
```

### Property Value

Type: Number

## See Also

StockPriceSeries Members
StockPriceSeries Class
MindFusion.Charting Namespace

# StockPriceSeries.labelPrefix Property

Gets or sets a prefix added in front of formatted DateTime labels.

**Namespace:** MindFusion.Charting
**File:** StockPriceSeries.js

## Syntax

**JavaScript**

```
get labelPrefix() {}
```

### Property Value

Type: String

## See Also

StockPriceSeries Members
StockPriceSeries Class
MindFusion.Charting Namespace

# StockPriceSeries.labelSuffix Property

Gets or sets a suffix appended to formatted DateTime labels.

**Namespace:** MindFusion.Charting
**File:** StockPriceSeries.js

## Syntax

**JavaScript**

```
get labelSuffix() {}
```

### Property Value

Type: String

## See Also

StockPriceSeries Members
StockPriceSeries Class
MindFusion.Charting Namespace

# StockPriceSeries.size Property

Implements Series.Size. Returns the number of elements in Values list.

**Namespace:** MindFusion.Charting
**File:** StockPriceSeries.js

## Syntax

```
JavaScript
get size() {}
```

### Property Value

Type: Number

## See Also

StockPriceSeries Members
StockPriceSeries Class
MindFusion.Charting Namespace

# StockPriceSeries.supportedLabels Property

Implements Series.SupportedLabels. Returns InnerLabel.

**Namespace:** MindFusion.Charting
**File:** StockPriceSeries.js

## Syntax

```
JavaScript
get supportedLabels() {}
```

### Property Value

Type: LabelKinds

## See Also

StockPriceSeries Members
StockPriceSeries Class
MindFusion.Charting Namespace

# StockPriceSeries.title Property

Implements Series.Title. Gets or sets the title of this series.

**Namespace:** MindFusion.Charting
**File:** StockPriceSeries.js

## Syntax

```
JavaScript
get title() {}
```

### Property Value

Type: String

## See Also

StockPriceSeries Members
StockPriceSeries Class
MindFusion.Charting Namespace

### StockPriceSeries.values Property

Gets or sets the series' points.

**Namespace:** MindFusion.Charting
**File:** StockPriceSeries.js

## Syntax

**JavaScript**

```
get values() {}
```

#### Property Value

Type: List

## See Also

StockPriceSeries Members
StockPriceSeries Class
MindFusion.Charting Namespace

# TextRenderer Class

Draws text and labels inside chart components.

## Syntax

**JavaScript**

```
// class
TextRenderer.prototype = {}
```

## Inheritance Hierarchy

## See Also

TextRenderer Members
MindFusion.Charting Namespace

### TextRenderer Members

Draws text and labels inside chart components.

The following tables list the members exposed by the TextRenderer type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | TextRenderer | Initializes a new instance of the TextRenderer class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | labelFont | Gets or sets the Font used to draw labels. |
| | textBrush | Gets or sets the brush used to draw labels. |

## Public Methods

| | Name | Description |
|---|---|---|
| | drawLabelAtPoint | Draws the specified label centered at specified location. |
| | drawLabelInRect | Draws the specified label in specified layout rectangle. |
| | drawLeftFromPoint | Draws the specified label on the left side of the specified point. |
| | drawRightFromPoint | Draws the specified label on the right side of specified point. |

## See Also

TextRenderer Class
MindFusion.Charting Namespace

# TextRenderer Constructor

Initializes a new instance of the TextRenderer class.

**Namespace:** MindFusion.Charting
**File:** TextRenderer.js

## Syntax

**JavaScript**

```
function TextRenderer (context, font, textBrush)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

*font*
Type: Font
A Font instance.

*textBrush*
Type: Brush
A Brush instance.

## See Also

TextRenderer Class
MindFusion.Charting Namespace

# TextRenderer Methods

## TextRenderer.drawLabelAtPoint Method

Draws the specified label centered at specified location.

**Namespace:** MindFusion.Charting
**File:** TextRenderer.js

## Syntax

**JavaScript**

```
function drawLabelAtPoint (series, index, point, labelKind, [format])
```

### Parameters

*series*
Type: Series
A Series whose label to draw.

*index*
Type: Number
An integer index of the label.

*point*
Type: Point
A Point specifying the label's location.

*labelKind*
Type: LabelKinds
A member of the LabelKinds enumeration.

*format*
Optional.
Type: StringFormat
A StringFormat instance.

## See Also

TextRenderer Members
TextRenderer Class
MindFusion.Charting Namespace

## TextRenderer.drawLabelInRect Method

Draws the specified label in specified layout rectangle.

**Namespace:** MindFusion.Charting
**File:** TextRenderer.js

## Syntax

JavaScript

```
function drawLabelInRect (series, index, bounds, labelKind, [format])
```

### Parameters

*series*
Type: Series
A Series whose label to draw.

*index*
Type: Number
An integer index of the label.

*bounds*
Type: Rect
A RectangleF specifying the layout rectangle.

*labelKind*
Type: LabelKinds
A member of the LabelKinds enumeration.

*format*
Optional.
Type: StringFormat
A StringFormat instance.

## See Also

TextRenderer Members
TextRenderer Class
MindFusion.Charting Namespace

# TextRenderer.drawLeftFromPoint Method

Draws the specified label on the left side of the specified point.

**Namespace:** MindFusion.Charting
**File:** TextRenderer.js

## Syntax

JavaScript

```
function drawLeftFromPoint (series, index, point, labelKind)
```

### Parameters

*series*
Type: Series
A Series whose label to draw.

*index*
Type: Number
An integer index of the label.

*point*
Type: Point
A Point specifying the label's location.

*labelKind*

Type: LabelKinds
A member of the LabelKinds enumeration.

## See Also

TextRenderer Members
TextRenderer Class
MindFusion.Charting Namespace

# TextRenderer.drawRightFromPoint Method

Draws the specified label on the right side of specified point.

**Namespace:** MindFusion.Charting
**File:** TextRenderer.js

## Syntax

```
JavaScript
function drawRightFromPoint (series, index, point, labelKind)
```

### Parameters

*series*
Type: Series
A Series whose label to draw.

*index*
Type: Number
An integer index of the label.

*point*
Type: Point
A Point specifying the label's location.

*labelKind*
Type: LabelKinds
A member of the LabelKinds enumeration.

## See Also

TextRenderer Members
TextRenderer Class
MindFusion.Charting Namespace

# TextRenderer Properties

# TextRenderer.labelFont Property

Gets or sets the Font used to draw labels.

**Namespace:** MindFusion.Charting
**File:** TextRenderer.js

## Syntax

JavaScript

```
get labelFont() {}
```

### Property Value

Type: Font

## See Also

TextRenderer Members
TextRenderer Class
MindFusion.Charting Namespace

# TextRenderer.textBrush Property

Gets or sets the brush used to draw labels.

**Namespace:** MindFusion.Charting
**File:** TextRenderer.js

## Syntax

JavaScript

```
get textBrush() {}
```

### Property Value

Type: Brush

## See Also

TextRenderer Members
TextRenderer Class
MindFusion.Charting Namespace

# TextStyle Class

Defines text appearance attributes.

## Syntax

JavaScript

```
// class
TextStyle.prototype = {}
```

## Inheritance Hierarchy

## See Also

TextStyle Members
MindFusion.Charting Namespace

# TextStyle Members

Defines text appearance attributes.

The following tables list the members exposed by the TextStyle type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | TextStyle | Initializes a new instance of the TextStyle class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | fontName | Gets or sets font name. |
|  | fontSize | Gets or sets font size. |
|  | fontStyle | Gets or sets font style. |
|  | textBrush | Gets or sets a Brush used to draw text. |

## See Also

TextStyle Class
MindFusion.Charting Namespace

# TextStyle Constructor

Initializes a new instance of the TextStyle class.

**Namespace:** MindFusion.Charting
**File:** TextStyle.js

## Syntax

| JavaScript |
|---|
| `function TextStyle ()` |

## See Also

TextStyle Class
MindFusion.Charting Namespace

# TextStyle Properties

# TextStyle.fontName Property

Gets or sets font name.

**Namespace:** MindFusion.Charting
**File:** TextStyle.js

## Syntax

```
JavaScript
get fontName() {}
```

### Property Value

Type: String

## See Also

TextStyle Members
TextStyle Class
MindFusion.Charting Namespace

# TextStyle.fontSize Property

Gets or sets font size.

**Namespace:** MindFusion.Charting
**File:** TextStyle.js

## Syntax

```
JavaScript
get fontSize() {}
```

### Property Value

Type: Number

## See Also

TextStyle Members
TextStyle Class
MindFusion.Charting Namespace

# TextStyle.fontStyle Property

Gets or sets font style.

**Namespace:** MindFusion.Charting
**File:** TextStyle.js

## Syntax

```
JavaScript
get fontStyle() {}
```

### Property Value

Type: FontStyle

## See Also

TextStyle Members
TextStyle Class
MindFusion.Charting Namespace

# TextStyle.textBrush Property

Gets or sets a Brush used to draw text.

**Namespace:** MindFusion.Charting
**File:** TextStyle.js

## Syntax

**JavaScript**

```
get textBrush() {}
```

### Property Value

Type: Brush

## See Also

TextStyle Members
TextStyle Class
MindFusion.Charting Namespace

# TextStyleHint Enumeration

Identifies text styles that can be accessed through RenderContent.

**Namespace:** MindFusion.Charting
**File:** TextStyleHint.js

## Syntax

**JavaScript**

```
// enum
TextStyleHint = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **AxisLabels** | Identifies style used to draw axis labels. |
| | **AxisTitle** | Identifies style used to draw axis title. |
| | **DataLabels** | Identifies style used to draw data labels. |
| | **LegendTitle** | Identifies style used to draw legend title. |
| | **Subtitle** | Identifies style used to draw chart's sub-title. |

| | | |
|---|---|---|
| | **Title** | Identifies style used to draw chart's title. |
| | **Widget** | Identifies style used to draw text in UI widgets. |

## See Also

MindFusion.Charting Namespace

# Theme Class

Represents a chart's theme defining all appearance attributes of its elements.

## Syntax

**JavaScript**

```
// class
Theme.prototype = {}
```

## Inheritance Hierarchy

## See Also

Theme Members
MindFusion.Charting Namespace

# Theme Members

Represents a chart's theme defining all appearance attributes of its elements.

The following tables list the members exposed by the Theme type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | Theme | Initializes a new instance of the Theme class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | axisLabelsBrush | Gets or sets the Brush used to draw axis labels. |
| | axisLabelsFontName | Gets or sets the name of font used to draw axis labels. |
| | axisLabelsFontSize | Gets or sets the size of font used to draw axis labels. |
| | axisLabelsFontStyle | Gets or sets the style of font used to draw axis labels. |
| | axisStroke | Gets or sets a Brush used to stroke axis lines. |
| | axisStrokeDashStyle | Gets or sets the dash style of axis lines. |
| | axisStrokeThickness | Gets or sets the thickness of axis lines. |

| | | |
|---|---|---|
| | axisTitleBrush | Gets or sets the Brush used to draw axis titles. |
| | axisTitleFontName | Gets or sets the name of font used to draw axis titles. |
| | axisTitleFontSize | Gets or sets the size of font used to draw axis titles. |
| | axisTitleFontStyle | Gets or sets the style of font used to draw axis titles. |
| | commonSeriesFills | Gets or sets a list of brushes, each Brush used to fill all elements of a series. |
| | commonSeriesStrokeDashStyles | Gets or sets a list of dash styles, each style applied to all elements of a series. |
| | commonSeriesStrokes | Gets or sets a list of brushes, each Brush used to stroke all elements of a series. |
| | commonSeriesStrokeThicknesses | Gets or sets a list of stroke thicknesses, each thickness applied to all elements of a series. |
| | dataLabelsBrush | Gets or sets the Brush used to draw data labels. |
| | dataLabelsFontName | Gets or sets the name of font used to draw data labels. |
| | dataLabelsFontSize | Gets or sets the size of font used to draw data labels. |
| | dataLabelsFontStyle | Gets or sets the style of font used to draw data labels. |
| | gaugeBackground | Gets or sets a Brush used to draw gauge backgrounds. |
| | gaugeFontName | Gets or sets the name of font used to draw text in gauges. |
| | gaugeFontSize | Gets or sets the size of font used to draw text in gauges. |
| | gaugeFontStyle | Gets or sets the style of font used to draw text in gauges. |
| | gaugePointerBackground | Gets or sets a Brush used to draw gauge pointer backgrounds. |
| | gaugePointerStroke | Gets or sets a Brush used to stroke gauge pointer borders. |
| | gaugePointerStrokeThickness | Gets or sets the thickness of gauge pointer borders. |
| | gaugeScaleBackground | Gets or sets a Brush used to draw gauge scale backgrounds. |
| | gaugeScaleStroke | Gets or sets a Brush used to stroke gauge scale borders. |
| | gaugeScaleStrokeThickness | Gets or sets the thickness of gauge scale borders. |
| | gaugeStroke | Gets or sets a Brush used to stroke gauge borders. |
| | gaugeStrokeThickness | Gets or sets the thickness of gauge borders. |
| | gaugeTickBackground | Gets or sets a Brush used to draw gauge tick backgrounds. |
| | gaugeTickStroke | Gets or sets a Brush used to stroke gauge tick borders. |
| | gaugeTickStrokeThickness | Gets or sets the thickness of gauge tick borders. |
| | gridColor1 | Gets or sets the main color of plot grid. |
| | gridColor2 | Gets or sets the alternating color of plot grid. |
| | gridLineColor | Gets or sets the line color of plot grid. |

| | | |
|---|---|---|
| | gridLineStyle | Gets or sets the style of plot grid lines. |
| | gridLineThickness | Gets or sets the thickness of plot grid lines. |
| | highlightStroke | Gets or sets a Brush used to stroke highlighted items. |
| | highlightStrokeDashStyle | Gets or sets the dash style of highlight strokes. |
| | highlightStrokeThickness | Gets or sets the thickness of highlight strokes. |
| | legendBackground | Gets or sets a Brush used to draw legend background. |
| | legendBorderStroke | Gets or sets a Brush used to draw legend borders. |
| | legendBorderStrokeDashStyle | Gets or sets the dash style of legend borders. |
| | legendBorderStrokeThickness | Gets or sets the thickness of legend borders. |
| | legendLabelsBrush | Gets or sets the Brush used to draw legend labels. |
| | legendTitleFontName | Gets or sets the name of font used to draw legend titles. |
| | legendTitleFontSize | Gets or sets the size of font used to draw legend titles. |
| | legendTitleFontStyle | Gets or sets the style of font used to draw legend titles. |
| | plotBackground | Gets or sets a Brush used to draw the background of plots. |
| | plotBorderStroke | Gets or sets a Brush used to stroke plot borders. |
| | plotBorderStrokeDashStyle | Gets or sets the thickness of plot borders. |
| | plotBorderStrokeThickness | Gets or sets the thickness of plot borders. |
| | propertyChanged | Occurs when a property value changes. |
| | seriesFills | Gets or sets a list of Brush lists, each list used to draw a different series, and each Brush used to fill individual elements of a series. |
| | seriesStrokeDashStyles | Gets or sets a list of DashStyle lists, each list used to stroke a different series, and each DashStyle value specifying style of individual elements of a series. |
| | seriesStrokes | Gets or sets a list of Brush lists, each list used to stroke a different series, and each Brush used to stroke individual elements of a series. |
| | seriesStrokeThicknesses | Gets or sets a list of number lists, each list used to stroke a different series, and each number value specifying thickness of individual elements of a series. |
| | subtitleBrush | Gets or sets the Brush used to draw the chart sub-title. |
| | subtitleFontName | Gets or sets the name of font used to draw the chart sub-title. |
| | subtitleFontSize | Gets or sets the size of font used to draw the chart sub-title. |
| | subtitleFontStyle | Gets or sets the style of font used to draw the chart sub-title. |
| | titleBrush | Gets or sets the Brush used to draw the chart title. |
| | titleFontName | Gets or sets the name of font used to draw the chart title. |

| | | |
|---|---|---|
| | titleFontSize | Gets or sets the size of font used to draw the chart title. |
| | titleFontStyle | Gets or sets the style of font used to draw the chart title. |
| | uniformSeriesFill | Gets or sets a Brush used to fill all elements of all series uniformly. |
| | uniformSeriesStroke | Gets or sets a Brush used to stroke all elements of all series uniformly. |
| | uniformSeriesStrokeDashStyle | Gets or sets uniform stroke dash style for all elements of all series. |
| | uniformSeriesStrokeThickness | Gets or sets uniform stroke thickness for all elements of all series. |
| | widgetBrush | Gets or sets the Brush used to draw text in UI widgets. |
| | widgetFontName | Gets or sets the name of font used to draw text in UI widgets. |
| | widgetFontSize | Gets or sets the size of font used to draw text in UI widgets. |
| | widgetFontStyle | Gets or sets the style of font used to draw text in UI widgets. |

## Public Methods

| | Name | Description |
|---|---|---|
| | loadFrom | Loads the theme values from an XML file. |
| | saveTo | Saves the theme values to specified file. |

## See Also

Theme Class
MindFusion.Charting Namespace

# Theme Constructor

Initializes a new instance of the Theme class.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

**JavaScript**

```
function Theme (fill, stroke, strokeThickness, strokeDashStyle)
```

### Parameters

*fill*
Type: Brush
A Brush used to fill all series elements.

*stroke*

Type: Brush
A Brush used to stroke all series elements.

*strokeThickness*
Type: Number
Uniform thickness of series elements' strokes.

*strokeDashStyle*
Type: DashStyle
Uniform dash style of series elements' strokes.

## See Also

Theme Class
MindFusion.Charting Namespace

# Theme Methods

# Theme.loadFrom Method

Loads the theme values from an XML file.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
function loadFrom (fileUrl)
```

### Parameters

*fileUrl*
Type: String
The URL of an XML file where the data should be read from.

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.saveTo Method

Saves the theme values to specified file.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
function saveTo (fileUrl)
```

### Parameters

*fileUrl*
Type: String
The URL where the theme's XML should be posted to.

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme Properties

## Theme.axisLabelsBrush Property

Gets or sets the Brush used to draw axis labels.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

**JavaScript**

```
get axisLabelsBrush() {}
```

### Property Value

Type: Brush

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

## Theme.axisLabelsFontName Property

Gets or sets the name of font used to draw axis labels.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

**JavaScript**

```
get axisLabelsFontName() {}
```

### Property Value

Type: String

## See Also

Theme Members
Theme Class

MindFusion.Charting Namespace

## Theme.axisLabelsFontSize Property

Gets or sets the size of font used to draw axis labels.

**Namespace:** MindFusion.Charting
**File:** Theme.js

### Syntax

```
JavaScript
get axisLabelsFontSize() {}
```

#### Property Value

Type: Number

### See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

## Theme.axisLabelsFontStyle Property

Gets or sets the style of font used to draw axis labels.

**Namespace:** MindFusion.Charting
**File:** Theme.js

### Syntax

```
JavaScript
get axisLabelsFontStyle() {}
```

#### Property Value

Type: FontStyle

### See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

## Theme.axisStroke Property

Gets or sets a Brush used to stroke axis lines.

**Namespace:** MindFusion.Charting
**File:** Theme.js

### Syntax

```
JavaScript
get axisStroke() {}
```

### Property Value

Type: Brush

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.axisStrokeDashStyle Property

Gets or sets the dash style of axis lines.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get axisStrokeDashStyle() {}
```

### Property Value

Type: DashStyle

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.axisStrokeThickness Property

Gets or sets the thickness of axis lines.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get axisStrokeThickness() {}
```

### Property Value

Type: Number

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

## Theme.axisTitleBrush Property

Gets or sets the Brush used to draw axis titles.

**Namespace:** MindFusion.Charting
**File:** Theme.js

### Syntax

```
JavaScript
get axisTitleBrush() {}
```

#### Property Value

Type: Brush

### See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

## Theme.axisTitleFontName Property

Gets or sets the name of font used to draw axis titles.

**Namespace:** MindFusion.Charting
**File:** Theme.js

### Syntax

```
JavaScript
get axisTitleFontName() {}
```

#### Property Value

Type: String

### See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

## Theme.axisTitleFontSize Property

Gets or sets the size of font used to draw axis titles.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get axisTitleFontSize() {}
```

### Property Value

Type: Number

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.axisTitleFontStyle Property

Gets or sets the style of font used to draw axis titles.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get axisTitleFontStyle() {}
```

### Property Value

Type: FontStyle

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.commonSeriesFills Property

Gets or sets a list of brushes, each Brush used to fill all elements of a series.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get commonSeriesFills() {}
```

### Property Value

Type: List

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.commonSeriesStrokeDashStyles Property

Gets or sets a list of dash styles, each style applied to all elements of a series.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get commonSeriesStrokeDashStyles() {}
```

### Property Value

Type: List

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.commonSeriesStrokes Property

Gets or sets a list of brushes, each Brush used to stroke all elements of a series.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get commonSeriesStrokes() {}
```

### Property Value

Type: List

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.commonSeriesStrokeThicknesses Property

Gets or sets a list of stroke thicknesses, each thickness applied to all elements of a series.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get commonSeriesStrokeThicknesses() {}
```

### Property Value

Type: List

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.dataLabelsBrush Property

Gets or sets the Brush used to draw data labels.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get dataLabelsBrush() {}
```

### Property Value

Type: Brush

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.dataLabelsFontName Property

Gets or sets the name of font used to draw data labels.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get dataLabelsFontName() {}
```

### Property Value

Type: String

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.dataLabelsFontSize Property

Gets or sets the size of font used to draw data labels.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

**JavaScript**

```
get dataLabelsFontSize() {}
```

### Property Value

Type: Number

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.dataLabelsFontStyle Property

Gets or sets the style of font used to draw data labels.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

**JavaScript**

```
get dataLabelsFontStyle() {}
```

### Property Value

Type: FontStyle

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.gaugeBackground Property

Gets or sets a Brush used to draw gauge backgrounds.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get gaugeBackground() {}
```

### Property Value

Type: Brush

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.gaugeFontName Property

Gets or sets the name of font used to draw text in gauges.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get gaugeFontName() {}
```

### Property Value

Type: String

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.gaugeFontSize Property

Gets or sets the size of font used to draw text in gauges.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
```

```
get gaugeFontSize() {}
```

### Property Value

Type: Number

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.gaugeFontStyle Property

Gets or sets the style of font used to draw text in gauges.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

**JavaScript**

```
get gaugeFontStyle() {}
```

### Property Value

Type: FontStyle

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.gaugePointerBackground Property

Gets or sets a Brush used to draw gauge pointer backgrounds.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

**JavaScript**

```
get gaugePointerBackground() {}
```

### Property Value

Type: Brush

## See Also

Theme Members
Theme Class

MindFusion.Charting Namespace

## Theme.gaugePointerStroke Property

Gets or sets a Brush used to stroke gauge pointer borders.

**Namespace:** MindFusion.Charting
**File:** Theme.js

### Syntax

```
JavaScript
get gaugePointerStroke() {}
```

#### Property Value

Type: Brush

### See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

## Theme.gaugePointerStrokeThickness Property

Gets or sets the thickness of gauge pointer borders.

**Namespace:** MindFusion.Charting
**File:** Theme.js

### Syntax

```
JavaScript
get gaugePointerStrokeThickness() {}
```

#### Property Value

Type: Number

### See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

## Theme.gaugeScaleBackground Property

Gets or sets a Brush used to draw gauge scale backgrounds.

**Namespace:** MindFusion.Charting
**File:** Theme.js

### Syntax

```
JavaScript
get gaugeScaleBackground() {}
```

### Property Value

Type: Brush

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.gaugeScaleStroke Property

Gets or sets a Brush used to stroke gauge scale borders.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get gaugeScaleStroke() {}
```

### Property Value

Type: Brush

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.gaugeScaleStrokeThickness Property

Gets or sets the thickness of gauge scale borders.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get gaugeScaleStrokeThickness() {}
```

### Property Value

Type: Number

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.gaugeStroke Property

Gets or sets a Brush used to stroke gauge borders.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get gaugeStroke() {}
```

### Property Value

Type: Brush

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.gaugeStrokeThickness Property

Gets or sets the thickness of gauge borders.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get gaugeStrokeThickness() {}
```

### Property Value

Type: Number

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.gaugeTickBackground Property

Gets or sets a Brush used to draw gauge tick backgrounds.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get gaugeTickBackground() {}
```

### Property Value

Type: Brush

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.gaugeTickStroke Property

Gets or sets a Brush used to stroke gauge tick borders.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get gaugeTickStroke() {}
```

### Property Value

Type: Brush

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.gaugeTickStrokeThickness Property

Gets or sets the thickness of gauge tick borders.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get gaugeTickStrokeThickness() {}
```

### Property Value

Type: Number

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.gridColor1 Property

Gets or sets the main color of plot grid.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get gridColor1() {}
```

### Property Value

Type: Color

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.gridColor2 Property

Gets or sets the alternating color of plot grid.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get gridColor2() {}
```

### Property Value

Type: Color

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.gridLineColor Property

Gets or sets the line color of plot grid.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get gridLineColor() {}
```

### Property Value

Type: Color

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.gridLineStyle Property

Gets or sets the style of plot grid lines.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get gridLineStyle() {}
```

### Property Value

Type: DashStyle

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.gridLineThickness Property

Gets or sets the thickness of plot grid lines.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get gridLineThickness() {}
```

### Property Value

Type: Number

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.highlightStroke Property

Gets or sets a Brush used to stroke highlighted items.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get highlightStroke() {}
```

### Property Value

Type: Brush

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.highlightStrokeDashStyle Property

Gets or sets the dash style of highlight strokes.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get highlightStrokeDashStyle() {}
```

### Property Value

Type: DashStyle

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

## Theme.highlightStrokeThickness Property

Gets or sets the thickness of highlight strokes.

**Namespace:** MindFusion.Charting
**File:** Theme.js

### Syntax

JavaScript

```
get highlightStrokeThickness() {}
```

#### Property Value

Type: Number

### See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

## Theme.legendBackground Property

Gets or sets a Brush used to draw legend background.

**Namespace:** MindFusion.Charting
**File:** Theme.js

### Syntax

JavaScript

```
get legendBackground() {}
```

#### Property Value

Type: Brush

### See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

## Theme.legendBorderStroke Property

Gets or sets a Brush used to draw legend borders.

**Namespace:** MindFusion.Charting
**File:** Theme.js

### Syntax

JavaScript

```
get legendBorderStroke() {}
```

### Property Value

Type: Brush

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.legendBorderStrokeDashStyle Property

Gets or sets the dash style of legend borders.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

**JavaScript**

```
get legendBorderStrokeDashStyle() {}
```

### Property Value

Type: DashStyle

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.legendBorderStrokeThickness Property

Gets or sets the thickness of legend borders.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

**JavaScript**

```
get legendBorderStrokeThickness() {}
```

### Property Value

Type: Number

## See Also

Theme Members
Theme Class

## Theme.legendLabelsBrush Property

Gets or sets the Brush used to draw legend labels.

**Namespace:** MindFusion.Charting
**File:** Theme.js

### Syntax

```
JavaScript
get legendLabelsBrush() {}
```

#### Property Value

Type: Brush

### See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

## Theme.legendTitleFontName Property

Gets or sets the name of font used to draw legend titles.

**Namespace:** MindFusion.Charting
**File:** Theme.js

### Syntax

```
JavaScript
get legendTitleFontName() {}
```

#### Property Value

Type: String

### See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

## Theme.legendTitleFontSize Property

Gets or sets the size of font used to draw legend titles.

**Namespace:** MindFusion.Charting
**File:** Theme.js

### Syntax

JavaScript

```
get legendTitleFontSize() {}
```

### Property Value

Type: Number

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.legendTitleFontStyle Property

Gets or sets the style of font used to draw legend titles.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

JavaScript

```
get legendTitleFontStyle() {}
```

### Property Value

Type: FontStyle

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.plotBackground Property

Gets or sets a Brush used to draw the background of plots.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

JavaScript

```
get plotBackground() {}
```

### Property Value

Type: Brush

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

## Theme.plotBorderStroke Property

Gets or sets a Brush used to stroke plot borders.

**Namespace:** MindFusion.Charting
**File:** Theme.js

### Syntax

```
JavaScript
get plotBorderStroke() {}
```

#### Property Value

Type: Brush

### See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

## Theme.plotBorderStrokeDashStyle Property

Gets or sets the thickness of plot borders.

**Namespace:** MindFusion.Charting
**File:** Theme.js

### Syntax

```
JavaScript
get plotBorderStrokeDashStyle() {}
```

#### Property Value

Type: DashStyle

### See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

## Theme.plotBorderStrokeThickness Property

Gets or sets the thickness of plot borders.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get plotBorderStrokeThickness() {}
```

### Property Value

Type: Number

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.propertyChanged Property

Occurs when a property value changes.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get propertyChanged() {}
```

### Property Value

Type: PropertyChangedEventDispatcher

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.seriesFills Property

Gets or sets a list of Brush lists, each list used to draw a different series, and each Brush used to fill individual elements of a series.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get seriesFills() {}
```

### Property Value

Type: List>

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.seriesStrokeDashStyles Property

Gets or sets a list of DashStyle lists, each list used to stroke a different series, and each DashStyle value specifying style of individual elements of a series.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

**JavaScript**

```
get seriesStrokeDashStyles() {}
```

### Property Value

Type: List>

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.seriesStrokes Property

Gets or sets a list of Brush lists, each list used to stroke a different series, and each Brush used to stroke individual elements of a series.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

**JavaScript**

```
get seriesStrokes() {}
```

### Property Value

Type: List>

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

## Theme.seriesStrokeThicknesses Property

Gets or sets a list of number lists, each list used to stroke a different series, and each number value specifying thickness of individual elements of a series.

**Namespace:** MindFusion.Charting
**File:** Theme.js

### Syntax

```
JavaScript
get seriesStrokeThicknesses() {}
```

#### Property Value

Type: List>

### See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

## Theme.subtitleBrush Property

Gets or sets the Brush used to draw the chart sub-title.

**Namespace:** MindFusion.Charting
**File:** Theme.js

### Syntax

```
JavaScript
get subtitleBrush() {}
```

#### Property Value

Type: Brush

### See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

## Theme.subtitleFontName Property

Gets or sets the name of font used to draw the chart sub-title.

**Namespace:** MindFusion.Charting
**File:** Theme.js

### Syntax

```
JavaScript
```

```
get subtitleFontName() {}
```

### Property Value

Type: String

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.subtitleFontSize Property

Gets or sets the size of font used to draw the chart sub-title.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

**JavaScript**

```
get subtitleFontSize() {}
```

### Property Value

Type: Number

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.subtitleFontStyle Property

Gets or sets the style of font used to draw the chart sub-title.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

**JavaScript**

```
get subtitleFontStyle() {}
```

### Property Value

Type: FontStyle

## See Also

Theme Members
Theme Class

MindFusion.Charting Namespace

## Theme.titleBrush Property

Gets or sets the Brush used to draw the chart title.

**Namespace:** MindFusion.Charting
**File:** Theme.js

### Syntax

```
JavaScript
get titleBrush() {}
```

#### Property Value

Type: Brush

### See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

## Theme.titleFontName Property

Gets or sets the name of font used to draw the chart title.

**Namespace:** MindFusion.Charting
**File:** Theme.js

### Syntax

```
JavaScript
get titleFontName() {}
```

#### Property Value

Type: String

### See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

## Theme.titleFontSize Property

Gets or sets the size of font used to draw the chart title.

**Namespace:** MindFusion.Charting
**File:** Theme.js

### Syntax

```
JavaScript
get titleFontSize() {}
```

### Property Value

Type: Number

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

## Theme.titleFontStyle Property

Gets or sets the style of font used to draw the chart title.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get titleFontStyle() {}
```

### Property Value

Type: FontStyle

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

## Theme.uniformSeriesFill Property

Gets or sets a Brush used to fill all elements of all series uniformly.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get uniformSeriesFill() {}
```

### Property Value

Type: Brush

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

## Theme.uniformSeriesStroke Property

Gets or sets a Brush used to stroke all elements of all series uniformly.

**Namespace:** MindFusion.Charting
**File:** Theme.js

### Syntax

```
JavaScript
get uniformSeriesStroke() {}
```

#### Property Value

Type: Brush

### See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

## Theme.uniformSeriesStrokeDashStyle Property

Gets or sets uniform stroke dash style for all elements of all series.

**Namespace:** MindFusion.Charting
**File:** Theme.js

### Syntax

```
JavaScript
get uniformSeriesStrokeDashStyle() {}
```

#### Property Value

Type: DashStyle

### See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

## Theme.uniformSeriesStrokeThickness Property

Gets or sets uniform stroke thickness for all elements of all series.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get uniformSeriesStrokeThickness() {}
```

### Property Value

Type: Number

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.widgetBrush Property

Gets or sets the Brush used to draw text in UI widgets.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get widgetBrush() {}
```

### Property Value

Type: Brush

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# Theme.widgetFontName Property

Gets or sets the name of font used to draw text in UI widgets.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

```
JavaScript
get widgetFontName() {}
```

### Property Value

Type: String

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

### Theme.widgetFontSize Property

Gets or sets the size of font used to draw text in UI widgets.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

**JavaScript**

```
get widgetFontSize() {}
```

#### Property Value

Type: Number

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

### Theme.widgetFontStyle Property

Gets or sets the style of font used to draw text in UI widgets.

**Namespace:** MindFusion.Charting
**File:** Theme.js

## Syntax

**JavaScript**

```
get widgetFontStyle() {}
```

#### Property Value

Type: FontStyle

## See Also

Theme Members
Theme Class
MindFusion.Charting Namespace

# ToolTip Class

Represents the current tooltip.

## Syntax

**JavaScript**

```
// class
ToolTip.prototype = {}
```

## Remarks

The properties of the ToolTip can be used as static. The tooltips can be set by including LabelKinds.ToolTip among the values of the supportedLabel property of a Series.

## Example

The following code creates a BarSeries and provides two arrays with label data - one is the data array, which will serve as labels and the other is an array with strings. Then it specifies that the two arrays will be used for tooltips and labels at the X-axis with the supportedLabel property. Finally, it customizes the tooltip:

**JavaScript**

```
var labels = [
    "Jan", "Feb", "Mar", "Apr", "May", "Jun",
    "Jul", "Aug", "Sep", "Oct", "Nov", "Dec"
];

var data = new Collections.List([1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]);

// create sample data series
var barSeries = new Charting.BarSeries(data, data, labels),

//specifies that the labels will be used for the X-axis and tooltips
barSeries.supportedLabels = Charting.LabelKinds.XAxisLabel | Charting.LabelKinds.ToolTip;

// customize tooltips
ToolTip.brush = new Drawing.Brush("#fafafa");
ToolTip.pen = new Drawing.Pen("#9caac6");
ToolTip.textBrush = new Drawing.Brush("#5050c0");
ToolTip.horizontalPadding = 6;
ToolTip.verticalPadding = 4;
ToolTip.font = new Charting.Drawing.Font("Verdana", 12, Charting.Drawing.FontStyle.Italic);
```

## Inheritance Hierarchy

## See Also

ToolTip Members
MindFusion.Charting Namespace

# ToolTip Members

Represents the current tooltip.

The following tables list the members exposed by the ToolTip type.

## Public Properties

| | Name | Description |
|---|---|---|
|  | brush | Gets or sets the tooltip background brush. |
|  | font | Gets or sets the tooltip font. |

| | | |
|---|---|---|
| | horizontalOffset | Gets or sets the horizontal offset from the pointer. |
| | horizontalPadding | Gets or sets the horizontal padding to the left and right of the tooltip text. |
| | pen | Gets or sets the tooltip border pen. |
| | position | Gets or sets the tooltip position. |
| | text | Gets or sets the tooltip text. |
| | textBrush | Gets or sets the tooltip text brush. |
| | verticalOffset | Gets or sets the horizontal offset from the pointer. |
| | verticalPadding | Gets or sets the vertical padding above and below the tooltip text. |

## See Also

ToolTip Class
MindFusion.Charting Namespace

# ToolTip Properties

# ToolTip.brush Property

Gets or sets the tooltip background brush.

**Namespace:** MindFusion.Charting
**File:** ToolTip.js

## Syntax

**JavaScript**

```
get brush() {}
```

### Property Value

Type: Brush

## Example

The following code customizes the ToolTip-s rendered over chart elements:

**JavaScript**

```
// customize tooltips
ToolTip.position = new Point (-5, -3);
ToolTip.text = "Click on a bar to see detailed chart";
ToolTip.brush = new Drawing.Brush("#fafafa");
ToolTip.pen = new Drawing.Pen("#9caac6");
ToolTip.textBrush = new Drawing.Brush("#5050c0");
ToolTip.horizontalPadding = 6;
ToolTip.verticalPadding = 4;
ToolTip.font = new Charting.Drawing.Font("Verdana", 12, Charting.Drawing.FontStyle.Italic);
```

## See Also

ToolTip Members
ToolTip Class
MindFusion.Charting Namespace

# ToolTip.font Property

Gets or sets the tooltip font.

**Namespace:** MindFusion.Charting
**File:** ToolTip.js

## Syntax

```
JavaScript
get font() {}
```

### Property Value

Type: Font

## Example

The following code customizes the ToolTip-s rendered over chart elements:

```
JavaScript
// customize tooltips
ToolTip.position = new Point (-5, -3);
ToolTip.text = "Click on a bar to see detailed chart";
ToolTip.brush = new Drawing.Brush("#fafafa");
ToolTip.pen = new Drawing.Pen("#9caac6");
ToolTip.textBrush = new Drawing.Brush("#5050c0");
ToolTip.horizontalPadding = 6;
ToolTip.verticalPadding = 4;
ToolTip.font = new Charting.Drawing.Font("Verdana", 12, Charting.Drawing.FontStyle.Italic);
```

## See Also

ToolTip Members
ToolTip Class
MindFusion.Charting Namespace

# ToolTip.horizontalOffset Property

Gets or sets the horizontal offset from the pointer.

**Namespace:** MindFusion.Charting
**File:** ToolTip.js

## Syntax

```
JavaScript
get horizontalOffset() {}
```

### Property Value

Type: Number

## Example

The following code customizes the ToolTip-s rendered over chart elements:

**JavaScript**

```
// customize tooltips
ToolTip.position = new Point (-5, -3);
ToolTip.text = "Click on a bar to see detailed chart";
ToolTip.brush = new Drawing.Brush("#fafafa");
ToolTip.pen = new Drawing.Pen("#9caac6");
ToolTip.textBrush = new Drawing.Brush("#5050c0");
ToolTip.horizontalOffset = 6;
ToolTip.verticalPadding = 4;
ToolTip.font = new Charting.Drawing.Font("Verdana", 12, Charting.Drawing.FontStyle.Italic);
```

## See Also

ToolTip Members
ToolTip Class
MindFusion.Charting Namespace

# ToolTip.horizontalPadding Property

Gets or sets the horizontal padding to the left and right of the tooltip text.

**Namespace:** MindFusion.Charting
**File:** ToolTip.js

## Syntax

**JavaScript**

```
get horizontalPadding() {}
```

### Property Value

Type: Number

## Example

The following code customizes the ToolTip-s rendered over chart elements:

**JavaScript**

```
// customize tooltips
ToolTip.position = new Point (-5, -3);
ToolTip.text = "Click on a bar to see detailed chart";
ToolTip.brush = new Drawing.Brush("#fafafa");
ToolTip.pen = new Drawing.Pen("#9caac6");
ToolTip.textBrush = new Drawing.Brush("#5050c0");
ToolTip.horizontalOffset = 6;
ToolTip.horizontalPadding = -4;
ToolTip.font = new Charting.Drawing.Font("Verdana", 12, Charting.Drawing.FontStyle.Italic);
```

## See Also

ToolTip Members
ToolTip Class
MindFusion.Charting Namespace

# ToolTip.pen Property

Gets or sets the tooltip border pen.

**Namespace:** MindFusion.Charting
**File:** ToolTip.js

## Syntax

**JavaScript**

```
get pen() {}
```

### Property Value

Type: Pen

## Example

The following code customizes the ToolTip-s rendered over chart elements:

**JavaScript**

```
// customize tooltips
ToolTip.position = new Point (-5, -3);
ToolTip.text = "Click on a bar to see detailed chart";
ToolTip.brush = new Drawing.Brush("#fafafa");
ToolTip.pen = new Drawing.Pen("#9caac6");
ToolTip.textBrush = new Drawing.Brush("#5050c0");
ToolTip.horizontalOffset = 6;
ToolTip.horizontalPadding = -4;
ToolTip.font = new Charting.Drawing.Font("Verdana", 12, Charting.Drawing.FontStyle.Italic);
```

## See Also

ToolTip Members
ToolTip Class
MindFusion.Charting Namespace

# ToolTip.position Property

Gets or sets the tooltip position.

**Namespace:** MindFusion.Charting
**File:** ToolTip.js

## Syntax

**JavaScript**

```
get position() {}
```

### Property Value

Type: Point

## Example

The following code customizes the ToolTip-s rendered over chart elements:

JavaScript

```
// customize tooltips
ToolTip.position = new Point (-5, -3);
ToolTip.brush = new Drawing.Brush("#fafafa");
ToolTip.pen = new Drawing.Pen("#9caac6");
ToolTip.textBrush = new Drawing.Brush("#5050c0");
ToolTip.horizontalPadding = 6;
ToolTip.verticalPadding = 4;
ToolTip.font = new Charting.Drawing.Font("Verdana", 12, Charting.Drawing.FontStyle.Italic);
```

## See Also

ToolTip Members
ToolTip Class
MindFusion.Charting Namespace

# ToolTip.text Property

Gets or sets the tooltip text.

**Namespace:** MindFusion.Charting
**File:** ToolTip.js

## Syntax

JavaScript

```
get text() {}
```

### Property Value

Type: String

## Remarks

In most cases you will use an array or List to set the labels that will render as tooltip-s and use the supportedLabels property to specify that they should be rendered as tooltips.

## Example

The following code customizes the ToolTips rendered over chart elements:

JavaScript

```
// customize tooltips
ToolTip.position = new Point (-5, -3);
ToolTip.text = "Click on a bar to see detailed chart";
ToolTip.brush = new Drawing.Brush("#fafafa");
ToolTip.pen = new Drawing.Pen("#9caac6");
ToolTip.textBrush = new Drawing.Brush("#5050c0");
ToolTip.horizontalPadding = 6;
ToolTip.verticalPadding = 4;
ToolTip.font = new Charting.Drawing.Font("Verdana", 12, Charting.Drawing.FontStyle.Italic);
```

## See Also

ToolTip Members
ToolTip Class
MindFusion.Charting Namespace

# ToolTip.textBrush Property

Gets or sets the tooltip text brush.

**Namespace:** MindFusion.Charting
**File:** ToolTip.js

## Syntax

```
JavaScript
get textBrush() {}
```

### Property Value

Type: Brush

## Example

The following code customizes the ToolTip-s rendered over chart elements:

```
JavaScript
// customize tooltips
ToolTip.position = new Point (-5, -3);
ToolTip.text = "Click on a bar to see detailed chart";
ToolTip.brush = new Drawing.Brush("#fafafa");
ToolTip.pen = new Drawing.Pen("#9caac6");
ToolTip.textBrush = new Drawing.Brush("#5050c0");
ToolTip.horizontalOffset = 6;
ToolTip.horizontalPadding = -4;
ToolTip.font = new Charting.Drawing.Font("Verdana", 12, Charting.Drawing.FontStyle.Italic);
```

## See Also

ToolTip Members
ToolTip Class
MindFusion.Charting Namespace

# ToolTip.verticalOffset Property

Gets or sets the horizontal offset from the pointer.

**Namespace:** MindFusion.Charting
**File:** ToolTip.js

## Syntax

```
JavaScript
get verticalOffset() {}
```

### Property Value

Type: Number

## Example

The following code customizes the ToolTip-s rendered over chart elements:

JavaScript

```
// customize tooltips
ToolTip.position = new Point (-5, -3);
ToolTip.text = "Click on a bar to see detailed chart";
ToolTip.brush = new Drawing.Brush("#fafafa");
ToolTip.pen = new Drawing.Pen("#9caac6");
ToolTip.textBrush = new Drawing.Brush("#5050c0");
ToolTip.verticallOffset = 6;
ToolTip.verticalPadding = -4;
ToolTip.font = new Charting.Drawing.Font("Verdana", 12, Charting.Drawing.FontStyle.Italic);
```

## See Also

ToolTip Members
ToolTip Class
MindFusion.Charting Namespace

# ToolTip.verticalPadding Property

Gets or sets the vertical padding above and below the tooltip text.

**Namespace:** MindFusion.Charting
**File:** ToolTip.js

## Syntax

JavaScript

```
get verticalPadding() {}
```

### Property Value

Type: Number

## Example

The following code customizes the ToolTip-s rendered over chart elements:

JavaScript

```
// customize tooltips
ToolTip.position = new Point (-5, -3);
ToolTip.text = "Click on a bar to see detailed chart";
ToolTip.brush = new Drawing.Brush("#fafafa");
ToolTip.pen = new Drawing.Pen("#9caac6");
ToolTip.textBrush = new Drawing.Brush("#5050c0");
ToolTip.verticallOffset = 6;
ToolTip.verticalPadding = -4;
ToolTip.font = new Charting.Drawing.Font("Verdana", 12, Charting.Drawing.FontStyle.Italic);
```

## See Also

ToolTip Members
ToolTip Class
MindFusion.Charting Namespace

# ToolTip Class

Represents the current chart tooltip.

## Syntax

**JavaScript**

```
// class
ToolTip.prototype = {}
```

## Remarks

The properties of the ToolTip can be used as static. The tooltips can be set by including LabelKinds.ToolTip among the values of the

## Example

The following code creates a BarSeries and provides two arrays with label data - one is the data array, which will serve as labels and the other is an array with strings. Then it specifies that the two arrays will be used for tooltips and labels at the X-axis with the supportedLabel property. Finally, it customizes the tooltip:

**JavaScript**

```
var labels = [
    "Jan", "Feb", "Mar", "Apr", "May", "Jun",
    "Jul", "Aug", "Sep", "Oct", "Nov", "Dec"
];

var data = new Collections.List([1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]);

// create sample data series
var barSeries = new Charting.BarSeries(data, data, labels),

//specifies that the labels will be used for the X-axis and tooltips
barSeries.supportedLabels = Charting.LabelKinds.XAxisLabel | Charting.LabelKinds.ToolTip;

// customize tooltips
ToolTip.brush = new Drawing.Brush("#fafafa");
ToolTip.pen = new Drawing.Pen("#9caac6");
ToolTip.textBrush = new Drawing.Brush("#5050c0");
ToolTip.horizontalPadding = 6;
ToolTip.verticalPadding = 4;
ToolTip.font = new Charting.Drawing.Font("Verdana", 12, Charting.Drawing.FontStyle.Italic);
```

## Inheritance Hierarchy

## See Also

ToolTip Members
MindFusion.Charting Namespace

# ToolTip Members

Represents the current chart tooltip.

The following tables list the members exposed by the ToolTip type.

## Public Properties

| | Name | Description |
|---|---|---|
|  | brush | Gets or sets the tooltip background brush. |
|  | font | Gets or sets the tooltip font. |

| | | |
|---|---|---|
| | horizontalOffset | Gets or sets the horizontal offset from the pointer. |
| | horizontalPadding | Gets or sets the horizontal padding to the left and right of the tooltip text. |
| | pen | Gets or sets the tooltip border pen. |
| | position | Gets or sets the tooltip position. |
| | text | Gets or sets the tooltip text. |
| | textBrush | Gets or sets the tooltip text brush. |
| | verticalOffset | Gets or sets the horizontal offset from the pointer. |
| | verticalPadding | Gets or sets the vertical padding above and below the tooltip text. |

## See Also

ToolTip Class
MindFusion.Charting Namespace

# ToolTip Properties

# ToolTip.brush Property

Gets or sets the tooltip background brush.

**Namespace:** MindFusion.Charting
**File:** MindFusion.Charting.js

## Syntax

**JavaScript**

```
get brush() {}
```

### Property Value

A Brush instance.

## Example

The following code customizes a tooltip rendered on a chart built with MindFusion.Charting for JavaScript:

**JavaScript**

```
var ToolTip = Charting.ToolTip;

// customize tooltips
ToolTip.brush = new Drawing.Brush("#fafafa");
ToolTip.pen = new Drawing.Pen("#9caac6");
ToolTip.textBrush = new Drawing.Brush("#5050c0");
ToolTip.horizontalPadding = 6;
ToolTip.verticalPadding = 4;
ToolTip.font = new Charting.Drawing.Font("Verdana", 12, Charting.Drawing.FontStyle.Italic);
```

## See Also

ToolTip Members
ToolTip Class
MindFusion.Charting Namespace

# ToolTip.font Property

Gets or sets the tooltip font.

**Namespace:** MindFusion.Charting
**File:** MindFusion.Charting.js

## Syntax

```
JavaScript
get font() {}
```

### Property Value

A Font instance.

## Example

The following code customizes a tooltip rendered on a chart built with MindFusion.Charting for JavaScript:

```
JavaScript
var ToolTip = Charting.ToolTip;

// customize tooltips
ToolTip.brush = new Drawing.Brush("#fafafa");
ToolTip.pen = new Drawing.Pen("#9caac6");
ToolTip.textBrush = new Drawing.Brush("#5050c0");
ToolTip.horizontalPadding = 6;
ToolTip.verticalPadding = 4;
ToolTip.font = new Charting.Drawing.Font("Verdana", 12, Charting.Drawing.FontStyle.Italic);
```

## See Also

ToolTip Members
ToolTip Class
MindFusion.Charting Namespace

# ToolTip.horizontalOffset Property

Gets or sets the horizontal offset from the pointer.

**Namespace:** MindFusion.Charting
**File:** MindFusion.Charting.js

## Syntax

```
JavaScript
get horizontalOffset() {}
```

### Property Value

A number.

## Example

The following code customizes a tooltip rendered on a chart built with MindFusion.Charting for JavaScript:

**JavaScript**

```
var ToolTip = Charting.ToolTip;

// customize tooltips
ToolTip.brush = new Drawing.Brush("#fafafa");
ToolTip.pen = new Drawing.Pen("#9caac6");
ToolTip.textBrush = new Drawing.Brush("#5050c0");
ToolTip.horizontalOffset = 2;
ToolTip.horizontalPadding = 6;
ToolTip.verticalPadding = 4;
ToolTip.font = new Charting.Drawing.Font("Verdana", 12, Charting.Drawing.FontStyle.Italic);
```

## See Also

ToolTip Members
ToolTip Class
MindFusion.Charting Namespace

# ToolTip.horizontalPadding Property

Gets or sets the horizontal padding to the left and right of the tooltip text.

**Namespace:** MindFusion.Charting
**File:** MindFusion.Charting.js

## Syntax

**JavaScript**

```
get horizontalPadding() {}
```

### Property Value

A number.

## Example

The following code customizes a tooltip rendered on a chart built with MindFusion.Charting for JavaScript:

**JavaScript**

```
var ToolTip = Charting.ToolTip;

// customize tooltips
ToolTip.brush = new Drawing.Brush("#fafafa");
ToolTip.pen = new Drawing.Pen("#9caac6");
ToolTip.textBrush = new Drawing.Brush("#5050c0");
ToolTip.horizontalOffset = 2;
ToolTip.horizontalPadding = 6;
ToolTip.verticalPadding = 4;
ToolTip.font = new Charting.Drawing.Font("Verdana", 12, Charting.Drawing.FontStyle.Italic);
```

## See Also

ToolTip Members
ToolTip Class
MindFusion.Charting Namespace

# ToolTip.pen Property

Gets or sets the tooltip border pen.

**Namespace:** MindFusion.Charting
**File:** MindFusion.Charting.js

## Syntax

```
JavaScript
get pen() {}
```

### Property Value

A Pen instance.

## Example

The following code customizes a tooltip rendered on a chart built with MindFusion.Charting for JavaScript:

```
JavaScript
var ToolTip = Charting.ToolTip;

// customize tooltips
ToolTip.brush = new Drawing.Brush("#fafafa");
ToolTip.pen = new Drawing.Pen("#9caac6");
ToolTip.textBrush = new Drawing.Brush("#5050c0");
ToolTip.horizontalOffset = 2;
ToolTip.horizontalPadding = 6;
ToolTip.verticalPadding = 4;
ToolTip.font = new Charting.Drawing.Font("Verdana", 12, Charting.Drawing.FontStyle.Italic);
```

## See Also

ToolTip Members
ToolTip Class
MindFusion.Charting Namespace

# ToolTip.position Property

Gets or sets the tooltip position.

**Namespace:** MindFusion.Charting
**File:** MindFusion.Charting.js

## Syntax

```
JavaScript
get position() {}
```

### Property Value

A Point instance.

## Example

The following code specifies that the tooltip will render at a Point(100,100):

**JavaScript**

```
var ToolTip = Charting.ToolTip;

// customize the tooltip
ToolTip.position = new MindFusion.Drawing.Point(100, 100);
```

## See Also

ToolTip Members
ToolTip Class
MindFusion.Charting Namespace

# ToolTip.text Property

Gets or sets the tooltip text.

**Namespace:** MindFusion.Charting
**File:** MindFusion.Charting.js

## Syntax

**JavaScript**

```
get text() {}
```

### Property Value

A String that specifies the text.

## Example

The following code customizes a tooltip rendered on a chart built with MindFusion.Charting for JavaScript:

**JavaScript**

```
var ToolTip = Charting.ToolTip;


// customize the tooltip
ToolTip.text = "Click on the chart to see detailed data";
ToolTip.brush = new Drawing.Brush("#fafafa");
ToolTip.pen = new Drawing.Pen("#9caac6");
ToolTip.textBrush = new Drawing.Brush("#5050c0");
ToolTip.horizontalOffset = -2;
ToolTip.verticallOffset = -2;
ToolTip.horizontalPadding = 6;
ToolTip.verticalPadding = 4;
ToolTip.font = new Charting.Drawing.Font("Verdana", 12, Charting.Drawing.FontStyle.Italic);
```

## See Also

ToolTip Members
ToolTip Class

# ToolTip.textBrush Property

Gets or sets the tooltip text brush.

**Namespace:** MindFusion.Charting
**File:** MindFusion.Charting.js

## Syntax

**JavaScript**

```
get textBrush() {}
```

### Property Value

Type: Brush

## Example

The following code customizes a tooltip rendered on a chart built with MindFusion.Charting for JavaScript:

**JavaScript**

```
var ToolTip = Charting.ToolTip;


// customize the tooltip
ToolTip.text = "Click on the chart to see detailed data";
ToolTip.brush = new Drawing.Brush("#fafafa");
ToolTip.pen = new Drawing.Pen("#9caac6");
ToolTip.textBrush = new Drawing.Brush("#5050c0");
ToolTip.horizontalOffset = -2;
ToolTip.verticallOffset = -2;
ToolTip.horizontalPadding = 6;
ToolTip.verticalPadding = 4;
ToolTip.font = new Charting.Drawing.Font("Verdana", 12, Charting.Drawing.FontStyle.Italic);
```

## See Also

ToolTip Members
ToolTip Class
MindFusion.Charting Namespace

# ToolTip.verticalOffset Property

Gets or sets the horizontal offset from the pointer.

**Namespace:** MindFusion.Charting
**File:** MindFusion.Charting.js

## Syntax

**JavaScript**

```
get verticalOffset() {}
```

### Property Value

A number.

## Example

The following code customizes a tooltip rendered on a chart built with MindFusion.Charting for JavaScript:

**JavaScript**

```
var ToolTip = Charting.ToolTip;

// customize the tooltip
ToolTip.text = "Click on the chart to see detailed data";
ToolTip.brush = new Drawing.Brush("#fafafa");
ToolTip.pen = new Drawing.Pen("#9caac6");
ToolTip.textBrush = new Drawing.Brush("#5050c0");
ToolTip.horizontalOffset = -2;
ToolTip.verticallOffset = -2;
ToolTip.horizontalPadding = 6;
ToolTip.verticalPadding = 4;
ToolTip.font = new Charting.Drawing.Font("Verdana", 12, Charting.Drawing.FontStyle.Italic);
```

## See Also

ToolTip Members
ToolTip Class
MindFusion.Charting Namespace

# ToolTip.verticalPadding Property

Gets or sets the vertical padding above and below the tooltip text.

**Namespace:** MindFusion.Charting
**File:** MindFusion.Charting.js

## Syntax

**JavaScript**

```
get verticalPadding() {}
```

### Property Value

A number.

## Example

The following code customizes a tooltip rendered on a chart built with MindFusion.Charting for JavaScript:

**JavaScript**

```
// customize the tooltip
ToolTip.text = "Click on the chart to see detailed data";
ToolTip.brush = new Drawing.Brush("#fafafa");
ToolTip.pen = new Drawing.Pen("#9caac6");
ToolTip.textBrush = new Drawing.Brush("#5050c0");
ToolTip.horizontalOffset = -2;
ToolTip.verticallOffset = -2;
```

```
ToolTip.horizontalPadding = 6;
ToolTip.verticalPadding = 4;
ToolTip.font = new Charting.Drawing.Font("Verdana", 12, Charting.Drawing.FontStyle.Italic);
```

## See Also

ToolTip Members
ToolTip Class
MindFusion.Charting Namespace

# TowerLayout Enumeration

Specifies how to arrange segments in tower charts.

**Namespace:** MindFusion.Charting
**File:** TowerLayout.js

## Syntax

**JavaScript**

```
// enum
TowerLayout = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **Interleave** | Segments from the two series are interleaved. |
| | **Stack** | Segments in each series are stacked on top of each other. |
| | **Timeline** | Segments are positioned proportionally to their time value. |

## See Also

MindFusion.Charting Namespace

# TowerRenderer Class

Implements a SeriesRenderer used to draw tower charts, rendering series side by side to allow comparing data sequence and sizes.

## Syntax

**JavaScript**

```
// class
TowerRenderer.prototype = {}
```

## Remarks

Assign the series to compare to leftSeries and rightSeries properties of the renderer. The chart segments are arranged according to the value of layout property, and their shape is drawn as specified by segmentShape.

This renderer requires three-dimensional series. The first dimension specifies event order or timing and is used to sort or position segments. Second dimension specifies duration and is rendered as segment length along the main axis. Third dimension specifies a value rendered as segment width along the cross axis. You could use the stock EventSeries class as data container, or implement the Series interface to provide data from your own data structures.

## Inheritance Hierarchy

**MindFusion.Charting.TowerRenderer**

## See Also

TowerRenderer Members
MindFusion.Charting Namespace

# TowerRenderer Members

Implements a SeriesRenderer used to draw tower charts, rendering series side by side to allow comparing data sequence and sizes.

The following tables list the members exposed by the TowerRenderer type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | TowerRenderer | Initializes a new instance of the TowerRenderer class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | layout | Gets or sets the layout of tower segments. |
|  | leftSeries | Gets or sets the left-side Series drawn by this TowerRenderer. |
|  | margins | Gets or sets the size of margins around chart graphics. |
|  | range | Gets or sets fixed data range size. |
|  | rightSeries | Gets or sets the right-side Series drawn by this TowerRenderer. |
|  | segmentShape | Gets or sets the shape of tower segments. |
|  | seriesPadding | Gets or sets the padding distance between series. |

## Public Methods

| | Name | Description |
|---|---|---|
|  | draw | SeriesRenderer.Draw override. Draws the series data in specified RenderContext. |
|  | drawHighlight | SeriesRenderer.DrawHighlight override. Draws highlighted slice in specified RenderContext. |

| | | |
|---|---|---|
| | enumSegments | Enumerates the segments of the tower chart. |
| | enumSeries | Implements the SeriesContainer interface. |
| | hitTest | SeriesRenderer.HitTest override. Hit-tests the slices representing data items. |

## See Also

TowerRenderer Class
MindFusion.Charting Namespace

# TowerRenderer Constructor

Initializes a new instance of the TowerRenderer class.

**Namespace:** MindFusion.Charting
**File:** TowerRenderer.js

## Syntax

**JavaScript**

```
function TowerRenderer (leftSeries, rightSeries)
```

### Parameters

*leftSeries*
Type: Series
Series. The Series that should be rendered on left side of the chart.

*rightSeries*
Type: Series
Series. The Series that should be rendered on right side of the chart.

## See Also

TowerRenderer Class
MindFusion.Charting Namespace

# TowerRenderer Methods

# TowerRenderer.draw Method

SeriesRenderer.Draw override. Draws the series data in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** TowerRenderer.js

## Syntax

**JavaScript**

```
function draw (context)
```

### Parameters

*context*
Type: RenderContext
RenderContext. A RenderContext instance.

## See Also

TowerRenderer Members
TowerRenderer Class
MindFusion.Charting Namespace

# TowerRenderer.drawHighlight Method

SeriesRenderer.DrawHighlight override. Draws highlighted slice in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** TowerRenderer.js

## Syntax

**JavaScript**

```
function drawHighlight (context, hitResult)
```

### Parameters

*context*
Type: RenderContext
RenderContext. A RenderContext instance.

*hitResult*
Type: HitResult
HitResult. A HitResult instance identifying highlighted data item.

## See Also

TowerRenderer Members
TowerRenderer Class
MindFusion.Charting Namespace

# TowerRenderer.enumSegments Method

Enumerates the segments of the tower chart.

**Namespace:** MindFusion.Charting
**File:** TowerRenderer.js

## Syntax

**JavaScript**

```
function enumSegments (context, process)
```

### Parameters

*context*
Type: RenderContext
RenderContext. A RenderContext instance.

*process*
Type: ProcessSegment
ProcessSegment. A ProcessSegment callback.

## See Also

TowerRenderer Members
TowerRenderer Class
MindFusion.Charting Namespace

# TowerRenderer.enumSeries Method

Implements the SeriesContainer interface.

**Namespace:** MindFusion.Charting
**File:** TowerRenderer.js

## Syntax

**JavaScript**

```
function enumSeries ()
```

### Return Value

Type:
IEnumerable.<Series>. An instance of the IEnumerable<Series> class.

## See Also

TowerRenderer Members
TowerRenderer Class
MindFusion.Charting Namespace

# TowerRenderer.hitTest Method

SeriesRenderer.HitTest override. Hit-tests the slices representing data items.

**Namespace:** MindFusion.Charting
**File:** TowerRenderer.js

## Syntax

**JavaScript**

```
function hitTest (context, location)
```

### Parameters

*context*
Type: RenderContext
RenderContext. A RenderContext instance.

*location*
Type: Point
Point. A Point specifying where to look for a tower segment.

### Return Value

Type: HitResult
HitResult. A HitResult instance identifying the found data item.

## See Also

TowerRenderer Members
TowerRenderer Class
MindFusion.Charting Namespace

# TowerRenderer Properties

## TowerRenderer.layout Property

Gets or sets the layout of tower segments.

**Namespace:** MindFusion.Charting
**File:** TowerRenderer.js

## Syntax

**JavaScript**

```
get layout() {}
set layout(value) {}
```

### Property Value

Type: TowerLayout
*TowerLayout*

A member of the TowerLayout enumeration. The default value is Interleave.

## See Also

TowerRenderer Members
TowerRenderer Class
MindFusion.Charting Namespace

## TowerRenderer.leftSeries Property

Gets or sets the left-side Series drawn by this TowerRenderer.

**Namespace:** MindFusion.Charting
**File:** TowerRenderer.js

## Syntax

**JavaScript**

```
get leftSeries() {}
set leftSeries(value) {}
```

### Property Value

Type: Series
*Series*

An object that implements the Series interface.

## See Also

TowerRenderer Members
TowerRenderer Class
MindFusion.Charting Namespace

# TowerRenderer.margins Property

Gets or sets the size of margins around chart graphics.

**Namespace:** MindFusion.Charting
**File:** TowerRenderer.js

## Syntax

**JavaScript**

```
get margins() {}
set margins(value) {}
```

### Property Value

Type: Margins
*Margins*

A Margins instance specifying margin size on each side of the plot.

## See Also

TowerRenderer Members
TowerRenderer Class
MindFusion.Charting Namespace

# TowerRenderer.range Property

Gets or sets fixed data range size.

**Namespace:** MindFusion.Charting
**File:** TowerRenderer.js

## Syntax

**JavaScript**

```
get range() {}
set range(value) {}
```

### Property Value

Type: Number
*Number*

An optional number. The default value is null.

## See Also

TowerRenderer Members
TowerRenderer Class
MindFusion.Charting Namespace

# TowerRenderer.rightSeries Property

Gets or sets the right-side Series drawn by this TowerRenderer.

**Namespace:** MindFusion.Charting
**File:** TowerRenderer.js

## Syntax

**JavaScript**

```
get rightSeries() {}
set rightSeries(value) {}
```

### Property Value

Type: Series
*Series*

An object that implements the Series interface.

## See Also

TowerRenderer Members
TowerRenderer Class
MindFusion.Charting Namespace

# TowerRenderer.segmentShape Property

Gets or sets the shape of tower segments.

**Namespace:** MindFusion.Charting
**File:** TowerRenderer.js

## Syntax

**JavaScript**

```
get segmentShape() {}
set segmentShape(value) {}
```

### Property Value

Type: TowerSegmentShape
*TowerSegmentShape*

A member of the TowerSegmentShape enumeration. The default value is Triangle.

## See Also

TowerRenderer Members
TowerRenderer Class

## TowerRenderer.seriesPadding Property

Gets or sets the padding distance between series.

**Namespace:** MindFusion.Charting
**File:** TowerRenderer.js

### Syntax

**JavaScript**

```
get seriesPadding() {}
set seriesPadding(value) {}
```

#### Property Value

Type: Number
*Number*

The distance between LeftSeries and RightSeries graphics.

### See Also

TowerRenderer Members
TowerRenderer Class
MindFusion.Charting Namespace

## TowerSegmentShape Enumeration

Identifies the shapes rendered by a ScatterRenderer.

**Namespace:** MindFusion.Charting
**File:** TowerSegmentShape.js

### Syntax

**JavaScript**

```
// enum
TowerSegmentShape = {}
```

### Members

| | Member name | Description |
|---|---|---|
| | **Rectangle** | Draw rectangular segments. |
| | **Triangle** | Draw triangular segments. |

### See Also

MindFusion.Charting Namespace

# UniformSeriesStyle Class

Implements SeriesStyle using uniform attributes for all elements in all series.

## Syntax

**JavaScript**

```
// class
UniformSeriesStyle.prototype = {}
```

## Inheritance Hierarchy

## See Also

UniformSeriesStyle Members
MindFusion.Charting Namespace

# UniformSeriesStyle Members

Implements SeriesStyle using uniform attributes for all elements in all series.

The following tables list the members exposed by the UniformSeriesStyle type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | UniformSeriesStyle | Initializes a new instance of the UniformSeriesStyle class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | uniformFill | Gets or sets a Brush used to fill all elements of all series uniformly. |
|  | uniformStroke | Gets or sets a Brush used to stroke all elements of all series uniformly. |
|  | uniformStrokeDashStyle | Gets or sets uniform stroke dash style for all elements of all series. |
|  | uniformStrokeThickness | Gets or sets uniform stroke thickness for all elements of all series. |

## Public Methods

| | Name | Description |
|---|---|---|
|  | fill | Implements SeriesStyle.Fill. Returns UniformFill for any data item. |
|  | stroke | Implements SeriesStyle.Stroke. Returns UniformStroke for any data item. |

| | | |
|---|---|---|
| | strokeDashStyle | Implements SeriesStyle.StrokeDashStyle. Returns UniformStrokeDashStyle for any data item. |
| | strokeThickness | Implements SeriesStyle.StrokeThickness. Returns UniformStrokeThickness for any data item. |

## See Also

UniformSeriesStyle Class
MindFusion.Charting Namespace

# UniformSeriesStyle Constructor

Initializes a new instance of the UniformSeriesStyle class.

**Namespace:** MindFusion.Charting
**File:** UniformSeriesStyle.js

## Syntax

**JavaScript**

```
function UniformSeriesStyle ([fill, [stroke, [strokeThickness, [strokeDashStyle]]]])
```

### Parameters

*fill*
Optional.
Type: Brush
A Brush used to fill all series elements.

*stroke*
Optional.
Type: Brush
A Brush used to stroke all series elements.

*strokeThickness*
Optional.
Type: Number
Uniform thickness of series elements' strokes.

*strokeDashStyle*
Optional.
Type: DashStyle
Uniform dash style of series elements' strokes.

## See Also

UniformSeriesStyle Class
MindFusion.Charting Namespace

# UniformSeriesStyle Methods

# UniformSeriesStyle.fill Method

Implements SeriesStyle.Fill. Returns UniformFill for any data item.

**Namespace:** MindFusion.Charting
**File:** UniformSeriesStyle.js

## Syntax

```
JavaScript
function fill (seriesIndex, dataIndex)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series.

*dataIndex*
Type: Number
An integer index of a data item in specified series.

### Return Value

Type: Brush
A Brush instance.

## See Also

UniformSeriesStyle Members
UniformSeriesStyle Class
MindFusion.Charting Namespace

# UniformSeriesStyle.stroke Method

Implements SeriesStyle.Stroke. Returns UniformStroke for any data item.

**Namespace:** MindFusion.Charting
**File:** UniformSeriesStyle.js

## Syntax

```
JavaScript
function stroke (seriesIndex, dataIndex)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series.

*dataIndex*
Type: Number
An integer index of a data item in specified series.

### Return Value

Type: Brush
A Brush instance.

## See Also

UniformSeriesStyle Members
UniformSeriesStyle Class
MindFusion.Charting Namespace

# UniformSeriesStyle.strokeDashStyle Method

Implements SeriesStyle.StrokeDashStyle. Returns UniformStrokeDashStyle for any data item.

**Namespace:** MindFusion.Charting
**File:** UniformSeriesStyle.js

## Syntax

```
JavaScript
function strokeDashStyle (seriesIndex, dataIndex)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series.

*dataIndex*
Type: Number
An integer index of a data item in specified series.

### Return Value

Type: DashStyle
A nullable DashStyle value specifying stroke dash style.

## See Also

UniformSeriesStyle Members
UniformSeriesStyle Class
MindFusion.Charting Namespace

# UniformSeriesStyle.strokeThickness Method

Implements SeriesStyle.StrokeThickness. Returns UniformStrokeThickness for any data item.

**Namespace:** MindFusion.Charting
**File:** UniformSeriesStyle.js

## Syntax

```
JavaScript
function strokeThickness (seriesIndex, dataIndex)
```

### Parameters

*seriesIndex*
Type: Number
An integer index of a Series.

*dataIndex*

Type: Number
An integer index of a data item in specified series.

### Return Value

Type: Number
A nullable double value specifying stroke thickness.

## See Also

UniformSeriesStyle Members
UniformSeriesStyle Class
MindFusion.Charting Namespace

# UniformSeriesStyle Properties

# UniformSeriesStyle.uniformFill Property

Gets or sets a Brush used to fill all elements of all series uniformly.

**Namespace:** MindFusion.Charting
**File:** UniformSeriesStyle.js

## Syntax

**JavaScript**

```
get uniformFill() {}
```

### Property Value

Type: Brush

## See Also

UniformSeriesStyle Members
UniformSeriesStyle Class
MindFusion.Charting Namespace

# UniformSeriesStyle.uniformStroke Property

Gets or sets a Brush used to stroke all elements of all series uniformly.

**Namespace:** MindFusion.Charting
**File:** UniformSeriesStyle.js

## Syntax

**JavaScript**

```
get uniformStroke() {}
```

### Property Value

Type: Brush

## See Also

UniformSeriesStyle Members
UniformSeriesStyle Class
MindFusion.Charting Namespace

### UniformSeriesStyle.uniformStrokeDashStyle Property

Gets or sets uniform stroke dash style for all elements of all series.

**Namespace:** MindFusion.Charting
**File:** UniformSeriesStyle.js

## Syntax

**JavaScript**

```
get uniformStrokeDashStyle() {}
```

#### Property Value

Type: DashStyle

## See Also

UniformSeriesStyle Members
UniformSeriesStyle Class
MindFusion.Charting Namespace

### UniformSeriesStyle.uniformStrokeThickness Property

Gets or sets uniform stroke thickness for all elements of all series.

**Namespace:** MindFusion.Charting
**File:** UniformSeriesStyle.js

## Syntax

**JavaScript**

```
get uniformStrokeThickness() {}
```

#### Property Value

Type: Number

## See Also

UniformSeriesStyle Members
UniformSeriesStyle Class
MindFusion.Charting Namespace

# Vector Class

Represents a two-dimensional vector.

## Syntax

| JavaScript |
| --- |
| `// class`<br>`Vector.prototype = {}` |

## Inheritance Hierarchy

## See Also

Vector Members
MindFusion.Charting Namespace

# Vector Members

Represents a two-dimensional vector.

The following tables list the members exposed by the Vector type.

## Public Constructors

| | Name | Description |
| --- | --- | --- |
|  | Vector | Initializes a new instance of the Vector structure. |

## Public Properties

| | Name | Description |
| --- | --- | --- |
|  | length | Gets the length of this vector. |
|  | lengthSquared | Gets the squared length of this vector. |
|  | x | Gets or sets the X component of this vector. |
|  | y | Gets or sets the Y component of this vector. |

## Public Methods

| | Name | Description |
| --- | --- | --- |
|  | add | Adds two vectors. |
|  | addPoint | Translates point by a vector. |
|  | angleBetween | Determines the angle between two vectors. |
|  | crossProduct | Returns the size of cross-product of two vectors. |
|  | determinant | Finds the determinant of two vectors. |
|  | divide | Divides a vector by a scalar. |
|  | multiply | Returns the dot product of two vectors. |
|  | multiplyScalarAndVector | Multiplies vector by a scalar. |

| | | |
|---|---|---|
| | multiplyVectorAndScalar | Multiplies vector by a scalar. |
| | negate | Reverses the direction of this vector. |
| | normalize | Makes this vector a unit vector. |
| | subtract | Subtracts two vectors. |

## See Also

Vector Class
MindFusion.Charting Namespace

# Vector Constructor

Initializes a new instance of the Vector structure.

**Namespace:** MindFusion.Charting
**File:** Vector.js

## Syntax

**JavaScript**

```
function Vector (x, y)
```

### Parameters

*x*
Type: Number
X component of the vector.

*y*
Type: Number
Y component of the vector.

## See Also

Vector Class
MindFusion.Charting Namespace

# Vector Methods

# Vector.add Method

Adds two vectors.

**Namespace:** MindFusion.Charting
**File:** Vector.js

## Syntax

**JavaScript**

```
function add (vector1, vector2)
```

### Parameters

*vector1*
Type: Vector
The first vector.

*vector2*
Type: Vector
The second vector.

### Return Value

Type: Vector
A vector representing the sum of specified vectors.

## See Also

Vector Members
Vector Class
MindFusion.Charting Namespace

# Vector.addPoint Method

Translates point by a vector.

**Namespace:** MindFusion.Charting
**File:** Vector.js

## Syntax

**JavaScript**

```
function addPoint (vector, point)
```

### Parameters

*vector*
Type: Vector
The vector to add.

*point*
Type: Point
The reference point.

### Return Value

Type: Point
The translated point.

## See Also

Vector Members
Vector Class
MindFusion.Charting Namespace

# Vector.angleBetween Method

Determines the angle between two vectors.

**Namespace:** MindFusion.Charting
**File:** Vector.js

## Syntax

```
JavaScript
function angleBetween (vector1, vector2)
```

### Parameters

*vector1*
Type: Vector
The first vector.

*vector2*
Type: Vector
The second vector.

### Return Value

Type: Number
The angle between specified vectors.

## See Also

Vector Members
Vector Class
MindFusion.Charting Namespace

# Vector.crossProduct Method

Returns the size of cross-product of two vectors.

**Namespace:** MindFusion.Charting
**File:** Vector.js

## Syntax

```
JavaScript
function crossProduct (vector1, vector2)
```

### Parameters

*vector1*
Type: Vector
The first vector.

*vector2*
Type: Vector
The second vector.

### Return Value

Type: Number
The size of cross-product.

## See Also

Vector Members
Vector Class
MindFusion.Charting Namespace

# Vector.determinant Method

Finds the determinant of two vectors.

**Namespace:** MindFusion.Charting
**File:** Vector.js

## Syntax

```
JavaScript
function determinant (vector1, vector2)
```

### Parameters

*vector1*
Type: Vector
The first vector.

*vector2*
Type: Vector
The second vector.

### Return Value

Type: Number
The determinant.

## See Also

Vector Members
Vector Class
MindFusion.Charting Namespace

# Vector.divide Method

Divides a vector by a scalar.

**Namespace:** MindFusion.Charting
**File:** Vector.js

## Syntax

```
JavaScript
function divide (vector, scalar)
```

### Parameters

*vector*
Type: Vector
The vector to divide.

*scalar*

Type: Number
The scalar value.

### Return Value

Type: Vector
A vector containing the division result.

## See Also

Vector Members
Vector Class
MindFusion.Charting Namespace

# Vector.multiply Method

Returns the dot product of two vectors.

**Namespace:** MindFusion.Charting
**File:** Vector.js

## Syntax

**JavaScript**

```
function multiply (vector1, vector2)
```

### Parameters

*vector1*

Type: Vector
The first vector.

*vector2*

Type: Vector
The second vector.

### Return Value

Type: Number
The dot product.

## See Also

Vector Members
Vector Class
MindFusion.Charting Namespace

# Vector.multiplyScalarAndVector Method

Multiplies vector by a scalar.

**Namespace:** MindFusion.Charting
**File:** Vector.js

## Syntax

```
JavaScript
function multiplyScalarAndVector (scalar, vector)
```

### Parameters

*scalar*
Type: Number
The scalar value.

*vector*
Type: Vector
The vector to multiply.

### Return Value

Type: Vector
A vector containing the multiplication result.

## See Also

Vector Members
Vector Class
MindFusion.Charting Namespace

# Vector.multiplyVectorAndScalar Method

Multiplies vector by a scalar.

**Namespace:** MindFusion.Charting
**File:** Vector.js

## Syntax

```
JavaScript
function multiplyVectorAndScalar (vector, scalar)
```

### Parameters

*vector*
Type: Vector
The vector to multiply.

*scalar*
Type: Number
The scalar value.

### Return Value

Type: Vector
A vector containing the multiplication result.

## See Also

Vector Members
Vector Class
MindFusion.Charting Namespace

# Vector.negate Method

Reverses the direction of this vector.

**Namespace:** MindFusion.Charting
**File:** Vector.js

## Syntax

**JavaScript**

```
function negate ()
```

## See Also

Vector Members
Vector Class
MindFusion.Charting Namespace

# Vector.normalize Method

Makes this vector a unit vector.

**Namespace:** MindFusion.Charting
**File:** Vector.js

## Syntax

**JavaScript**

```
function normalize ()
```

## See Also

Vector Members
Vector Class
MindFusion.Charting Namespace

# Vector.subtract Method

Subtracts two vectors.

**Namespace:** MindFusion.Charting
**File:** Vector.js

## Syntax

**JavaScript**

```
function subtract (vector1, vector2)
```

### Parameters

*vector1*
Type: Vector
The first vector.

*vector2*
Type: Vector
The second vector.

### Return Value

Type: Vector
A vector representing the difference between specified vectors.

## See Also

Vector Members
Vector Class
MindFusion.Charting Namespace

# Vector Properties

# Vector.length Property

Gets the length of this vector.

**Namespace:** MindFusion.Charting
**File:** Vector.js

## Syntax

**JavaScript**

```
get length() {}
```

### Property Value

Type: Number

## See Also

Vector Members
Vector Class
MindFusion.Charting Namespace

# Vector.lengthSquared Property

Gets the squared length of this vector.

**Namespace:** MindFusion.Charting
**File:** Vector.js

## Syntax

**JavaScript**

```
get lengthSquared() {}
```

### Property Value

Type: Number

## See Also

Vector Members
Vector Class
MindFusion.Charting Namespace

# Vector.x Property

Gets or sets the X component of this vector.

**Namespace:** MindFusion.Charting
**File:** Vector.js

## Syntax

```
JavaScript
get x() {}
```

### Property Value

Type: Number

## See Also

Vector Members
Vector Class
MindFusion.Charting Namespace

# Vector.y Property

Gets or sets the Y component of this vector.

**Namespace:** MindFusion.Charting
**File:** Vector.js

## Syntax

```
JavaScript
get y() {}
```

### Property Value

Type: Number

## See Also

Vector Members
Vector Class
MindFusion.Charting Namespace

# WebComponents Class

Implements registration of web components.

## Syntax

| JavaScript |
| --- |
| `// class`<br>`WebComponents.prototype = {}` |

## Inheritance Hierarchy

**MindFusion.Charting.WebComponents**

## See Also

WebComponents Members
MindFusion.Charting Namespace

# WebComponents Members

Implements registration of web components.

The following tables list the members exposed by the WebComponents type.

## Public Methods

| | Name | Description |
| --- | --- | --- |
|  | register | Registers the controls in the package as web components. |

## See Also

WebComponents Class
MindFusion.Charting Namespace

# WebComponents Methods

## WebComponents.register Method

Registers the controls in the package as web components.

**Namespace:** MindFusion.Charting
**File:** WebComponents.js

## Syntax

| JavaScript |
| --- |
| `function register ()` |

## Remarks

Call this method to register the control classes from Charting package as web components. After registration, the browser instantiates corresponding classes for following tags in HTML code:

| HTML | JS Class |
| --- | --- |

| | |
|---|---|
| <mindfusion-areachart> | AreaChart |
| <mindfusion-barchart> | BarChart |
| <mindfusion-barchart3d> | BarChart3D |
| <mindfusion-bubblechart> | BubbleChart |
| <mindfusion-candlestickchart> | CandlestickChart |
| <mindfusion-funnelchart> | FunnelChart |
| <mindfusion-linechart> | LineChart |
| <mindfusion-piechart> | PieChart |
| <mindfusion-radarchart> | RadarChart |
| <mindfusion-scatterchart> | ScatterChart |
| <mindfusion-towerchart> | TowerChart |

You can get a reference to objects created by the browser by calling the find method of corresponding class with **id** attribute's value as argument.

## See Also

WebComponents Members
WebComponents Class
MindFusion.Charting Namespace

# XAxisRenderer Class

A Component that renders horizontal Axis ranges.

## Syntax

**JavaScript**

```
// class
XAxisRenderer.prototype = {}
```

## Inheritance Hierarchy

## See Also

XAxisRenderer Members
MindFusion.Charting Namespace

### XAxisRenderer Members

A Component that renders horizontal Axis ranges.

The following tables list the members exposed by the XAxisRenderer type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | XAxisRenderer | Initializes a new instance of the XAxisRenderer class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | plotBottomSide | Gets or sets whether this XAxisRenderer is shown at the bottom side of a plot. |

## Public Methods

| | Name | Description |
|---|---|---|
| | draw | Component.draw override. Draws this component in specified RenderContext. |
| | effectiveAxis | AxisRenderer.effectiveAxis override. Gets the effective horizontal Axis in current context,getting one from Plot2D or chart control if there's no local Axis associated with this renderer. |
| | measure | Component.measure override. Measures the desired size of this component. |

## See Also

XAxisRenderer Class
MindFusion.Charting Namespace

# XAxisRenderer Constructor

Initializes a new instance of the XAxisRenderer class.

**Namespace:** MindFusion.Charting
**File:** XAxisRenderer.js

## Syntax

**JavaScript**

```
function XAxisRenderer (axis)
```

### Parameters

*axis*
Type: Axis
The Axis that will be drawn by this object.

## See Also

XAxisRenderer Class
MindFusion.Charting Namespace

# XAxisRenderer Methods

## XAxisRenderer.draw Method

Component.draw override. Draws this component in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** XAxisRenderer.js

## Syntax

**JavaScript**

```
function draw (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

XAxisRenderer Members
XAxisRenderer Class
MindFusion.Charting Namespace

## XAxisRenderer.effectiveAxis Method

AxisRenderer.effectiveAxis override. Gets the effective horizontal Axis in current context,getting one from Plot2D or chart control if there's no local Axis associated with this renderer.

**Namespace:** MindFusion.Charting
**File:** XAxisRenderer.js

## Syntax

**JavaScript**

```
function effectiveAxis (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: Axis
An Axis instance.

## See Also

XAxisRenderer Members
XAxisRenderer Class
MindFusion.Charting Namespace

## XAxisRenderer.measure Method

Component.measure override. Measures the desired size of this component.

**Namespace:** MindFusion.Charting
**File:** XAxisRenderer.js

## Syntax

```
JavaScript
function measure (maxWidth, maxHeight, context)
```

### Parameters

*maxWidth*
Type: Number
The maximum width provided by parent component.

*maxHeight*
Type: Number
The maximum height provided by parent component.

*context*
Type: RenderContext
A RenderContext instance.

## See Also

XAxisRenderer Members
XAxisRenderer Class
MindFusion.Charting Namespace

## XAxisRenderer Properties

## XAxisRenderer.plotBottomSide Property

Gets or sets whether this XAxisRenderer is shown at the bottom side of a plot.

**Namespace:** MindFusion.Charting
**File:** XAxisRenderer.js

## Syntax

```
JavaScript
get plotBottomSide() {}
```

### Property Value

Type: Boolean

## See Also

XAxisRenderer Members
XAxisRenderer Class
MindFusion.Charting Namespace

# XmlPersistContext Class

Contains methods that facilitate serializing and deserializing objects to and from XML documents.

## Syntax

**JavaScript**

```
// class
XmlPersistContext.prototype = {}
```

## Inheritance Hierarchy

## See Also

XmlPersistContext Members
MindFusion.Charting Namespace

## XmlPersistContext Members

Contains methods that facilitate serializing and deserializing objects to and from XML documents.

The following tables list the members exposed by the XmlPersistContext type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | XmlPersistContext | Initializes a new instance of the XmlPersistContext class. |

## Public Methods

| | Name | Description |
|---|---|---|
| | addChildElement | Adds a new child node with the specified name andvalue to the specified parent node. |
| | readBool | Reads a boolean value with the specified name. |
| | readBrush | Reads a MindFusion.Charting.Drawing.Brush object with the specified name. |
| | readBrushes | Reads all brushes. |
| | readBrushList | Reads a list of brushes. |
| | readColor | Reads a color with the specified name. |
| | readDashStyleList | Reads a list of dash styles. |
| | readEnum | Reads an enum value with the specified name. |

| | | |
|---|---|---|
| | readFloat | Reads a float value with the specified name. |
| | readFloatList | Reads a list of floats. |
| | readInt | Reads an integer value with the specified name. |
| | readObject | Reads an object with the specified name. |
| | readString | Reads a string value with the specified name. |
| | readStringFormat | Reads a StringFormat object with the specified name. |
| | writeBool | Writes a boolean value with the specified name. |
| | writeBrush | Writes a reference to the specified brush. The brushis registered within the internal hashtable forsubsequent serialization. |
| | writeBrushes | Writes all currently accumulated brushes. |
| | writeBrushList | Writes a list of brushes. |
| | writeColor | Writes the specified color. |
| | writeColorList | Writes a list of colors. |
| | writeDashStyleList | Writes a list of dash styles. |
| | writeEnum | Writes the specified enum value. |
| | writeFloat | Writes a float value with the specified name. |
| | writeFloatList | Writes a list of floating-point numbers. |
| | writeInt | Writes an integer value with the specified name. |
| | writeObject | Writes the specified object under the specified name. |
| | writeString | Writes a string value with the specified name. |

## See Also

XmlPersistContext Class
MindFusion.Charting Namespace

# XmlPersistContext Constructor

Initializes a new instance of the XmlPersistContext class.

**Namespace:** MindFusion.Charting
**File:** XmlPersistContext.js

## Syntax

**JavaScript**

```
function XmlPersistContext (document, fileVersion)
```

### Parameters

*document*

Type: XMLDocument
The associated XMLDocument.

*fileVersion*
Type: number
Specifies the current file format number.

## See Also

XmlPersistContext Class
MindFusion.Charting Namespace

# XmlPersistContext Methods

## XmlPersistContext.addChildElement Method

Adds a new child node with the specified name andvalue to the specified parent node.

**Namespace:** MindFusion.Charting
**File:** XmlPersistContext.js

## Syntax

**JavaScript**

```
function addChildElement (elementName, parentElement, innerText)
```

### Parameters

*elementName*
Type: void

*parentElement*
Type: void

*innerText*
Type: void

## See Also

XmlPersistContext Members
XmlPersistContext Class
MindFusion.Charting Namespace

## XmlPersistContext.readBool Method

Reads a boolean value with the specified name.

**Namespace:** MindFusion.Charting
**File:** XmlPersistContext.js

## Syntax

**JavaScript**

```
function readBool (elementName, parentElement, defaultValue)
```

### Parameters

*elementName*
Type: void

*parentElement*
Type: void

*defaultValue*
Type: void

## See Also

XmlPersistContext Members
XmlPersistContext Class
MindFusion.Charting Namespace

# XmlPersistContext.readBrush Method

Reads a MindFusion.Charting.Drawing.Brush object with the specified name.

**Namespace:** MindFusion.Charting
**File:** XmlPersistContext.js

## Syntax

**JavaScript**

```
function readBrush (elementName, parentElement, defaultValue)
```

### Parameters

*elementName*
Type: void

*parentElement*
Type: void

*defaultValue*
Type: void

## See Also

XmlPersistContext Members
XmlPersistContext Class
MindFusion.Charting Namespace

# XmlPersistContext.readBrushes Method

Reads all brushes.

**Namespace:** MindFusion.Charting
**File:** XmlPersistContext.js

## Syntax

**JavaScript**

```
function readBrushes (parentElement)
```

### Parameters

*parentElement*
Type: void

## See Also

XmlPersistContext Members
XmlPersistContext Class
MindFusion.Charting Namespace

# XmlPersistContext.readBrushList Method

Reads a list of brushes.

**Namespace:** MindFusion.Charting
**File:** XmlPersistContext.js

## Syntax

**JavaScript**

```
function readBrushList (elementName, subElementName, parentElement)
```

### Parameters

*elementName*
Type: void

*subElementName*
Type: void

*parentElement*
Type: void

## See Also

XmlPersistContext Members
XmlPersistContext Class
MindFusion.Charting Namespace

# XmlPersistContext.readColor Method

Reads a color with the specified name.

**Namespace:** MindFusion.Charting
**File:** XmlPersistContext.js

## Syntax

**JavaScript**

```
function readColor (elementName, parentElement, defaultValue)
```

### Parameters

*elementName*
Type: void

*parentElement*
Type: void

*defaultValue*
Type: void

## See Also

XmlPersistContext Members
XmlPersistContext Class
MindFusion.Charting Namespace

# XmlPersistContext.readDashStyleList Method

Reads a list of dash styles.

**Namespace:** MindFusion.Charting
**File:** XmlPersistContext.js

## Syntax

**JavaScript**

```
function readDashStyleList (elementName, subElementName, parentElement)
```

### Parameters

*elementName*
Type: void

*subElementName*
Type: void

*parentElement*
Type: void

## See Also

XmlPersistContext Members
XmlPersistContext Class
MindFusion.Charting Namespace

# XmlPersistContext.readEnum Method

Reads an enum value with the specified name.

**Namespace:** MindFusion.Charting
**File:** XmlPersistContext.js

## Syntax

**JavaScript**

```
function readEnum (elementName, parentElement, defaultValue)
```

### Parameters

*elementName*

Type: void

*parentElement*
Type: void

*defaultValue*
Type: void

## See Also

XmlPersistContext Members
XmlPersistContext Class
MindFusion.Charting Namespace

# XmlPersistContext.readFloat Method

Reads a float value with the specified name.

**Namespace:** MindFusion.Charting
**File:** XmlPersistContext.js

## Syntax

**JavaScript**

```
function readFloat (elementName, parentElement, defaultValue)
```

### Parameters

*elementName*
Type: void

*parentElement*
Type: void

*defaultValue*
Type: void

## See Also

XmlPersistContext Members
XmlPersistContext Class
MindFusion.Charting Namespace

# XmlPersistContext.readFloatList Method

Reads a list of floats.

**Namespace:** MindFusion.Charting
**File:** XmlPersistContext.js

## Syntax

**JavaScript**

```
function readFloatList (elementName, subElementName, parentElement)
```

### Parameters

*elementName*
Type: void

*subElementName*
Type: void

*parentElement*
Type: void

## See Also

XmlPersistContext Members
XmlPersistContext Class
MindFusion.Charting Namespace

# XmlPersistContext.readInt Method

Reads an integer value with the specified name.

**Namespace:** MindFusion.Charting
**File:** XmlPersistContext.js

## Syntax

**JavaScript**

```
function readInt (elementName, parentElement, defaultValue)
```

### Parameters

*elementName*
Type: void

*parentElement*
Type: void

*defaultValue*
Type: void

## See Also

XmlPersistContext Members
XmlPersistContext Class
MindFusion.Charting Namespace

# XmlPersistContext.readObject Method

Reads an object with the specified name.

**Namespace:** MindFusion.Charting
**File:** XmlPersistContext.js

## Syntax

**JavaScript**

```
function readObject (elementName, parentElement)
```

### Parameters

*elementName*
Type: void

*parentElement*
Type: void

## See Also

XmlPersistContext Members
XmlPersistContext Class
MindFusion.Charting Namespace

# XmlPersistContext.readString Method

Reads a string value with the specified name.

**Namespace:** MindFusion.Charting
**File:** XmlPersistContext.js

## Syntax

**JavaScript**

```
function readString (elementName, parentElement, defaultValue)
```

### Parameters

*elementName*
Type: void

*parentElement*
Type: void

*defaultValue*
Type: void

## See Also

XmlPersistContext Members
XmlPersistContext Class
MindFusion.Charting Namespace

# XmlPersistContext.readStringFormat Method

Reads a StringFormat object with the specified name.

**Namespace:** MindFusion.Charting
**File:** XmlPersistContext.js

## Syntax

**JavaScript**

```
function readStringFormat (elementName, parentElement, defultFormat)
```

### Parameters

*elementName*

Type: void

*parentElement*
Type: void

*defultFormat*
Type: void

## See Also

XmlPersistContext Members
XmlPersistContext Class
MindFusion.Charting Namespace

# XmlPersistContext.writeBool Method

Writes a boolean value with the specified name.

**Namespace:** MindFusion.Charting
**File:** XmlPersistContext.js

## Syntax

**JavaScript**

```
function writeBool (boolValue, elementName, parentElement)
```

### Parameters

*boolValue*
Type: void

*elementName*
Type: void

*parentElement*
Type: void

## See Also

XmlPersistContext Members
XmlPersistContext Class
MindFusion.Charting Namespace

# XmlPersistContext.writeBrush Method

Writes a reference to the specified brush. The brushis registered within the internal hashtable forsubsequent serialization.

**Namespace:** MindFusion.Charting
**File:** XmlPersistContext.js

## Syntax

**JavaScript**

```
function writeBrush (brush, elementName, parentElement)
```

### Parameters

*brush*
Type: void

*elementName*
Type: void

*parentElement*
Type: void

## See Also

XmlPersistContext Members
XmlPersistContext Class
MindFusion.Charting Namespace

# XmlPersistContext.writeBrushes Method

Writes all currently accumulated brushes.

**Namespace:** MindFusion.Charting
**File:** XmlPersistContext.js

## Syntax

**JavaScript**

```
function writeBrushes (parentElement)
```

### Parameters

*parentElement*
Type: void

## See Also

XmlPersistContext Members
XmlPersistContext Class
MindFusion.Charting Namespace

# XmlPersistContext.writeBrushList Method

Writes a list of brushes.

**Namespace:** MindFusion.Charting
**File:** XmlPersistContext.js

## Syntax

**JavaScript**

```
function writeBrushList (list, elementName, subElementName, parentElement)
```

### Parameters

*list*
Type: void

*elementName*

Type: void

*subElementName*
Type: void

*parentElement*
Type: void

## See Also

XmlPersistContext Members
XmlPersistContext Class
MindFusion.Charting Namespace

# XmlPersistContext.writeColor Method

Writes the specified color.

**Namespace:** MindFusion.Charting
**File:** XmlPersistContext.js

## Syntax

**JavaScript**

```
function writeColor (color, elementName, parentElement)
```

### Parameters

*color*
Type: void

*elementName*
Type: void

*parentElement*
Type: void

## See Also

XmlPersistContext Members
XmlPersistContext Class
MindFusion.Charting Namespace

# XmlPersistContext.writeColorList Method

Writes a list of colors.

**Namespace:** MindFusion.Charting
**File:** XmlPersistContext.js

## Syntax

**JavaScript**

```
function writeColorList (list, elementName, subElementName, parentElement)
```

### Parameters

*list*
Type: void

*elementName*
Type: void

*subElementName*
Type: void

*parentElement*
Type: void

## See Also

XmlPersistContext Members
XmlPersistContext Class
MindFusion.Charting Namespace

# XmlPersistContext.writeDashStyleList Method

Writes a list of dash styles.

**Namespace:** MindFusion.Charting
**File:** XmlPersistContext.js

## Syntax

**JavaScript**

```
function writeDashStyleList (list, elementName, subElementName, parentElement)
```

### Parameters

*list*
Type: void

*elementName*
Type: void

*subElementName*
Type: void

*parentElement*
Type: void

## See Also

XmlPersistContext Members
XmlPersistContext Class
MindFusion.Charting Namespace

# XmlPersistContext.writeEnum Method

Writes the specified enum value.

**Namespace:** MindFusion.Charting
**File:** XmlPersistContext.js

## Syntax

JavaScript

```
function writeEnum (enumValue, elementName, parentElement)
```

### Parameters

*enumValue*
Type: void

*elementName*
Type: void

*parentElement*
Type: void

## See Also

XmlPersistContext Members
XmlPersistContext Class
MindFusion.Charting Namespace

# XmlPersistContext.writeFloat Method

Writes a float value with the specified name.

**Namespace:** MindFusion.Charting
**File:** XmlPersistContext.js

## Syntax

JavaScript

```
function writeFloat (floatValue, elementName, parentElement)
```

### Parameters

*floatValue*
Type: void

*elementName*
Type: void

*parentElement*
Type: void

## See Also

XmlPersistContext Members
XmlPersistContext Class
MindFusion.Charting Namespace

# XmlPersistContext.writeFloatList Method

Writes a list of floating-point numbers.

**Namespace:** MindFusion.Charting
**File:** XmlPersistContext.js

## Syntax

**JavaScript**

```
function writeFloatList (list, elementName, subElementName, parentElement)
```

### Parameters

*list*
Type: void

*elementName*
Type: void

*subElementName*
Type: void

*parentElement*
Type: void

## See Also

XmlPersistContext Members
XmlPersistContext Class
MindFusion.Charting Namespace

# XmlPersistContext.writeInt Method

Writes an integer value with the specified name.

**Namespace:** MindFusion.Charting
**File:** XmlPersistContext.js

## Syntax

**JavaScript**

```
function writeInt (intValue, elementName, parentElement)
```

### Parameters

*intValue*
Type: void

*elementName*
Type: void

*parentElement*
Type: void

## See Also

XmlPersistContext Members
XmlPersistContext Class
MindFusion.Charting Namespace

# XmlPersistContext.writeObject Method

Writes the specified object under the specified name.

**Namespace:** MindFusion.Charting
**File:** XmlPersistContext.js

## Syntax

```
JavaScript
function writeObject (objectValue, elementName, parentElement)
```

### Parameters

*objectValue*
Type: void

*elementName*
Type: void

*parentElement*
Type: void

## See Also

XmlPersistContext Members
XmlPersistContext Class
MindFusion.Charting Namespace

# XmlPersistContext.writeString Method

Writes a string value with the specified name.

**Namespace:** MindFusion.Charting
**File:** XmlPersistContext.js

## Syntax

```
JavaScript
function writeString (stringValue, elementName, parentElement)
```

### Parameters

*stringValue*
Type: void

*elementName*
Type: void

*parentElement*
Type: void

## See Also

XmlPersistContext Members
XmlPersistContext Class
MindFusion.Charting Namespace

# YAxisRenderer Class

A Component that renders vertical Axis ranges.

## Syntax

```
// class
YAxisRenderer.prototype = {}
```

## Inheritance Hierarchy

## See Also

YAxisRenderer Members
MindFusion.Charting Namespace

# YAxisRenderer Members

A Component that renders vertical Axis ranges.

The following tables list the members exposed by the YAxisRenderer type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | YAxisRenderer | Initializes a new instance of the YAxisRenderer class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | plotLeftSide | Gets or sets whether this YAxisRenderer is shown at the left side of a plot. |

## Public Methods

| | Name | Description |
|---|---|---|
| | draw | Component.draw override. Draws this component in specified RenderContext. |
| | effectiveAxis | AxisRenderer.effectiveAxis override. Gets the effective vertical Axis in current context,getting one from Plot2D or chart control if there's no local Axis associated with this renderer. |
| | measure | Component.measure override. Measures the desired size of this component. |

## See Also

YAxisRenderer Class
MindFusion.Charting Namespace

# YAxisRenderer Constructor

Initializes a new instance of the YAxisRenderer class.

**Namespace:** MindFusion.Charting
**File:** YAxisRenderer.js

## Syntax

**JavaScript**

```
function YAxisRenderer (axis, [xAxis])
```

### Parameters

*axis*
Type: Axis
The Axis that will be drawn by this object.

*xAxis*
Optional.
Type: Axis
The Axis used to check data item visibility.

## See Also

YAxisRenderer Class
MindFusion.Charting Namespace

# YAxisRenderer Methods

# YAxisRenderer.draw Method

Component.draw override. Draws this component in specified RenderContext.

**Namespace:** MindFusion.Charting
**File:** YAxisRenderer.js

## Syntax

**JavaScript**

```
function draw (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

YAxisRenderer Members
YAxisRenderer Class
MindFusion.Charting Namespace

# YAxisRenderer.effectiveAxis Method

AxisRenderer.effectiveAxis override. Gets the effective vertical Axis in current context,getting one from Plot2D or chart control if there's no local Axis associated with this renderer.

**Namespace:** MindFusion.Charting
**File:** YAxisRenderer.js

## Syntax

**JavaScript**

```
function effectiveAxis (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: Axis
An Axis instance.

## See Also

YAxisRenderer Members
YAxisRenderer Class
MindFusion.Charting Namespace

# YAxisRenderer.measure Method

Component.measure override. Measures the desired size of this component.

**Namespace:** MindFusion.Charting
**File:** YAxisRenderer.js

## Syntax

**JavaScript**

```
function measure (maxWidth, maxHeight, context)
```

### Parameters

*maxWidth*
Type: Number
The maximum width provided by parent component.

*maxHeight*
Type: Number
The maximum height provided by parent component.

*context*
Type: RenderContext
A RenderContext instance.

## See Also

YAxisRenderer Members
YAxisRenderer Class
MindFusion.Charting Namespace

# YAxisRenderer Properties

## YAxisRenderer.plotLeftSide Property

Gets or sets whether this YAxisRenderer is shown at the left side of a plot.

**Namespace:** MindFusion.Charting
**File:** YAxisRenderer.js

## Syntax

**JavaScript**

```
get plotLeftSide() {}
```

### Property Value

Type: Boolean

## See Also

YAxisRenderer Members
YAxisRenderer Class
MindFusion.Charting Namespace

# ZoomController Class

Represents a controller that zooms into plot's data range.

## Syntax

**JavaScript**

```
// class
ZoomController.prototype = {}
```

## Inheritance Hierarchy

## See Also

ZoomController Members
MindFusion.Charting Namespace

# ZoomController Members

Represents a controller that zooms into plot's data range.

The following tables list the members exposed by the ZoomController type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | ZoomController | Initializes a new instance of the ZoomController class. |

## Public Methods

| | Name | Description |
|---|---|---|
| | drawInteraction | PlotController.drawInteraction override. Draws the currently selected data range that that controller will zoom into when the user releases the mouse button. |
| | onMouseDown | Plot2DController.OnMouseDown override. Called when the user presses a mouse button. |
| | onMouseMove | PlotController.OnMouseMove override. Called when the user moves the mouse. |
| | onMouseUp | PlotController.OnMouseUp override. Called when the user releases a mouse button. |
| | onMouseWheel | PlotController.OnMouseWheel override. Called when the user releases a mouse button. |

## See Also

ZoomController Class
MindFusion.Charting Namespace

# ZoomController Constructor

Initializes a new instance of the ZoomController class.

**Namespace:** MindFusion.Charting
**File:** ZoomController.js

## Syntax

**JavaScript**

```
function ZoomController (renderContext, vertical)
```

### Parameters

*renderContext*
Type: RenderContext
A RenderContext instance.

*vertical*
Type: Boolean
true to implement interaction along the Y axis, or false for the X axis.

## See Also

ZoomController Class
MindFusion.Charting Namespace

# ZoomController Methods

## ZoomController.drawInteraction Method

PlotController.drawInteraction override. Draws the currently selected data range that that controller will zoom into when the user releases the mouse button.

**Namespace:** MindFusion.Charting
**File:** ZoomController.js

## Syntax

**JavaScript**

```
function drawInteraction (graphics)
```

### Parameters

*graphics*
Type: Graphics
A Graphics instance.

## See Also

ZoomController Members
ZoomController Class
MindFusion.Charting Namespace

## ZoomController.onMouseDown Method

Plot2DController.OnMouseDown override. Called when the user presses a mouse button.

**Namespace:** MindFusion.Charting
**File:** ZoomController.js

## Syntax

**JavaScript**

```
function onMouseDown (x, y)
```

### Parameters

*x*
Type: Number
A number value specifying the horizontal position of mouse pointer.

*y*
Type: Number
A number value specifying the vertical position of mouse pointer.

## See Also

ZoomController Members
ZoomController Class
MindFusion.Charting Namespace

# ZoomController.onMouseMove Method

PlotController.OnMouseMove override. Called when the user moves the mouse.

**Namespace:** MindFusion.Charting
**File:** ZoomController.js

## Syntax

**JavaScript**

```
function onMouseMove (x, y)
```

### Parameters

*x*
Type: Number
A number value specifying the horizontal position of mouse pointer.

*y*
Type: Number
A number value specifying the vertical position of mouse pointer.

## See Also

ZoomController Members
ZoomController Class
MindFusion.Charting Namespace

# ZoomController.onMouseUp Method

PlotController.OnMouseUp override. Called when the user releases a mouse button.

**Namespace:** MindFusion.Charting
**File:** ZoomController.js

## Syntax

**JavaScript**

```
function onMouseUp (x, y)
```

### Parameters

*x*
Type: Number
A number value specifying the horizontal position of mouse pointer.

*y*
Type: Number
A number value specifying the vertical position of mouse pointer.

## See Also

ZoomController Members
ZoomController Class
MindFusion.Charting Namespace

### ZoomController.onMouseWheel Method

PlotController.OnMouseWheel override. Called when the user releases a mouse button.

**Namespace:** MindFusion.Charting
**File:** ZoomController.js

## Syntax

**JavaScript**

```
function onMouseWheel (x, y, delta)
```

#### Parameters

*x*
Type: Number
A number value specifying the horizontal position of mouse pointer.

*y*
Type: Number
A number value specifying the vertical position of mouse pointer.

*delta*
Type: Number
A number value specifying the direction and the amount of scrolling needed.

## See Also

ZoomController Members
ZoomController Class
MindFusion.Charting Namespace

# MindFusion.Charting.Commands Namespace

### Command Class

Represents an action that modifies an object and whose effects can be undone.

## Syntax

**JavaScript**

```
// class
Command.prototype = {}
```

## Inheritance Hierarchy

## See Also

Command Members
MindFusion.Charting.Commands Namespace

# Command Members

Represents an action that modifies an object and whose effects can be undone.

The following tables list the members exposed by the Command type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | Command | Initializes a new instance of the Command class. |

## Public Methods

| | Name | Description |
|---|---|---|
| | commit | A shortcut method for executing the command through the undo engine. |

## See Also

Command Class
MindFusion.Charting.Commands Namespace

# Command Constructor

Initializes a new instance of the Command class.

**Namespace:** MindFusion.Charting.Commands
**File:** Command.js

## Syntax

**JavaScript**

```
function Command (manager)
```

### Parameters

*manager*
Type: UndoManager
The UndoManager that contains this command.

## See Also

Command Class
MindFusion.Charting.Commands Namespace

# Command Methods

## Command.commit Method

A shortcut method for executing the command through the undo engine.

**Namespace:** MindFusion.Charting.Commands
**File:** Command.js

### Syntax

```
JavaScript
function commit ()
```

### See Also

Command Members
Command Class
MindFusion.Charting.Commands Namespace

# CompositeCommand Class

Represents a set of Command instances as a single operation.Undoing or redoing the composite action, respectively undoes or redoes all its constituent subactions at the same time.

### Syntax

```
JavaScript
// class
CompositeCommand.prototype = {}
```

### Inheritance Hierarchy

### See Also

CompositeCommand Members
MindFusion.Charting.Commands Namespace

## CompositeCommand Members

Represents a set of Command instances as a single operation.Undoing or redoing the composite action, respectively undoes or redoes all its constituent subactions at the same time.

The following tables list the members exposed by the CompositeCommand type.

### Public Constructors

| | Name | Description |
|---|---|---|
|  | CompositeCommand | Initializes a new instance of the CompositeCommand class. |

### Public Methods

| | Name | Description |
|---|---|---|
| | execute | DisposableCommand.Execute override. |
| | redo | Command.Redo override. |
| | undo | Command.Undo override. |

## See Also

CompositeCommand Class
MindFusion.Charting.Commands Namespace

# CompositeCommand Constructor

Initializes a new instance of the CompositeCommand class.

**Namespace:** MindFusion.Charting.Commands
**File:** CompositeCommand.js

## Syntax

**JavaScript**

```
function CompositeCommand (manager)
```

### Parameters

*manager*
Type: UndoManager
The UndoManager that contains this command.

## See Also

CompositeCommand Class
MindFusion.Charting.Commands Namespace

# CompositeCommand Methods

# CompositeCommand.execute Method

DisposableCommand.Execute override.

**Namespace:** MindFusion.Charting.Commands
**File:** CompositeCommand.js

## Syntax

**JavaScript**

```
function execute ()
```

## See Also

CompositeCommand Members
CompositeCommand Class
MindFusion.Charting.Commands Namespace

## CompositeCommand.redo Method

Command.Redo override.

**Namespace:** MindFusion.Charting.Commands
**File:** CompositeCommand.js

### Syntax

**JavaScript**

```
function redo ()
```

### See Also

CompositeCommand Members
CompositeCommand Class
MindFusion.Charting.Commands Namespace

## CompositeCommand.undo Method

Command.Undo override.

**Namespace:** MindFusion.Charting.Commands
**File:** CompositeCommand.js

### Syntax

**JavaScript**

```
function undo ()
```

### See Also

CompositeCommand Members
CompositeCommand Class
MindFusion.Charting.Commands Namespace

# DisposableCommand Class

Represents a command that implements the IDisposable interface.

### Syntax

**JavaScript**

```
// class
DisposableCommand.prototype = {}
```

### Inheritance Hierarchy

## See Also

DisposableCommand Members
MindFusion.Charting.Commands Namespace

# DisposableCommand Members

Represents a command that implements the IDisposable interface.

The following tables list the members exposed by the DisposableCommand type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | DisposableCommand | Initializes a new instance of the DisposableCommand class. |

## Public Methods

| | Name | Description |
|---|---|---|
| | cancel | Cancels the command. |
| | clean | Disposes of this command. |
| | dispose | IDisposable.Dispose implementation. |
| | execute | Command.Execute override. |
| | redo | Command.Redo override. |
| | undo | Command.Undo override. |

## See Also

DisposableCommand Class
MindFusion.Charting.Commands Namespace

# DisposableCommand Constructor

Initializes a new instance of the DisposableCommand class.

**Namespace:** MindFusion.Charting.Commands
**File:** DisposableCommand.js

## Syntax

**JavaScript**

```
function DisposableCommand (manager)
```

### Parameters

*manager*
Type: UndoManager
The UndoManager that contains this command.

## See Also

DisposableCommand Class
MindFusion.Charting.Commands Namespace

# DisposableCommand Methods

## DisposableCommand.cancel Method

Cancels the command.

**Namespace:** MindFusion.Charting.Commands
**File:** DisposableCommand.js

## Syntax

**JavaScript**

```
function cancel ()
```

## See Also

DisposableCommand Members
DisposableCommand Class
MindFusion.Charting.Commands Namespace

## DisposableCommand.clean Method

Disposes of this command.

**Namespace:** MindFusion.Charting.Commands
**File:** DisposableCommand.js

## Syntax

**JavaScript**

```
function clean ()
```

## See Also

DisposableCommand Members
DisposableCommand Class
MindFusion.Charting.Commands Namespace

## DisposableCommand.dispose Method

IDisposable.Dispose implementation.

**Namespace:** MindFusion.Charting.Commands
**File:** DisposableCommand.js

## Syntax

```
JavaScript
function dispose ()
```

## See Also

DisposableCommand Members
DisposableCommand Class
MindFusion.Charting.Commands Namespace

# DisposableCommand.execute Method

Command.Execute override.

**Namespace:** MindFusion.Charting.Commands
**File:** DisposableCommand.js

## Syntax

```
JavaScript
function execute ()
```

## See Also

DisposableCommand Members
DisposableCommand Class
MindFusion.Charting.Commands Namespace

# DisposableCommand.redo Method

Command.Redo override.

**Namespace:** MindFusion.Charting.Commands
**File:** DisposableCommand.js

## Syntax

```
JavaScript
function redo ()
```

## See Also

DisposableCommand Members
DisposableCommand Class
MindFusion.Charting.Commands Namespace

# DisposableCommand.undo Method

Command.Undo override.

**Namespace:** MindFusion.Charting.Commands
**File:** DisposableCommand.js

## Syntax

```
JavaScript
function undo ()
```

## See Also

DisposableCommand Members
DisposableCommand Class
MindFusion.Charting.Commands Namespace

# TrackChangesCommand Class

Represents a command that can track and undo changes of INotifyPropertyValueChanged objects.

## Syntax

```
JavaScript
// class
TrackChangesCommand.prototype = {}
```

## Inheritance Hierarchy

## See Also

TrackChangesCommand Members
MindFusion.Charting.Commands Namespace

# TrackChangesCommand Members

Represents a command that can track and undo changes of INotifyPropertyValueChanged objects.

The following tables list the members exposed by the TrackChangesCommand type.

## Public Methods

| | Name | Description |
|---|---|---|
| | beginChange | Called before undoing or redoing the command. |
| | endChange | Called after undoing or redoing the command. |
| | redo | DisposableCommand.Redo override. |
| | undo | DisposableCommand.Undo override. |

## See Also

TrackChangesCommand Class
MindFusion.Charting.Commands Namespace

# TrackChangesCommand Methods

## TrackChangesCommand.beginChange Method

Called before undoing or redoing the command.

**Namespace:** MindFusion.Charting.Commands
**File:** TrackChangesCommand.js

### Syntax

```
JavaScript
function beginChange ()
```

### See Also

TrackChangesCommand Members
TrackChangesCommand Class
MindFusion.Charting.Commands Namespace

## TrackChangesCommand.endChange Method

Called after undoing or redoing the command.

**Namespace:** MindFusion.Charting.Commands
**File:** TrackChangesCommand.js

### Syntax

```
JavaScript
function endChange ()
```

### See Also

TrackChangesCommand Members
TrackChangesCommand Class
MindFusion.Charting.Commands Namespace

## TrackChangesCommand.redo Method

DisposableCommand.Redo override.

**Namespace:** MindFusion.Charting.Commands
**File:** TrackChangesCommand.js

### Syntax

```
JavaScript
function redo ()
```

### See Also

TrackChangesCommand Members
TrackChangesCommand Class
MindFusion.Charting.Commands Namespace

## TrackChangesCommand.undo Method

DisposableCommand.Undo override.

**Namespace:** MindFusion.Charting.Commands
**File:** TrackChangesCommand.js

### Syntax

**JavaScript**

```
function undo ()
```

### See Also

TrackChangesCommand Members
TrackChangesCommand Class
MindFusion.Charting.Commands Namespace

# UndoEventArgs Class

Contains the arguments passed to the ActionUndone and ActionRedone event handlers.

### Syntax

**JavaScript**

```
// class
UndoEventArgs.prototype = {}
```

### Inheritance Hierarchy

### See Also

UndoEventArgs Members
MindFusion.Charting.Commands Namespace

## UndoEventArgs Members

Contains the arguments passed to the ActionUndone and ActionRedone event handlers.

The following tables list the members exposed by the UndoEventArgs type.

### Public Constructors

| | Name | Description |
|---|---|---|
|  | UndoEventArgs | Initializes a new instance of the UndoEventArgs class. |

## See Also

UndoEventArgs Class
MindFusion.Charting.Commands Namespace

### UndoEventArgs Constructor

Initializes a new instance of the UndoEventArgs class.

**Namespace:** MindFusion.Charting.Commands
**File:** UndoEventArgs.js

## Syntax

**JavaScript**

```
function UndoEventArgs (command)
```

#### Parameters

*command*
Type: Command
The Command related to the event.

## See Also

UndoEventArgs Class
MindFusion.Charting.Commands Namespace

# MindFusion.Charting.Components Namespace

## BorderComponent Class

Represents a border UI component.

## Syntax

**JavaScript**

```
// class
BorderComponent.prototype = {}
```

## Inheritance Hierarchy

## See Also

BorderComponent Members
MindFusion.Charting.Components Namespace

### BorderComponent Members

Represents a border UI component.

The following tables list the members exposed by the BorderComponent type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | BorderComponent | Initializes a new instance of the BorderComponent class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | borderBrush | Gets or sets the Brush used to draw the border outlines. |
| | borderThickness | Gets or sets the border thickness. |
| | content | Gets or sets the Component displayed inside this border. |
| | padding | Gets or sets the padding space between this border and its Content. |

## Public Methods

| | Name | Description |
|---|---|---|
| | arrange | Component.Arrange override. Sets the location and size of this border and arranges its Content. |
| | draw | Component.Draw override. Draws this border in specified RenderContext. |
| | measure | Component.Measure override. Measures the desired size of this border and its Content. |

## See Also

BorderComponent Class
MindFusion.Charting.Components Namespace

# BorderComponent Constructor

Initializes a new instance of the BorderComponent class.

**Namespace:** MindFusion.Charting.Components
**File:** BorderComponent.js

## Syntax

| JavaScript |
|---|
| `function BorderComponent ()` |

## See Also

BorderComponent Class
MindFusion.Charting.Components Namespace

# BorderComponent Methods

## BorderComponent.arrange Method

Component.Arrange override. Sets the location and size of this border and arranges its Content.

**Namespace:** MindFusion.Charting.Components
**File:** BorderComponent.js

## Syntax

**JavaScript**

```
function arrange (x, y, w, h, context)
```

### Parameters

*x*
Type: Number
A number value specifying horizontal position.

*y*
Type: Number
A number value specifying vertical position.

*w*
Type: Number
A number value specifying the component's width.

*h*
Type: Number
A number value specifying the component's height.

*context*
Type: RenderContext
A RenderContext instance.

## See Also

BorderComponent Members
BorderComponent Class
MindFusion.Charting.Components Namespace

## BorderComponent.draw Method

Component.Draw override. Draws this border in specified RenderContext.

**Namespace:** MindFusion.Charting.Components
**File:** BorderComponent.js

## Syntax

**JavaScript**

```
function draw (context)
```

### Parameters

*context*

Type: RenderContext
A RenderContext instance.

## See Also

BorderComponent Members
BorderComponent Class
MindFusion.Charting.Components Namespace

# BorderComponent.measure Method

Component.Measure override. Measures the desired size of this border and its Content.

**Namespace:** MindFusion.Charting.Components
**File:** BorderComponent.js

## Syntax

**JavaScript**

```
function measure (maxWidth, maxHeight, context)
```

### Parameters

*maxWidth*
Type: Number
The maximum width provided by parent component.

*maxHeight*
Type: Number
The maximum height provided by parent component.

*context*
Type: RenderContext
A RenderContext instance.

## See Also

BorderComponent Members
BorderComponent Class
MindFusion.Charting.Components Namespace

# BorderComponent Properties

# BorderComponent.borderBrush Property

Gets or sets the Brush used to draw the border outlines.

**Namespace:** MindFusion.Charting.Components
**File:** BorderComponent.js

## Syntax

**JavaScript**

```
get borderBrush() {}
```

### Property Value

Type: Brush

## See Also

BorderComponent Members
BorderComponent Class
MindFusion.Charting.Components Namespace

# BorderComponent.borderThickness Property

Gets or sets the border thickness.

**Namespace:** MindFusion.Charting.Components
**File:** BorderComponent.js

## Syntax

**JavaScript**

```
get borderThickness() {}
```

### Property Value

Type: Number

## See Also

BorderComponent Members
BorderComponent Class
MindFusion.Charting.Components Namespace

# BorderComponent.content Property

Gets or sets the Component displayed inside this border.

**Namespace:** MindFusion.Charting.Components
**File:** BorderComponent.js

## Syntax

**JavaScript**

```
get content() {}
```

### Property Value

Type: Component

## See Also

BorderComponent Members
BorderComponent Class
MindFusion.Charting.Components Namespace

## BorderComponent.padding Property

Gets or sets the padding space between this border and its Content.

**Namespace:** MindFusion.Charting.Components
**File:** BorderComponent.js

### Syntax

| JavaScript |
| --- |
| `get padding() {}` |

#### Property Value

Type: Number

### See Also

BorderComponent Members
BorderComponent Class
MindFusion.Charting.Components Namespace

# ButtonComponent Class

Represents a button UI component.

### Syntax

| JavaScript |
| --- |
| `// class`<br>`ButtonComponent.prototype = {}` |

### Inheritance Hierarchy

### See Also

ButtonComponent Members
MindFusion.Charting.Components Namespace

## ButtonComponent Members

Represents a button UI component.

The following tables list the members exposed by the ButtonComponent type.

### Public Constructors

| | Name | Description |
| --- | --- | --- |
|  | ButtonComponent | Initializes a new instance of the ButtonComponent class. |

### Public Properties

| | Name | Description |
|---|---|---|
| | clicked | Raised when the user clicks on this button. |
| | content | Gets or sets the Component displayed inside this border. |
| | customDraw | Raised to let you custom-draw button graphics. |
| | padding | Gets or sets the padding space between this border and its Content. |

## Public Methods

| | Name | Description |
|---|---|---|
| | arrange | Component.Arrange override. Sets the location and size of this button relatively to its parent. |
| | createController | Component.CreateController override. Returns a controller used to interact with this button. |
| | draw | Component.Draw override. Draws this button in specified RenderContext. |
| | measure | Component.Measure override. Measures the desired size of this button. |
| | onClicked | Raises the Clicked event. |
| | onCustomDraw | Raises the CustomDraw event. |

## See Also

ButtonComponent Class
MindFusion.Charting.Components Namespace

# ButtonComponent Constructor

Initializes a new instance of the ButtonComponent class.

**Namespace:** MindFusion.Charting.Components
**File:** ButtonComponent.js

## Syntax

| JavaScript |
|---|
| `function ButtonComponent ()` |

## See Also

ButtonComponent Class
MindFusion.Charting.Components Namespace

# ButtonComponent Methods

# ButtonComponent.arrange Method

Component.Arrange override. Sets the location and size of this button relatively to its parent.

**Namespace:** MindFusion.Charting.Components
**File:** ButtonComponent.js

## Syntax

**JavaScript**

```
function arrange (x, y, w, h, context)
```

### Parameters

*x*
Type: Number
A number value specifying horizontal position.

*y*
Type: Number
A number value specifying vertical position.

*w*
Type: Number
A number value specifying the component's width.

*h*
Type: Number
A number value specifying the component's height.

*context*
Type: RenderContext
A RenderContext instance.

## See Also

ButtonComponent Members
ButtonComponent Class
MindFusion.Charting.Components Namespace

# ButtonComponent.createController Method

Component.CreateController override. Returns a controller used to interact with this button.

**Namespace:** MindFusion.Charting.Components
**File:** ButtonComponent.js

## Syntax

**JavaScript**

```
function createController (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: ComponentController
Instance of a ComponentController -derived class.

## See Also

ButtonComponent Members
ButtonComponent Class
MindFusion.Charting.Components Namespace

# ButtonComponent.draw Method

Component.Draw override. Draws this button in specified RenderContext.

**Namespace:** MindFusion.Charting.Components
**File:** ButtonComponent.js

## Syntax

**JavaScript**

```
function draw (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

ButtonComponent Members
ButtonComponent Class
MindFusion.Charting.Components Namespace

# ButtonComponent.measure Method

Component.Measure override. Measures the desired size of this button.

**Namespace:** MindFusion.Charting.Components
**File:** ButtonComponent.js

## Syntax

**JavaScript**

```
function measure (maxWidth, maxHeight, context)
```

### Parameters

*maxWidth*
Type: Number
The maximum width provided by parent component.

*maxHeight*

Type: Number
The maximum height provided by parent component.

*context*
Type: RenderContext
A RenderContext instance.

## See Also

ButtonComponent Members
ButtonComponent Class
MindFusion.Charting.Components Namespace

# ButtonComponent.onClicked Method

Raises the Clicked event.

**Namespace:** MindFusion.Charting.Components
**File:** ButtonComponent.js

## Syntax

**JavaScript**

```
function onClicked (e)
```

### Parameters

*e*
Type: EventArgs
An instance of the EventArgs class.

## See Also

ButtonComponent Members
ButtonComponent Class
MindFusion.Charting.Components Namespace

# ButtonComponent.onCustomDraw Method

Raises the CustomDraw event.

**Namespace:** MindFusion.Charting.Components
**File:** ButtonComponent.js

## Syntax

**JavaScript**

```
function onCustomDraw (e)
```

### Parameters

*e*
Type: ButtonDrawEventArgs
An instance of the ButtonDrawEventArgs class.

## See Also

ButtonComponent Members
ButtonComponent Class
MindFusion.Charting.Components Namespace

# ButtonComponent Properties

## ButtonComponent.clicked Property

Raised when the user clicks on this button.

**Namespace:** MindFusion.Charting.Components
**File:** ButtonComponent.js

## Syntax

```
JavaScript
get clicked() {}
```

### Property Value

Type: EventDispatcher

## See Also

ButtonComponent Members
ButtonComponent Class
MindFusion.Charting.Components Namespace

## ButtonComponent.content Property

Gets or sets the Component displayed inside this border.

**Namespace:** MindFusion.Charting.Components
**File:** ButtonComponent.js

## Syntax

```
JavaScript
get content() {}
```

### Property Value

Type: Component

## See Also

ButtonComponent Members
ButtonComponent Class
MindFusion.Charting.Components Namespace

## ButtonComponent.customDraw Property

Raised to let you custom-draw button graphics.

**Namespace:** MindFusion.Charting.Components
**File:** ButtonComponent.js

### Syntax

```
JavaScript
get customDraw() {}
```

#### Property Value

Type: ButtonDrawEventDispatcher

### See Also

ButtonComponent Members
ButtonComponent Class
MindFusion.Charting.Components Namespace

## ButtonComponent.padding Property

Gets or sets the padding space between this border and its Content.

**Namespace:** MindFusion.Charting.Components
**File:** ButtonComponent.js

### Syntax

```
JavaScript
get padding() {}
```

#### Property Value

Type: Number

### See Also

ButtonComponent Members
ButtonComponent Class
MindFusion.Charting.Components Namespace

# ButtonDrawEventArgs Class

Contains arguments passed to the CustomDraw event of buttons.

### Syntax

```
JavaScript
// class
ButtonDrawEventArgs.prototype = {}
```

## Inheritance Hierarchy

## See Also

ButtonDrawEventArgs Members
MindFusion.Charting.Components Namespace

# ButtonDrawEventArgs Members

Contains arguments passed to the CustomDraw event of buttons.

The following tables list the members exposed by the ButtonDrawEventArgs type.

## Public Properties

| | Name | Description |
|---|---|---|
|  | clipBounds | Gets the current clip rectangle. |
|  | graphics | Gets the Graphics surface where the event handler should draw. |

## See Also

ButtonDrawEventArgs Class
MindFusion.Charting.Components Namespace

# ButtonDrawEventArgs Properties

# ButtonDrawEventArgs.clipBounds Property

Gets the current clip rectangle.

**Namespace:** MindFusion.Charting.Components
**File:** ButtonDrawEventArgs.js

## Syntax

**JavaScript**

```
get clipBounds() {}
```

### Property Value

Type: Rect

## See Also

ButtonDrawEventArgs Members
ButtonDrawEventArgs Class
MindFusion.Charting.Components Namespace

## ButtonDrawEventArgs.graphics Property

Gets the Graphics surface where the event handler should draw.

**Namespace:** MindFusion.Charting.Components
**File:** ButtonDrawEventArgs.js

### Syntax

```
JavaScript
get graphics() {}
```

#### Property Value

Type: Graphics

### See Also

ButtonDrawEventArgs Members
ButtonDrawEventArgs Class
MindFusion.Charting.Components Namespace

# ButtonDrawEventDispatcher Class

Represents a dispatcher for ButtonDraw events.

### Syntax

```
JavaScript
// class
ButtonDrawEventDispatcher.prototype = {}
```

### Inheritance Hierarchy

### See Also

ButtonDrawEventDispatcher Members
MindFusion.Charting.Components Namespace

## ButtonDrawEventDispatcher Members

Represents a dispatcher for ButtonDraw events.

The following tables list the members exposed by the ButtonDrawEventDispatcher type.

### See Also

ButtonDrawEventDispatcher Class
MindFusion.Charting.Components Namespace

# Component Class

Represents a user interface component.

## Syntax

| JavaScript |
| --- |
| `// class`<br>`Component.prototype = {}` |

## Inheritance Hierarchy

## See Also

Component Members
MindFusion.Charting.Components Namespace

# Component Members

Represents a user interface component.

The following tables list the members exposed by the Component type.

## Public Constructors

| | Name | Description |
| --- | --- | --- |
| | Component | Initializes a new instance of the Component class. |

## Public Properties

| | Name | Description |
| --- | --- | --- |
| | actualHeight | Gets a number value specifying the component's assigned height. Valid only after calling arrange. |
| | actualWidth | Gets a number value specifying the component's assigned width. Valid only after calling arrange. |
| | desiredHeight | Gets a nullable number value specifying the component's desired height. Valid only after calling measure. |
| | desiredWidth | Gets a nullable number value specifying the component's desired width. Valid only after calling measure. |
| | gridColumn | Gets or sets the column index of this component when placed inside a GridPanel. |
| | gridRow | Gets or sets the row index of this component when placed inside a GridPanel. |
| | height | Gets or sets a fixed height for this component. |
| | hitTestVisibility | Gets or sets the hit-test visibility of this component. |
| | horizontalAlignment | Gets or sets the horizontal alignment of this component inside the layout rectangle allocated by its parent panel. |

| | | |
|---|---|---|
| | margin | Gets the margin space around this component relative to the layout rectangle allocated by its parent panel. |
| | rectInParent | Gets the boundaries of this component relative to its parent. |
| | toolTip | Gets or sets the component's tooltip text. |
| | verticalAlignment | Gets or sets the vertical alignment of this component inside the layout rectangle allocated by its parent panel. |
| | visibility | Gets or sets the visibility of this component. |
| | width | Gets or sets a fixed width for this component. |
| | xInParent | Gets a number value specifying the component's horizontal position relative to its parent. |
| | yInParent | Gets a number value specifying the component's vertical position relative to its parent. |

## Public Methods

| | Name | Description |
|---|---|---|
| | add | Sums specified nullable number values. |
| | arrange | Sets the location and size of this component relatively to its parent. |
| | arrangeInRect | Arranges a child component in specified layout rectangle. |
| | createController | Returns a ComponentController used to interact with this component. |
| | draw | Draws this component in specified RenderContext. |
| | getCursorHint | Invoked while the mouse is moved to let your application set the mouse cursor. |
| | hitTest | Returns the component containing specified point. |
| | invalidate | Invalidates the appearance of this component and calls its Draw method at next draw operation. |
| | invalidateLayout | Invalidates the current layout and runs a new layout pass before next draw operation. |
| | localToRoot | Transforms the specified point to the coordinate system of the root panel. |
| | max | Returns the larger of specified values. |
| | measure | Measures the desired size of this component. |
| | rootToLocal | Transforms the specified point to the coordinate system of this component. |
| | visit | Implements the visitor design pattern. |

## See Also

Component Class
MindFusion.Charting.Components Namespace

## Component Constructor

Initializes a new instance of the Component class.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

### Syntax

```
JavaScript
function Component ()
```

### See Also

Component Class
MindFusion.Charting.Components Namespace

## Component Methods

## Component.add Method

Sums specified nullable number values.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

### Syntax

```
JavaScript
function add (value1, value2)
```

#### Parameters

*value1*
Type: Number
The first term to sum.

*value2*
Type: Number
The second term to sum.

#### Return Value

Type: Number
A nullable number value representing the sum.

### See Also

Component Members
Component Class
MindFusion.Charting.Components Namespace

# Component.arrange Method

Sets the location and size of this component relatively to its parent.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

## Syntax

**JavaScript**

```
function arrange (x, y, w, h, context)
```

### Parameters

*x*
Type: Number
A number value specifying horizontal position.

*y*
Type: Number
A number value specifying vertical position.

*w*
Type: Number
A number value specifying the component's width.

*h*
Type: Number
A number value specifying the component's height.

*context*
Type: RenderContext
A RenderContext instance.

## See Also

Component Members
Component Class
MindFusion.Charting.Components Namespace

# Component.arrangeInRect Method

Arranges a child component in specified layout rectangle.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

## Syntax

**JavaScript**

```
function arrangeInRect (child, context, x, y, w, h)
```

### Parameters

*child*
Type: Component
A Component instance specifying the child to arrange.

*context*

Type: RenderContext
A RenderContext instance.

*x*
Type: Number
Layout rectangle's horizontal position.

*y*
Type: Number
Layout rectangle's vertical position.

*w*
Type: Number
Layout rectangle's width.

*h*
Type: Number
Layout rectangle's height.

## See Also

Component Members
Component Class
MindFusion.Charting.Components Namespace

# Component.createController Method

Returns a ComponentController used to interact with this component.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

## Syntax

**JavaScript**

```
function createController (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: ComponentController
Instance of a ComponentController -derived class.

## See Also

Component Members
Component Class
MindFusion.Charting.Components Namespace

# Component.draw Method

Draws this component in specified RenderContext.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

## Syntax

```
JavaScript
function draw (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

Component Members
Component Class
MindFusion.Charting.Components Namespace

# Component.getCursorHint Method

Invoked while the mouse is moved to let your application set the mouse cursor.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

## Syntax

```
JavaScript
function getCursorHint (x, y)
```

### Parameters

*x*
Type: Number
A double value specifying the horizontal position of mouse pointer.

*y*
Type: Number
A double value specifying the vertical position of mouse pointer.

### Return Value

Type: CursorHint
A member of the CursorHint enumeration.

## See Also

Component Members
Component Class
MindFusion.Charting.Components Namespace

# Component.hitTest Method

Returns the component containing specified point.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

## Syntax

```
JavaScript
function hitTest (x, y)
```

### Parameters

*x*
Type: Number
X coordinate of the point to test.

*y*
Type: Number
Y coordinate of the point to test.

### Return Value

Type: Component
A Component instance if one contains the point, or null otherwise.

## See Also

Component Members
Component Class
MindFusion.Charting.Components Namespace

# Component.invalidate Method

Invalidates the appearance of this component and calls its Draw method at next draw operation.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

## Syntax

```
JavaScript
function invalidate ([rect])
```

### Parameters

*rect*
Optional.
Type: Rect
A RectD instance specifying the invalid rectangle.

## See Also

Component Members
Component Class
MindFusion.Charting.Components Namespace

## Component.invalidateLayout Method

Invalidates the current layout and runs a new layout pass before next draw operation.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

### Syntax

**JavaScript**

```
function invalidateLayout ()
```

### See Also

Component Members
Component Class
MindFusion.Charting.Components Namespace

## Component.localToRoot Method

Transforms the specified point to the coordinate system of the root panel.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

### Syntax

**JavaScript**

```
function localToRoot (point)
```

#### Parameters

*point*
Type: Point
A Point instance containing coordinates local to this component.

#### Return Value

Type: Point
A Point instance containing coordinates relative to the root panel.

### See Also

Component Members
Component Class
MindFusion.Charting.Components Namespace

## Component.max Method

Returns the larger of specified values.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

## Syntax

```
JavaScript
function max (value1, value2)
```

### Parameters

*value1*
Type: Number
The first value to compare.

*value2*
Type: Number
The second value to compare.

### Return Value

Type: Number
A nullable number value representing the larger value.

## See Also

Component Members
Component Class
MindFusion.Charting.Components Namespace

# Component.measure Method

Measures the desired size of this component.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

## Syntax

```
JavaScript
function measure (maxWidth, maxHeight, context)
```

### Parameters

*maxWidth*
Type: Number
The maximum width provided by parent component.

*maxHeight*
Type: Number
The maximum height provided by parent component.

*context*
Type: RenderContext
A RenderContext instance.

## See Also

Component Members
Component Class
MindFusion.Charting.Components Namespace

# Component.rootToLocal Method

Transforms the specified point to the coordinate system of this component.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

## Syntax

```
JavaScript
function rootToLocal (point)
```

### Parameters

*point*
Type: Point
A Point instance containing coordinates relative to the root panel.

### Return Value

Type: Point
A Point instance containing coordinates local to this component.

## See Also

Component Members
Component Class
MindFusion.Charting.Components Namespace

# Component.visit Method

Implements the visitor design pattern.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

## Syntax

```
JavaScript
function visit (visitor)
```

### Parameters

*visitor*
Type: ComponentVisitor
An instance of a ComponentVisitor -derived class.

## See Also

Component Members
Component Class

MindFusion.Charting.Components Namespace

## Component Properties

## Component.actualHeight Property

Gets a number value specifying the component's assigned height. Valid only after calling arrange.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

### Syntax

```
JavaScript
get actualHeight() {}
```

#### Property Value

Type: Number

### See Also

Component Members
Component Class
MindFusion.Charting.Components Namespace

## Component.actualWidth Property

Gets a number value specifying the component's assigned width. Valid only after calling arrange.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

### Syntax

```
JavaScript
get actualWidth() {}
```

#### Property Value

Type: Number

### See Also

Component Members
Component Class
MindFusion.Charting.Components Namespace

## Component.desiredHeight Property

Gets a nullable number value specifying the component's desired height. Valid only after calling measure.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

## Syntax

```
JavaScript
get desiredHeight() {}
```

### Property Value

Type: Number

## See Also

Component Members
Component Class
MindFusion.Charting.Components Namespace

# Component.desiredWidth Property

Gets a nullable number value specifying the component's desired width. Valid only after calling measure.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

## Syntax

```
JavaScript
get desiredWidth() {}
```

### Property Value

Type: Number

## See Also

Component Members
Component Class
MindFusion.Charting.Components Namespace

# Component.gridColumn Property

Gets or sets the column index of this component when placed inside a GridPanel.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

## Syntax

```
JavaScript
get gridColumn() {}
```

### Property Value

Type: Number

## See Also

Component Members
Component Class
MindFusion.Charting.Components Namespace

# Component.gridRow Property

Gets or sets the row index of this component when placed inside a GridPanel.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

## Syntax

**JavaScript**

```
get gridRow() {}
```

### Property Value

Type: Number

## See Also

Component Members
Component Class
MindFusion.Charting.Components Namespace

# Component.height Property

Gets or sets a fixed height for this component.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

## Syntax

**JavaScript**

```
get height() {}
```

### Property Value

Type: Number

## See Also

Component Members
Component Class
MindFusion.Charting.Components Namespace

## Component.hitTestVisibility Property

Gets or sets the hit-test visibility of this component.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

### Syntax

```
JavaScript
get hitTestVisibility() {}
```

#### Property Value

Type: HitTestVisibility

### See Also

Component Members
Component Class
MindFusion.Charting.Components Namespace

## Component.horizontalAlignment Property

Gets or sets the horizontal alignment of this component inside the layout rectangle allocated by its parent panel.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

### Syntax

```
JavaScript
get horizontalAlignment() {}
```

#### Property Value

Type: LayoutAlignment

### See Also

Component Members
Component Class
MindFusion.Charting.Components Namespace

## Component.margin Property

Gets the margin space around this component relative to the layout rectangle allocated by its parent panel.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

### Syntax

```
JavaScript
```

```
get margin() {}
```

### Property Value

Type: Margins

## See Also

Component Members
Component Class
MindFusion.Charting.Components Namespace

# Component.rectInParent Property

Gets the boundaries of this component relative to its parent.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

## Syntax

**JavaScript**

```
get rectInParent() {}
```

### Property Value

Type: Number

## See Also

Component Members
Component Class
MindFusion.Charting.Components Namespace

# Component.toolTip Property

Gets or sets the component's tooltip text.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

## Syntax

**JavaScript**

```
get toolTip() {}
```

### Property Value

Type: String

## See Also

Component Members
Component Class

MindFusion.Charting.Components Namespace

## Component.verticalAlignment Property

Gets or sets the vertical alignment of this component inside the layout rectangle allocated by its parent panel.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

### Syntax

**JavaScript**

```
get verticalAlignment() {}
```

#### Property Value

Type: LayoutAlignment

### See Also

Component Members
Component Class
MindFusion.Charting.Components Namespace

## Component.visibility Property

Gets or sets the visibility of this component.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

### Syntax

**JavaScript**

```
get visibility() {}
```

#### Property Value

Type: Visibility

### See Also

Component Members
Component Class
MindFusion.Charting.Components Namespace

## Component.width Property

Gets or sets a fixed width for this component.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

## Syntax

```
JavaScript
get width() {}
```

### Property Value

Type: Number

## See Also

Component Members
Component Class
MindFusion.Charting.Components Namespace

# Component.xInParent Property

Gets a number value specifying the component's horizontal position relative to its parent.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

## Syntax

```
JavaScript
get xInParent() {}
```

### Property Value

Type: Number

## See Also

Component Members
Component Class
MindFusion.Charting.Components Namespace

# Component.yInParent Property

Gets a number value specifying the component's vertical position relative to its parent.

**Namespace:** MindFusion.Charting.Components
**File:** Component.js

## Syntax

```
JavaScript
get yInParent() {}
```

### Property Value

Type: Number

## See Also

Component Members
Component Class
MindFusion.Charting.Components Namespace

# ComponentAnimation Interface

Represents an animation.

## Syntax

**JavaScript**

```
// interface
ComponentAnimation.prototype = {}
```

## See Also

ComponentAnimation Members
MindFusion.Charting.Components Namespace

### ComponentAnimation Members

Represents an animation.

The following tables list the members exposed by the ComponentAnimation type.

## Public Methods

| | Name | Description |
|---|---|---|
|  | stop | Stops this animation. |

## See Also

ComponentAnimation Interface
MindFusion.Charting.Components Namespace

### ComponentAnimation Methods

### ComponentAnimation.stop Method

Stops this animation.

**Namespace:** MindFusion.Charting.Components
**File:** ComponentAnimation.js

## Syntax

**JavaScript**

```
function stop ()
```

## See Also

ComponentAnimation Members
ComponentAnimation Interface
MindFusion.Charting.Components Namespace

# ComponentController Interface

Defines the interface that controller classes should implement to get user input from Chart controls.

## Syntax

**JavaScript**

```
// interface
ComponentController.prototype = {}
```

## See Also

ComponentController Members
MindFusion.Charting.Components Namespace

# ComponentController Members

Defines the interface that controller classes should implement to get user input from Chart controls.

The following tables list the members exposed by the ComponentController type.

## Public Properties

| | Name | Description |
|---|---|---|
|  | component | Gets the component modified by this ComponentController. |

## Public Methods

| | Name | Description |
|---|---|---|
|  | drawInteraction | Draws a representation of the current state of user interaction on specified IGraphics surface. |
|  | getRunningAnimation | For internal use. |
|  | onMouseDown | Called when the user presses a mouse button. |
|  | onMouseMove | Called when the user moves the mouse. |
|  | onMouseUp | Called when the user releases a mouse button. |
|  | onMouseWheel | Called when the user scrolls with the middle mouse button. |

## See Also

# ComponentController Methods

## ComponentController.drawInteraction Method

Draws a representation of the current state of user interaction on specified IGraphics surface.

**Namespace:** MindFusion.Charting.Components
**File:** ComponentController.js

### Syntax

**JavaScript**

```
function drawInteraction (graphics)
```

#### Parameters

*graphics*
Type: Graphics
A Graphics instance.

### See Also

ComponentController Members
ComponentController Interface
MindFusion.Charting.Components Namespace

## ComponentController.getRunningAnimation Method

For internal use.

**Namespace:** MindFusion.Charting.Components
**File:** ComponentController.js

### Syntax

**JavaScript**

```
function getRunningAnimation ()
```

#### Return Value

Type: ComponentAnimation
An instance of a ComponentAnimation -derived class.

### See Also

ComponentController Members
ComponentController Interface
MindFusion.Charting.Components Namespace

# ComponentController.onMouseDown Method

Called when the user presses a mouse button.

**Namespace:** MindFusion.Charting.Components
**File:** ComponentController.js

## Syntax

**JavaScript**

```
function onMouseDown (x, y)
```

### Parameters

*x*
Type: Number
A number value specifying the horizontal position of mouse pointer.

*y*
Type: Number
A number value specifying the vertical position of mouse pointer.

## See Also

ComponentController Members
ComponentController Interface
MindFusion.Charting.Components Namespace

# ComponentController.onMouseMove Method

Called when the user moves the mouse.

**Namespace:** MindFusion.Charting.Components
**File:** ComponentController.js

## Syntax

**JavaScript**

```
function onMouseMove (x, y)
```

### Parameters

*x*
Type: Number
A number value specifying the horizontal position of mouse pointer.

*y*
Type: Number
A number value specifying the vertical position of mouse pointer.

## See Also

ComponentController Members
ComponentController Interface
MindFusion.Charting.Components Namespace

# ComponentController.onMouseUp Method

Called when the user releases a mouse button.

**Namespace:** MindFusion.Charting.Components
**File:** ComponentController.js

## Syntax

**JavaScript**

```
function onMouseUp (x, y)
```

### Parameters

*x*
Type: Number
A number value specifying the horizontal position of mouse pointer.

*y*
Type: Number
A number value specifying the vertical position of mouse pointer.

## See Also

ComponentController Members
ComponentController Interface
MindFusion.Charting.Components Namespace

# ComponentController.onMouseWheel Method

Called when the user scrolls with the middle mouse button.

**Namespace:** MindFusion.Charting.Components
**File:** ComponentController.js

## Syntax

**JavaScript**

```
function onMouseWheel (x, y, delta)
```

### Parameters

*x*
Type: Number
A number value specifying the horizontal position of mouse pointer.

*y*
Type: Number
A number value specifying the vertical position of mouse pointer.

*delta*
Type: Number
A number value specifying the amount and the direction for the mouse scroll.

## See Also

ComponentController Members
ComponentController Interface

MindFusion.Charting.Components Namespace

## ComponentController Properties

## ComponentController.component Property

Gets the component modified by this ComponentController.

**Namespace:** MindFusion.Charting.Components
**File:** ComponentController.js

### Syntax

```
JavaScript
get component() {}
```

#### Property Value

Type: Component

### See Also

ComponentController Members
ComponentController Interface
MindFusion.Charting.Components Namespace

# ComponentVisitor Class

Defines the base visitor class for hierarchy of Component objects.

### Syntax

```
JavaScript
// class
ComponentVisitor.prototype = {}
```

### Inheritance Hierarchy

### See Also

ComponentVisitor Members
MindFusion.Charting.Components Namespace

## ComponentVisitor Members

Defines the base visitor class for hierarchy of Component objects.

The following tables list the members exposed by the ComponentVisitor type.

### Public Methods

| | Name | Description |
|---|---|---|
| | visitPanel | Visits a Panel component. |
| | visitPlot | Visits a Plot component. |

## See Also

ComponentVisitor Class
MindFusion.Charting.Components Namespace

# ComponentVisitor Methods

# ComponentVisitor.visitPanel Method

Visits a Panel component.

**Namespace:** MindFusion.Charting.Components
**File:** ComponentVisitor.js

## Syntax

**JavaScript**

```
function visitPanel (panel)
```

### Parameters

*panel*
Type: void
A Panel instance.

## See Also

ComponentVisitor Members
ComponentVisitor Class
MindFusion.Charting.Components Namespace

# ComponentVisitor.visitPlot Method

Visits a Plot component.

**Namespace:** MindFusion.Charting.Components
**File:** ComponentVisitor.js

## Syntax

**JavaScript**

```
function visitPlot (plot)
```

### Parameters

*plot*
Type: Plot
A Plot instance.

## See Also

ComponentVisitor Members
ComponentVisitor Class
MindFusion.Charting.Components Namespace

# CursorHint Enumeration

Specifies what mouse cursor to display while a user interacts with the control.

**Namespace:** MindFusion.Charting.Components
**File:** CursorHint.js

## Syntax

**JavaScript**

```
// enum
CursorHint = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **CounterDiagonalResize** | Indicates the cursor specified by the CounterDiagonalResize property. |
| | **DiagonalResize** | Indicates the cursor specified by the DiagonalResize property. |
| | **Disallow** | Indicates the cursor specified by the DisallowCursor property. |
| | **DontChange** | Indicates the cursor specified by the Cursor property. |
| | **HorizontalResize** | Indicates the cursor specified by the HorizontalResizeCursor property. |
| | **Move** | Indicates the cursor specified by the MoveCursor property. |
| | **Pointer** | Indicates the cursor specified by the PointerCursor property. |
| | **Rotate** | Indicates the cursor specified by the RotateCursor property. |
| | **VerticalResize** | Indicates the cursor specified by the VerticalResize property. |

## See Also

MindFusion.Charting.Components Namespace

# GridColumn Class

Represents a column in a GridPanel.

## Syntax

JavaScript

```
// class
GridColumn.prototype = {}
```

## Inheritance Hierarchy

## See Also

GridColumn Members
MindFusion.Charting.Components Namespace

# GridColumn Members

Represents a column in a GridPanel.

The following tables list the members exposed by the GridColumn type.

## Public Properties

| | Name | Description |
|---|---|---|
|  | lengthType | Gets or sets the column's sizing mode. |
|  | width | Gets or sets fixed width for this column. |

## Public Methods

| | Name | Description |
|---|---|---|
|  | measuredSize | Returns the column's measured width. |
|  | relativeSize | Gets whether the column should be sized relatively to other columns in the grid panel. |
|  | setPos | Sets the column's position. |

## See Also

GridColumn Class
MindFusion.Charting.Components Namespace

# GridColumn Methods

## GridColumn.measuredSize Method

Returns the column's measured width.

**Namespace:** MindFusion.Charting.Components
**File:** GridColumn.js

## Syntax

JavaScript

```
function measuredSize ()
```

### Return Value

Type: Number
A number value representing the column's width.

## See Also

GridColumn Members
GridColumn Class
MindFusion.Charting.Components Namespace

# GridColumn.relativeSize Method

Gets whether the column should be sized relatively to other columns in the grid panel.

**Namespace:** MindFusion.Charting.Components
**File:** GridColumn.js

## Syntax

**JavaScript**

```
function relativeSize ()
```

### Return Value

Type: Boolean
true to apply relative size to this column, or false otherwise.

## See Also

GridColumn Members
GridColumn Class
MindFusion.Charting.Components Namespace

# GridColumn.setPos Method

Sets the column's position.

**Namespace:** MindFusion.Charting.Components
**File:** GridColumn.js

## Syntax

**JavaScript**

```
function setPos (value)
```

### Parameters

*value*
Type: Number
X coordinate of the column.

## See Also

GridColumn Members
GridColumn Class
MindFusion.Charting.Components Namespace

# GridColumn Properties

## GridColumn.lengthType Property

Gets or sets the column's sizing mode.

**Namespace:** MindFusion.Charting.Components
**File:** GridColumn.js

## Syntax

```
JavaScript
get lengthType() {}
```

### Property Value

Type: LengthType

## See Also

GridColumn Members
GridColumn Class
MindFusion.Charting.Components Namespace

## GridColumn.width Property

Gets or sets fixed width for this column.

**Namespace:** MindFusion.Charting.Components
**File:** GridColumn.js

## Syntax

```
JavaScript
get width() {}
```

### Property Value

Type: Number

## See Also

GridColumn Members
GridColumn Class
MindFusion.Charting.Components Namespace

# GridPanel Class

Represents a layout panel that arranges its child components in a grid.

## Syntax

**JavaScript**

```
// class
GridPanel.prototype = {}
```

## Inheritance Hierarchy

## See Also

GridPanel Members
MindFusion.Charting.Components Namespace

# GridPanel Members

Represents a layout panel that arranges its child components in a grid.

The following tables list the members exposed by the GridPanel type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | GridPanel | Initializes a new instance of the GridPanel class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | columns | Gets or sets a list of GridColumn objects specifying column attributes. |
| | rows | Gets or sets a list of GridRow objects specifying row attributes. |

## Public Methods

| | Name | Description |
|---|---|---|
| | arrange | Component.Arrange override. Sets the location and size of this panel and arranges its child components. |
| | measure | Panel.Measure override. Measures the desired size of this panel and its child components. |

## See Also

GridPanel Class
MindFusion.Charting.Components Namespace

# GridPanel Constructor

Initializes a new instance of the GridPanel class.

**Namespace:** MindFusion.Charting.Components
**File:** GridPanel.js

## Syntax

**JavaScript**

```
function GridPanel ()
```

## See Also

GridPanel Class
MindFusion.Charting.Components Namespace

# GridPanel Methods

# GridPanel.arrange Method

Component.Arrange override. Sets the location and size of this panel and arranges its child components.

**Namespace:** MindFusion.Charting.Components
**File:** GridPanel.js

## Syntax

**JavaScript**

```
function arrange (x, y, w, h, context)
```

### Parameters

*x*
Type: void
A number value specifying horizontal position.

*y*
Type: void
A number value specifying vertical position.

*w*
Type: void
A number value specifying the component's width.

*h*
Type: void
A number value specifying the component's height.

*context*
Type: void
A RenderContext instance.

## See Also

GridPanel Members
GridPanel Class

# GridPanel.measure Method

Panel.Measure override. Measures the desired size of this panel and its child components.

**Namespace:** MindFusion.Charting.Components
**File:** GridPanel.js

## Syntax

**JavaScript**

```
function measure (maxWidth, maxHeight, context)
```

### Parameters

*maxWidth*
Type: void
The maximum width provided by parent component.

*maxHeight*
Type: void
The maximum height provided by parent component.

*context*
Type: void
A RenderContext instance.

## See Also

GridPanel Members
GridPanel Class
MindFusion.Charting.Components Namespace

# GridPanel Properties

# GridPanel.columns Property

Gets or sets a list of GridColumn objects specifying column attributes.

**Namespace:** MindFusion.Charting.Components
**File:** GridPanel.js

## Syntax

**JavaScript**

```
get columns() {}
```

### Property Value

Type: List

## See Also

GridPanel Members
GridPanel Class
MindFusion.Charting.Components Namespace

## GridPanel.rows Property

Gets or sets a list of GridRow objects specifying row attributes.

**Namespace:** MindFusion.Charting.Components
**File:** GridPanel.js

### Syntax

```
JavaScript
get rows() {}
```

#### Property Value

Type: List

### See Also

GridPanel Members
GridPanel Class
MindFusion.Charting.Components Namespace

# GridRow Class

Represents a row in a GridPanel.

### Syntax

```
JavaScript
// class
GridRow.prototype = {}
```

### Inheritance Hierarchy

### See Also

GridRow Members
MindFusion.Charting.Components Namespace

## GridRow Members

Represents a row in a GridPanel.

The following tables list the members exposed by the GridRow type.

### Public Properties

| | Name | Description |
|---|---|---|

| | | |
|---|---|---|
| | height | Gets or sets fixed height for this row. |
| | lengthType | Gets or sets the row's sizing mode. |

## Public Methods

| | Name | Description |
|---|---|---|
| | measuredSize | Returns the row's measured height. |
| | relativeSize | Gets whether the row should be sized relatively to other rows in the grid panel. |
| | setPos | Sets the row's position. |

## See Also

GridRow Class
MindFusion.Charting.Components Namespace

# GridRow Methods

## GridRow.measuredSize Method

Returns the row's measured height.

**Namespace:** MindFusion.Charting.Components
**File:** GridRow.js

## Syntax

**JavaScript**

```
function measuredSize ()
```

## See Also

GridRow Members
GridRow Class
MindFusion.Charting.Components Namespace

## GridRow.relativeSize Method

Gets whether the row should be sized relatively to other rows in the grid panel.

**Namespace:** MindFusion.Charting.Components
**File:** GridRow.js

## Syntax

**JavaScript**

```
function relativeSize ()
```

## See Also

GridRow Members
GridRow Class
MindFusion.Charting.Components Namespace

# GridRow.setPos Method

Sets the row's position.

**Namespace:** MindFusion.Charting.Components
**File:** GridRow.js

## Syntax

**JavaScript**

```
function setPos (value)
```

### Parameters

*value*
Type: void
Y coordinate of the row.

## See Also

GridRow Members
GridRow Class
MindFusion.Charting.Components Namespace

# GridRow Properties

# GridRow.height Property

Gets or sets fixed height for this row.

**Namespace:** MindFusion.Charting.Components
**File:** GridRow.js

## Syntax

**JavaScript**

```
get height() {}
```

### Property Value

Type: Number

## See Also

GridRow Members
GridRow Class
MindFusion.Charting.Components Namespace

## GridRow.lengthType Property

Gets or sets the row's sizing mode.

**Namespace:** MindFusion.Charting.Components
**File:** GridRow.js

### Syntax

**JavaScript**

```
get lengthType() {}
```

#### Property Value

Type: LengthType

### See Also

GridRow Members
GridRow Class
MindFusion.Charting.Components Namespace

# HitTestVisibility Enumeration

Identifies hit-test visibility of a component.

**Namespace:** MindFusion.Charting.Components
**File:** HitTestVisibility.js

### Syntax

**JavaScript**

```
// enum
HitTestVisibility = {}
```

### Members

| | Member name | Description |
|---|---|---|
| | **Children** | Only hit-test child components. |
| | **HitTestVisibility** | Do not participate in hit-testing. |
| | **SelfAndChildren** | Hit-test this component and its children. |

### See Also

MindFusion.Charting.Components Namespace

# ImageComponent Class

Represents a component that draws a bitmap image.

## Syntax

| JavaScript |
| --- |
| `// class`<br>`ImageComponent.prototype = {}` |

## Inheritance Hierarchy

## See Also

ImageComponent Members
MindFusion.Charting.Components Namespace

# ImageComponent Members

Represents a component that draws a bitmap image.

The following tables list the members exposed by the ImageComponent type.

## Public Constructors

| | Name | Description |
| --- | --- | --- |
|  | ImageComponent | Initializes a new instance of the ImageComponent class. |

## Public Properties

| | Name | Description |
| --- | --- | --- |
|  | autoSize | Gets or sets a value indicating whether the component should auto-size to match the dimensions of its Image. |
|  | imageAlign | Gets or sets the image alignment relative to the component. |
|  | location | Gets or sets Image by its file location. |

## Public Methods

| | Name | Description |
| --- | --- | --- |
|  | draw | Component.Draw override. Draws associated Image in specified RenderContext. |
|  | measure | Component.Measure override. Measures the desired size of this component. |

## See Also

ImageComponent Class
MindFusion.Charting.Components Namespace

# ImageComponent Constructor

Initializes a new instance of the ImageComponent class.

**Namespace:** MindFusion.Charting.Components
**File:** ImageComponent.js

## Syntax

```
JavaScript
function ImageComponent ()
```

## See Also

ImageComponent Class
MindFusion.Charting.Components Namespace

# ImageComponent Methods

# ImageComponent.draw Method

Component.Draw override. Draws associated Image in specified RenderContext.

**Namespace:** MindFusion.Charting.Components
**File:** ImageComponent.js

## Syntax

```
JavaScript
function draw (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

ImageComponent Members
ImageComponent Class
MindFusion.Charting.Components Namespace

# ImageComponent.measure Method

Component.Measure override. Measures the desired size of this component.

**Namespace:** MindFusion.Charting.Components
**File:** ImageComponent.js

## Syntax

```
JavaScript
function measure (maxWidth, maxHeight, context)
```

### Parameters

*maxWidth*
Type: Number
The maximum width provided by parent component.

*maxHeight*
Type: Number
The maximum height provided by parent component.

*context*
Type: RenderContext
A RenderContext instance.

## See Also

ImageComponent Members
ImageComponent Class
MindFusion.Charting.Components Namespace

# ImageComponent Properties

# ImageComponent.autoSize Property

Gets or sets a value indicating whether the component should auto-size to match the dimensions of its Image.

**Namespace:** MindFusion.Charting.Components
**File:** ImageComponent.js

## Syntax

```
JavaScript
get autoSize() {}
```

### Property Value

Type: Boolean

## See Also

ImageComponent Members
ImageComponent Class
MindFusion.Charting.Components Namespace

# ImageComponent.imageAlign Property

Gets or sets the image alignment relative to the component.

**Namespace:** MindFusion.Charting.Components
**File:** ImageComponent.js

## Syntax

```
JavaScript
get imageAlign() {}
```

### Property Value

Type: ImageAlign

## See Also

ImageComponent Members
ImageComponent Class
MindFusion.Charting.Components Namespace

# ImageComponent.location Property

Gets or sets Image by its file location.

**Namespace:** MindFusion.Charting.Components
**File:** ImageComponent.js

## Syntax

**JavaScript**

```
get location() {}
```

### Property Value

Type: String

## See Also

ImageComponent Members
ImageComponent Class
MindFusion.Charting.Components Namespace

# LayoutAlignment Enumeration

Indicates how child components are aligned within the layout rectangle allocated to them by their parent component.

**Namespace:** MindFusion.Charting.Components
**File:** LayoutAlignment.js

## Syntax

**JavaScript**

```
// enum
LayoutAlignment = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **Center** | Center the child component. |
| | **Far** | Align the child component to the right or bottom side. |

| | | |
|---|---|---|
| | **Near** | Align the child component to the left or top side. |
| | **Stretch** | Stretch the child component. |

## See Also

MindFusion.Charting.Components Namespace

# LengthType Enumeration

Indicates how GridPanel determines dimensions of its rows and columns.

**Namespace:** MindFusion.Charting.Components
**File:** LengthType.js

## Syntax

**JavaScript**

```
// enum
LengthType = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **Auto** | The row or column is auto-sized to fit its child components. |
| | **Relative** | The size of relative elements is calculated from available space in the GridPanel proportionally to the number of other relative elements. |

## See Also

MindFusion.Charting.Components Namespace

# Orientation Enumeration

Identifies layout orientation.

**Namespace:** MindFusion.Charting.Components
**File:** Orientation.js

## Syntax

**JavaScript**

```
// enum
Orientation = {}
```

## Members

| | Member name | Description |
|---|---|---|

| | | |
|---|---|---|
| | **Horizontal** | Horizontal orientation. |
| | **Vertical** | Vertical orientation. |

## See Also

MindFusion.Charting.Components Namespace

# Panel Class

Represents a component that contains multiple child components and manages their layout.

## Syntax

**JavaScript**

```
// class
Panel.prototype = {}
```

## Inheritance Hierarchy

## See Also

Panel Members
MindFusion.Charting.Components Namespace

### Panel Members

Represents a component that contains multiple child components and manages their layout.

The following tables list the members exposed by the Panel type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | Panel | Initializes a new instance of the Panel class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | children | Gets the list of child components of this panel. |
| | parentControl | Gets or sets a reference to the control containing this panel. |

## Public Methods

| | Name | Description |
|---|---|---|
| | draw | Component.Draw override. Draws child components in specified RenderContext. |

| | | |
|---|---|---|
| | hitTest | Component.HitTest override. Returns the component containing specified point. |
| | invalidate | Component.Invalidate override. Invalidates the appearance this panel and calls its Draw method at next draw operation. |
| | invalidateLayout | Component.InvalidateLayout override. Invalidates the current layout and runs a new layout pass before next draw operation. |
| | measure | Component.Measure override. Measures the desired size of this panel and its child components. |
| | visit | Component.Visit override. Implements the visitor design pattern. |

## See Also

Panel Class
MindFusion.Charting.Components Namespace

# Panel Constructor

Initializes a new instance of the Panel class.

**Namespace:** MindFusion.Charting.Components
**File:** Panel.js

## Syntax

**JavaScript**

```
function Panel ()
```

## See Also

Panel Class
MindFusion.Charting.Components Namespace

# Panel Methods

# Panel.draw Method

Component.Draw override. Draws child components in specified RenderContext.

**Namespace:** MindFusion.Charting.Components
**File:** Panel.js

## Syntax

**JavaScript**

```
function draw (context)
```

### Parameters

*context*
Type: void
A RenderContext instance.

## See Also

Panel Members
Panel Class
MindFusion.Charting.Components Namespace

# Panel.hitTest Method

Component.HitTest override. Returns the component containing specified point.

**Namespace:** MindFusion.Charting.Components
**File:** Panel.js

## Syntax

**JavaScript**

```
function hitTest (x, y)
```

### Parameters

*x*
Type: Number
X coordinate of the point to test.

*y*
Type: Number
Y coordinate of the point to test.

### Return Value

Type: Component
A Component instance if one contains the point, or null otherwise.

## See Also

Panel Members
Panel Class
MindFusion.Charting.Components Namespace

# Panel.invalidate Method

Component.Invalidate override. Invalidates the appearance this panel and calls its Draw method at next draw operation.

**Namespace:** MindFusion.Charting.Components
**File:** Panel.js

## Syntax

**JavaScript**

```
function invalidate ([rect])
```

### Parameters

*rect*
Optional.
Type: Rect
A Rect instance specifying the invalid rectangle.

## See Also

Panel Members
Panel Class
MindFusion.Charting.Components Namespace

# Panel.invalidateLayout Method

Component.InvalidateLayout override. Invalidates the current layout and runs a new layout pass before next draw operation.

**Namespace:** MindFusion.Charting.Components
**File:** Panel.js

## Syntax

**JavaScript**

```
function invalidateLayout ()
```

## See Also

Panel Members
Panel Class
MindFusion.Charting.Components Namespace

# Panel.measure Method

Component.Measure override. Measures the desired size of this panel and its child components.

**Namespace:** MindFusion.Charting.Components
**File:** Panel.js

## Syntax

**JavaScript**

```
function measure (maxWidth, maxHeight, context)
```

### Parameters

*maxWidth*
Type: void
The maximum width provided by parent component.

*maxHeight*
Type: void
The maximum height provided by parent component.

*context*

Type: void
A RenderContext instance.

## See Also

Panel Members
Panel Class
MindFusion.Charting.Components Namespace

# Panel.visit Method

Component.Visit override. Implements the visitor design pattern.

**Namespace:** MindFusion.Charting.Components
**File:** Panel.js

## Syntax

```
JavaScript
function visit (visitor)
```

### Parameters

*visitor*
Type: void
An instance of a ComponentVisitor -derived class.

## See Also

Panel Members
Panel Class
MindFusion.Charting.Components Namespace

# Panel Properties

# Panel.children Property

Gets the list of child components of this panel.

**Namespace:** MindFusion.Charting.Components
**File:** Panel.js

## Syntax

```
JavaScript
get children() {}
```

### Property Value

Type: ObservableCollection

## See Also

Panel Members
Panel Class
MindFusion.Charting.Components Namespace

## Panel.parentControl Property

Gets or sets a reference to the control containing this panel.

**Namespace:** MindFusion.Charting.Components
**File:** Panel.js

### Syntax

```
JavaScript
get parentControl() {}
```

#### Property Value

Type: RootControl

### See Also

Panel Members
Panel Class
MindFusion.Charting.Components Namespace

# RangeSelector Class

Represents a control that handles zooming and scrolling of axes.

### Syntax

```
JavaScript
// class
RangeSelector.prototype = {}
```

### Inheritance Hierarchy

### See Also

RangeSelector Members
MindFusion.Charting.Components Namespace

## RangeSelector Members

Represents a control that handles zooming and scrolling of axes.

The following tables list the members exposed by the RangeSelector type.

### Public Constructors

| | Name | Description |
|---|---|---|

| | | |
|---|---|---|
| | RangeSelector | Initializes a new instance of the RangeSelector class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | axesSource | Gets or sets the object whose Axes will be handled by this component. |
| | axis | Gets or sets the Axis that will be handled by this component. |
| | brush | Gets or sets the Brush used to paint the component's background. |
| | handleBrush | Gets or sets the Brush used to paint resize handles. |
| | handleSize | Gets or sets the width or height of resize handles. |
| | maxValue | Gets or sets the largest value of the displayed range. |
| | minValue | Gets or sets the smallest value of the displayed range. |
| | orientation | Gets or sets the orientation of the component. |
| | size | Gets or sets the width or height of the component. |
| | thumbBrush | Gets or sets the Brush used to paint the thumb. |

## Public Methods

| | Name | Description |
|---|---|---|
| | arrange | Component.Arrange override. Sets the location and size of this component relatively to its parent. |
| | createController | Component.CreateController override. Returns a controller used to interact with this component. |
| | draw | Component.Draw override. Draws this component in specified RenderContext. |
| | measure | Component.Measure override. Measures the desired size of this component. |

## See Also

RangeSelector Class
MindFusion.Charting.Components Namespace

### RangeSelector Constructor

Initializes a new instance of the RangeSelector class.

**Namespace:** MindFusion.Charting.Components
**File:** RangeSelector.js

## Syntax

```
JavaScript
function RangeSelector (axis, minValue, maxValue)
```

### Parameters

*axis*
Type: void

*minValue*
Type: void

*maxValue*
Type: void

## See Also

RangeSelector Class
MindFusion.Charting.Components Namespace

# RangeSelector Methods

## RangeSelector.arrange Method

Component.Arrange override. Sets the location and size of this component relatively to its parent.

**Namespace:** MindFusion.Charting.Components
**File:** RangeSelector.js

## Syntax

```
JavaScript
function arrange (x, y, w, h, context)
```

### Parameters

*x*
Type: Number
A number value specifying horizontal position.

*y*
Type: Number
A number value specifying vertical position.

*w*
Type: Number
A number value specifying the component's width.

*h*
Type: Number
A number value specifying the component's height.

*context*
Type: RenderContext
A RenderContext instance.

## See Also

RangeSelector Members
RangeSelector Class
MindFusion.Charting.Components Namespace

# RangeSelector.createController Method

Component.CreateController override. Returns a controller used to interact with this component.

**Namespace:** MindFusion.Charting.Components
**File:** RangeSelector.js

## Syntax

```
JavaScript
function createController (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: ComponentController
Instance of a ComponentController -derived class.

## See Also

RangeSelector Members
RangeSelector Class
MindFusion.Charting.Components Namespace

# RangeSelector.draw Method

Component.Draw override. Draws this component in specified RenderContext.

**Namespace:** MindFusion.Charting.Components
**File:** RangeSelector.js

## Syntax

```
JavaScript
function draw (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

RangeSelector Members
RangeSelector Class

# RangeSelector.measure Method

Component.Measure override. Measures the desired size of this component.

**Namespace:** MindFusion.Charting.Components
**File:** RangeSelector.js

## Syntax

**JavaScript**

```
function measure (maxWidth, maxHeight, context)
```

### Parameters

*maxWidth*
Type: Number
The maximum width provided by parent component.

*maxHeight*
Type: Number
The maximum height provided by parent component.

*context*
Type: RenderContext
A RenderContext instance.

## See Also

RangeSelector Members
RangeSelector Class
MindFusion.Charting.Components Namespace

# RangeSelector Properties

# RangeSelector.axesSource Property

Gets or sets the object whose Axes will be handled by this component.

**Namespace:** MindFusion.Charting.Components
**File:** RangeSelector.js

## Syntax

**JavaScript**

```
get axesSource() {}
```

### Property Value

Type: SeriesContainer

## See Also

RangeSelector Members
RangeSelector Class
MindFusion.Charting.Components Namespace

## RangeSelector.axis Property

Gets or sets the Axis that will be handled by this component.

**Namespace:** MindFusion.Charting.Components
**File:** RangeSelector.js

### Syntax

```
JavaScript
get axis() {}
```

#### Property Value

Type: Axis

### See Also

RangeSelector Members
RangeSelector Class
MindFusion.Charting.Components Namespace

## RangeSelector.brush Property

Gets or sets the Brush used to paint the component's background.

**Namespace:** MindFusion.Charting.Components
**File:** RangeSelector.js

### Syntax

```
JavaScript
get brush() {}
```

#### Property Value

Type: Brush

### See Also

RangeSelector Members
RangeSelector Class
MindFusion.Charting.Components Namespace

## RangeSelector.handleBrush Property

Gets or sets the Brush used to paint resize handles.

**Namespace:** MindFusion.Charting.Components
**File:** RangeSelector.js

## Syntax

```
JavaScript
get handleBrush() {}
```

### Property Value

Type: Brush

## See Also

RangeSelector Members
RangeSelector Class
MindFusion.Charting.Components Namespace

# RangeSelector.handleSize Property

Gets or sets the width or height of resize handles.

**Namespace:** MindFusion.Charting.Components
**File:** RangeSelector.js

## Syntax

```
JavaScript
get handleSize() {}
```

### Property Value

Type: Number

## See Also

RangeSelector Members
RangeSelector Class
MindFusion.Charting.Components Namespace

# RangeSelector.maxValue Property

Gets or sets the largest value of the displayed range.

**Namespace:** MindFusion.Charting.Components
**File:** RangeSelector.js

## Syntax

```
JavaScript
get maxValue() {}
```

### Property Value

Type: Number

## See Also

RangeSelector Members
RangeSelector Class
MindFusion.Charting.Components Namespace

# RangeSelector.minValue Property

Gets or sets the smallest value of the displayed range.

**Namespace:** MindFusion.Charting.Components
**File:** RangeSelector.js

## Syntax

**JavaScript**

```
get minValue() {}
```

### Property Value

Type: Number

## See Also

RangeSelector Members
RangeSelector Class
MindFusion.Charting.Components Namespace

# RangeSelector.orientation Property

Gets or sets the orientation of the component.

**Namespace:** MindFusion.Charting.Components
**File:** RangeSelector.js

## Syntax

**JavaScript**

```
get orientation() {}
```

### Property Value

Type: Orientation

## See Also

RangeSelector Members
RangeSelector Class
MindFusion.Charting.Components Namespace

# RangeSelector.size Property

Gets or sets the width or height of the component.

**Namespace:** MindFusion.Charting.Components
**File:** RangeSelector.js

## Syntax

```
JavaScript
get size() {}
```

### Property Value

Type: Number

## See Also

RangeSelector Members
RangeSelector Class
MindFusion.Charting.Components Namespace

# RangeSelector.thumbBrush Property

Gets or sets the Brush used to paint the thumb.

**Namespace:** MindFusion.Charting.Components
**File:** RangeSelector.js

## Syntax

```
JavaScript
get thumbBrush() {}
```

### Property Value

Type: Brush

## See Also

RangeSelector Members
RangeSelector Class
MindFusion.Charting.Components Namespace

# RootControl Interface

Interface used to communicate with host DOM element.

## Syntax

```
JavaScript
// interface
RootControl.prototype = {}
```

## See Also

# RootControl Members

Interface used to communicate with host DOM element.

The following tables list the members exposed by the RootControl type.

## Public Methods

| | Name | Description |
|---|---|---|
|  | invalidate | Invalidates the specified region of a component. |
|  | invalidateLayout | Invalidates layout of specified component. |

## See Also

RootControl Interface
MindFusion.Charting.Components Namespace

# RootControl Methods

# RootControl.invalidate Method

Invalidates the specified region of a component.

**Namespace:** MindFusion.Charting.Components
**File:** RootControl.js

## Syntax

**JavaScript**

```
function invalidate (rect, panel)
```

### Parameters

*rect*

Type: void
The area to invalidate and redraw.

*panel*

Type: void
The reference Component.

## See Also

RootControl Members
RootControl Interface
MindFusion.Charting.Components Namespace

## RootControl.invalidateLayout Method

Invalidates layout of specified component.

**Namespace:** MindFusion.Charting.Components
**File:** RootControl.js

### Syntax

**JavaScript**

```
function invalidateLayout (panel)
```

#### Parameters

*panel*
Type: void
The component to invalidate.

### See Also

RootControl Members
RootControl Interface
MindFusion.Charting.Components Namespace

# SimplePanel Class

Represents a layout panel that arranges its child components over each other.

### Syntax

**JavaScript**

```
// class
SimplePanel.prototype = {}
```

### Inheritance Hierarchy

### See Also

SimplePanel Members
MindFusion.Charting.Components Namespace

## SimplePanel Members

Represents a layout panel that arranges its child components over each other.

The following tables list the members exposed by the SimplePanel type.

### Public Methods

| | Name | Description |
|---|---|---|
|  | arrange | Component.Arrange override. Sets the location and size of this panel and arranges its child components. |

| | | |
|---|---|---|
| | measure | Panel.Measure override. Measures the desired size of this panel and its child components. |

## See Also

SimplePanel Class
MindFusion.Charting.Components Namespace

# SimplePanel Methods

## SimplePanel.arrange Method

Component.Arrange override. Sets the location and size of this panel and arranges its child components.

**Namespace:** MindFusion.Charting.Components
**File:** SimplePanel.js

## Syntax

**JavaScript**

```
function arrange (x, y, w, h, context)
```

### Parameters

*x*
Type: Number
A number value specifying horizontal position.

*y*
Type: Number
A number value specifying vertical position.

*w*
Type: Number
A number value specifying the component's width.

*h*
Type: Number
A number value specifying the component's height.

*context*
Type: RenderContext
A RenderContext instance.

## See Also

SimplePanel Members
SimplePanel Class
MindFusion.Charting.Components Namespace

## SimplePanel.measure Method

Panel.Measure override. Measures the desired size of this panel and its child components.

**Namespace:** MindFusion.Charting.Components
**File:** SimplePanel.js

## Syntax

| JavaScript |
| --- |
| `function measure (maxWidth, maxHeight, context)` |

### Parameters

*maxWidth*
Type: Number
The maximum width provided by parent component.

*maxHeight*
Type: Number
The maximum height provided by parent component.

*context*
Type: RenderContext
A RenderContext instance.

## See Also

SimplePanel Members
SimplePanel Class
MindFusion.Charting.Components Namespace

# StackPanel Class

Represents a layout panel that arranges its child components in a stack.

## Syntax

| JavaScript |
| --- |
| `// class`<br>`StackPanel.prototype = {}` |

## Inheritance Hierarchy

## See Also

StackPanel Members
MindFusion.Charting.Components Namespace

### StackPanel Members

Represents a layout panel that arranges its child components in a stack.

The following tables list the members exposed by the StackPanel type.

## Public Properties

| | Name | Description |
| --- | --- | --- |
|  | orientation | Gets or sets the stack orientation. |

## Public Methods

| | Name | Description |
|---|---|---|
| | arrange | Component.Arrange override. Sets the location and size of this panel and arranges its child components. |
| | measure | Panel.Measure override. Measures the desired size of this panel and its child components. |

## See Also

StackPanel Class
MindFusion.Charting.Components Namespace

# StackPanel Methods

## StackPanel.arrange Method

Component.Arrange override. Sets the location and size of this panel and arranges its child components.

**Namespace:** MindFusion.Charting.Components
**File:** StackPanel.js

## Syntax

**JavaScript**

```
function arrange (x, y, w, h, context)
```

### Parameters

*x*
Type: Number
A number value specifying horizontal position.

*y*
Type: Number
A number value specifying vertical position.

*w*
Type: Number
A number value specifying the component's width.

*h*
Type: Number
A number value specifying the component's height.

*context*
Type: RenderContext
A RenderContext instance.

## See Also

StackPanel Members
StackPanel Class
MindFusion.Charting.Components Namespace

# StackPanel.measure Method

Panel.Measure override. Measures the desired size of this panel and its child components.

**Namespace:** MindFusion.Charting.Components
**File:** StackPanel.js

## Syntax

**JavaScript**

```
function measure (maxWidth, maxHeight, context)
```

### Parameters

*maxWidth*
Type: Number
The maximum width provided by parent component.

*maxHeight*
Type: Number
The maximum height provided by parent component.

*context*
Type: RenderContext
A RenderContext instance.

## See Also

StackPanel Members
StackPanel Class
MindFusion.Charting.Components Namespace

# StackPanel Properties

# StackPanel.orientation Property

Gets or sets the stack orientation.

**Namespace:** MindFusion.Charting.Components
**File:** StackPanel.js

## Syntax

**JavaScript**

```
get orientation() {}
```

### Property Value

Type: Orientation

## See Also

StackPanel Members
StackPanel Class
MindFusion.Charting.Components Namespace

# TextComponent Class

Represents a Component that draws text.

## Syntax

```
// class
TextComponent.prototype = {}
```

## Inheritance Hierarchy

## See Also

TextComponent Members
MindFusion.Charting.Components Namespace

# TextComponent Members

Represents a Component that draws text.

The following tables list the members exposed by the TextComponent type.

## Public Properties

| | Name | Description |
|---|---|---|
| | fontName | Gets or sets the name of the font that should be used to draw the component's text. |
| | fontSize | Gets or sets the size of the font that should be used to draw the component's text. |
| | fontStyle | Gets or sets the style of the font that should be used to draw the component's text. |
| | styleHint | Gets or sets a value indicating which attribute values this component should inherit from current Theme if its local text appearance properties are not set. |
| | text | Gets or sets the text that should be drawn inside this component. |
| | textBrush | Gets or sets the Brush that should be used to draw the component's text. |

## Public Methods

| | Name | Description |
|---|---|---|
| | draw | Component.Draw override. Draws associated Text in specified RenderContext. |
| | measure | Component.Measure override. Measures the desired size of this component. |

## See Also

TextComponent Class
MindFusion.Charting.Components Namespace

# TextComponent Methods

## TextComponent.draw Method

Component.Draw override. Draws associated Text in specified RenderContext.

**Namespace:** MindFusion.Charting.Components
**File:** TextComponent.js

## Syntax

**JavaScript**

```
function draw (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

TextComponent Members
TextComponent Class
MindFusion.Charting.Components Namespace

## TextComponent.measure Method

Component.Measure override. Measures the desired size of this component.

**Namespace:** MindFusion.Charting.Components
**File:** TextComponent.js

## Syntax

**JavaScript**

```
function measure (maxWidth, maxHeight, context)
```

### Parameters

*maxWidth*
Type: Number
The maximum width provided by parent component.

*maxHeight*
Type: Number
The maximum height provided by parent component.

*context*

Type: RenderContext
A RenderContext instance.

## See Also

TextComponent Members
TextComponent Class
MindFusion.Charting.Components Namespace

# TextComponent Properties

## TextComponent.fontName Property

Gets or sets the name of the font that should be used to draw the component's text.

**Namespace:** MindFusion.Charting.Components
**File:** TextComponent.js

## Syntax

```
JavaScript
get fontName() {}
```

### Property Value

Type: String

## See Also

TextComponent Members
TextComponent Class
MindFusion.Charting.Components Namespace

## TextComponent.fontSize Property

Gets or sets the size of the font that should be used to draw the component's text.

**Namespace:** MindFusion.Charting.Components
**File:** TextComponent.js

## Syntax

```
JavaScript
get fontSize() {}
```

### Property Value

Type: Number

## See Also

TextComponent Members
TextComponent Class
MindFusion.Charting.Components Namespace

## TextComponent.fontStyle Property

Gets or sets the style of the font that should be used to draw the component's text.

**Namespace:** MindFusion.Charting.Components
**File:** TextComponent.js

### Syntax

```
JavaScript
get fontStyle() {}
```

#### Property Value

Type: FontStyle

### See Also

TextComponent Members
TextComponent Class
MindFusion.Charting.Components Namespace

## TextComponent.styleHint Property

Gets or sets a value indicating which attribute values this component should inherit from current Theme if its local text appearance properties are not set.

**Namespace:** MindFusion.Charting.Components
**File:** TextComponent.js

### Syntax

```
JavaScript
get styleHint() {}
```

#### Property Value

Type: TextStyleHint

### See Also

TextComponent Members
TextComponent Class
MindFusion.Charting.Components Namespace

## TextComponent.text Property

Gets or sets the text that should be drawn inside this component.

**Namespace:** MindFusion.Charting.Components
**File:** TextComponent.js

## Syntax

```
JavaScript
get text() {}
```

### Property Value

Type: String

## See Also

TextComponent Members
TextComponent Class
MindFusion.Charting.Components Namespace

# TextComponent.textBrush Property

Gets or sets the Brush that should be used to draw the component's text.

**Namespace:** MindFusion.Charting.Components
**File:** TextComponent.js

## Syntax

```
JavaScript
get textBrush() {}
```

### Property Value

Type: Brush

## See Also

TextComponent Members
TextComponent Class
MindFusion.Charting.Components Namespace

# Visibility Enumeration

Identifies visibility of components.

**Namespace:** MindFusion.Charting.Components
**File:** Visibility.js

## Syntax

```
JavaScript
// enum
Visibility = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **Collapsed** | The component is hidden and does not participate in layout measurements. |
| | **Hidden** | The component is hidden but participates in layout measurements.c |
| | **Visible** | The component is visible. |

## See Also

MindFusion.Charting.Components Namespace

# MindFusion.Charting.Controls Namespace

## AreaChart Class

A control used to draw area charts.

## Syntax

```
JavaScript
// class
AreaChart.prototype = {}
```

## Inheritance Hierarchy

## See Also

AreaChart Members
MindFusion.Charting.Controls Namespace

## AreaChart Members

A control used to draw area charts.

The following tables list the members exposed by the AreaChart type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | AreaChart | Initializes a new instance of the AreaChart class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | areaOpacity | Gets or sets the opacity of area polygons. |

| | | |
|---|---|---|
| | lineType | Gets or sets what type of line segments to draw between data points. |
| | series | BiaxialChart.Series override. Gets or sets the list of series whose data is drawn in this chart. |

## See Also

AreaChart Class
MindFusion.Charting.Controls Namespace

# AreaChart Constructor

Initializes a new instance of the AreaChart class.

**Namespace:** MindFusion.Charting.Controls
**File:** AreaChart.js

## Syntax

**JavaScript**

```
function AreaChart (element, [lineType, [seriesRenderer]])
```

### Parameters

*element*
Type: HTMLCanvasElement
The canvas DOM element to associate this chart with.

*lineType*
Optional.
Type: LineType
A member of the LineType enumeration.

*seriesRenderer*
Optional.
Type: AreaRenderer
An AreaRenderer used to draw chart's data series.

## See Also

AreaChart Class
MindFusion.Charting.Controls Namespace

# AreaChart Properties

# AreaChart.areaOpacity Property

Gets or sets the opacity of area polygons.

**Namespace:** MindFusion.Charting.Controls
**File:** AreaChart.js

## Syntax

JavaScript

```
get areaOpacity() {}
```

### Property Value

Type: Number

## See Also

AreaChart Members
AreaChart Class
MindFusion.Charting.Controls Namespace

# AreaChart.lineType Property

Gets or sets what type of line segments to draw between data points.

**Namespace:** MindFusion.Charting.Controls
**File:** AreaChart.js

## Syntax

JavaScript

```
get lineType() {}
```

### Property Value

Type: LineType

## See Also

AreaChart Members
AreaChart Class
MindFusion.Charting.Controls Namespace

# AreaChart.series Property

BiaxialChart.Series override. Gets or sets the list of series whose data is drawn in this chart.

**Namespace:** MindFusion.Charting.Controls
**File:** AreaChart.js

## Syntax

JavaScript

```
get series() {}
```

### Property Value

Type: ObservableCollection

## See Also

# BarChart Class

A control used to draw bar charts.

## Syntax

```
// class
BarChart.prototype = {}
```

## Inheritance Hierarchy

## See Also

BarChart Members
MindFusion.Charting.Controls Namespace

# BarChart Members

A control used to draw bar charts.

The following tables list the members exposed by the BarChart type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | BarChart | Initializes a new instance of the BarChart class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | barLayout | Gets or sets how to arrange bars when rendering multiple series. |
|  | barSpacingRatio | Gets or sets the ratio of empty space between bars to space occupied by bars. |
|  | horizontalBars | Gets or sets a value identifying whether bars should be horizontal. |
|  | innerLabelRotation | Gets the rotation angle of inner labels. If not specified, labels are rotated at predefiend angles depending on bars' orientation and layout. |
|  | outerLabelRotation | Gets the rotation angle of outer labels. If not specified, labels are rotated at predefiend angles depending on bars' orientation and layout. |

| | | |
|---|---|---|
| | series | BiaxialChart.Series override. Gets or sets the list of series whose data is drawn in this chart. |
| | stackOuterLabels | Gets whether outer labels are displayed stacked on top of stacked bars, instead of showing them on the bar sides. |

## Public Methods

| | Name | Description |
|---|---|---|
| | createPlot | Chart.CreatePlot override. Creates the Plot type corresponding to this chart. |

## See Also

BarChart Class
MindFusion.Charting.Controls Namespace

# BarChart Constructor

Initializes a new instance of the BarChart class.

**Namespace:** MindFusion.Charting.Controls
**File:** BarChart.js

## Syntax

**JavaScript**

```
function BarChart (element, [barLayout, [seriesRenderer]])
```

### Parameters

*element*
Type: HTMLCanvasElement
The canvas DOM element to associate this chart with.

*barLayout*
Optional.
Type: BarLayout
A member of the BarLayout enumeration.

*seriesRenderer*
Optional.
Type: Renderer2D
A Renderer2D used to draw chart's data series.

## See Also

BarChart Class
MindFusion.Charting.Controls Namespace

# BarChart Methods

## BarChart.createPlot Method

Chart.CreatePlot override. Creates the Plot type corresponding to this chart.

**Namespace:** MindFusion.Charting.Controls
**File:** BarChart.js

### Syntax

```
JavaScript
function createPlot ()
```

#### Return Value

Type: Plot
An instance of the Plot2D class.

### See Also

BarChart Members
BarChart Class
MindFusion.Charting.Controls Namespace

## BarChart Properties

## BarChart.barLayout Property

Gets or sets how to arrange bars when rendering multiple series.

**Namespace:** MindFusion.Charting.Controls
**File:** BarChart.js

### Syntax

```
JavaScript
get barLayout() {}
```

#### Property Value

Type: BarLayout

### See Also

BarChart Members
BarChart Class
MindFusion.Charting.Controls Namespace

## BarChart.barSpacingRatio Property

Gets or sets the ratio of empty space between bars to space occupied by bars.

**Namespace:** MindFusion.Charting.Controls
**File:** BarChart.js

## Syntax

```
JavaScript
get barSpacingRatio() {}
```

### Property Value

Type: Number

## See Also

BarChart Members
BarChart Class
MindFusion.Charting.Controls Namespace

# BarChart.horizontalBars Property

Gets or sets a value identifying whether bars should be horizontal.

**Namespace:** MindFusion.Charting.Controls
**File:** BarChart.js

## Syntax

```
JavaScript
get horizontalBars() {}
```

### Property Value

Type: Boolean

## See Also

BarChart Members
BarChart Class
MindFusion.Charting.Controls Namespace

# BarChart.innerLabelRotation Property

Gets the rotation angle of inner labels. If not specified, labels are rotated at predefiend angles depending on bars' orientation and layout.

**Namespace:** MindFusion.Charting.Controls
**File:** BarChart.js

## Syntax

```
JavaScript
get innerLabelRotation() {}
```

### Property Value

Type: Number

## Example

The following code customizes a BarChart instance:

**JavaScript**

```
barChart.backColor = new Drawing.Color("gray");
barChart.innerLabelRotation = 30;
```

## See Also

BarChart Members
BarChart Class
MindFusion.Charting.Controls Namespace

# BarChart.outerLabelRotation Property

Gets the rotation angle of outer labels. If not specified, labels are rotated at predefiend angles depending on bars' orientation and layout.

**Namespace:** MindFusion.Charting.Controls
**File:** BarChart.js

## Syntax

**JavaScript**

```
get outerLabelRotation() {}
```

### Property Value

Type: Number

## Example

The following code customizes a BarChart instance:

**JavaScript**

```
barChart.backColor = new Drawing.Color("gray");
barChart.outerLabelRotation = 30;
```

## See Also

BarChart Members
BarChart Class
MindFusion.Charting.Controls Namespace

# BarChart.series Property

BiaxialChart.Series override. Gets or sets the list of series whose data is drawn in this chart.

**Namespace:** MindFusion.Charting.Controls
**File:** BarChart.js

## Syntax

JavaScript

```
get series() {}
```

### Property Value

Type: ObservableCollection

## See Also

BarChart Members
BarChart Class
MindFusion.Charting.Controls Namespace

# BarChart.stackOuterLabels Property

Gets whether outer labels are displayed stacked on top of stacked bars, instead of showing them on the bar sides.

**Namespace:** MindFusion.Charting.Controls
**File:** BarChart.js

## Syntax

JavaScript

```
get stackOuterLabels() {}
```

### Property Value

Type: Boolean

## Example

The following code creates two BarSeries with data and outer labels. They are assigned to the series property of a BarChart control, whose barLayout is set to BarLayout.Stack. The stackOuterLabels property is used to specify that outer labels must be drawn on top of the bar stacks:

JavaScript

```
//the second argument specifies the inner labels, the third argument specifies the top(outer)
labels
var data1 = new Charting.BarSeries(
 new Collections.List([13, 13, 14, 16, 23, 44]), null, new Collections.List(["Jan", "Feb",
"Mar", "Apr",

"May", "Jun"]));
var data2 = new Charting.BarSeries(
 new Collections.List([22, 18, 25, 34, 51, 44, 8]), null, new Collections.List(["Jan", "Feb",
"Mar", "Apr",

"May", "Jun"]));

barChart.series = new Collections.ObservableCollection([data1, data2]);

barChart.barLayout = Charting.BarLayout.Stack;
barRenderer.stackOuterLabels = true;
```

## See Also

BarChart Members
BarChart Class

MindFusion.Charting.Controls Namespace

# BarChart3D Class

A control used to draw 3D bar charts.

## Syntax

```javascript
// class
BarChart3D.prototype = {}
```

## Inheritance Hierarchy

## See Also

BarChart3D Members
MindFusion.Charting.Controls Namespace

# BarChart3D Members

A control used to draw 3D bar charts.

The following tables list the members exposed by the BarChart3D type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | BarChart3D | Initializes a new instance of the BarChart3D class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | barLayout | Gets or sets how to arrange bars when rendering multiple series. |
|  | barModel | Gets or sets how to visualize the bars when rendering multiple series. |
|  | series | BiaxialChart.Series override. Gets or sets the list of series whose data is drawn in this chart. |

## Public Methods

| | Name | Description |
|---|---|---|
|  | createPlot | Chart.CreatePlot override. Creates the Plot type corresponding to this chart. |

## See Also

BarChart3D Class
MindFusion.Charting.Controls Namespace

# BarChart3D Constructor

Initializes a new instance of the BarChart3D class.

**Namespace:** MindFusion.Charting.Controls
**File:** BarChart3D.js

## Syntax

**JavaScript**

```
function BarChart3D (element, [barLayout, [barModel, [seriesRenderer]]])
```

### Parameters

*element*
Type: HTMLCanvasElement
The canvas DOM element to associate this chart with.

*barLayout*
Optional.
Type: BarLayout
A member of the BarLayout enumeration.

*barModel*
Optional.
Type: BarModel3D
A member of the BarModel3D enumeration.

*seriesRenderer*
Optional.
Type: Renderer2D
A Renderer2D used to draw chart's data series.

## See Also

BarChart3D Class
MindFusion.Charting.Controls Namespace

# BarChart3D Methods

# BarChart3D.createPlot Method

Chart.CreatePlot override. Creates the Plot type corresponding to this chart.

**Namespace:** MindFusion.Charting.Controls
**File:** BarChart3D.js

## Syntax

**JavaScript**

```
function createPlot ()
```

### Return Value

Type: Plot
An instance of the Plot3D class.

## See Also

BarChart3D Members
BarChart3D Class
MindFusion.Charting.Controls Namespace

# BarChart3D Properties

## BarChart3D.barLayout Property

Gets or sets how to arrange bars when rendering multiple series.

**Namespace:** MindFusion.Charting.Controls
**File:** BarChart3D.js

## Syntax

```
JavaScript
get barLayout() {}
```

### Property Value

Type: BarLayout

## See Also

BarChart3D Members
BarChart3D Class
MindFusion.Charting.Controls Namespace

## BarChart3D.barModel Property

Gets or sets how to visualize the bars when rendering multiple series.

**Namespace:** MindFusion.Charting.Controls
**File:** BarChart3D.js

## Syntax

```
JavaScript
get barModel() {}
```

### Property Value

Type: BarModel3D

## See Also

BarChart3D Members
BarChart3D Class
MindFusion.Charting.Controls Namespace

## BarChart3D.series Property

BiaxialChart.Series override. Gets or sets the list of series whose data is drawn in this chart.

**Namespace:** MindFusion.Charting.Controls
**File:** BarChart3D.js

### Syntax

```
JavaScript
get series() {}
```

#### Property Value

Type: ObservableCollection

### See Also

BarChart3D Members
BarChart3D Class
MindFusion.Charting.Controls Namespace

# BiaxialChart Class

A base class for charts that display X and Y axes.

### Syntax

```
JavaScript
// class
BiaxialChart.prototype = {}
```

### Inheritance Hierarchy

### See Also

BiaxialChart Members
MindFusion.Charting.Controls Namespace

## BiaxialChart Members

A base class for charts that display X and Y axes.

The following tables list the members exposed by the BiaxialChart type.

### Public Constructors

| | Name | Description |
|---|---|---|

| | | |
|---|---|---|
| | BiaxialChart | Initializes a new instance of the BiaxialChart class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | allowPan | Gets or sets a value indicating whether users are allowed to pan the chart's plot. |
| | annotations | Gets or sets the list of series whose labels are drawn as annotations. |
| | gridType | Gets or sets the type of grid to draw in this chart. |
| | pinGrid | Gets or sets a value indicating whether grid stripes should be pinned in place or scroll together with the plot when users pan it. |
| | series | Gets or sets the list of series whose data is drawn in this chart. |
| | showScatter | Gets or sets a value indicating whether the chart should render scatter shapes. |
| | showXCoordinates | Gets or sets a value indicating whether to show X axis coordinates. |
| | showXRangeSelector | Gets or sets a value indicating whether to show a RangeSelector for the X axis, to let users scroll or resize the currently visible horizontal data window. |
| | showXTicks | Gets or sets a value indicating whether to show X axis ticks. |
| | showYCoordinates | Gets or sets a value indicating whether to show Y axis coordinates. |
| | showYRangeSelector | Gets or sets a value indicating whether to show a RangeSelector for the Y axis, to let users scroll or resize the currently visible vertical data window. |
| | showYTicks | Gets or sets a value indicating whether to show Y axis ticks. |
| | xAxis | Gets or sets an Axis object representing horizontal data range. |
| | xAxisLabelRotationAngle | Gets or sets the rotation angle of the X axis labels. |
| | xAxisTickLength | Gets or sets the length of X axis ticks. |
| | xScrollRangeMax | Gets or sets the largest value allowed to scroll to using the X axis' RangeSelector. |
| | xScrollRangeMin | Gets or sets the smallest value allowed to scroll to using the X axis' RangeSelector. |
| | yAxis | Gets or sets an Axis object representing vertical data range. |
| | yAxisLabelRotationAngle | Gets or sets the rotation angle of the Y axis labels. |
| | yAxisTickLength | Gets or sets the length of Y axis ticks. |
| | yLabelAlignment | Gets the horizontal alignment of Y-axis labels. |

| | | |
|---|---|---|
| | yScrollRangeMax | Gets or sets the largest value allowed to scroll to using the Y axis' RangeSelector. |
| | yScrollRangeMin | Gets or sets the smallest value allowed to scroll to using the Y axis' RangeSelector. |

## Public Methods

| | Name | Description |
|---|---|---|
| | createRenderContext | ChartBase.createRenderContext override. Creates a RenderContext instance. |
| | resetZoom | Chart.resetZoom override. Resets zoom level to original data range. |
| | zoomOut | Chart.zoomOut override. Zooms out of current data range. |

## See Also

BiaxialChart Class
MindFusion.Charting.Controls Namespace

# BiaxialChart Constructor

Initializes a new instance of the BiaxialChart class.

**Namespace:** MindFusion.Charting.Controls
**File:** BiaxialChart.js

## Syntax

**JavaScript**

```
function BiaxialChart (element, [seriesRenderer])
```

### Parameters

*element*
Type: HTMLCanvasElement
The canvas DOM element to associate this chart with.

*seriesRenderer*
Optional.
Type: Renderer2D
A SeriesRenderer used to draw chart's data series.

## See Also

BiaxialChart Class
MindFusion.Charting.Controls Namespace

# BiaxialChart Methods

# BiaxialChart.createRenderContext Method

ChartBase.createRenderContext override. Creates a RenderContext instance.

**Namespace:** MindFusion.Charting.Controls
**File:** BiaxialChart.js

## Syntax

**JavaScript**

```
function createRenderContext (graphics, clipRect)
```

### Parameters

*graphics*
Type: Graphics
A Graphics surface where dashboard elements should be rendered.

*clipRect*
Type: Rect
The current clip rectangle.

### Return Value

Type: RenderContext
A RenderContext instance.

## See Also

BiaxialChart Members
BiaxialChart Class
MindFusion.Charting.Controls Namespace

# BiaxialChart.resetZoom Method

Chart.resetZoom override. Resets zoom level to original data range.

**Namespace:** MindFusion.Charting.Controls
**File:** BiaxialChart.js

## Syntax

**JavaScript**

```
function resetZoom ()
```

## See Also

BiaxialChart Members
BiaxialChart Class
MindFusion.Charting.Controls Namespace

# BiaxialChart.zoomOut Method

Chart.zoomOut override. Zooms out of current data range.

**Namespace:** MindFusion.Charting.Controls
**File:** BiaxialChart.js

## Syntax

```
JavaScript
function zoomOut ()
```

## See Also

BiaxialChart Members
BiaxialChart Class
MindFusion.Charting.Controls Namespace

# BiaxialChart Properties

## BiaxialChart.allowPan Property

Gets or sets a value indicating whether users are allowed to pan the chart's plot.

**Namespace:** MindFusion.Charting.Controls
**File:** BiaxialChart.js

## Syntax

```
JavaScript
get allowPan() {}
```

### Property Value

Type: Boolean

## See Also

BiaxialChart Members
BiaxialChart Class
MindFusion.Charting.Controls Namespace

## BiaxialChart.annotations Property

Gets or sets the list of series whose labels are drawn as annotations.

**Namespace:** MindFusion.Charting.Controls
**File:** BiaxialChart.js

## Syntax

```
JavaScript
get annotations() {}
```

### Property Value

Type: ObservableCollection

## See Also

BiaxialChart Members
BiaxialChart Class
MindFusion.Charting.Controls Namespace

# BiaxialChart.gridType Property

Gets or sets the type of grid to draw in this chart.

**Namespace:** MindFusion.Charting.Controls
**File:** BiaxialChart.js

## Syntax

**JavaScript**

```
get gridType() {}
```

### Property Value

Type: GridType

## See Also

BiaxialChart Members
BiaxialChart Class
MindFusion.Charting.Controls Namespace

# BiaxialChart.pinGrid Property

Gets or sets a value indicating whether grid stripes should be pinned in place or scroll together with the plot when users pan it.

**Namespace:** MindFusion.Charting.Controls
**File:** BiaxialChart.js

## Syntax

**JavaScript**

```
get pinGrid() {}
```

### Property Value

Type: Boolean

## See Also

BiaxialChart Members
BiaxialChart Class
MindFusion.Charting.Controls Namespace

# BiaxialChart.series Property

Gets or sets the list of series whose data is drawn in this chart.

**Namespace:** MindFusion.Charting.Controls
**File:** BiaxialChart.js

## Syntax

```
JavaScript
get series() {}
```

### Property Value

Type: ObservableCollection

## See Also

BiaxialChart Members
BiaxialChart Class
MindFusion.Charting.Controls Namespace

# BiaxialChart.showScatter Property

Gets or sets a value indicating whether the chart should render scatter shapes.

**Namespace:** MindFusion.Charting.Controls
**File:** BiaxialChart.js

## Syntax

```
JavaScript
get showScatter() {}
```

### Property Value

Type: Boolean

## See Also

BiaxialChart Members
BiaxialChart Class
MindFusion.Charting.Controls Namespace

# BiaxialChart.showXCoordinates Property

Gets or sets a value indicating whether to show X axis coordinates.

**Namespace:** MindFusion.Charting.Controls
**File:** BiaxialChart.js

## Syntax

```
JavaScript
```

```
get showXCoordinates() {}
```

### Property Value

Type: Boolean

## See Also

BiaxialChart Members
BiaxialChart Class
MindFusion.Charting.Controls Namespace

# BiaxialChart.showXRangeSelector Property

Gets or sets a value indicating whether to show a RangeSelector for the X axis, to let users scroll or resize the currently visible horizontal data window.

**Namespace:** MindFusion.Charting.Controls
**File:** BiaxialChart.js

## Syntax

**JavaScript**

```
get showXRangeSelector() {}
```

### Property Value

Type: Boolean

## See Also

BiaxialChart Members
BiaxialChart Class
MindFusion.Charting.Controls Namespace

# BiaxialChart.showXTicks Property

Gets or sets a value indicating whether to show X axis ticks.

**Namespace:** MindFusion.Charting.Controls
**File:** BiaxialChart.js

## Syntax

**JavaScript**

```
get showXTicks() {}
```

### Property Value

Type: Boolean

## See Also

BiaxialChart Members
BiaxialChart Class
MindFusion.Charting.Controls Namespace

## BiaxialChart.showYCoordinates Property

Gets or sets a value indicating whether to show Y axis coordinates.

**Namespace:** MindFusion.Charting.Controls
**File:** BiaxialChart.js

### Syntax

```
JavaScript
get showYCoordinates() {}
```

#### Property Value

Type: Boolean

### See Also

BiaxialChart Members
BiaxialChart Class
MindFusion.Charting.Controls Namespace

## BiaxialChart.showYRangeSelector Property

Gets or sets a value indicating whether to show a RangeSelector for the Y axis, to let users scroll or resize the currently visible vertical data window.

**Namespace:** MindFusion.Charting.Controls
**File:** BiaxialChart.js

### Syntax

```
JavaScript
get showYRangeSelector() {}
```

#### Property Value

Type: Boolean

### See Also

BiaxialChart Members
BiaxialChart Class
MindFusion.Charting.Controls Namespace

## BiaxialChart.showYTicks Property

Gets or sets a value indicating whether to show Y axis ticks.

**Namespace:** MindFusion.Charting.Controls
**File:** BiaxialChart.js

## Syntax

```
JavaScript
get showYTicks() {}
```

### Property Value

Type: Boolean

## See Also

BiaxialChart Members
BiaxialChart Class
MindFusion.Charting.Controls Namespace

# BiaxialChart.xAxis Property

Gets or sets an Axis object representing horizontal data range.

**Namespace:** MindFusion.Charting.Controls
**File:** BiaxialChart.js

## Syntax

```
JavaScript
get xAxis() {}
```

### Property Value

Type: Axis

## See Also

BiaxialChart Members
BiaxialChart Class
MindFusion.Charting.Controls Namespace

# BiaxialChart.xAxisLabelRotationAngle Property

Gets or sets the rotation angle of the X axis labels.

**Namespace:** MindFusion.Charting.Controls
**File:** BiaxialChart.js

## Syntax

```
JavaScript
get xAxisLabelRotationAngle() {}
```

### Property Value

Type: Number

## See Also

BiaxialChart Members
BiaxialChart Class
MindFusion.Charting.Controls Namespace

# BiaxialChart.xAxisTickLength Property

Gets or sets the length of X axis ticks.

**Namespace:** MindFusion.Charting.Controls
**File:** BiaxialChart.js

## Syntax

**JavaScript**

```
get xAxisTickLength() {}
```

### Property Value

Type: Number

## See Also

BiaxialChart Members
BiaxialChart Class
MindFusion.Charting.Controls Namespace

# BiaxialChart.xScrollRangeMax Property

Gets or sets the largest value allowed to scroll to using the X axis' RangeSelector.

**Namespace:** MindFusion.Charting.Controls
**File:** BiaxialChart.js

## Syntax

**JavaScript**

```
get xScrollRangeMax() {}
```

### Property Value

Type: Number

## See Also

BiaxialChart Members
BiaxialChart Class
MindFusion.Charting.Controls Namespace

## BiaxialChart.xScrollRangeMin Property

Gets or sets the smallest value allowed to scroll to using the X axis' RangeSelector.

**Namespace:** MindFusion.Charting.Controls
**File:** BiaxialChart.js

## Syntax

**JavaScript**

```
get xScrollRangeMin() {}
```

### Property Value

Type: Number

## See Also

BiaxialChart Members
BiaxialChart Class
MindFusion.Charting.Controls Namespace

## BiaxialChart.yAxis Property

Gets or sets an Axis object representing vertical data range.

**Namespace:** MindFusion.Charting.Controls
**File:** BiaxialChart.js

## Syntax

**JavaScript**

```
get yAxis() {}
```

### Property Value

Type: Axis

## See Also

BiaxialChart Members
BiaxialChart Class
MindFusion.Charting.Controls Namespace

## BiaxialChart.yAxisLabelRotationAngle Property

Gets or sets the rotation angle of the Y axis labels.

**Namespace:** MindFusion.Charting.Controls
**File:** BiaxialChart.js

## Syntax

**JavaScript**

```
get yAxisLabelRotationAngle() {}
```

### Property Value

Type: Number

## See Also

BiaxialChart Members
BiaxialChart Class
MindFusion.Charting.Controls Namespace

# BiaxialChart.yAxisTickLength Property

Gets or sets the length of Y axis ticks.

**Namespace:** MindFusion.Charting.Controls
**File:** BiaxialChart.js

## Syntax

**JavaScript**

```
get yAxisTickLength() {}
```

### Property Value

Type: Number

## See Also

BiaxialChart Members
BiaxialChart Class
MindFusion.Charting.Controls Namespace

# BiaxialChart.yLabelAlignment Property

Gets the horizontal alignment of Y-axis labels.

**Namespace:** MindFusion.Charting.Controls
**File:** BiaxialChart.js

## Syntax

**JavaScript**

```
get yLabelAlignment() {}
```

### Property Value

Type: StringAlignment

## Example

The following code aligns the label of the Y-axis "Near", which means the labels will be drawn to the left size of the Y-axis.

```
JavaScript
lineChart.yLabelAlignment = Drawing.StringAlignment.Near;
```

## See Also

BiaxialChart Members
BiaxialChart Class
MindFusion.Charting.Controls Namespace

# BiaxialChart.yScrollRangeMax Property

Gets or sets the largest value allowed to scroll to using the Y axis' RangeSelector.

**Namespace:** MindFusion.Charting.Controls
**File:** BiaxialChart.js

## Syntax

```
JavaScript
get yScrollRangeMax() {}
```

### Property Value

Type: Number

## See Also

BiaxialChart Members
BiaxialChart Class
MindFusion.Charting.Controls Namespace

# BiaxialChart.yScrollRangeMin Property

Gets or sets the smallest value allowed to scroll to using the Y axis' RangeSelector.

**Namespace:** MindFusion.Charting.Controls
**File:** BiaxialChart.js

## Syntax

```
JavaScript
get yScrollRangeMin() {}
```

### Property Value

Type: Number

## See Also

BiaxialChart Members
BiaxialChart Class
MindFusion.Charting.Controls Namespace

# BubbleChart Class

A control used to draw bubble charts.

## Syntax

```
// class
BubbleChart.prototype = {}
```

## Inheritance Hierarchy

## See Also

BubbleChart Members
MindFusion.Charting.Controls Namespace

# BubbleChart Members

A control used to draw bubble charts.

The following tables list the members exposed by the BubbleChart type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | BubbleChart | Initializes a new instance of the BubbleChart class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | labelAlignment | Gets or sets the alignment of labels relative to their associated bubbles. |
|  | series | BiaxialChart.Series override. Gets or sets the list of series whose data is drawn in this chart. |

## See Also

BubbleChart Class
MindFusion.Charting.Controls Namespace

# BubbleChart Constructor

Initializes a new instance of the BubbleChart class.

**Namespace:** MindFusion.Charting.Controls
**File:** BubbleChart.js

## Syntax

```
JavaScript
function BubbleChart (element, [seriesRenderer])
```

### Parameters

*element*
Type: HTMLCanvasElement
The canvas DOM element to associate this chart with.

*seriesRenderer*
Optional.
Type: BubbleRenderer
A LineRenderer used to draw chart's data series.

## See Also

BubbleChart Class
MindFusion.Charting.Controls Namespace

# BubbleChart Properties

## BubbleChart.labelAlignment Property

Gets or sets the alignment of labels relative to their associated bubbles.

**Namespace:** MindFusion.Charting.Controls
**File:** BubbleChart.js

## Syntax

```
JavaScript
get labelAlignment() {}
```

### Property Value

Type: BubbleLabelAlignment

## See Also

BubbleChart Members
BubbleChart Class
MindFusion.Charting.Controls Namespace

## BubbleChart.series Property

BiaxialChart.Series override. Gets or sets the list of series whose data is drawn in this chart.

**Namespace:** MindFusion.Charting.Controls
**File:** BubbleChart.js

## Syntax

```
JavaScript
```

```
get series() {}
```

### Property Value

Type: ObservableCollection

## See Also

BubbleChart Members
BubbleChart Class
MindFusion.Charting.Controls Namespace

# CandlestickChart Class

A control used to draw candlestick charts.

## Syntax

**JavaScript**

```
// class
CandlestickChart.prototype = {}
```

## Inheritance Hierarchy

## See Also

CandlestickChart Members
MindFusion.Charting.Controls Namespace

# CandlestickChart Members

A control used to draw candlestick charts.

The following tables list the members exposed by the CandlestickChart type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | CandlestickChart | Initializes a new instance of the CandlestickChart class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | candlestickWidth | Gets or sets the width of the candlesticks. |
|  | series | BiaxialChart.Series override. Gets or sets the list of series whose data is drawn in this chart. |

## See Also

CandlestickChart Class
MindFusion.Charting.Controls Namespace

# CandlestickChart Constructor

Initializes a new instance of the CandlestickChart class.

**Namespace:** MindFusion.Charting.Controls
**File:** CandlestickChart.js

## Syntax

**JavaScript**

```
function CandlestickChart (element, [seriesRenderer])
```

### Parameters

*element*

Type: HTMLCanvasElement
The canvas DOM element to associate this chart with.

*seriesRenderer*

Optional.
Type: CandlestickRenderer
A CandlestickRenderer used to draw chart's data series.

## See Also

CandlestickChart Class
MindFusion.Charting.Controls Namespace

# CandlestickChart Properties

# CandlestickChart.candlestickWidth Property

Gets or sets the width of the candlesticks.

**Namespace:** MindFusion.Charting.Controls
**File:** CandlestickChart.js

## Syntax

**JavaScript**

```
get candlestickWidth() {}
```

### Property Value

Type: Number

## See Also

CandlestickChart Members
CandlestickChart Class
MindFusion.Charting.Controls Namespace

## CandlestickChart.series Property

BiaxialChart.Series override. Gets or sets the list of series whose data is drawn in this chart.

**Namespace:** MindFusion.Charting.Controls
**File:** CandlestickChart.js

### Syntax

| JavaScript |
| --- |
| `get series() {}` |

#### Property Value

Type: ObservableCollection

### See Also

CandlestickChart Members
CandlestickChart Class
MindFusion.Charting.Controls Namespace

# Chart Class

A base class for chart controls.

### Syntax

| JavaScript |
| --- |
| `// class`<br>`Chart.prototype = {}` |

### Inheritance Hierarchy

### See Also

Chart Members
MindFusion.Charting.Controls Namespace

## Chart Members

A base class for chart controls.

The following tables list the members exposed by the Chart type.

### Public Constructors

| | Name | Description |
| --- | --- | --- |
| | Chart | Initializes a new instance of the Chart class. |

### Public Properties

| | Name | Description |
|---|---|---|
| | allowMoveLegend | Gets or sets a value indicating whether users are allowed to move the legend. |
| | chartPanel | Gets or sets the GridPanel that contains the chart's PlotPanel and axis renderers. |
| | dataBoundSeries | Contains series generated from DataSource data. |
| | dataSource | Gets or sets the chart's data source. |
| | innerLabelsDataFields | Gets or sets the names of fields in the data source whose values are shown as inner labels of rendered graphical elements. |
| | legendHorizontalAlignment | Gets or sets the horizontal alignment of the legend. |
| | legendMargin | Gets or sets the margin space around the legend. |
| | legendRenderer | Gets the LegendRenderer component used to draw the chart's legend. |
| | legendSpacing | Gets the distance between adjacent entries in the legend. |
| | legendTitle | Gets or sets the legend title. |
| | legendVerticalAlignment | Gets or sets the vertical alignment of the legend. |
| | outerLabelsDataFields | Gets or sets the names of fields in the data source whose values are shown as outer labels of rendered graphical elements. |
| | plot | Gets the chart's Plot component. |
| | plotImageAlign | Gets or sets the alignment of PlotImage relatively to the plot boundaries. |
| | plotImageAutoSize | Gets or sets whether auto-size mode is enabled for PlotImage. |
| | plotImageLocation | Gets or sets the Image drawn inside chart's plot area. |
| | plotPanel | Gets the Panel that contains the chart's plot and associated elements. |
| | showDataLabels | Gets or sets what kind of labels from data series should be drawn. |
| | showHighlight | Gets a value indicating whether to show highlights on data items. |
| | showLegend | Gets or sets a value indicating whether the chart should draw a legend. |
| | showLegendTitle | Gets a value indicating whether to display the legend title. |
| | showToolTips | Gets a value indicating whether to show tooltips. |
| | showZoomWidgets | Gets or sets a value indicating whether the plot should show zoom buttons. |
| | subtitle | Gets or sets the sub-title of this chart. |
| | subtitleBrush | Gets or sets the Brush used to draw chart's Subtitle. |

| | | |
|---|---|---|
| | subtitleFontName | Gets or sets the name of Font used to draw chart's Subtitle. |
| | subtitleFontSize | Gets or sets the size of Font used to draw chart's Subtitle. |
| | subtitleFontStyle | Gets or sets the style of Font used to draw chart's Subtitle. |
| | subtitleMargin | Gets or sets the margin space around Subtitle. |
| | title | Gets or sets the title of this chart. |
| | titleBrush | Gets or sets the Brush used to draw chart's Title. |
| | titleFontName | Gets or sets the name of Font used to draw chart's Title. |
| | titleFontSize | Gets or sets the size of Font used to draw chart's Title. |
| | titleFontStyle | Gets or sets the style of Font used to draw chart's Title. |
| | titleMargin | Gets or sets the margin space around Title. |
| | toolTipsDataFields | Gets or sets the names of fields in the data source whose values are shown as tooltips of rendered graphical elements. |
| | xAxisLabelsDataFields | Gets or sets the names of fields in the data source whose values are shown as X axis labels of rendered graphical elements. |
| | xDataFields | Gets or sets the names of fields in the data source whose values are assigned to X coordinates of rendered graphical elements. |
| | yAxisLabelsDataFields | Gets or sets the names of fields in the data source whose values are shown as Y axis labels of rendered graphical elements. |
| | yDataFields | Gets or sets the names of fields in the data source whose values are assigned to Y coordinates of rendered graphical elements. |
| | zAxisLabelsDataFields | Gets or sets the names of fields in the data source whose values are shown as Z axis labels of rendered graphical elements. |
| | zDataFields | Gets or sets the names of fields in the data source whose values are assigned to Z coordinates of rendered graphical elements. |

## Public Methods

| | Name | Description |
|---|---|---|
| | createPlot | Creates the Plot type corresponding to this chart. |
| | onPaddingChanged | Control.OnPaddingChanged override. |
| | onSizeChanged | Control.OnSizeChanged override. |
| | resetZoom | Resets zoom level to original data range. |
| | zoomOut | Zooms out of current data range. |

## See Also

Chart Class
MindFusion.Charting.Controls Namespace

## Chart Constructor

Initializes a new instance of the Chart class.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

### Syntax

```
JavaScript
function Chart (element, [renderer])
```

#### Parameters

*element*
Type: HTMLCanvasElement
The canvas DOM element to associate this chart with.

*renderer*
Optional.
Type: SeriesRenderer
A SeriesRenderer used to draw chart's data series.

### See Also

Chart Class
MindFusion.Charting.Controls Namespace

## Chart Methods

## Chart.createPlot Method

Creates the Plot type corresponding to this chart.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

### Syntax

```
JavaScript
function createPlot ()
```

#### Return Value

Type: Plot
An instance of Plot -derived class.

### See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

## Chart.onPaddingChanged Method

Control.OnPaddingChanged override.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

### Syntax

**JavaScript**

```
function onPaddingChanged (e)
```

#### Parameters

*e*
Type: EventArgs
An EventArgs instance.

### See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

## Chart.onSizeChanged Method

Control.OnSizeChanged override.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

### Syntax

**JavaScript**

```
function onSizeChanged (e)
```

#### Parameters

*e*
Type: EventArgs
An EventArgs instance.

### See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

## Chart.resetZoom Method

Resets zoom level to original data range.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

### Syntax

**JavaScript**

```
function resetZoom ()
```

### See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

## Chart.zoomOut Method

Zooms out of current data range.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

### Syntax

**JavaScript**

```
function zoomOut ()
```

### See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

# Chart Properties

## Chart.allowMoveLegend Property

Gets or sets a value indicating whether users are allowed to move the legend.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

### Syntax

**JavaScript**

```
get allowMoveLegend() {}
```

#### Property Value

Type: Boolean

## See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

# Chart.chartPanel Property

Gets or sets the GridPanel that contains the chart's PlotPanel and axis renderers.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

## Syntax

**JavaScript**

```
get chartPanel() {}
```

### Property Value

Type: GridPanel

## See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

# Chart.dataBoundSeries Property

Contains series generated from DataSource data.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

## Syntax

**JavaScript**

```
get dataBoundSeries() {}
```

### Property Value

Type: List

## See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

# Chart.dataSource Property

Gets or sets the chart's data source.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

## Syntax

```
JavaScript
get dataSource() {}
```

### Property Value

Type: Object

## See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

# Chart.innerLabelsDataFields Property

Gets or sets the names of fields in the data source whose values are shown as inner labels of rendered graphical elements.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

## Syntax

```
JavaScript
get innerLabelsDataFields() {}
```

### Property Value

Type: ObservableCollection

## See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

# Chart.legendHorizontalAlignment Property

Gets or sets the horizontal alignment of the legend.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

## Syntax

```
JavaScript
get legendHorizontalAlignment() {}
```

### Property Value

Type: LayoutAlignment

## See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

# Chart.legendMargin Property

Gets or sets the margin space around the legend.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

## Syntax

**JavaScript**

```
get legendMargin() {}
```

### Property Value

Type: Margins

## See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

# Chart.legendRenderer Property

Gets the LegendRenderer component used to draw the chart's legend.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

## Syntax

**JavaScript**

```
get legendRenderer() {}
```

### Property Value

Type: LegendRenderer

## See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

## Chart.legendSpacing Property

Gets the distance between adjacent entries in the legend.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

### Syntax

```
JavaScript
get legendSpacing() {}
```

#### Property Value

Type: Number

### See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

## Chart.legendTitle Property

Gets or sets the legend title.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

### Syntax

```
JavaScript
get legendTitle() {}
```

#### Property Value

Type: String

### See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

## Chart.legendVerticalAlignment Property

Gets or sets the vertical alignment of the legend.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

### Syntax

```
JavaScript
```

```
get legendVerticalAlignment() {}
```

### Property Value

Type: LayoutAlignment

## See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

# Chart.outerLabelsDataFields Property

Gets or sets the names of fields in the data source whose values are shown as outer labels of rendered graphical elements.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

## Syntax

**JavaScript**

```
get outerLabelsDataFields() {}
```

### Property Value

Type: ObservableCollection

## See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

# Chart.plot Property

Gets the chart's Plot component.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

## Syntax

**JavaScript**

```
get plot() {}
```

### Property Value

Type: Plot

## See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

## Chart.plotImageAlign Property

Gets or sets the alignment of PlotImage relatively to the plot boundaries.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

### Syntax

```
JavaScript
get plotImageAlign() {}
```

#### Property Value

Type: ImageAlign

### See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

## Chart.plotImageAutoSize Property

Gets or sets whether auto-size mode is enabled for PlotImage.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

### Syntax

```
JavaScript
get plotImageAutoSize() {}
```

#### Property Value

Type: Boolean

### See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

## Chart.plotImageLocation Property

Gets or sets the Image drawn inside chart's plot area.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

## Syntax

```
JavaScript
get plotImageLocation() {}
```

### Property Value

Type: String

## See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

# Chart.plotPanel Property

Gets the Panel that contains the chart's plot and associated elements.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

## Syntax

```
JavaScript
get plotPanel() {}
```

### Property Value

Type: Panel

## See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

# Chart.showDataLabels Property

Gets or sets what kind of labels from data series should be drawn.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

## Syntax

```
JavaScript
get showDataLabels() {}
```

### Property Value

Type: Boolean

## See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

# Chart.showHighlight Property

Gets a value indicating whether to show highlights on data items.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

## Syntax

**JavaScript**

```
get showHighlight() {}
```

### Property Value

Type: Boolean

## See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

# Chart.showLegend Property

Gets or sets a value indicating whether the chart should draw a legend.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

## Syntax

**JavaScript**

```
get showLegend() {}
```

### Property Value

Type: Boolean

## See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

# Chart.showLegendTitle Property

Gets a value indicating whether to display the legend title.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

## Syntax

```
JavaScript
get showLegendTitle() {}
```

### Property Value

Type: Boolean

## See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

# Chart.showToolTips Property

Gets a value indicating whether to show tooltips.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

## Syntax

```
JavaScript
get showToolTips() {}
```

### Property Value

Type: Boolean

## See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

# Chart.showZoomWidgets Property

Gets or sets a value indicating whether the plot should show zoom buttons.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

## Syntax

```
JavaScript
get showZoomWidgets() {}
```

### Property Value

Type: Boolean

## See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

# Chart.subtitle Property

Gets or sets the sub-title of this chart.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

## Syntax

| JavaScript |
| --- |
| `get subtitle() {}` |

### Property Value

Type: String

## See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

# Chart.subtitleBrush Property

Gets or sets the Brush used to draw chart's Subtitle.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

## Syntax

| JavaScript |
| --- |
| `get subtitleBrush() {}` |

### Property Value

Type: Brush

## See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

## Chart.subtitleFontName Property

Gets or sets the name of Font used to draw chart's Subtitle.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

### Syntax

```
JavaScript
get subtitleFontName() {}
```

#### Property Value

Type: String

### See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

## Chart.subtitleFontSize Property

Gets or sets the size of Font used to draw chart's Subtitle.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

### Syntax

```
JavaScript
get subtitleFontSize() {}
```

#### Property Value

Type: Number

### See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

## Chart.subtitleFontStyle Property

Gets or sets the style of Font used to draw chart's Subtitle.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

### Syntax

```
JavaScript
```

```
get subtitleFontStyle() {}
```

### Property Value

Type: FontStyle

## See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

# Chart.subtitleMargin Property

Gets or sets the margin space around Subtitle.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

## Syntax

**JavaScript**

```
get subtitleMargin() {}
```

### Property Value

Type: Margins

## See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

# Chart.title Property

Gets or sets the title of this chart.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

## Syntax

**JavaScript**

```
get title() {}
```

### Property Value

Type: String

## See Also

Chart Members
Chart Class

MindFusion.Charting.Controls Namespace

## Chart.titleBrush Property

Gets or sets the Brush used to draw chart's Title.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

### Syntax

```
JavaScript
get titleBrush() {}
```

#### Property Value

Type: Brush

### See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

## Chart.titleFontName Property

Gets or sets the name of Font used to draw chart's Title.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

### Syntax

```
JavaScript
get titleFontName() {}
```

#### Property Value

Type: String

### See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

## Chart.titleFontSize Property

Gets or sets the size of Font used to draw chart's Title.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

### Syntax

```
JavaScript
get titleFontSize() {}
```

### Property Value

Type: Number

## See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

# Chart.titleFontStyle Property

Gets or sets the style of Font used to draw chart's Title.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

## Syntax

```
JavaScript
get titleFontStyle() {}
```

### Property Value

Type: FontStyle

## See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

# Chart.titleMargin Property

Gets or sets the margin space around Title.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

## Syntax

```
JavaScript
get titleMargin() {}
```

### Property Value

Type: Margins

## See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

## Chart.toolTipsDataFields Property

Gets or sets the names of fields in the data source whose values are shown as tooltips of rendered graphical elements.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

### Syntax

```
JavaScript
get toolTipsDataFields() {}
```

#### Property Value

Type: ObservableCollection

### See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

## Chart.xAxisLabelsDataFields Property

Gets or sets the names of fields in the data source whose values are shown as X axis labels of rendered graphical elements.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

### Syntax

```
JavaScript
get xAxisLabelsDataFields() {}
```

#### Property Value

Type: ObservableCollection

### See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

## Chart.xDataFields Property

Gets or sets the names of fields in the data source whose values are assigned to X coordinates of rendered graphical elements.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

## Syntax

```
JavaScript
get xDataFields() {}
```

### Property Value

Type: ObservableCollection

## See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

# Chart.yAxisLabelsDataFields Property

Gets or sets the names of fields in the data source whose values are shown as Y axis labels of rendered graphical elements.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

## Syntax

```
JavaScript
get yAxisLabelsDataFields() {}
```

### Property Value

Type: ObservableCollection

## See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

# Chart.yDataFields Property

Gets or sets the names of fields in the data source whose values are assigned to Y coordinates of rendered graphical elements.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

## Syntax

```
JavaScript
get yDataFields() {}
```

### Property Value

Type: ObservableCollection

## See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

# Chart.zAxisLabelsDataFields Property

Gets or sets the names of fields in the data source whose values are shown as Z axis labels of rendered graphical elements.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

## Syntax

**JavaScript**

```
get zAxisLabelsDataFields() {}
```

### Property Value

Type: ObservableCollection

## See Also

Chart Members
Chart Class
MindFusion.Charting.Controls Namespace

# Chart.zDataFields Property

Gets or sets the names of fields in the data source whose values are assigned to Z coordinates of rendered graphical elements.

**Namespace:** MindFusion.Charting.Controls
**File:** Chart.js

## Syntax

**JavaScript**

```
get zDataFields() {}
```

### Property Value

Type: ObservableCollection

## See Also

# ChartBase Class

A control that contains multiple chart plots and gauges

## Syntax

```
// class
ChartBase.prototype = {}
```

## Inheritance Hierarchy

## See Also

ChartBase Members
MindFusion.Charting.Controls Namespace

# ChartBase Members

A control that contains multiple chart plots and gauges

The following tables list the members exposed by the ChartBase type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | ChartBase | Initializes a new instance of the **ChartBase** class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | allowZoom | Gets or sets a value indicating whether users are allowed to zoom into plots' data ranges. |
|  | backColor | Gets or sets dashboard's back color. |
|  | backgroundImageAlign | Gets or sets the alignment of BackgroundImage relatively to dashboard boundaries. |
|  | backgroundImageAutoSize | Gets or sets whether auto-size mode is enabled for BackgroundImage. |
|  | backgroundImageLocation | Gets or sets dashboard's background image. |
|  | dataItemClicked | Raised when the user clicks on a data item inside a plot. |
|  | layoutBuilder | Gets a LayoutBuilder instance that provides shortcut methods for building fragments of dashboard's user interface. |

| | | |
|---|---|---|
| | layoutPanel | Gets a Panel containing dashboard components that should participate in layout measurements and be arranged relatively to each other. |
| | licenseKey | Gets or sets the license key of the control. |
| | licenseLocation | Gets or sets the path to the license file. |
| | rootPanel | Gets the root Panel in the hierarchy of dashboard components. |
| | theme | Gets or sets a Theme specifying appearance of dashboard elements. |

## Public Methods

| | Name | Description |
|---|---|---|
| | createRenderContext | Creates a RenderContext instance. |
| | dataBind | Binds the chart to current DataSource. |
| | draw | Draws the dashboard contents. |
| | fromJson | Deserializes the control's data from JSON string. |
| | invalidate | Implements RootControl.InvalidateLayout. Invalidates the specified region of a component. |
| | invalidateLayout | Implements RootControl.InvalidateLayout. Invalidates layout of specified component. |
| | onClick | Raises the DataItemClicked event if the user has clicked on data element in a plot. |
| | onMouseDown | Handles the HTMLCanvasElement mousedown event. |
| | onMouseMove | Handles the HTMLCanvasElement mousemove event. |
| | onMouseUp | Handles the HTMLCanvasElement mouseup event. |
| | onResize | Control.OnResize override. Invalidates the layout of child components and runs a new layout pass. |
| | toJson | Serializes the control's data to JSON string. |

## See Also

ChartBase Class
MindFusion.Charting.Controls Namespace

### ChartBase Constructor

Initializes a new instance of the ChartBase class.

**Namespace:** MindFusion.Charting.Controls
**File:** Dashboard.js

## Syntax

JavaScript

```
function ChartBase (element)
```

### Parameters

*element*
Type: HTMLCanvasElement
The canvas DOM element to associate this dashboard with.

## See Also

ChartBase Class
MindFusion.Charting.Controls Namespace

# ChartBase Methods

## ChartBase.createRenderContext Method

Creates a RenderContext instance.

**Namespace:** MindFusion.Charting.Controls
**File:** Dashboard.js

## Syntax

JavaScript

```
function createRenderContext (graphics, clipRect)
```

### Parameters

*graphics*
Type: Graphics
A Graphics surface where dashboard elements should be rendered.

*clipRect*
Type: Rect
The current clip rectangle.

### Return Value

Type: RenderContext
A RenderContext instance.

## See Also

ChartBase Members
ChartBase Class
MindFusion.Charting.Controls Namespace

## ChartBase.dataBind Method

Binds the chart to current DataSource.

**Namespace:** MindFusion.Charting.Controls
**File:** Dashboard.js

## Syntax

```
JavaScript
function dataBind ()
```

## See Also

ChartBase Members
ChartBase Class
MindFusion.Charting.Controls Namespace

# ChartBase.draw Method

Draws the dashboard contents.

**Namespace:** MindFusion.Charting.Controls
**File:** Dashboard.js

## Syntax

```
JavaScript
function draw ()
```

## See Also

ChartBase Members
ChartBase Class
MindFusion.Charting.Controls Namespace

# ChartBase.fromJson Method

Deserializes the control's data from JSON string.

**Namespace:** MindFusion.Charting.Controls
**File:** Dashboard.js

## Syntax

```
JavaScript
function fromJson (json)
```

### Parameters

*json*
Type: void

## See Also

ChartBase Members
ChartBase Class
MindFusion.Charting.Controls Namespace

# ChartBase.invalidate Method

Implements RootControl.InvalidateLayout. Invalidates the specified region of a component.

**Namespace:** MindFusion.Charting.Controls
**File:** Dashboard.js

## Syntax

**JavaScript**

```
function invalidate ([rect, [panel]])
```

### Parameters

*rect*
Optional.
Type: RectD
The area to invalidate and redraw.

*panel*
Optional.
Type: Component
The reference Component.

## See Also

ChartBase Members
ChartBase Class
MindFusion.Charting.Controls Namespace

# ChartBase.invalidateLayout Method

Implements RootControl.InvalidateLayout. Invalidates layout of specified component.

**Namespace:** MindFusion.Charting.Controls
**File:** Dashboard.js

## Syntax

**JavaScript**

```
function invalidateLayout (panel)
```

### Parameters

*panel*
Type: Component
The component to invalidate.

## See Also

ChartBase Members
ChartBase Class
MindFusion.Charting.Controls Namespace

# ChartBase.onClick Method

Raises the DataItemClicked event if the user has clicked on data element in a plot.

**Namespace:** MindFusion.Charting.Controls
**File:** Dashboard.js

## Syntax

```
JavaScript
function onClick (e)
```

### Parameters

*e*
Type: MouseEvent
A MouseEvent instance.

## See Also

ChartBase Members
ChartBase Class
MindFusion.Charting.Controls Namespace

# ChartBase.onMouseDown Method

Handles the HTMLCanvasElement mousedown event.

**Namespace:** MindFusion.Charting.Controls
**File:** Dashboard.js

## Syntax

```
JavaScript
function onMouseDown (e)
```

### Parameters

*e*
Type: MouseEvent
A MouseEvent instance.

## See Also

ChartBase Members
ChartBase Class
MindFusion.Charting.Controls Namespace

# ChartBase.onMouseMove Method

Handles the HTMLCanvasElement mousemove event.

**Namespace:** MindFusion.Charting.Controls
**File:** Dashboard.js

## Syntax

```
JavaScript
function onMouseMove (e)
```

### Parameters

*e*
Type: MouseEvent
A MouseEvent instance.

## See Also

ChartBase Members
ChartBase Class
MindFusion.Charting.Controls Namespace

# ChartBase.onMouseUp Method

Handles the HTMLCanvasElement mouseup event.

**Namespace:** MindFusion.Charting.Controls
**File:** Dashboard.js

## Syntax

```
JavaScript
function onMouseUp (e)
```

### Parameters

*e*
Type: MouseEvent
A MouseEvent instance.

## See Also

ChartBase Members
ChartBase Class
MindFusion.Charting.Controls Namespace

# ChartBase.onResize Method

Control.OnResize override. Invalidates the layout of child components and runs a new layout pass.

**Namespace:** MindFusion.Charting.Controls
**File:** Dashboard.js

## Syntax

```
JavaScript
function onResize (e)
```

### Parameters

*e*
Type: EventArgs
An EventArgs instance.

## See Also

ChartBase Members
ChartBase Class
MindFusion.Charting.Controls Namespace

# ChartBase.toJson Method

Serializes the control's data to JSON string.

**Namespace:** MindFusion.Charting.Controls
**File:** Dashboard.js

## Syntax

**JavaScript**

```
function toJson ()
```

## See Also

ChartBase Members
ChartBase Class
MindFusion.Charting.Controls Namespace

# ChartBase Properties

# ChartBase.allowZoom Property

Gets or sets a value indicating whether users are allowed to zoom into plots' data ranges.

**Namespace:** MindFusion.Charting.Controls
**File:** Dashboard.js

## Syntax

**JavaScript**

```
get allowZoom() {}
```

### Property Value

Type: Boolean

## See Also

ChartBase Members
ChartBase Class
MindFusion.Charting.Controls Namespace

## ChartBase.backColor Property

Gets or sets dashboard's back color.

**Namespace:** MindFusion.Charting.Controls
**File:** Dashboard.js

### Syntax

```
JavaScript
get backColor() {}
```

#### Property Value

Type: Color

### See Also

ChartBase Members
ChartBase Class
MindFusion.Charting.Controls Namespace

## ChartBase.backgroundImageAlign Property

Gets or sets the alignment of BackgroundImage relatively to dashboard boundaries.

**Namespace:** MindFusion.Charting.Controls
**File:** Dashboard.js

### Syntax

```
JavaScript
get backgroundImageAlign() {}
```

#### Property Value

Type: ImageAlign

### See Also

ChartBase Members
ChartBase Class
MindFusion.Charting.Controls Namespace

## ChartBase.backgroundImageAutoSize Property

Gets or sets whether auto-size mode is enabled for BackgroundImage.

**Namespace:** MindFusion.Charting.Controls
**File:** Dashboard.js

### Syntax

```
JavaScript
```

```
get backgroundImageAutoSize() {}
```

### Property Value

Type: Boolean

## See Also

ChartBase Members
ChartBase Class
MindFusion.Charting.Controls Namespace

# ChartBase.backgroundImageLocation Property

Gets or sets dashboard's background image.

**Namespace:** MindFusion.Charting.Controls
**File:** Dashboard.js

## Syntax

**JavaScript**

```
get backgroundImageLocation() {}
```

### Property Value

Type: String

## See Also

ChartBase Members
ChartBase Class
MindFusion.Charting.Controls Namespace

# ChartBase.dataItemClicked Property

Raised when the user clicks on a data item inside a plot.

**Namespace:** MindFusion.Charting.Controls
**File:** Dashboard.js

## Syntax

**JavaScript**

```
get dataItemClicked() {}
```

### Property Value

Type: EventDispatcher

## See Also

ChartBase Members
ChartBase Class

MindFusion.Charting.Controls Namespace

## ChartBase.layoutBuilder Property

Gets a LayoutBuilder instance that provides shortcut methods for building fragments of dashboard's user interface.

**Namespace:** MindFusion.Charting.Controls
**File:** Dashboard.js

### Syntax

```
JavaScript
get layoutBuilder() {}
```

#### Property Value

Type: LayoutBuilder

### See Also

ChartBase Members
ChartBase Class
MindFusion.Charting.Controls Namespace

## ChartBase.layoutPanel Property

Gets a Panel containing dashboard components that should participate in layout measurements and be arranged relatively to each other.

**Namespace:** MindFusion.Charting.Controls
**File:** Dashboard.js

### Syntax

```
JavaScript
get layoutPanel() {}
```

#### Property Value

Type: StackPanel

### See Also

ChartBase Members
ChartBase Class
MindFusion.Charting.Controls Namespace

## ChartBase.licenseKey Property

Gets or sets the license key of the control.

**Namespace:** MindFusion.Charting.Controls
**File:** Dashboard.js

## Syntax

```
JavaScript
get licenseKey() {}
```

### Property Value

Type: String

## See Also

ChartBase Members
ChartBase Class
MindFusion.Charting.Controls Namespace

# ChartBase.licenseLocation Property

Gets or sets the path to the license file.

**Namespace:** MindFusion.Charting.Controls
**File:** Dashboard.js

## Syntax

```
JavaScript
get licenseLocation() {}
```

### Property Value

A string specifying license file URL.

## See Also

ChartBase Members
ChartBase Class
MindFusion.Charting.Controls Namespace

# ChartBase.rootPanel Property

Gets the root Panel in the hierarchy of dashboard components.

**Namespace:** MindFusion.Charting.Controls
**File:** Dashboard.js

## Syntax

```
JavaScript
get rootPanel() {}
```

### Property Value

Type: Panel

## See Also

ChartBase Members
ChartBase Class
MindFusion.Charting.Controls Namespace

### ChartBase.theme Property

Gets or sets a Theme specifying appearance of dashboard elements.

**Namespace:** MindFusion.Charting.Controls
**File:** Dashboard.js

## Syntax

**JavaScript**

```
get theme() {}
```

#### Property Value

Type: Theme

## See Also

ChartBase Members
ChartBase Class
MindFusion.Charting.Controls Namespace

# FunnelChart Class

A control used to draw funnel charts. Funnel charts represent the stages of a process and show relative values or measurements associated with each stage.

## Syntax

**JavaScript**

```
// class
FunnelChart.prototype = {}
```

## Inheritance Hierarchy

## See Also

FunnelChart Members
MindFusion.Charting.Controls Namespace

### FunnelChart Members

A control used to draw funnel charts. Funnel charts represent the stages of a process and show relative values or measurements associated with each stage.

The following tables list the members exposed by the FunnelChart type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | FunnelChart | Initializes a new instance of the FunnelChart class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | dimension | Gets or sets the dimension index that should be used to read data from the Series. |
| | segmentSpacing | Gets or sets the spacing between segments. |
| | series | Gets or sets the series whose data is drawn in this chart. |
| | stemWidth | Gets or sets the width of the funnel stem. |

## See Also

FunnelChart Class
MindFusion.Charting.Controls Namespace

# FunnelChart Constructor

Initializes a new instance of the FunnelChart class.

**Namespace:** MindFusion.Charting.Controls
**File:** FunnelChart.js

## Syntax

**JavaScript**

```
function FunnelChart (element, [seriesRenderer])
```

### Parameters

*element*
Type: HTMLCanvasElement
The canvas DOM element to associate this chart with.

*seriesRenderer*
Optional.
Type: FunnelRenderer
A FunnelRenderer used to draw chart's data series.

## See Also

FunnelChart Class
MindFusion.Charting.Controls Namespace

# FunnelChart Properties

# FunnelChart.dimension Property

Gets or sets the dimension index that should be used to read data from the Series.

**Namespace:** MindFusion.Charting.Controls
**File:** FunnelChart.js

## Syntax

```
JavaScript
get dimension() {}
```

### Property Value

Type: Number

## See Also

FunnelChart Members
FunnelChart Class
MindFusion.Charting.Controls Namespace

# FunnelChart.segmentSpacing Property

Gets or sets the spacing between segments.

**Namespace:** MindFusion.Charting.Controls
**File:** FunnelChart.js

## Syntax

```
JavaScript
get segmentSpacing() {}
```

### Property Value

Type: Number

## See Also

FunnelChart Members
FunnelChart Class
MindFusion.Charting.Controls Namespace

# FunnelChart.series Property

Gets or sets the series whose data is drawn in this chart.

**Namespace:** MindFusion.Charting.Controls
**File:** FunnelChart.js

## Syntax

```
JavaScript
```

```
get series() {}
```

### Property Value

Type: Series

## See Also

FunnelChart Members
FunnelChart Class
MindFusion.Charting.Controls Namespace

# FunnelChart.stemWidth Property

Gets or sets the width of the funnel stem.

**Namespace:** MindFusion.Charting.Controls
**File:** FunnelChart.js

## Syntax

**JavaScript**

```
get stemWidth() {}
```

### Property Value

Type: Number

## See Also

FunnelChart Members
FunnelChart Class
MindFusion.Charting.Controls Namespace

# LayoutBuilder Class

Provides shortcut methods for building fragments of dashboard's user interface.

## Syntax

**JavaScript**

```
// class
LayoutBuilder.prototype = {}
```

## Inheritance Hierarchy

## See Also

LayoutBuilder Members
MindFusion.Charting.Controls Namespace

# LayoutBuilder Members

Provides shortcut methods for building fragments of dashboard's user interface.

The following tables list the members exposed by the LayoutBuilder type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | LayoutBuilder | Initializes a new instance of the LayoutBuilder class. |

## Public Methods

| | Name | Description |
|---|---|---|
| | create1By2ColumnLayout | Creates a two-column stack panel whose first column is a vertical stack of two components. |
| | create1By2RowLayout | Creates a two-row stack panel whose second row is a horizontal stack of two components. |
| | create2By1ColumnLayout | Creates a two-column stack panel whose first column is a vertical stack of two components. |
| | create2By1RowLayout | Creates a two-row stack panel whose first row is a horizontal stack of two components. |
| | create2By2Layout | Creates horizontal stack panels for components on same row and adds them to a vertical stack panel. |
| | createAndAdd1By2ColumnLayout | Creates a two-column stack panel whose first column is a vertical stack of two components.The panel is automatically added to the dashboard's LayoutPanel. |
| | createAndAdd1By2RowLayout | Creates a two-row stack panel whose second row is a horizontal stack of two components.The panel is automatically added to the dashboard's LayoutPanel. |
| | createAndAdd2By1ColumnLayout | Creates a two-column stack panel whose first column is a vertical stack of two components.The panel is automatically added to the dashboard's LayoutPanel. |
| | createAndAdd2By1RowLayout | Creates a two-row stack panel whose first row is a horizontal stack of two components.The panel is automatically added to the dashboard's LayoutPanel. |
| | createAndAdd2By2Layout | Creates horizontal stack panels for components on same row and adds them to a vertical stack panel.The panel is automatically added to the dashboard's LayoutPanel. |
| | createAndAddPlotAndAxes | Creates a GridPanel containing the specified plot and axis renderers.The panel is automatically added to the dashboard's LayoutPanel. |

| | | |
|---|---|---|
| | createAndAddPlotWithBottomAndLeftAxes | Creates a GridPanel containing the specified plot and axis renderers.The panel is automatically added to the dashboard's LayoutPanel. |
| | createAndAddPlotWithBottomAndRightAxes | Creates a GridPanel containing the specified plot and axis renderers.The panel is automatically added to the dashboard's LayoutPanel. |
| | createAndAddPlotWithTopAndLeftAxes | Creates a GridPanel containing the specified plot and axis renderers.The panel is automatically added to the dashboard's LayoutPanel. |
| | createAndAddPlotWithTopAndRightAxes | Creates a GridPanel containing the specified plot and axis renderers.The panel is automatically added to the dashboard's LayoutPanel. |
| | createLayout | Creates a StackPanel for each array of components and adds it to a parent StackPanel with opposite orientation. |
| | createPlotAndAxes | Creates a GridPanel containing the specified plot and axis renderers. |
| | createPlotWithBottomAndLeftAxes | Creates a GridPanel containing the specified plot and axis renderers. |
| | createPlotWithBottomAndRightAxes | Creates a GridPanel containing the specified plot and axis renderers. |
| | createPlotWithTopAndLeftAxes | Creates a GridPanel containing the specified plot and axis renderers. |
| | createPlotWithTopAndRightAxes | Creates a GridPanel containing the specified plot and axis renderers. |

## See Also

LayoutBuilder Class
MindFusion.Charting.Controls Namespace

# LayoutBuilder Constructor

Initializes a new instance of the LayoutBuilder class.

**Namespace:** MindFusion.Charting.Controls
**File:** LayoutBuilder.js

## Syntax

**JavaScript**

```
function LayoutBuilder (board)
```

### Parameters

*board*
Type: Dashboard
A Dashboard instance.

## See Also

# LayoutBuilder Methods

## LayoutBuilder.create1By2ColumnLayout Method

Creates a two-column stack panel whose first column is a vertical stack of two components.

**Namespace:** MindFusion.Charting.Controls
**File:** LayoutBuilder.js

## Syntax

**JavaScript**

```
function create1By2ColumnLayout (col1, col2row1, col2row2)
```

### Parameters

*col1*
Type: Component
The component in first column.

*col2row1*
Type: Component
The first component in second column.

*col2row2*
Type: Component
The second component in second column.

### Return Value

Type: StackPanel
A StackPanel instance.

## See Also

LayoutBuilder Members
LayoutBuilder Class
MindFusion.Charting.Controls Namespace

## LayoutBuilder.create1By2RowLayout Method

Creates a two-row stack panel whose second row is a horizontal stack of two components.

**Namespace:** MindFusion.Charting.Controls
**File:** LayoutBuilder.js

## Syntax

**JavaScript**

```
function create1By2RowLayout (row1, row2col1, row2col2)
```

### Parameters

*row1*
Type: Component
The component on first row.

*row2col1*
Type: Component
The first component on second row.

*row2col2*
Type: Component
The second component on second row.

### Return Value

Type: StackPanel
A StackPanel instance.

## See Also

LayoutBuilder Members
LayoutBuilder Class
MindFusion.Charting.Controls Namespace

# LayoutBuilder.create2By1ColumnLayout Method

Creates a two-column stack panel whose first column is a vertical stack of two components.

**Namespace:** MindFusion.Charting.Controls
**File:** LayoutBuilder.js

## Syntax

**JavaScript**

```
function create2By1ColumnLayout (col1row1, col1row2, col2)
```

### Parameters

*col1row1*
Type: Component
The first component in first column.

*col1row2*
Type: Component
The second component in first column.

*col2*
Type: Component
The component in second column.

### Return Value

Type: StackPanel
A StackPanel instance.

## See Also

LayoutBuilder Members
LayoutBuilder Class
MindFusion.Charting.Controls Namespace

# LayoutBuilder.create2By1RowLayout Method

Creates a two-row stack panel whose first row is a horizontal stack of two components.

**Namespace:** MindFusion.Charting.Controls
**File:** LayoutBuilder.js

## Syntax

**JavaScript**

```
function create2By1RowLayout (row1col1, row1col2, row2)
```

### Parameters

*row1col1*
Type: Component
The first component on first row.

*row1col2*
Type: Component
The second component on first row.

*row2*
Type: Component
The component on first row.

### Return Value

Type: StackPanel
A StackPanel instance.

## See Also

LayoutBuilder Members
LayoutBuilder Class
MindFusion.Charting.Controls Namespace

# LayoutBuilder.create2By2Layout Method

Creates horizontal stack panels for components on same row and adds them to a vertical stack panel.

**Namespace:** MindFusion.Charting.Controls
**File:** LayoutBuilder.js

## Syntax

**JavaScript**

```
function create2By2Layout (row1col1, row1col2, row2col1, row2col2)
```

### Parameters

*row1col1*

Type: Component
The first component on first row.

*row1col2*
Type: Component
The second component on first row.

*row2col1*
Type: Component
The first component on second row.

*row2col2*
Type: Component
The second component on second row.

### Return Value

Type: StackPanel
A StackPanel instance.

## See Also

LayoutBuilder Members
LayoutBuilder Class
MindFusion.Charting.Controls Namespace

# LayoutBuilder.createAndAdd1By2ColumnLayout Method

Creates a two-column stack panel whose first column is a vertical stack of two components.The panel is automatically added to the dashboard's LayoutPanel.

**Namespace:** MindFusion.Charting.Controls
**File:** LayoutBuilder.js

## Syntax

**JavaScript**

```
function createAndAdd1By2ColumnLayout (col1, col2row1, col2row2)
```

### Parameters

*col1*
Type: Component
The component in first column.

*col2row1*
Type: Component
The first component in second column.

*col2row2*
Type: Component
The second component in second column.

### Return Value

Type: StackPanel
A StackPanel instance.

## See Also

LayoutBuilder Members
LayoutBuilder Class
MindFusion.Charting.Controls Namespace

## LayoutBuilder.createAndAdd1By2RowLayout Method

Creates a two-row stack panel whose second row is a horizontal stack of two components.The panel is automatically added to the dashboard's LayoutPanel.

**Namespace:** MindFusion.Charting.Controls
**File:** LayoutBuilder.js

### Syntax

**JavaScript**

```
function createAndAdd1By2RowLayout (row1, row2col1, row2col2)
```

#### Parameters

*row1*
Type: Component
The component on first row.

*row2col1*
Type: Component
The first component on second row.

*row2col2*
Type: Component
The second component on second row.

#### Return Value

Type: StackPanel
A StackPanel instance.

### See Also

LayoutBuilder Members
LayoutBuilder Class
MindFusion.Charting.Controls Namespace

## LayoutBuilder.createAndAdd2By1ColumnLayout Method

Creates a two-column stack panel whose first column is a vertical stack of two components.The panel is automatically added to the dashboard's LayoutPanel.

**Namespace:** MindFusion.Charting.Controls
**File:** LayoutBuilder.js

### Syntax

**JavaScript**

```
function createAndAdd2By1ColumnLayout (col1row1, col1row2, col2)
```

#### Parameters

*col1row1*
Type: Component
The first component in first column.

*col1row2*
Type: Component
The second component in first column.

*col2*
Type: Component
The component in second column.

### Return Value

Type: StackPanel
A StackPanel instance.

## See Also

LayoutBuilder Members
LayoutBuilder Class
MindFusion.Charting.Controls Namespace

# LayoutBuilder.createAndAdd2By1RowLayout Method

Creates a two-row stack panel whose first row is a horizontal stack of two components.The panel is automatically added to the dashboard's LayoutPanel.

**Namespace:** MindFusion.Charting.Controls
**File:** LayoutBuilder.js

## Syntax

**JavaScript**

```
function createAndAdd2By1RowLayout (row1col1, row1col2, row2)
```

### Parameters

*row1col1*
Type: Component
The first component on first row.

*row1col2*
Type: Component
The second component on first row.

*row2*
Type: Component
The component on first row.

### Return Value

Type: StackPanel
A StackPanel instance.

## See Also

LayoutBuilder Members
LayoutBuilder Class

# LayoutBuilder.createAndAdd2By2Layout Method

Creates horizontal stack panels for components on same row and adds them to a vertical stack panel.The panel is automatically added to the dashboard's LayoutPanel.

**Namespace:** MindFusion.Charting.Controls
**File:** LayoutBuilder.js

## Syntax

**JavaScript**

```
function createAndAdd2By2Layout (row1col1, row1col2, row2col1, row2col2)
```

### Parameters

*row1col1*

Type: Component
The first component on first row.

*row1col2*

Type: Component
The second component on first row.

*row2col1*

Type: Component
The first component on second row.

*row2col2*

Type: Component
The second component on second row.

### Return Value

Type: StackPanel
A StackPanel instance.

## See Also

LayoutBuilder Members
LayoutBuilder Class
MindFusion.Charting.Controls Namespace

# LayoutBuilder.createAndAddPlotAndAxes Method

Creates a GridPanel containing the specified plot and axis renderers.The panel is automatically added to the dashboard's LayoutPanel.

**Namespace:** MindFusion.Charting.Controls
**File:** LayoutBuilder.js

## Syntax

**JavaScript**

```
function createAndAddPlotAndAxes (plot, top, left, bottom, right)
```

### Parameters

*plot*
Type: Plot2D
A Plot2D instance.

*top*

*left*

*bottom*

*right*

### Return Value

Type: GridPanel
A GridPanel instance.

## See Also

LayoutBuilder Members
LayoutBuilder Class
MindFusion.Charting.Controls Namespace

# LayoutBuilder.createAndAddPlotWithBottomAndLeftAxes Method

Creates a GridPanel containing the specified plot and axis renderers.The panel is automatically added to the dashboard's LayoutPanel.

**Namespace:** MindFusion.Charting.Controls
**File:** LayoutBuilder.js

## Syntax

**JavaScript**

```
function createAndAddPlotWithBottomAndLeftAxes (plot, bottom, left)
```

### Parameters

*plot*
Type: Plot2D
A Plot2D instance.

*bottom*
Type: XAxisRenderer
An XAxisRenderer that should be placed at bottom side of the plot.

*left*
Type: YAxisRenderer
A YAxisRenderer that should be placed at left side of the plot.

### Return Value

Type: GridPanel
A GridPanel instance.

## See Also

LayoutBuilder Members
LayoutBuilder Class

## LayoutBuilder.createAndAddPlotWithBottomAndRightAxes Method

Creates a GridPanel containing the specified plot and axis renderers.The panel is automatically added to the dashboard's LayoutPanel.

**Namespace:** MindFusion.Charting.Controls
**File:** LayoutBuilder.js

# Syntax

**JavaScript**

```
function createAndAddPlotWithBottomAndRightAxes (plot, bottom, right)
```

### Parameters

*plot*
Type: Plot2D
A Plot2D instance.

*bottom*
Type: XAxisRenderer
An XAxisRenderer that should be placed at bottom side of the plot.

*right*
Type: YAxisRenderer
A YAxisRenderer that should be placed at right side of the plot.

### Return Value

Type: GridPanel
A GridPanel instance.

# See Also

LayoutBuilder Members
LayoutBuilder Class
MindFusion.Charting.Controls Namespace

## LayoutBuilder.createAndAddPlotWithTopAndLeftAxes Method

Creates a GridPanel containing the specified plot and axis renderers.The panel is automatically added to the dashboard's LayoutPanel.

**Namespace:** MindFusion.Charting.Controls
**File:** LayoutBuilder.js

# Syntax

**JavaScript**

```
function createAndAddPlotWithTopAndLeftAxes (plot, top, left)
```

### Parameters

*plot*

Type: Plot2D
A Plot2D instance.

*top*
Type: XAxisRenderer
An XAxisRenderer that should be placed at top side of the plot.

*left*
Type: YAxisRenderer
A YAxisRenderer that should be placed at left side of the plot.

### Return Value

Type: GridPanel
A GridPanel instance.

## See Also

LayoutBuilder Members
LayoutBuilder Class
MindFusion.Charting.Controls Namespace

# LayoutBuilder.createAndAddPlotWithTopAndRightAxes Method

Creates a GridPanel containing the specified plot and axis renderers.The panel is automatically added to the dashboard's LayoutPanel.

**Namespace:** MindFusion.Charting.Controls
**File:** LayoutBuilder.js

## Syntax

**JavaScript**

```
function createAndAddPlotWithTopAndRightAxes (plot, top, right)
```

### Parameters

*plot*
Type: Plot2D
A Plot2D instance.

*top*
Type: XAxisRenderer
An XAxisRenderer that should be placed at top side of the plot.

*right*
Type: YAxisRenderer
A YAxisRenderer that should be placed at right side of the plot.

### Return Value

Type: GridPanel
A GridPanel instance.

## See Also

LayoutBuilder Members
LayoutBuilder Class
MindFusion.Charting.Controls Namespace

# LayoutBuilder.createLayout Method

Creates a StackPanel for each array of components and adds it to a parent StackPanel with opposite orientation.

**Namespace:** MindFusion.Charting.Controls
**File:** LayoutBuilder.js

## Syntax

**JavaScript**

```
function createLayout (horizontal, c1, c2, c3)
```

### Parameters

*horizontal*
Type: Boolean
The orientation of the parent panel.

*c1*

*c2*

*c3*

### Return Value

Type: StackPanel
A StackPanel instance.

## See Also

LayoutBuilder Members
LayoutBuilder Class
MindFusion.Charting.Controls Namespace

# LayoutBuilder.createPlotAndAxes Method

Creates a GridPanel containing the specified plot and axis renderers.

**Namespace:** MindFusion.Charting.Controls
**File:** LayoutBuilder.js

## Syntax

**JavaScript**

```
function createPlotAndAxes (plot, top, left, bottom, right)
```

### Parameters

*plot*
Type: Plot2D
A Plot2D instance.

*top*

*left*

*bottom*

*right*

### Return Value

Type: GridPanel
A GridPanel instance.

## See Also

LayoutBuilder Members
LayoutBuilder Class
MindFusion.Charting.Controls Namespace

# LayoutBuilder.createPlotWithBottomAndLeftAxes Method

Creates a GridPanel containing the specified plot and axis renderers.

**Namespace:** MindFusion.Charting.Controls
**File:** LayoutBuilder.js

## Syntax

**JavaScript**

```
function createPlotWithBottomAndLeftAxes (plot, bottom, left)
```

### Parameters

*plot*
Type: Plot2D
A Plot2D instance.

*bottom*
Type: XAxisRenderer
An XAxisRenderer that should be placed at bottom side of the plot.

*left*
Type: YAxisRenderer
A YAxisRenderer that should be placed at left side of the plot.

### Return Value

Type: GridPanel
A GridPanel instance.

## See Also

LayoutBuilder Members
LayoutBuilder Class
MindFusion.Charting.Controls Namespace

# LayoutBuilder.createPlotWithBottomAndRightAxes Method

Creates a GridPanel containing the specified plot and axis renderers.

**Namespace:** MindFusion.Charting.Controls
**File:** LayoutBuilder.js

## Syntax

JavaScript

```
function createPlotWithBottomAndRightAxes (plot, bottom, right)
```

### Parameters

*plot*
Type: Plot2D
A Plot2D instance.

*bottom*
Type: XAxisRenderer
An XAxisRenderer that should be placed at bottom side of the plot.

*right*
Type: YAxisRenderer
A YAxisRenderer that should be placed at right side of the plot.

### Return Value

Type: GridPanel
A GridPanel instance.

## See Also

LayoutBuilder Members
LayoutBuilder Class
MindFusion.Charting.Controls Namespace

# LayoutBuilder.createPlotWithTopAndLeftAxes Method

Creates a GridPanel containing the specified plot and axis renderers.

**Namespace:** MindFusion.Charting.Controls
**File:** LayoutBuilder.js

## Syntax

JavaScript

```
function createPlotWithTopAndLeftAxes (plot, top, left)
```

### Parameters

*plot*
Type: Plot2D
A Plot2D instance.

*top*
Type: XAxisRenderer
An XAxisRenderer that should be placed at top side of the plot.

*left*
Type: YAxisRenderer
A YAxisRenderer that should be placed at left side of the plot.

### Return Value

Type: GridPanel
A GridPanel instance.

## See Also

LayoutBuilder Members
LayoutBuilder Class
MindFusion.Charting.Controls Namespace

# LayoutBuilder.createPlotWithTopAndRightAxes Method

Creates a GridPanel containing the specified plot and axis renderers.

**Namespace:** MindFusion.Charting.Controls
**File:** LayoutBuilder.js

## Syntax

**JavaScript**

```
function createPlotWithTopAndRightAxes (plot, top, right)
```

### Parameters

*plot*
Type: Plot2D
A Plot2D instance.

*top*
Type: XAxisRenderer
An XAxisRenderer that should be placed at top side of the plot.

*right*
Type: YAxisRenderer
A YAxisRenderer that should be placed at right side of the plot.

### Return Value

Type: GridPanel
A GridPanel instance.

## See Also

LayoutBuilder Members
LayoutBuilder Class
MindFusion.Charting.Controls Namespace

# LineChart Class

A control used to draw line charts.

## Syntax

**JavaScript**

```
// class
LineChart.prototype = {}
```

## Inheritance Hierarchy

## See Also

LineChart Members
MindFusion.Charting.Controls Namespace

# LineChart Members

A control used to draw line charts.

The following tables list the members exposed by the LineChart type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | LineChart | Initializes a new instance of the LineChart class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | lineType | Gets or sets what type of line segments to draw between data points. |
|  | series | BiaxialChart.Series override. Gets or sets the list of series whose data is drawn in this chart. |

## See Also

LineChart Class
MindFusion.Charting.Controls Namespace

# LineChart Constructor

Initializes a new instance of the LineChart class.

**Namespace:** MindFusion.Charting.Controls
**File:** LineChart.js

## Syntax

| JavaScript |
|---|
| `function LineChart (element, [seriesRenderer])` |

### Parameters

*element*
Type: HTMLCanvasElement
The canvas DOM element to associate this chart with.

*seriesRenderer*
Optional.
Type: LineRenderer
A LineRenderer used to draw chart's data series.

## See Also

LineChart Class
MindFusion.Charting.Controls Namespace

# LineChart Properties

## LineChart.lineType Property

Gets or sets what type of line segments to draw between data points.

**Namespace:** MindFusion.Charting.Controls
**File:** LineChart.js

## Syntax

```
JavaScript
get lineType() {}
```

### Property Value

Type: LineType

## See Also

LineChart Members
LineChart Class
MindFusion.Charting.Controls Namespace

## LineChart.series Property

BiaxialChart.Series override. Gets or sets the list of series whose data is drawn in this chart.

**Namespace:** MindFusion.Charting.Controls
**File:** LineChart.js

## Syntax

```
JavaScript
get series() {}
```

### Property Value

Type: ObservableCollection

## See Also

LineChart Members
LineChart Class
MindFusion.Charting.Controls Namespace

# PieChart Class

A control used to draw pie charts.

## Syntax

**JavaScript**

```
// class
PieChart.prototype = {}
```

## Inheritance Hierarchy

## See Also

PieChart Members
MindFusion.Charting.Controls Namespace

# PieChart Members

A control used to draw pie charts.

The following tables list the members exposed by the PieChart type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | PieChart | Initializes a new instance of the PieChart class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | allowRotate | Gets or sets a value indicating whether users are allowed to rotate this chart. |
| | chartPadding | Gets or set padding space between the plot's border and series graphics. |
| | detachedSlices | Gets or sets indices of detached slices. |
| | doughnut | Gets or sets a value indicating whether the pie should be rendered as a doughnut. |
| | series | Gets or sets the series whose data is drawn in this chart. |
| | startAngle | Gets or sets the angle where first data item of series should be drawn. |

## Public Methods

| | Name | Description |
|---|---|---|
| | createPlot | Chart.CreatePlot override. Creates the Plot type corresponding to this chart. |

## See Also

PieChart Class
MindFusion.Charting.Controls Namespace

# PieChart Constructor

Initializes a new instance of the PieChart class.

**Namespace:** MindFusion.Charting.Controls
**File:** PieChart.js

## Syntax

**JavaScript**

```
function PieChart (element, [seriesRenderer])
```

### Parameters

*element*
Type: HTMLCanvasElement
The canvas DOM element to associate this chart with.

*seriesRenderer*
Optional.
Type: PieRenderer
A PieRenderer used to draw chart's data series.

## See Also

PieChart Class
MindFusion.Charting.Controls Namespace

# PieChart Methods

# PieChart.createPlot Method

Chart.CreatePlot override. Creates the Plot type corresponding to this chart.

**Namespace:** MindFusion.Charting.Controls
**File:** PieChart.js

## Syntax

**JavaScript**

```
function createPlot ()
```

### Return Value

Type: Plot
An instance of the PolarPlot class.

## See Also

PieChart Members
PieChart Class
MindFusion.Charting.Controls Namespace

# PieChart Properties

## PieChart.allowRotate Property

Gets or sets a value indicating whether users are allowed to rotate this chart.

**Namespace:** MindFusion.Charting.Controls
**File:** PieChart.js

### Syntax

```
JavaScript
get allowRotate() {}
```

#### Property Value

Type: Boolean

### See Also

PieChart Members
PieChart Class
MindFusion.Charting.Controls Namespace

## PieChart.chartPadding Property

Gets or set padding space between the plot's border and series graphics.

**Namespace:** MindFusion.Charting.Controls
**File:** PieChart.js

### Syntax

```
JavaScript
get chartPadding() {}
```

#### Property Value

Type: Number

### See Also

PieChart Members
PieChart Class
MindFusion.Charting.Controls Namespace

## PieChart.detachedSlices Property

Gets or sets indices of detached slices.

**Namespace:** MindFusion.Charting.Controls
**File:** PieChart.js

## Syntax

```
JavaScript
get detachedSlices() {}
```

### Property Value

Type: List

## See Also

PieChart Members
PieChart Class
MindFusion.Charting.Controls Namespace

# PieChart.doughnut Property

Gets or sets a value indicating whether the pie should be rendered as a doughnut.

**Namespace:** MindFusion.Charting.Controls
**File:** PieChart.js

## Syntax

```
JavaScript
get doughnut() {}
```

### Property Value

Type: Boolean

## See Also

PieChart Members
PieChart Class
MindFusion.Charting.Controls Namespace

# PieChart.series Property

Gets or sets the series whose data is drawn in this chart.

**Namespace:** MindFusion.Charting.Controls
**File:** PieChart.js

## Syntax

```
JavaScript
get series() {}
```

### Property Value

Type: Series

## See Also

PieChart Members
PieChart Class
MindFusion.Charting.Controls Namespace

### PieChart.startAngle Property

Gets or sets the angle where first data item of series should be drawn.

**Namespace:** MindFusion.Charting.Controls
**File:** PieChart.js

## Syntax

**JavaScript**

```
get startAngle() {}
```

#### Property Value

Type: Number

## See Also

PieChart Members
PieChart Class
MindFusion.Charting.Controls Namespace

# RadarChart Class

A control used to draw radar charts.

## Syntax

**JavaScript**

```
// class
RadarChart.prototype = {}
```

## Inheritance Hierarchy

## See Also

RadarChart Members
MindFusion.Charting.Controls Namespace

### RadarChart Members

A control used to draw radar charts.

The following tables list the members exposed by the RadarChart type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | RadarChart | Initializes a new instance of the RadarChart class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | alignToAxis | Gets or sets whether pie-radar sectors should be aligned to axes in the radar grid or centered around them. |
| | allowRotate | Gets or sets a value indicating whether users are allowed to rotate this chart. |
| | areaOpacity | Gets or sets the opacity of radar polygons. |
| | axes | Gets the list of Axis objects representing ranges of variables represented in the radar chart. |
| | chartPadding | Gets or set padding space between the plot's border and series graphics. |
| | defaultAxis | Gets a default Axis object used when data item index does not have corresponding element in the Axes collection. |
| | gridColor1 | Gets or sets the main color of the grid. |
| | gridColor2 | Gets or sets the alternating color of the grid. |
| | gridDivisions | Gets or sets the number of grid divisions. |
| | gridType | Gets or sets the type of grid drawn in this chart. |
| | radarType | Gets or sets the type of radar graphics drawn in this chart. |
| | series | Gets or sets the list of series whose data is drawn in this chart. |
| | showCoordinates | Gets or sets a value indicating whether the chart should show axis coordinates. |
| | showScatter | Gets or sets a value indicating whether the chart should render scatter shapes. |
| | startAngle | Gets or sets the angle where first data item of series should be drawn. |
| | uniformScale | Gets or sets a value indicating whether all axes should display same data range. |

## Public Methods

| | Name | Description |
|---|---|---|
| | createPlot | Chart.CreatePlot override. Creates the Plot type corresponding to this chart. |

## See Also

RadarChart Class
MindFusion.Charting.Controls Namespace

# RadarChart Constructor

Initializes a new instance of the RadarChart class.

**Namespace:** MindFusion.Charting.Controls
**File:** RadarChart.js

## Syntax

```
JavaScript
function RadarChart (element, [seriesRenderer])
```

### Parameters

*element*
Type: HTMLCanvasElement
The canvas DOM element to associate this chart with.

*seriesRenderer*
Optional.
Type: RadarRenderer
A RadarRenderer used to draw chart's data series.

## See Also

RadarChart Class
MindFusion.Charting.Controls Namespace

# RadarChart Methods

# RadarChart.createPlot Method

Chart.CreatePlot override. Creates the Plot type corresponding to this chart.

**Namespace:** MindFusion.Charting.Controls
**File:** RadarChart.js

## Syntax

```
JavaScript
function createPlot ()
```

### Return Value

Type: Plot
An instance of the RadarPlot class.

## See Also

RadarChart Members
RadarChart Class
MindFusion.Charting.Controls Namespace

# RadarChart Properties

## RadarChart.alignToAxis Property

Gets or sets whether pie-radar sectors should be aligned to axes in the radar grid or centered around them.

**Namespace:** MindFusion.Charting.Controls
**File:** RadarChart.js

### Syntax

```
JavaScript
get alignToAxis() {}
```

#### Property Value

Type: Boolean

### See Also

RadarChart Members
RadarChart Class
MindFusion.Charting.Controls Namespace

## RadarChart.allowRotate Property

Gets or sets a value indicating whether users are allowed to rotate this chart.

**Namespace:** MindFusion.Charting.Controls
**File:** RadarChart.js

### Syntax

```
JavaScript
get allowRotate() {}
```

#### Property Value

Type: Boolean

### See Also

RadarChart Members
RadarChart Class
MindFusion.Charting.Controls Namespace

## RadarChart.areaOpacity Property

Gets or sets the opacity of radar polygons.

**Namespace:** MindFusion.Charting.Controls
**File:** RadarChart.js

## Syntax

```
JavaScript
get areaOpacity() {}
```

### Property Value

Type: Number

## See Also

RadarChart Members
RadarChart Class
MindFusion.Charting.Controls Namespace

# RadarChart.axes Property

Gets the list of Axis objects representing ranges of variables represented in the radar chart.

**Namespace:** MindFusion.Charting.Controls
**File:** RadarChart.js

## Syntax

```
JavaScript
get axes() {}
```

### Property Value

Type: ObservableCollection

## See Also

RadarChart Members
RadarChart Class
MindFusion.Charting.Controls Namespace

# RadarChart.chartPadding Property

Gets or set padding space between the plot's border and series graphics.

**Namespace:** MindFusion.Charting.Controls
**File:** RadarChart.js

## Syntax

```
JavaScript
get chartPadding() {}
```

### Property Value

Type: Number

## See Also

RadarChart Members
RadarChart Class
MindFusion.Charting.Controls Namespace

# RadarChart.defaultAxis Property

Gets a default Axis object used when data item index does not have corresponding element in the Axes collection.

**Namespace:** MindFusion.Charting.Controls
**File:** RadarChart.js

## Syntax

**JavaScript**

```
get defaultAxis() {}
```

### Property Value

Type: Axis

## See Also

RadarChart Members
RadarChart Class
MindFusion.Charting.Controls Namespace

# RadarChart.gridColor1 Property

Gets or sets the main color of the grid.

**Namespace:** MindFusion.Charting.Controls
**File:** RadarChart.js

## Syntax

**JavaScript**

```
get gridColor1() {}
```

### Property Value

Type: Color

## See Also

RadarChart Members
RadarChart Class
MindFusion.Charting.Controls Namespace

# RadarChart.gridColor2 Property

Gets or sets the alternating color of the grid.

**Namespace:** MindFusion.Charting.Controls
**File:** RadarChart.js

## Syntax

```
JavaScript
get gridColor2() {}
```

### Property Value

Type: Color

## See Also

RadarChart Members
RadarChart Class
MindFusion.Charting.Controls Namespace

# RadarChart.gridDivisions Property

Gets or sets the number of grid divisions.

**Namespace:** MindFusion.Charting.Controls
**File:** RadarChart.js

## Syntax

```
JavaScript
get gridDivisions() {}
```

### Property Value

Type: Number

## See Also

RadarChart Members
RadarChart Class
MindFusion.Charting.Controls Namespace

# RadarChart.gridType Property

Gets or sets the type of grid drawn in this chart.

**Namespace:** MindFusion.Charting.Controls
**File:** RadarChart.js

## Syntax

```
JavaScript
```

```
get gridType() {}
```

### Property Value

Type: RadarGridType

## See Also

RadarChart Members
RadarChart Class
MindFusion.Charting.Controls Namespace

# RadarChart.radarType Property

Gets or sets the type of radar graphics drawn in this chart.

**Namespace:** MindFusion.Charting.Controls
**File:** RadarChart.js

## Syntax

**JavaScript**

```
get radarType() {}
```

### Property Value

Type: RadarType

## See Also

RadarChart Members
RadarChart Class
MindFusion.Charting.Controls Namespace

# RadarChart.series Property

Gets or sets the list of series whose data is drawn in this chart.

**Namespace:** MindFusion.Charting.Controls
**File:** RadarChart.js

## Syntax

**JavaScript**

```
get series() {}
```

### Property Value

Type: ObservableCollection

## See Also

RadarChart Members
RadarChart Class

MindFusion.Charting.Controls Namespace

## RadarChart.showCoordinates Property

Gets or sets a value indicating whether the chart should show axis coordinates.

**Namespace:** MindFusion.Charting.Controls
**File:** RadarChart.js

### Syntax

**JavaScript**

```
get showCoordinates() {}
```

#### Property Value

Type: Boolean

### See Also

RadarChart Members
RadarChart Class
MindFusion.Charting.Controls Namespace

## RadarChart.showScatter Property

Gets or sets a value indicating whether the chart should render scatter shapes.

**Namespace:** MindFusion.Charting.Controls
**File:** RadarChart.js

### Syntax

**JavaScript**

```
get showScatter() {}
```

#### Property Value

Type: Boolean

### See Also

RadarChart Members
RadarChart Class
MindFusion.Charting.Controls Namespace

## RadarChart.startAngle Property

Gets or sets the angle where first data item of series should be drawn.

**Namespace:** MindFusion.Charting.Controls
**File:** RadarChart.js

### Syntax

```
JavaScript
get startAngle() {}
```

### Property Value

Type: Number

## See Also

RadarChart Members
RadarChart Class
MindFusion.Charting.Controls Namespace

# RadarChart.uniformScale Property

Gets or sets a value indicating whether all axes should display same data range.

**Namespace:** MindFusion.Charting.Controls
**File:** RadarChart.js

## Syntax

```
JavaScript
get uniformScale() {}
```

### Property Value

Type: Boolean

## See Also

RadarChart Members
RadarChart Class
MindFusion.Charting.Controls Namespace

# ScatterChart Class

A control used to draw scatter charts.

## Syntax

```
JavaScript
// class
ScatterChart.prototype = {}
```

## Inheritance Hierarchy

## See Also

ScatterChart Members
MindFusion.Charting.Controls Namespace

# ScatterChart Members

A control used to draw scatter charts.

The following tables list the members exposed by the ScatterChart type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | ScatterChart | Initializes a new instance of the ScatterChart class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | series | BiaxialChart.Series override. Gets or sets the list of series whose data is drawn in this chart. |
|  | shape | Gets or sets the type of scatter shapes. |
|  | shapeSize | Gets or sets the size of scatter shapes. |

## See Also

ScatterChart Class
MindFusion.Charting.Controls Namespace

# ScatterChart Constructor

Initializes a new instance of the ScatterChart class.

**Namespace:** MindFusion.Charting.Controls
**File:** ScatterChart.js

## Syntax

**JavaScript**

```
function ScatterChart (element, [seriesRenderer])
```

### Parameters

*element*
Type: HTMLCanvasElement
The canvas DOM element to associate this chart with.

*seriesRenderer*
Optional.
Type: ScatterRenderer
A ScatterRenderer used to draw chart's data series.

## See Also

ScatterChart Class
MindFusion.Charting.Controls Namespace

# ScatterChart Properties

## ScatterChart.series Property

BiaxialChart.Series override. Gets or sets the list of series whose data is drawn in this chart.

**Namespace:** MindFusion.Charting.Controls
**File:** ScatterChart.js

### Syntax

```
JavaScript
get series() {}
```

#### Property Value

Type: ObservableCollection

### See Also

ScatterChart Members
ScatterChart Class
MindFusion.Charting.Controls Namespace

## ScatterChart.shape Property

Gets or sets the type of scatter shapes.

**Namespace:** MindFusion.Charting.Controls
**File:** ScatterChart.js

### Syntax

```
JavaScript
get shape() {}
```

#### Property Value

Type: ScatterType

### See Also

ScatterChart Members
ScatterChart Class
MindFusion.Charting.Controls Namespace

## ScatterChart.shapeSize Property

Gets or sets the size of scatter shapes.

**Namespace:** MindFusion.Charting.Controls
**File:** ScatterChart.js

### Syntax

```
JavaScript
get shapeSize() {}
```

### Property Value

Type: Number

## See Also

ScatterChart Members
ScatterChart Class
MindFusion.Charting.Controls Namespace

# TowerChart Class

A control used to draw tower charts, rendering series side by side to allow comparing data sequence and sizes.

## Syntax

```
JavaScript
// class
TowerChart.prototype = {}
```

## Remarks

Assign the series to compare to leftSeries and rightSeries properties of the chart. The chart segments are arranged according to the value of towerLayout property, and their shape is drawn as specified by segmentShape.

This control requires three-dimensional series. The first dimension specifies event order or timing and is used to sort or position segments. Second dimension specifies duration and is rendered as segment length along the main axis. Third dimension specifies a value rendered as segment width along the cross axis. You could use the stock EventSeries class as data container, or implement the Series interface to provide data from your own data structures.

## Inheritance Hierarchy

**MindFusion.Charting.Controls.TowerChart**

## See Also

TowerChart Members
MindFusion.Charting.Controls Namespace

# TowerChart Members

A control used to draw tower charts, rendering series side by side to allow comparing data sequence and sizes.

The following tables list the members exposed by the TowerChart type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | TowerChart | Initializes a new instance of the TowerChart class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | leftSeries | Gets or sets the series whose data is drawn on the left side of this chart. |
|  | margins | Gets or sets the size of margins around chart graphics. |
|  | range | Gets or sets fixed data range size. |
|  | rightSeries | Gets or sets the series whose data is drawn on the right side of this chart. |
|  | segmentShape | Gets or sets the shape of tower segments. |
|  | seriesPadding | Gets or sets the padding distance between series. |
|  | towerLayout | Gets or sets the layout of tower segments. |

## See Also

TowerChart Class
MindFusion.Charting.Controls Namespace

# TowerChart Constructor

Initializes a new instance of the TowerChart class.

**Namespace:** MindFusion.Charting.Controls
**File:** TowerChart.js

## Syntax

**JavaScript**

```
function TowerChart (element, [seriesRenderer])
```

### Parameters

*element*
Type: HTMLCanvasElement
HTMLCanvasElement. The canvas DOM element to associate this chart with.

*seriesRenderer*
Optional.
Type: CandlestickRenderer
CandlestickRenderer. A TowerRenderer used to draw chart's data series.

## See Also

TowerChart Class
MindFusion.Charting.Controls Namespace

# TowerChart Properties

## TowerChart.leftSeries Property

Gets or sets the series whose data is drawn on the left side of this chart.

**Namespace:** MindFusion.Charting.Controls
**File:** TowerChart.js

### Syntax

```
JavaScript
get leftSeries() {}
set leftSeries(value) {}
```

#### Property Value

An object that implements the Series interface.

### Remarks

This property must be set to a three-dimensional series. The first dimension specifies event order or timing and is used to sort or position segments. Second dimension specifies duration and is rendered as segment length along the main axis. Third dimension specifies a value rendered as segment width along the cross axis. You could use the stock EventSeries class as data container, or implement the Series interface to provide data from your own data structures.

### See Also

TowerChart Members
TowerChart Class
MindFusion.Charting.Controls Namespace

## TowerChart.margins Property

Gets or sets the size of margins around chart graphics.

**Namespace:** MindFusion.Charting.Controls
**File:** TowerChart.js

### Syntax

```
JavaScript
get margins() {}
set margins(value) {}
```

#### Property Value

Type: Margins
*Margins*

A Margins instance specifying margin size on each side of the plot.

### See Also

TowerChart Members
TowerChart Class
MindFusion.Charting.Controls Namespace

# TowerChart.range Property

Gets or sets fixed data range size.

**Namespace:** MindFusion.Charting.Controls
**File:** TowerChart.js

## Syntax

```
JavaScript
get range() {}
set range(value) {}
```

### Property Value

Type: Number
*Number*

An optional number. The default value is null.

## See Also

TowerChart Members
TowerChart Class
MindFusion.Charting.Controls Namespace

# TowerChart.rightSeries Property

Gets or sets the series whose data is drawn on the right side of this chart.

**Namespace:** MindFusion.Charting.Controls
**File:** TowerChart.js

## Syntax

```
JavaScript
get rightSeries() {}
set rightSeries(value) {}
```

### Property Value

An object that implements the Series interface.

## Remarks

This property must be set to a three-dimensional series. The first dimension specifies event order or timing and is used to sort or position segments. Second dimension specifies duration and is rendered as segment length along the main axis. Third dimension specifies a value rendered as segment width along the cross axis. You could use the stock EventSeries class as data container, or implement the Series interface to provide data from your own data structures.

## See Also

TowerChart Members
TowerChart Class
MindFusion.Charting.Controls Namespace

## TowerChart.segmentShape Property

Gets or sets the shape of tower segments.

**Namespace:** MindFusion.Charting.Controls
**File:** TowerChart.js

## Syntax

**JavaScript**

```
get segmentShape() {}
set segmentShape(value) {}
```

### Property Value

A member of the TowerSegmentShape enumeration. The default value is **Triangle**.

## See Also

TowerChart Members
TowerChart Class
MindFusion.Charting.Controls Namespace

## TowerChart.seriesPadding Property

Gets or sets the padding distance between series.

**Namespace:** MindFusion.Charting.Controls
**File:** TowerChart.js

## Syntax

**JavaScript**

```
get seriesPadding() {}
set seriesPadding(value) {}
```

### Property Value

Type: Number
*Number*

The distance between LeftSeries and RightSeries graphics.

## See Also

TowerChart Members
TowerChart Class
MindFusion.Charting.Controls Namespace

## TowerChart.towerLayout Property

Gets or sets the layout of tower segments.

**Namespace:** MindFusion.Charting.Controls
**File:** TowerChart.js

### Syntax

```
JavaScript
get towerLayout() {}
set towerLayout(value) {}
```

#### Property Value

A member of the TowerLayout enumeration. The default value is **Interleave**.

### See Also

TowerChart Members
TowerChart Class
MindFusion.Charting.Controls Namespace

# MindFusion.Charting.Gauges Namespace

## GaugeRenderer Class

A Component that renders gauges in the dashboard.

### Syntax

```
JavaScript
// class
GaugeRenderer.prototype = {}
```

### Inheritance Hierarchy

### See Also

GaugeRenderer Members
MindFusion.Charting.Gauges Namespace

## GaugeRenderer Members

A Component that renders gauges in the dashboard.

The following tables list the members exposed by the GaugeRenderer type.

### Public Constructors

| | Name | Description |
|---|---|---|

| | | |
|---|---|---|
| | GaugeRenderer | Initializes a new instance of the GaugeRenderer class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | background | Gets or sets a Brush used to draw the background of this gauge. |
| | fontName | Gets or sets the name of font used to draw text in this gauge. |
| | fontSize | Gets or sets the size of font used to draw text in this gauge. |
| | fontStyle | Gets or sets the style of font used to draw text in this gauge. |
| | pointerBackground | Gets or sets a Brush used to draw the background of gauge pointers. |
| | pointerStroke | Gets or sets a Brush used to stroke gauge pointers. |
| | pointerStrokeThickness | Gets or sets the thickness of the Pen used to stroke the gauge pointers. |
| | scaleBackground | Gets or sets a Brush used to draw the background of the gauge scales. |
| | scaleStroke | Gets or sets a Brush used to stroke the gauge scales. |
| | scaleStrokeThickness | Gets or sets the thickness of the Pen used to stroke the gauge scales. |
| | stroke | Gets or sets a Brush used to stroke this gauge. |
| | strokeThickness | Gets or sets the thickness of the Pen used to stroke this gauge. |
| | tickBackground | Gets or sets a Brush used to draw the background of gauge ticks. |
| | tickStroke | Gets or sets a Brush used to stroke gauge ticks. |
| | tickStrokeThickness | Gets or sets the thickness of the Pen used to stroke the gauge ticks. |

## Public Methods

| | Name | Description |
|---|---|---|
| | getDefaultPointerFill | Gets the default pointer fill. |
| | getDefaultPointerStroke | Gets the default pointer stroke. |

## See Also

GaugeRenderer Class
MindFusion.Charting.Gauges Namespace

## GaugeRenderer Constructor

Initializes a new instance of the GaugeRenderer class.

**Namespace:** MindFusion.Charting.Gauges
**File:** GaugeRenderer.js

### Syntax

```
JavaScript
function GaugeRenderer ()
```

### See Also

GaugeRenderer Class
MindFusion.Charting.Gauges Namespace

## GaugeRenderer Methods

## GaugeRenderer.getDefaultPointerFill Method

Gets the default pointer fill.

**Namespace:** MindFusion.Charting.Gauges
**File:** GaugeRenderer.js

### Syntax

```
JavaScript
function getDefaultPointerFill (pointer)
```

#### Parameters

*pointer*
Type: void

### See Also

GaugeRenderer Members
GaugeRenderer Class
MindFusion.Charting.Gauges Namespace

## GaugeRenderer.getDefaultPointerStroke Method

Gets the default pointer stroke.

**Namespace:** MindFusion.Charting.Gauges
**File:** GaugeRenderer.js

### Syntax

```
JavaScript
function getDefaultPointerStroke (pointer)
```

### Parameters

*pointer*
Type: void

## See Also

GaugeRenderer Members
GaugeRenderer Class
MindFusion.Charting.Gauges Namespace

# GaugeRenderer Properties

## GaugeRenderer.background Property

Gets or sets a Brush used to draw the background of this gauge.

**Namespace:** MindFusion.Charting.Gauges
**File:** GaugeRenderer.js

## Syntax

**JavaScript**

```
get background() {}
```

### Property Value

Type: Brush

## See Also

GaugeRenderer Members
GaugeRenderer Class
MindFusion.Charting.Gauges Namespace

## GaugeRenderer.fontName Property

Gets or sets the name of font used to draw text in this gauge.

**Namespace:** MindFusion.Charting.Gauges
**File:** GaugeRenderer.js

## Syntax

**JavaScript**

```
get fontName() {}
```

### Property Value

Type: String

## See Also

GaugeRenderer Members
GaugeRenderer Class
MindFusion.Charting.Gauges Namespace

## GaugeRenderer.fontSize Property

Gets or sets the size of font used to draw text in this gauge.

**Namespace:** MindFusion.Charting.Gauges
**File:** GaugeRenderer.js

### Syntax

```
JavaScript
get fontSize() {}
```

#### Property Value

Type: Number

### See Also

GaugeRenderer Members
GaugeRenderer Class
MindFusion.Charting.Gauges Namespace

## GaugeRenderer.fontStyle Property

Gets or sets the style of font used to draw text in this gauge.

**Namespace:** MindFusion.Charting.Gauges
**File:** GaugeRenderer.js

### Syntax

```
JavaScript
get fontStyle() {}
```

#### Property Value

Type: FontStyle

### See Also

GaugeRenderer Members
GaugeRenderer Class
MindFusion.Charting.Gauges Namespace

## GaugeRenderer.pointerBackground Property

Gets or sets a Brush used to draw the background of gauge pointers.

**Namespace:** MindFusion.Charting.Gauges
**File:** GaugeRenderer.js

## Syntax

```
JavaScript
get pointerBackground() {}
```

### Property Value

Type: Brush

## See Also

GaugeRenderer Members
GaugeRenderer Class
MindFusion.Charting.Gauges Namespace

# GaugeRenderer.pointerStroke Property

Gets or sets a Brush used to stroke gauge pointers.

**Namespace:** MindFusion.Charting.Gauges
**File:** GaugeRenderer.js

## Syntax

```
JavaScript
get pointerStroke() {}
```

### Property Value

Type: Brush

## See Also

GaugeRenderer Members
GaugeRenderer Class
MindFusion.Charting.Gauges Namespace

# GaugeRenderer.pointerStrokeThickness Property

Gets or sets the thickness of the Pen used to stroke the gauge pointers.

**Namespace:** MindFusion.Charting.Gauges
**File:** GaugeRenderer.js

## Syntax

```
JavaScript
get pointerStrokeThickness() {}
```

### Property Value

Type: Number

## See Also

GaugeRenderer Members
GaugeRenderer Class
MindFusion.Charting.Gauges Namespace

# GaugeRenderer.scaleBackground Property

Gets or sets a Brush used to draw the background of the gauge scales.

**Namespace:** MindFusion.Charting.Gauges
**File:** GaugeRenderer.js

## Syntax

**JavaScript**

```
get scaleBackground() {}
```

### Property Value

Type: Brush

## See Also

GaugeRenderer Members
GaugeRenderer Class
MindFusion.Charting.Gauges Namespace

# GaugeRenderer.scaleStroke Property

Gets or sets a Brush used to stroke the gauge scales.

**Namespace:** MindFusion.Charting.Gauges
**File:** GaugeRenderer.js

## Syntax

**JavaScript**

```
get scaleStroke() {}
```

### Property Value

Type: Brush

## See Also

GaugeRenderer Members
GaugeRenderer Class
MindFusion.Charting.Gauges Namespace

# GaugeRenderer.scaleStrokeThickness Property

Gets or sets the thickness of the Pen used to stroke the gauge scales.

**Namespace:** MindFusion.Charting.Gauges
**File:** GaugeRenderer.js

## Syntax

```
JavaScript
get scaleStrokeThickness() {}
```

### Property Value

Type: Number

## See Also

GaugeRenderer Members
GaugeRenderer Class
MindFusion.Charting.Gauges Namespace

# GaugeRenderer.stroke Property

Gets or sets a Brush used to stroke this gauge.

**Namespace:** MindFusion.Charting.Gauges
**File:** GaugeRenderer.js

## Syntax

```
JavaScript
get stroke() {}
```

### Property Value

Type: Brush

## See Also

GaugeRenderer Members
GaugeRenderer Class
MindFusion.Charting.Gauges Namespace

# GaugeRenderer.strokeThickness Property

Gets or sets the thickness of the Pen used to stroke this gauge.

**Namespace:** MindFusion.Charting.Gauges
**File:** GaugeRenderer.js

## Syntax

```
JavaScript
get strokeThickness() {}
```

### Property Value

Type: Number

## See Also

GaugeRenderer Members
GaugeRenderer Class
MindFusion.Charting.Gauges Namespace

# GaugeRenderer.tickBackground Property

Gets or sets a Brush used to draw the background of gauge ticks.

**Namespace:** MindFusion.Charting.Gauges
**File:** GaugeRenderer.js

## Syntax

**JavaScript**

```
get tickBackground() {}
```

### Property Value

Type: Brush

## See Also

GaugeRenderer Members
GaugeRenderer Class
MindFusion.Charting.Gauges Namespace

# GaugeRenderer.tickStroke Property

Gets or sets a Brush used to stroke gauge ticks.

**Namespace:** MindFusion.Charting.Gauges
**File:** GaugeRenderer.js

## Syntax

**JavaScript**

```
get tickStroke() {}
```

### Property Value

Type: Brush

## See Also

GaugeRenderer Members
GaugeRenderer Class
MindFusion.Charting.Gauges Namespace

## GaugeRenderer.tickStrokeThickness Property

Gets or sets the thickness of the Pen used to stroke the gauge ticks.

**Namespace:** MindFusion.Charting.Gauges
**File:** GaugeRenderer.js

### Syntax

| JavaScript |
| --- |
| `get tickStrokeThickness() {}` |

#### Property Value

Type: Number

### See Also

GaugeRenderer Members
GaugeRenderer Class
MindFusion.Charting.Gauges Namespace

# LinearGaugeController Class

Controls user interaction with linear gauges.

### Syntax

| JavaScript |
| --- |
| `// class`<br>`LinearGaugeController.prototype = {}` |

### Inheritance Hierarchy

### See Also

LinearGaugeController Members
MindFusion.Charting.Gauges Namespace

## LinearGaugeController Members

Controls user interaction with linear gauges.

The following tables list the members exposed by the LinearGaugeController type.

### Public Constructors

| | Name | Description |
| --- | --- | --- |
| | LinearGaugeController | Initializes a new instance of the LinearGaugeController class. |

### Public Properties

| | Name | Description |
|---|---|---|
| | component | Implements ComponentController.Component. Gets the component modified by this controller. |

## Public Methods

| | Name | Description |
|---|---|---|
| | drawInteraction | Implements ComponentController.DrawInteraction. Draws a representation of the current state of user interaction on specified IGraphics surface. |
| | getCursorHint | For internal use. |
| | getRunningAnimation | For internal use. |
| | onMouseDown | Implements ComponentController.OnMouseDown. Called when the user presses a mouse button. |
| | onMouseMove | Implements ComponentController.OnMouseMove. Called when the user moves the mouse. |
| | onMouseUp | Implements ComponentController.OnMouseUp. Called when the user releases a mouse button. |
| | onMouseWheel | Implements ComponentController.OnMouseWheel. Called when the user scrolls with the middle mouse button. |

## See Also

LinearGaugeController Class
MindFusion.Charting.Gauges Namespace

# LinearGaugeController Constructor

Initializes a new instance of the LinearGaugeController class.

**Namespace:** MindFusion.Charting.Gauges
**File:** LinearGaugeController.js

## Syntax

**JavaScript**

```
function LinearGaugeController (r, context)
```

### Parameters

*r*
Type: LinearGaugeRenderer
A LinearGaugeRenderer instance controlled by this object.

*context*
Type: RenderContext
A RenderContext instance.

## See Also

LinearGaugeController Class
MindFusion.Charting.Gauges Namespace

# LinearGaugeController Methods

## LinearGaugeController.drawInteraction Method

Implements ComponentController.DrawInteraction. Draws a representation of the current state of user interaction on specified IGraphics surface.

**Namespace:** MindFusion.Charting.Gauges
**File:** LinearGaugeController.js

### Syntax

**JavaScript**

```
function drawInteraction (graphics)
```

#### Parameters

*graphics*
A Graphics instance.

### See Also

LinearGaugeController Members
LinearGaugeController Class
MindFusion.Charting.Gauges Namespace

## LinearGaugeController.getCursorHint Method

For internal use.

**Namespace:** MindFusion.Charting.Gauges
**File:** LinearGaugeController.js

### Syntax

**JavaScript**

```
function getCursorHint (x, y)
```

#### Parameters

*x*
Type: void

*y*
Type: void

#### Return Value

Type: CursorHint
A member of the CursorHint enumeration.

### See Also

LinearGaugeController Members
LinearGaugeController Class
MindFusion.Charting.Gauges Namespace

## LinearGaugeController.getRunningAnimation Method

For internal use.

**Namespace:** MindFusion.Charting.Gauges
**File:** LinearGaugeController.js

### Syntax

**JavaScript**

```
function getRunningAnimation ()
```

#### Return Value

Type: ComponentAnimation
An instance of a ComponentAnimation -derived class.

### See Also

LinearGaugeController Members
LinearGaugeController Class
MindFusion.Charting.Gauges Namespace

## LinearGaugeController.onMouseDown Method

Implements ComponentController.OnMouseDown. Called when the user presses a mouse button.

**Namespace:** MindFusion.Charting.Gauges
**File:** LinearGaugeController.js

### Syntax

**JavaScript**

```
function onMouseDown (x, y)
```

#### Parameters

*x*
Type: Number
A number value specifying the horizontal position of mouse pointer.

*y*
Type: Number
A number value specifying the vertical position of mouse pointer.

### See Also

LinearGaugeController Members
LinearGaugeController Class
MindFusion.Charting.Gauges Namespace

# LinearGaugeController.onMouseMove Method

Implements ComponentController.OnMouseMove. Called when the user moves the mouse.

**Namespace:** MindFusion.Charting.Gauges
**File:** LinearGaugeController.js

## Syntax

**JavaScript**

```
function onMouseMove (x, y)
```

### Parameters

*x*
Type: Number
A number value specifying the horizontal position of mouse pointer.

*y*
Type: Number
A number value specifying the vertical position of mouse pointer.

## See Also

LinearGaugeController Members
LinearGaugeController Class
MindFusion.Charting.Gauges Namespace

# LinearGaugeController.onMouseUp Method

Implements ComponentController.OnMouseUp. Called when the user releases a mouse button.

**Namespace:** MindFusion.Charting.Gauges
**File:** LinearGaugeController.js

## Syntax

**JavaScript**

```
function onMouseUp (x, y)
```

### Parameters

*x*
Type: Number
A number value specifying the horizontal position of mouse pointer.

*y*
Type: Number
A number value specifying the vertical position of mouse pointer.

## See Also

LinearGaugeController Members
LinearGaugeController Class
MindFusion.Charting.Gauges Namespace

## LinearGaugeController.onMouseWheel Method

Implements ComponentController.OnMouseWheel. Called when the user scrolls with the middle mouse button.

**Namespace:** MindFusion.Charting.Gauges
**File:** LinearGaugeController.js

## Syntax

**JavaScript**

```
function onMouseWheel (x, y, delta)
```

### Parameters

*x*
Type: Number
A number value specifying the horizontal position of mouse pointer.

*y*
Type: Number
A number value specifying the vertical position of mouse pointer.

*delta*
Type: Number
A number value specifying the amount and the direction for the mouse scroll.

## See Also

LinearGaugeController Members
LinearGaugeController Class
MindFusion.Charting.Gauges Namespace

## LinearGaugeController Properties

## LinearGaugeController.component Property

Implements ComponentController.Component. Gets the component modified by this controller.

**Namespace:** MindFusion.Charting.Gauges
**File:** LinearGaugeController.js

## Syntax

**JavaScript**

```
get component() {}
```

### Property Value

Type: Component

## See Also

# LinearGaugeRenderer Class

A Component that renders linear gauges in the dashboard.

## Syntax

**JavaScript**

```
// class
LinearGaugeRenderer.prototype = {}
```

## Inheritance Hierarchy

## See Also

LinearGaugeRenderer Members
MindFusion.Charting.Gauges Namespace

# LinearGaugeRenderer Members

A Component that renders linear gauges in the dashboard.

The following tables list the members exposed by the LinearGaugeRenderer type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | LinearGaugeRenderer | Initializes a new instance of the LinearGaugeRenderer class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | gauge | Gets or sets the LinearGauge represented by this renderer. |

## Public Methods

| | Name | Description |
|---|---|---|
| | arrange | Component.Arrange override. Sets the location and size of this component relatively to its parent. |
| | createController | Component.CreateController override. Returns a ComponentController used to interact with this component. |
| | draw | Component.Draw override. Draws this component in specified RenderContext. |
| | measure | Component.Measure override. Measures the desired size of this component. |

## See Also

LinearGaugeRenderer Class
MindFusion.Charting.Gauges Namespace

# LinearGaugeRenderer Constructor

Initializes a new instance of the LinearGaugeRenderer class.

**Namespace:** MindFusion.Charting.Gauges
**File:** LinearGaugeRenderer.js

## Syntax

**JavaScript**

```
function LinearGaugeRenderer (element)
```

### Parameters

*element*
Type: void

## See Also

LinearGaugeRenderer Class
MindFusion.Charting.Gauges Namespace

# LinearGaugeRenderer Methods

# LinearGaugeRenderer.arrange Method

Component.Arrange override. Sets the location and size of this component relatively to its parent.

**Namespace:** MindFusion.Charting.Gauges
**File:** LinearGaugeRenderer.js

## Syntax

**JavaScript**

```
function arrange (x, y, w, h, context)
```

### Parameters

*x*
Type: Number
A number value specifying horizontal position.

*y*
Type: Number
A number value specifying vertical position.

*w*
Type: Number
A number value specifying the component's width.

*h*
Type: Number
A number value specifying the component's height.

*context*
Type: RenderContext
A RenderContext instance.

## See Also

LinearGaugeRenderer Members
LinearGaugeRenderer Class
MindFusion.Charting.Gauges Namespace

# LinearGaugeRenderer.createController Method

Component.CreateController override. Returns a ComponentController used to interact with this component.

**Namespace:** MindFusion.Charting.Gauges
**File:** LinearGaugeRenderer.js

## Syntax

**JavaScript**

```
function createController (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: ComponentController
An instance of the LinearGaugeController class.

## See Also

LinearGaugeRenderer Members
LinearGaugeRenderer Class
MindFusion.Charting.Gauges Namespace

# LinearGaugeRenderer.draw Method

Component.Draw override. Draws this component in specified RenderContext.

**Namespace:** MindFusion.Charting.Gauges
**File:** LinearGaugeRenderer.js

## Syntax

**JavaScript**

```
function draw (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

LinearGaugeRenderer Members
LinearGaugeRenderer Class
MindFusion.Charting.Gauges Namespace

# LinearGaugeRenderer.measure Method

Component.Measure override. Measures the desired size of this component.

**Namespace:** MindFusion.Charting.Gauges
**File:** LinearGaugeRenderer.js

## Syntax

**JavaScript**

```
function measure (maxWidth, maxHeight, context)
```

### Parameters

*maxWidth*
Type: Number
The maximum width provided by parent component.

*maxHeight*
Type: Number
The maximum height provided by parent component.

*context*
Type: RenderContext
A RenderContext instance.

## See Also

LinearGaugeRenderer Members
LinearGaugeRenderer Class
MindFusion.Charting.Gauges Namespace

# LinearGaugeRenderer Properties

# LinearGaugeRenderer.gauge Property

Gets or sets the LinearGauge represented by this renderer.

**Namespace:** MindFusion.Charting.Gauges
**File:** LinearGaugeRenderer.js

## Syntax

```
JavaScript
get gauge() {}
```

### Property Value

Type: object

## See Also

LinearGaugeRenderer Members
LinearGaugeRenderer Class
MindFusion.Charting.Gauges Namespace

# OvalGaugeController Class

Controls user interaction with oval gauges.

## Syntax

```
JavaScript
// class
OvalGaugeController.prototype = {}
```

## Inheritance Hierarchy

## See Also

OvalGaugeController Members
MindFusion.Charting.Gauges Namespace

## OvalGaugeController Members

Controls user interaction with oval gauges.

The following tables list the members exposed by the OvalGaugeController type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | OvalGaugeController | Initializes a new instance of the OvalGaugeController class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | component | Implements ComponentController.Component. Gets the component modified by this controller. |

## Public Methods

| | Name | Description |
|---|---|---|
| | drawInteraction | Implements ComponentController.DrawInteraction. Draws a representation of the current state of user interaction on specified IGraphics surface. |
| | getCursorHint | For internal use. |
| | getRunningAnimation | For internal use. |
| | onMouseDown | Implements ComponentController.OnMouseDown. Called when the user presses a mouse button. |
| | onMouseMove | Implements ComponentController.OnMouseMove. Called when the user moves the mouse. |
| | onMouseUp | Implements ComponentController.OnMouseUp. Called when the user releases a mouse button. |
| | onMouseWheel | Implements ComponentController.OnMouseWheel. Called when the user scrolls with the middle mouse button. |

## See Also

OvalGaugeController Class
MindFusion.Charting.Gauges Namespace

# OvalGaugeController Constructor

Initializes a new instance of the OvalGaugeController class.

**Namespace:** MindFusion.Charting.Gauges
**File:** OvalGaugeController.js

## Syntax

**JavaScript**

```
function OvalGaugeController (r, context)
```

### Parameters

*r*
Type: OvalGaugeRenderer
An OvalGaugeRenderer instance controlled by this object.

*context*
Type: RenderContext
A RenderContext instance.

## See Also

OvalGaugeController Class
MindFusion.Charting.Gauges Namespace

# OvalGaugeController Methods

# OvalGaugeController.drawInteraction Method

Implements ComponentController.DrawInteraction. Draws a representation of the current state of user interaction on specified IGraphics surface.

**Namespace:** MindFusion.Charting.Gauges
**File:** OvalGaugeController.js

## Syntax

**JavaScript**

```
function drawInteraction (graphics)
```

### Parameters

*graphics*
A Graphics instance.

## See Also

OvalGaugeController Members
OvalGaugeController Class
MindFusion.Charting.Gauges Namespace

# OvalGaugeController.getCursorHint Method

For internal use.

**Namespace:** MindFusion.Charting.Gauges
**File:** OvalGaugeController.js

## Syntax

**JavaScript**

```
function getCursorHint (x, y)
```

### Parameters

*x*
Type: void

*y*
Type: void

### Return Value

Type: CursorHint
A member of the CursorHint enumeration.

## See Also

OvalGaugeController Members
OvalGaugeController Class
MindFusion.Charting.Gauges Namespace

# OvalGaugeController.getRunningAnimation Method

For internal use.

**Namespace:** MindFusion.Charting.Gauges
**File:** OvalGaugeController.js

## Syntax

**JavaScript**

```
function getRunningAnimation ()
```

### Return Value

Type: ComponentAnimation
An instance of a ComponentAnimation -derived class.

## See Also

OvalGaugeController Members
OvalGaugeController Class
MindFusion.Charting.Gauges Namespace

# OvalGaugeController.onMouseDown Method

Implements ComponentController.OnMouseDown. Called when the user presses a mouse button.

**Namespace:** MindFusion.Charting.Gauges
**File:** OvalGaugeController.js

## Syntax

**JavaScript**

```
function onMouseDown (x, y)
```

### Parameters

*x*
Type: Number
A number value specifying the horizontal position of mouse pointer.

*y*
Type: Number
A number value specifying the vertical position of mouse pointer.

## See Also

OvalGaugeController Members
OvalGaugeController Class
MindFusion.Charting.Gauges Namespace

# OvalGaugeController.onMouseMove Method

Implements ComponentController.OnMouseMove. Called when the user moves the mouse.

**Namespace:** MindFusion.Charting.Gauges
**File:** OvalGaugeController.js

## Syntax

```
JavaScript
function onMouseMove (x, y)
```

### Parameters

*x*
Type: Number
A number value specifying the horizontal position of mouse pointer.

*y*
Type: Number
A number value specifying the vertical position of mouse pointer.

## See Also

OvalGaugeController Members
OvalGaugeController Class
MindFusion.Charting.Gauges Namespace

# OvalGaugeController.onMouseUp Method

Implements ComponentController.OnMouseUp. Called when the user releases a mouse button.

**Namespace:** MindFusion.Charting.Gauges
**File:** OvalGaugeController.js

## Syntax

```
JavaScript
function onMouseUp (x, y)
```

### Parameters

*x*
Type: Number
A number value specifying the horizontal position of mouse pointer.

*y*
Type: Number
A number value specifying the vertical position of mouse pointer.

## See Also

OvalGaugeController Members
OvalGaugeController Class
MindFusion.Charting.Gauges Namespace

## OvalGaugeController.onMouseWheel Method

Implements ComponentController.OnMouseWheel. Called when the user scrolls with the middle mouse button.

**Namespace:** MindFusion.Charting.Gauges
**File:** OvalGaugeController.js

### Syntax

```
JavaScript
function onMouseWheel (x, y, delta)
```

#### Parameters

*x*
Type: Number
A number value specifying the horizontal position of mouse pointer.

*y*
Type: Number
A number value specifying the vertical position of mouse pointer.

*delta*
Type: Number
A number value specifying the amount and the direction for the mouse scroll.

### See Also

OvalGaugeController Members
OvalGaugeController Class
MindFusion.Charting.Gauges Namespace

## OvalGaugeController Properties

## OvalGaugeController.component Property

Implements ComponentController.Component. Gets the component modified by this controller.

**Namespace:** MindFusion.Charting.Gauges
**File:** OvalGaugeController.js

### Syntax

```
JavaScript
get component() {}
```

#### Property Value

Type: Component

### See Also

# OvalGaugeRenderer Class

A Component that renders oval gauges in the dashboard.

## Syntax

**JavaScript**

```
// class
OvalGaugeRenderer.prototype = {}
```

## Inheritance Hierarchy

## See Also

OvalGaugeRenderer Members
MindFusion.Charting.Gauges Namespace

# OvalGaugeRenderer Members

A Component that renders oval gauges in the dashboard.

The following tables list the members exposed by the OvalGaugeRenderer type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | OvalGaugeRenderer | Initializes a new instance of the OvalGaugeRenderer class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | gauge | Gets or sets the OvalGauge represented by this renderer. |

## Public Methods

| | Name | Description |
|---|---|---|
| | arrange | Component.Arrange override. Sets the location and size of this component relatively to its parent. |
| | createController | Component.CreateController override. Returns a ComponentController used to interact with this component. |
| | draw | Component.Draw override. Draws this component in specified RenderContext. |
| | measure | Component.Measure override. Measures the desired size of this component. |

## See Also

OvalGaugeRenderer Class
MindFusion.Charting.Gauges Namespace

# OvalGaugeRenderer Constructor

Initializes a new instance of the OvalGaugeRenderer class.

**Namespace:** MindFusion.Charting.Gauges
**File:** OvalGaugeRenderer.js

## Syntax

**JavaScript**

```
function OvalGaugeRenderer (element)
```

### Parameters

*element*
Type: void

## See Also

OvalGaugeRenderer Class
MindFusion.Charting.Gauges Namespace

# OvalGaugeRenderer Methods

# OvalGaugeRenderer.arrange Method

Component.Arrange override. Sets the location and size of this component relatively to its parent.

**Namespace:** MindFusion.Charting.Gauges
**File:** OvalGaugeRenderer.js

## Syntax

**JavaScript**

```
function arrange (x, y, w, h, context)
```

### Parameters

*x*
Type: Number
A number value specifying horizontal position.

*y*
Type: Number
A number value specifying vertical position.

*w*
Type: Number
A number value specifying the component's width.

*h*
Type: Number
A number value specifying the component's height.

*context*
Type: RenderContext
A RenderContext instance.

## See Also

OvalGaugeRenderer Members
OvalGaugeRenderer Class
MindFusion.Charting.Gauges Namespace

# OvalGaugeRenderer.createController Method

Component.CreateController override. Returns a ComponentController used to interact with this component.

**Namespace:** MindFusion.Charting.Gauges
**File:** OvalGaugeRenderer.js

## Syntax

**JavaScript**

```
function createController (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### Return Value

Type: ComponentController
An instance of the OvalGaugeController class.

## See Also

OvalGaugeRenderer Members
OvalGaugeRenderer Class
MindFusion.Charting.Gauges Namespace

# OvalGaugeRenderer.draw Method

Component.Draw override. Draws this component in specified RenderContext.

**Namespace:** MindFusion.Charting.Gauges
**File:** OvalGaugeRenderer.js

## Syntax

**JavaScript**

```
function draw (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

OvalGaugeRenderer Members
OvalGaugeRenderer Class
MindFusion.Charting.Gauges Namespace

# OvalGaugeRenderer.measure Method

Component.Measure override. Measures the desired size of this component.

**Namespace:** MindFusion.Charting.Gauges
**File:** OvalGaugeRenderer.js

## Syntax

**JavaScript**

```
function measure (maxWidth, maxHeight, context)
```

### Parameters

*maxWidth*
Type: Number
The maximum width provided by parent component.

*maxHeight*
Type: Number
The maximum height provided by parent component.

*context*
Type: RenderContext
A RenderContext instance.

## See Also

OvalGaugeRenderer Members
OvalGaugeRenderer Class
MindFusion.Charting.Gauges Namespace

# OvalGaugeRenderer Properties

# OvalGaugeRenderer.gauge Property

Gets or sets the OvalGauge represented by this renderer.

**Namespace:** MindFusion.Charting.Gauges
**File:** OvalGaugeRenderer.js

## Syntax

```
JavaScript
get gauge() {}
```

### Property Value

Type: object

## See Also

OvalGaugeRenderer Members
OvalGaugeRenderer Class
MindFusion.Charting.Gauges Namespace

# MindFusion.Charting.ThreeD Namespace

## Label3D Class

Represents a text label in 3D scene.

## Syntax

```
JavaScript
// class
Label3D.prototype = {}
```

## Inheritance Hierarchy

## See Also

Label3D Members
MindFusion.Charting.ThreeD Namespace

## Label3D Members

Represents a text label in 3D scene.

The following tables list the members exposed by the Label3D type.

## Public Properties

| | Name | Description |
|---|---|---|
|  | bottomLeft | Gets or sets the bottom-left corner of label's layout rectangle. |
|  | font | Gets or sets the label's font. |
|  | rotationAngle | Gets or sets the label's rotation angle. |
|  | text | Gets or sets the label's text. |
|  | topRight | Gets or sets the top-right corner of label's layout rectangle. |

## See Also

Label3D Class
MindFusion.Charting.ThreeD Namespace

# Label3D Properties

## Label3D.bottomLeft Property

Gets or sets the bottom-left corner of label's layout rectangle.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Label3D.js

## Syntax

```
JavaScript
get bottomLeft() {}
```

### Property Value

Type: Point3D

## See Also

Label3D Members
Label3D Class
MindFusion.Charting.ThreeD Namespace

## Label3D.font Property

Gets or sets the label's font.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Label3D.js

## Syntax

```
JavaScript
get font() {}
```

### Property Value

Type: Font

## See Also

Label3D Members
Label3D Class
MindFusion.Charting.ThreeD Namespace

# Label3D.rotationAngle Property

Gets or sets the label's rotation angle.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Label3D.js

## Syntax

**JavaScript**

```
get rotationAngle() {}
```

### Property Value

Type: Number

## See Also

Label3D Members
Label3D Class
MindFusion.Charting.ThreeD Namespace

# Label3D.text Property

Gets or sets the label's text.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Label3D.js

## Syntax

**JavaScript**

```
get text() {}
```

### Property Value

Type: String

## See Also

Label3D Members
Label3D Class
MindFusion.Charting.ThreeD Namespace

# Label3D.topRight Property

Gets or sets the top-right corner of label's layout rectangle.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Label3D.js

## Syntax

**JavaScript**

```
get topRight() {}
```

### Property Value

Type: Point3D

## See Also

Label3D Members
Label3D Class
MindFusion.Charting.ThreeD Namespace

# LabelProjection Class

Represents the projection of a Label3D on the projection plane.

## Syntax

**JavaScript**

```
// class
LabelProjection.prototype = {}
```

## Inheritance Hierarchy

## See Also

LabelProjection Members
MindFusion.Charting.ThreeD Namespace

## LabelProjection Members

Represents the projection of a Label3D on the projection plane.

The following tables list the members exposed by the LabelProjection type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | LabelProjection | Initializes a new instance of the LabelProjection class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | label | Gets or sets the projected Label3D. |
|  | projectionBottomRight | Gets or sets the projection of bottom-right corner of label's layout rectangle. |
|  | projectionTopLeft | Gets or sets the projection of top-left corner of label's layout rectangle. |

| | | |
|---|---|---|
| | zSort | Gets or sets a value by which projections are sorted in depth buffer. |

## Public Methods

| | Name | Description |
|---|---|---|
| | drawAsProjection | Draws this projection in specified RenderContext. |

## See Also

LabelProjection Class
MindFusion.Charting.ThreeD Namespace

# LabelProjection Constructor

Initializes a new instance of the LabelProjection class.

**Namespace:** MindFusion.Charting.ThreeD
**File:** LabelProjection.js

## Syntax

**JavaScript**

```
function LabelProjection (label, projTL, projBR)
```

### Parameters

*label*
Type: Label3D
The projected Label3D.

*projTL*
Type: Point3D
Projection of top-left corner of Label3D's layout rectangle.

*projBR*
Type: Point3D
Projection of bottom-right corner of Label3D's layout rectangle.

## See Also

LabelProjection Class
MindFusion.Charting.ThreeD Namespace

# LabelProjection Methods

# LabelProjection.drawAsProjection Method

Draws this projection in specified RenderContext.

**Namespace:** MindFusion.Charting.ThreeD
**File:** LabelProjection.js

## Syntax

```
JavaScript
function drawAsProjection (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

LabelProjection Members
LabelProjection Class
MindFusion.Charting.ThreeD Namespace

# LabelProjection Properties

## LabelProjection.label Property

Gets or sets the projected Label3D.

**Namespace:** MindFusion.Charting.ThreeD
**File:** LabelProjection.js

## Syntax

```
JavaScript
get label() {}
```

### Property Value

Type: Label3D

## See Also

LabelProjection Members
LabelProjection Class
MindFusion.Charting.ThreeD Namespace

## LabelProjection.projectionBottomRight Property

Gets or sets the projection of bottom-right corner of label's layout rectangle.

**Namespace:** MindFusion.Charting.ThreeD
**File:** LabelProjection.js

## Syntax

```
JavaScript
get projectionBottomRight() {}
```

### Property Value

Type: Point3D

## See Also

LabelProjection Members
LabelProjection Class
MindFusion.Charting.ThreeD Namespace

# LabelProjection.projectionTopLeft Property

Gets or sets the projection of top-left corner of label's layout rectangle.

**Namespace:** MindFusion.Charting.ThreeD
**File:** LabelProjection.js

## Syntax

**JavaScript**

```
get projectionTopLeft() {}
```

### Property Value

Type: Point3D

## See Also

LabelProjection Members
LabelProjection Class
MindFusion.Charting.ThreeD Namespace

# LabelProjection.zSort Property

Gets or sets a value by which projections are sorted in depth buffer.

**Namespace:** MindFusion.Charting.ThreeD
**File:** LabelProjection.js

## Syntax

**JavaScript**

```
get zSort() {}
```

### Property Value

Type: Number

## See Also

LabelProjection Members
LabelProjection Class
MindFusion.Charting.ThreeD Namespace

# Matrix3D Class

Represents a a mathematical matrix, specialized in 3D transformations.

## Syntax

**JavaScript**

```
// class
Matrix3D.prototype = {}
```

## Inheritance Hierarchy

## See Also

Matrix3D Members
MindFusion.Charting.ThreeD Namespace

# Matrix3D Members

Represents a a mathematical matrix, specialized in 3D transformations.

The following tables list the members exposed by the Matrix3D type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | Matrix3D | Initializes a new instance of the Matrix3D class. |

## Public Methods

| | Name | Description |
|---|---|---|
|  | fill | Fills the matrix with an array, starting from the top left, filling horisontally. |
|  | Identety | A 4x4 Identety Matrix |
|  | multiply | Overloaded. |
|  | nullify | Fills the matrix with zeros. |
|  | Perspective | A 4x4 matrix, allowing perspective and vercitile coordinates. |
|  | RotateX | A 4x4 matrix, allowing rotation around the X axis. |
|  | RotateY | A 4x4 matrix, allowing rotation around the Y axis. |
|  | RotateZ | A 4x4 matrix, allowing rotation around the Z axis. |
|  | Scale | A 4x4 matrix, allowing scaling. |
|  | Translate | A 4x4 matrix, allowing translation. |

## See Also

Matrix3D Class
MindFusion.Charting.ThreeD Namespace

# Matrix3D Constructor

Initializes a new instance of the Matrix3D class.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Matrix3D.js

## Syntax

**JavaScript**

```
function Matrix3D (width, height)
```

### Parameters

*width*

Type: number
The width of the matrix.

*height*

Type: number
The height of the matrix.

## See Also

Matrix3D Class
MindFusion.Charting.ThreeD Namespace

# Matrix3D Methods

# Matrix3D.fill Method

Fills the matrix with an array, starting from the top left, filling horisontally.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Matrix3D.js

## Syntax

**JavaScript**

```
function fill (array)
```

### Parameters

*array*

## See Also

Matrix3D Members
Matrix3D Class
MindFusion.Charting.ThreeD Namespace

# Matrix3D.Identety Method

A 4x4 Identety Matrix

**Namespace:** MindFusion.Charting.ThreeD
**File:** Matrix3D.js

## Syntax

| JavaScript |
| --- |
| `function Identety ()` |

## See Also

Matrix3D Members
Matrix3D Class
MindFusion.Charting.ThreeD Namespace

# Matrix3D.multiply Method

## Overload List

| Name | Description |
| --- | --- |
| Matrix3D.multiply | Multiplys the current matrix times a second matrix. |
| Matrix3D.multiply | Multiplys 2 matrices. |

## See Also

Matrix3D Members
Matrix3D Class
MindFusion.Charting.ThreeD Namespace

# Matrix3D.multiply Method

Multiplys the current matrix times a second matrix.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Matrix3D.js

## Syntax

| JavaScript |
| --- |
| `function multiply (mat)` |

### Parameters

*mat*
Type: Matrix3D
The right matrix.

## See Also

multiply Method Overload List
Matrix3D Members
Matrix3D Class
MindFusion.Charting.ThreeD Namespace

# Matrix3D.multiply Method

Multiplys 2 matrices.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Matrix3D.js

## Syntax

**JavaScript**

```
function multiply (m, n)
```

### Parameters

*m*
Type: Matrix3D
The left matrix.

*n*
Type: Matrix3D
The right matrix.

## See Also

multiply Method Overload List
Matrix3D Members
Matrix3D Class
MindFusion.Charting.ThreeD Namespace

# Matrix3D.nullify Method

Fills the matrix with zeros.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Matrix3D.js

## Syntax

**JavaScript**

```
function nullify ()
```

## See Also

Matrix3D Members
Matrix3D Class
MindFusion.Charting.ThreeD Namespace

# Matrix3D.Perspective Method

A 4x4 matrix, allowing perspective and vercitile coordinates.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Matrix3D.js

## Syntax

**JavaScript**

```
function Perspective (filedOfView, aspectRatio, near, far)
```

### Parameters

*filedOfView*
Type: number
The angle of the camera.

*aspectRatio*
Type: number
The width devided by height of the viewing window.

*near*
Type: number
The nearest coordinate, where the object will still be drawn.

*far*
Type: number
The furthest coordinate, where the object will still be drawn.

## See Also

Matrix3D Members
Matrix3D Class
MindFusion.Charting.ThreeD Namespace

# Matrix3D.RotateX Method

A 4x4 matrix, allowing rotation around the X axis.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Matrix3D.js

## Syntax

**JavaScript**

```
function RotateX (angle)
```

### Parameters

*angle*
Type: number
The amount of rotation around the X axis in radians.

## See Also

Matrix3D Members
Matrix3D Class

# Matrix3D.RotateY Method

A 4x4 matrix, allowing rotation around the Y axis.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Matrix3D.js

## Syntax

**JavaScript**

```
function RotateY (angle)
```

### Parameters

*angle*
Type: number
The amount of rotation around the Y axis in radians.

## See Also

Matrix3D Members
Matrix3D Class
MindFusion.Charting.ThreeD Namespace

# Matrix3D.RotateZ Method

A 4x4 matrix, allowing rotation around the Z axis.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Matrix3D.js

## Syntax

**JavaScript**

```
function RotateZ (angle)
```

### Parameters

*angle*
Type: number
The amount of rotation around the Z axis in radians.

## See Also

Matrix3D Members
Matrix3D Class
MindFusion.Charting.ThreeD Namespace

# Matrix3D.Scale Method

A 4x4 matrix, allowing scaling.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Matrix3D.js

## Syntax

```
JavaScript
function Scale (width, height, depth)
```

### Parameters

*width*
Type: number
The amount of scaling on the X axis in percent/100.

*height*
Type: number
The amount of scaling on the Y axis in percent/100.

*depth*
Type: number
The amount of scaling on the Z axis in percent/100.

## See Also

Matrix3D Members
Matrix3D Class
MindFusion.Charting.ThreeD Namespace

# Matrix3D.Translate Method

A 4x4 matrix, allowing translation.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Matrix3D.js

## Syntax

```
JavaScript
function Translate (x, y, z)
```

### Parameters

*x*
Type: number
The amount of translation on the X axis.

*y*
Type: number
The amount of translation on the Y axis.

*z*
Type: number
The amount of translation on the Z axis.

## See Also

Matrix3D Members
Matrix3D Class

MindFusion.Charting.ThreeD Namespace

# Mesh3D Class

Represents a 3D mesh.

## Syntax

```
JavaScript
// class
Mesh3D.prototype = {}
```

## Inheritance Hierarchy

## See Also

Mesh3D Members
MindFusion.Charting.ThreeD Namespace

# Mesh3D Members

Represents a 3D mesh.

The following tables list the members exposed by the Mesh3D type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | Mesh3D | Initializes a new instance of the Mesh3D class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | faces | Gets or sets the mesh faces. |

## See Also

Mesh3D Class
MindFusion.Charting.ThreeD Namespace

# Mesh3D Constructor

Initializes a new instance of the Mesh3D class.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Mesh3D.js

## Syntax

```
JavaScript
```

```
function Mesh3D (facePoints, brush, [normals])
```

### Parameters

*facePoints*
Type: List>
Lists of points defining mesh faces.

*brush*
Type: Brush
The Brush used to paint this mesh.

*normals*
Optional.
Type: Array>
An array with the normal vectors in each vertex.

## See Also

Mesh3D Class
MindFusion.Charting.ThreeD Namespace

# Mesh3D Properties

## Mesh3D.faces Property

Gets or sets the mesh faces.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Mesh3D.js

## Syntax

**JavaScript**

```
get faces() {}
```

### Property Value

Type: List

## See Also

Mesh3D Members
Mesh3D Class
MindFusion.Charting.ThreeD Namespace

# Model3D Class

Represents a three-dimensional model.

## Syntax

**JavaScript**

```
// class
Model3D.prototype = {}
```

## Inheritance Hierarchy

## See Also

Model3D Members
MindFusion.Charting.ThreeD Namespace

# Model3D Members

Represents a three-dimensional model.

The following tables list the members exposed by the Model3D type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | Model3D | Initializes a new instance of the Model3D class. |

## See Also

Model3D Class
MindFusion.Charting.ThreeD Namespace

# Model3D Constructor

Initializes a new instance of the Model3D class.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Model3D.js

## Syntax

| JavaScript |
|---|
| `function Model3D ([brush])` |

### Parameters

*brush*
Optional.
Type: Brush
The Brush used to fill this model's projection.

## See Also

Model3D Class
MindFusion.Charting.ThreeD Namespace

# Polygon3D Class

Represents a polygon in 3D space.

## Syntax

**JavaScript**

```
// class
Polygon3D.prototype = {}
```

## Inheritance Hierarchy

## See Also

Polygon3D Members
MindFusion.Charting.ThreeD Namespace

# Polygon3D Members

Represents a polygon in 3D space.

The following tables list the members exposed by the Polygon3D type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | Polygon3D | Initializes a new instance of the Polygon3D class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | points | Gets or sets the polygon's vertices. |
| | zSort | Gets or sets a value by which projections are sorted in depth buffer. |

## Public Methods

| | Name | Description |
|---|---|---|
| | drawAsProjection | Draws this projection in specified RenderContext. |
| | midPoint | Returns the polygon's middle point. |
| | normal | Gets the polygon's normal vector. |
| S | pointToPlaneDist | Overloaded. |

## See Also

Polygon3D Class
MindFusion.Charting.ThreeD Namespace

# Polygon3D Constructor

Initializes a new instance of the Polygon3D class.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Polygon3D.js

## Syntax

**JavaScript**

```
function Polygon3D ([points, [brush, [normals]]])
```

### Parameters

*points*
Optional.
Type: IEnumerable
A list of 3D points.

*brush*
Optional.
Type: Brush
A Brush used to fill the polygon.

*normals*
Optional.
Type: Array
An array with the normal vectors in each vertex.

## See Also

Polygon3D Class
MindFusion.Charting.ThreeD Namespace

# Polygon3D Methods

# Polygon3D.drawAsProjection Method

Draws this projection in specified RenderContext.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Polygon3D.js

## Syntax

**JavaScript**

```
function drawAsProjection (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

Polygon3D Members
Polygon3D Class
MindFusion.Charting.ThreeD Namespace

# Polygon3D.midPoint Method

Returns the polygon's middle point.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Polygon3D.js

## Syntax

```
JavaScript
function midPoint ()
```

### Return Value

Type: Point3D
A Point3D instance representing the middle point.

## See Also

Polygon3D Members
Polygon3D Class
MindFusion.Charting.ThreeD Namespace

# Polygon3D.normal Method

Gets the polygon's normal vector.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Polygon3D.js

## Syntax

```
JavaScript
function normal ()
```

### Return Value

Type: Vector3D
A Vector3D representing the polygon's normal vector.

## See Also

Polygon3D Members
Polygon3D Class
MindFusion.Charting.ThreeD Namespace

# Polygon3D.pointToPlaneDist Method

## Overload List

| Name | Description |
| --- | --- |
| Polygon3D.pointToPlaneDist | Gets the distance from specified point to polygon's plane. |
| Polygon3D.pointToPlaneDist | Gets the distance from specified point to specified plane. |

## See Also

Polygon3D Members
Polygon3D Class
MindFusion.Charting.ThreeD Namespace

# Polygon3D.pointToPlaneDist Method

Gets the distance from specified point to polygon's plane.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Polygon3D.js

## Syntax

**JavaScript**

```
function pointToPlaneDist (p)
```

### Parameters

*p*
Type: Point3D
A Point3D instance.

### Return Value

Type: Number
A number value representing the distance.

## See Also

pointToPlaneDist Method Overload List
Polygon3D Members
Polygon3D Class
MindFusion.Charting.ThreeD Namespace

# Polygon3D.pointToPlaneDist Method

Gets the distance from specified point to specified plane.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Polygon3D.js

## Syntax

**JavaScript**

```
function pointToPlaneDist (p, planePoint, planeNormal)
```

### Parameters

*p*
Type: Point3D
A Point3D instance.

*planePoint*
Type: Point3D
A point from the plane.

*planeNormal*
Type: Vector3D
The plane's normal vector.

### Return Value

Type: Number
A number value representing the distance.

## See Also

pointToPlaneDist Method Overload List
Polygon3D Members
Polygon3D Class
MindFusion.Charting.ThreeD Namespace

# Polygon3D Properties

# Polygon3D.points Property

Gets or sets the polygon's vertices.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Polygon3D.js

## Syntax

**JavaScript**

```
get points() {}
```

### Property Value

Type: List

## See Also

Polygon3D Members
Polygon3D Class
MindFusion.Charting.ThreeD Namespace

# Polygon3D.zSort Property

Gets or sets a value by which projections are sorted in depth buffer.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Polygon3D.js

## Syntax

```
JavaScript
get zSort() {}
```

### Property Value

Type: Number

## See Also

Polygon3D Members
Polygon3D Class
MindFusion.Charting.ThreeD Namespace

# Projection Interface

Defines the interface of 3D model projections.

## Syntax

```
JavaScript
// interface
Projection.prototype = {}
```

## See Also

Projection Members
MindFusion.Charting.ThreeD Namespace

### Projection Members

Defines the interface of 3D model projections.

The following tables list the members exposed by the Projection type.

## Public Properties

| | Name | Description |
|---|---|---|
|  | zSort | A value by which projections are sorted in depth buffer. |

## Public Methods

| | Name | Description |
|---|---|---|
|  | drawAsProjection | Draws the projections in specified RenderContext. |

## See Also

# Projection Methods

## Projection.drawAsProjection Method

Draws the projections in specified RenderContext.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Projection.js

### Syntax

```
JavaScript
function drawAsProjection (context)
```

#### Parameters

*context*
Type: RenderContext
A RenderContext instance.

### See Also

Projection Members
Projection Interface
MindFusion.Charting.ThreeD Namespace

# Projection Properties

## Projection.zSort Property

A value by which projections are sorted in depth buffer.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Projection.js

### Syntax

```
JavaScript
get zSort() {}
```

#### Property Value

Type: Number

### See Also

Projection Members
Projection Interface
MindFusion.Charting.ThreeD Namespace

# Scene3D Class

Represents a 3D scene.

## Syntax

| JavaScript |
| --- |
| `// class`<br>`Scene3D.prototype = {}` |

## Inheritance Hierarchy

## See Also

Scene3D Members
MindFusion.Charting.ThreeD Namespace

# Scene3D Members

Represents a 3D scene.

The following tables list the members exposed by the Scene3D type.

## Public Constructors

| | Name | Description |
| --- | --- | --- |
| | Scene3D | Initializes a new instance of the Scene3D class. |

## Public Properties

| | Name | Description |
| --- | --- | --- |
| | cameraOrientation | Gets or sets the camera orientation. |
| | cameraPosition | Gets or sets the camera position. |
| | focalLength | Gets or sets the camera focal length. |
| | models | Gets or sets a list of models in this scene. |
| | projections | Gets or sets a list of projections in this scene. |
| | viewportCenter | Gets or sets the viewport center. |

## Public Methods

| | Name | Description |
| --- | --- | --- |
| | buildCuboid | Creates a cuboid model. |
| | draw | Draws a projection of this scene in specified RenderContext. |

## See Also

Scene3D Class
MindFusion.Charting.ThreeD Namespace

# Scene3D Constructor

Initializes a new instance of the Scene3D class.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Scene3D.js

## Syntax

**JavaScript**

```
function Scene3D ()
```

## See Also

Scene3D Class
MindFusion.Charting.ThreeD Namespace

# Scene3D Methods

# Scene3D.buildCuboid Method

Creates a cuboid model.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Scene3D.js

## Syntax

**JavaScript**

```
function buildCuboid (x1, y1, z1, x2, y2, z2, brush)
```

### Parameters

*x1*
Type: Number
X coordinate of a corner point of the cuboid.

*y1*
Type: Number
Y coordinate of a corner point of the cuboid.

*z1*
Type: Number
Z coordinate of a corner point of the cuboid.

*x2*
Type: Number
X coordinate of diametrically opposite corner.

*y2*
Type: Number
Y coordinate of diametrically opposite corner.

*z2*

Type: Number
Z coordinate of diametrically opposite corner.

*brush*
Type: Brush
The Brush used to paint the cuboid model.

### Return Value

Type: Mesh3D
A Mesh3D containing the cuboid faces.

## See Also

Scene3D Members
Scene3D Class
MindFusion.Charting.ThreeD Namespace

# Scene3D.draw Method

Draws a projection of this scene in specified RenderContext.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Scene3D.js

## Syntax

**JavaScript**

```
function draw (context)
```

### Parameters

*context*
Type: RenderContext
A RenderContext instance.

## See Also

Scene3D Members
Scene3D Class
MindFusion.Charting.ThreeD Namespace

# Scene3D Properties

# Scene3D.cameraOrientation Property

Gets or sets the camera orientation.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Scene3D.js

## Syntax

**JavaScript**

```
get cameraOrientation() {}
```

### Property Value

Type: Point3D

## See Also

Scene3D Members
Scene3D Class
MindFusion.Charting.ThreeD Namespace

# Scene3D.cameraPosition Property

Gets or sets the camera position.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Scene3D.js

## Syntax

```
JavaScript
get cameraPosition() {}
```

### Property Value

Type: Point3D

## See Also

Scene3D Members
Scene3D Class
MindFusion.Charting.ThreeD Namespace

# Scene3D.focalLength Property

Gets or sets the camera focal length.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Scene3D.js

## Syntax

```
JavaScript
get focalLength() {}
```

### Property Value

Type: Number

## See Also

Scene3D Members
Scene3D Class
MindFusion.Charting.ThreeD Namespace

# Scene3D.models Property

Gets or sets a list of models in this scene.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Scene3D.js

## Syntax

```
JavaScript
get models() {}
```

### Property Value

Type: List

## See Also

Scene3D Members
Scene3D Class
MindFusion.Charting.ThreeD Namespace

# Scene3D.projections Property

Gets or sets a list of projections in this scene.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Scene3D.js

## Syntax

```
JavaScript
get projections() {}
```

### Property Value

Type: List

## See Also

Scene3D Members
Scene3D Class
MindFusion.Charting.ThreeD Namespace

# Scene3D.viewportCenter Property

Gets or sets the viewport center.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Scene3D.js

## Syntax

```
JavaScript
```

```
get viewportCenter() {}
```

### Property Value

Type: Point

## See Also

Scene3D Members
Scene3D Class
MindFusion.Charting.ThreeD Namespace

# Vector3D Class

Represents a three-dimensional vector.

## Syntax

**JavaScript**

```
// class
Vector3D.prototype = {}
```

## Inheritance Hierarchy

## See Also

Vector3D Members
MindFusion.Charting.ThreeD Namespace

# Vector3D Members

Represents a three-dimensional vector.

The following tables list the members exposed by the Vector3D type.

## Public Properties

| | Name | Description |
|---|---|---|
| | x | Gets or sets the X component of this vector. |
| | y | Gets or sets the Y component of this vector. |
| | z | Gets or sets the Z component of this vector. |

## Public Methods

| | Name | Description |
|---|---|---|
| S | crossProduct | Calculates the cross-product of two vectors. |
| | div | Divides the vector by specified measure. |
| | length | Calculates the length of this vector. |

## See Also

Vector3D Class
MindFusion.Charting.ThreeD Namespace

# Vector3D Methods

## Vector3D.crossProduct Method

Calculates the cross-product of two vectors.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Vector3D.js

## Syntax

**JavaScript**

```
function crossProduct (a, b)
```

### Parameters

*a*
Type: Vector3D
A Vector3D instance.

*b*
Type: Vector3D
A Vector3D instance.

### Return Value

Type: Vector3D
A Vector3D representing the cross-product of specified vectors.

## See Also

Vector3D Members
Vector3D Class
MindFusion.Charting.ThreeD Namespace

## Vector3D.div Method

Divides the vector by specified measure.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Vector3D.js

## Syntax

**JavaScript**

```
function div (s)
```

### Parameters

*s*

Type: Number
A double value representing the divisor.

### Return Value

Type: Vector3D
A Vector3D representing the result.

## See Also

Vector3D Members
Vector3D Class
MindFusion.Charting.ThreeD Namespace

# Vector3D.length Method

Calculates the length of this vector.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Vector3D.js

## Syntax

**JavaScript**

```
function length ()
```

### Return Value

Type: Number
A double value representing the vector length.

## See Also

Vector3D Members
Vector3D Class
MindFusion.Charting.ThreeD Namespace

# Vector3D Properties

# Vector3D.x Property

Gets or sets the X component of this vector.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Vector3D.js

## Syntax

**JavaScript**

```
get x() {}
```

### Property Value

Type: Number

## See Also

Vector3D Members
Vector3D Class
MindFusion.Charting.ThreeD Namespace

# Vector3D.y Property

Gets or sets the Y component of this vector.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Vector3D.js

## Syntax

**JavaScript**

```
get y() {}
```

### Property Value

Type: Number

## See Also

Vector3D Members
Vector3D Class
MindFusion.Charting.ThreeD Namespace

# Vector3D.z Property

Gets or sets the Z component of this vector.

**Namespace:** MindFusion.Charting.ThreeD
**File:** Vector3D.js

## Syntax

**JavaScript**

```
get z() {}
```

### Property Value

Type: Number

## See Also

Vector3D Members
Vector3D Class
MindFusion.Charting.ThreeD Namespace

# MindFusion.Common Namespace

# Control Class

A base class for UI controls.

## Syntax

**JavaScript**

```
// class
Control.prototype = {}
```

## Example

The following code creates a new instance of the **ListView** control, which inherits the **Control** class. Note that the constructor uses a DOM element with an id listView:

**JavaScript**

```
var ui = MindFusion.UI;

var listView;
var host;
var theme = "gray";

document.addEventListener("DOMContentLoaded", function ()
{
 listView = new ui.ListView(document.getElementById("listView"));
 listView.width = listView.height = ui.Unit.percentage(100);
 listView.theme = theme;
 listView.itemSize = ui.Unit.pixel(100);
 listView.itemDoubleClick.addEventListener(handleItemDoubleClick);
 ....................................
 ....................................
}
```

## Inheritance Hierarchy

**MindFusion.Common.Control**

## See Also

Control Members
MindFusion.Common Namespace

# Control Members

A base class for UI controls.

The following tables list the members exposed by the Control type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | Control | Initializes a new instance of the Control class. |

## Public Properties

| | Name | Description |
|---|---|---|

| | | |
|---|---|---|
| | bounds | Gets the bounds of this control. |
| | cssClass | Gets or sets the css class of the control. |
| | data | Gets or sets an object, holding custom user data. |
| | element | Gets a reference to the control's element. |
| | enabled | Gets or sets a value indicating whether user interactions are allowed for this control. |
| | height | Gets or sets the height of this **control**. |
| | left | Gets or sets the X-coordinate of the location of this **control**. |
| | licenseLocation | Gets or sets the URL of the **control's** license file. |
| | loaded | Gets a value indicating whether this control is loaded and ready for interaction. |
| | rect | Gets the bounding rect of this **control**. |
| | theme | Gets or sets the current theme of the control. |
| | top | Gets or sets the Y-coordinate of the location of this **control**. |
| | visible | Gets or sets the visibility of this **control**. |
| | width | Gets or sets the width of this **control**. |

## Public Methods

| | Name | Description |
|---|---|---|
| | attach | Attach control event handlers. |
| | detach | Detach control event handlers. |
| | dispose | Dispose the control. |
| | draw | Draws the control. |
| S | find | Returns the control with the specified ID. |
| | render | Draws the control and prepares it for user interaction. |

## Public Events

| | Name | Description |
|---|---|---|
| | controlLoad | Raised when the **control** is loaded. |
| | controlUnload | Raised when the **control** is unloaded. |
| | focus | Raised when the **control** is focused. |

## See Also

Control Class
MindFusion.Common Namespace

# Control Constructor

Initializes a new instance of the Control class.

**Namespace:** MindFusion.Common
**File:** Control.js

## Syntax

**JavaScript**

```
function Control ([element])
```

### Parameters

*element*
Optional.
Type: HTMLElement
HTMLElement. The control's associated Dom element.

## Example

The following code creates a new instance of the **ListView** control, which inherits the Control class. Note that the constructor uses a DOM element with an id listView:

**JavaScript**

```
var ui = MindFusion.UI;

var listView;
var host;
var theme = "gray";

document.addEventListener("DOMContentLoaded", function ()
{
 listView = new ui.ListView(document.getElementById("listView"));
 listView.width = listView.height = ui.Unit.percentage(100);
 listView.theme = theme;
 listView.itemSize = ui.Unit.pixel(100);
 listView.itemDoubleClick.addEventListener(handleItemDoubleClick);
 ....................................
 ....................................
}
```

## See Also

Control Class
MindFusion.Common Namespace

# Control Methods

# Control.attach Method

Attach control event handlers.

**Namespace:** MindFusion.Common
**File:** Control.js

## Syntax

JavaScript

```
function attach ()
```

## Remarks

The attach method attaches the event handlers of a UI element. It is used when creating UI controls dynamically in code.

## Example

The following code uses the attach method in a method that creates a **ToolStrip** programmatically:

JavaScript

```
function loadChatWindow(sender)
{
 var tools = new ui.ToolStrip();
 tools.theme = theme;
 tools.scrollable = false;
 tools.allowDrag = false;
 tools.itemClick.addEventListener(onToolClick);

 button = new ui.ToolStripItem(ui.ToolStripItemType.Icon);
 button.imageSrc = "./images/icon_home.png";
 button.tooltip = "home";
 tools.items.add(button);
 ....................................
 ....................................
 sender.element.querySelector("#toolstrip").appendChild(tools.draw());
 tools.attach();
}
```

## See Also

Control Members
Control Class
MindFusion.Common Namespace

# Control.detach Method

Detach control event handlers.

**Namespace:** MindFusion.Common
**File:** Control.js

## Syntax

JavaScript

```
function detach ()
```

## Remarks

The detach method dettaches the event handlers of a UI Control.

## See Also

Control Members
Control Class

# Control.dispose Method

Dispose the control.

**Namespace:** MindFusion.Common
**File:** Control.js

## Syntax

**JavaScript**

```
function dispose ()
```

## Remarks

Detaches control event handlers and removes its element from the DOM, if it was internally created.

## Example

The following code uses the dispose method to hide a Clock gauge that is no longer rendered on the web page.

**JavaScript**

```
function stopClock()
{
 clearInterval(timerId);

 document.getElementById("clock").parentNode.style.visibility = "hidden";

 var clock = g.OvalGauge.find("clock");
 if (clock)
  clock.dispose();
}
```

## See Also

Control Members
Control Class
MindFusion.Common Namespace

# Control.draw Method

Draws the control.

**Namespace:** MindFusion.Common
**File:** Control.js

## Syntax

**JavaScript**

```
function draw ()
```

### Return Value

Type: HTMLElement
HTMLElement. The control DOM element.

## Example

The following code creates a new **ListView** control and calls its draw method to render it on the web page.

**JavaScript**

```
// Create a new ListView control.
  // This list view will serve as a drag source.
  var list1 = new ui.ListView();
  list1.theme = "business";
  // Load the items data
  list1.fromObject(data);

  // Add event handlers
  list1.itemDragStart.addEventListener(listDragStart);
  list1.itemDragEnd.addEventListener(listDragEnd);

  // Draw the control and append it to the page DOM.
  document.body.appendChild(list1.draw());
  // Prepare the control for user interaction.
  list1.attach();
```

## See Also

Control Members
Control Class
MindFusion.Common Namespace

# Control.find Method

Returns the control with the specified ID.

**Namespace:** MindFusion.Common
**File:** Control.js

## Syntax

**JavaScript**

```
static find (id)
```

### Parameters

*id*
Type: String
String. The ID.

### Return Value

Control. The control with the specified ID if found, or **null** otherwise.

## Example

The following code uses the find method to search for an oval Gauge that must be disposed because it is no longer rendered on the web page.

**JavaScript**

```
function stopClock()
{
 clearInterval(timerId);

 document.getElementById("clock").parentNode.style.visibility = "hidden";

 var clock = g.OvalGauge.find("clock");
 if (clock)
  clock.dispose();
}
```

## See Also

Control Members
Control Class
MindFusion.Common Namespace

# Control.render Method

Draws the control and prepares it for user interaction.

**Namespace:** MindFusion.Common
**File:** Control.js

## Syntax

**JavaScript**

```
function render ()
```

## Example

The following code uses the render method to show a **ListView** that was created programmatically:

**JavaScript**

```
var ui = MindFusion.UI;

var listView;
var host;
var theme = "gray";

document.addEventListener("DOMContentLoaded", function ()
{
 listView = new ui.ListView(document.getElementById("listView"));
 listView.width = listView.height = ui.Unit.percentage(100);
 listView.theme = theme;
 listView.itemSize = ui.Unit.pixel(100);
 listView.itemDoubleClick.addEventListener(handleItemDoubleClick);
 ....................................
 ....................................
 listView.render();
}
```

## See Also

Control Members
Control Class
MindFusion.Common Namespace

# Control Properties

## Control.bounds Property

Gets the bounds of this control.

**Namespace:** MindFusion.Common
**File:** Control.js

### Syntax

```
JavaScript
get bounds() {}
```

#### Property Value

Type: Rect
Rect. The bounds.

### Example

The following code uses the bounds property of a **ToolStrip** to render a menu right before the tool item, over which the mouse hovers:

```
JavaScript
// This is the handler function of the toolstrip.itemMouseEnter event.
// If the menu is currently visible it will be removed.
// Otherwise, the menu will be created right below the hovered item.
function itemMouseEnter(sender, args)
{
 if (menu.loaded)
 {
  menu.dispose();
 }
 else if (args.item.title == "Theme")
 {
  menu.theme = sender.theme;
  menu.left = ui.Unit.pixel(sender.bounds.x + args.item.bounds.x);
  menu.top = ui.Unit.pixel(sender.bounds.y + args.item.bounds.height);

  document.body.appendChild(menu.draw());
  menu.attach();
 }
}
```

### See Also

Control Members
Control Class
MindFusion.Common Namespace

## Control.cssClass Property

Gets or sets the css class of the control.

**Namespace:** MindFusion.Common
**File:** Control.js

### Syntax

```
JavaScript
get cssClass() {}
```

### Property Value

Type: String
String. The name of the css class of the control.

## Example

The following code uses the cssClass property to specifies custom classes to **Item** instances of a **schedule**.

```
JavaScript
for (var i = 0; i < 5; i++)
 {
  var item = new p.Item();
  item.startTime = date.clone().addHours(Math.floor(Math.random() * 10) + 6);
  item.endTime = item.startTime.clone().addHours(1);
  var activity = calendar.schedule.resources.items()[0];
  item.subject = activity.name;
  item.resources.add(activity);
  item.cssClass = "itemClass" + activity.id;
  calendar.schedule.items.add(item);
 }
```

## See Also

Control Members
Control Class
MindFusion.Common Namespace

# Control.data Property

Gets or sets an object, holding custom user data.

**Namespace:** MindFusion.Common
**File:** Control.js

## Syntax

```
JavaScript
get data() {}
```

### Property Value

Type: Object
Object. The custom user data.

## Example

The following code uses the data property in the event handler of the **itemDrag** event of a **TreeView** to assign the data of the event sender to a **ListItem**:

```
JavaScript
function nodeDrag(sender, args)
{
```

```
 if (sender.data)
 {
  var item = new ui.ListItem(sender.title);
  item.data = sender.data;
  listView.acceptDrop = item;
 }
}
```

## See Also

Control Members
Control Class
MindFusion.Common Namespace

# Control.element Property

Gets a reference to the control's element.

**Namespace:** MindFusion.Common
**File:** Control.js

## Syntax

**JavaScript**

```
get element() {}
```

### Property Value

Type: HTMLElement
HTMLElement

## Example

The following code uses the element of a **TreeView** to create a ToolTip instance for the tree view.

**JavaScript**

```
tip = new ui.Tooltip(treeView.element, text);
tip.position = ui.TooltipPosition.Left;
tip.theme = theme;
tip.render();
```

## See Also

Control Members
Control Class
MindFusion.Common Namespace

# Control.enabled Property

Gets or sets a value indicating whether user interactions are allowed for this control.

**Namespace:** MindFusion.Common
**File:** Control.js

## Syntax

JavaScript

```
get enabled() {}
```

### Property Value

Type: Boolean
Boolean. True if user interactions are allowed, otherwise false.

## Example

The following code creates a new **TabPage** and disables it:

JavaScript

```
var tab = new ui.TabPage("&#xab;");
tab.header.size = new ui.Unit(20);
tab.header.interactive = false;
tab.header.tooltip = "Previous Tab";
tab.data = -1;
tab.enabled = false;
tab.header.cssClass = "tab-back";
tabControl.tabs.add(tab);
```

## See Also

Control Members
Control Class
MindFusion.Common Namespace

# Control.height Property

Gets or sets the height of this control.

**Namespace:** MindFusion.Common
**File:** Control.js

## Syntax

JavaScript

```
get height() {}
```

### Property Value

Unit. The height.

## Example

The following code creates a new instance of the ListView control and sets its width and height to 100 % using the Unit class.

JavaScript

```
var ui = MindFusion.UI;

var listView;
var host;
var theme = "gray";

document.addEventListener("DOMContentLoaded", function ()
{
```

```
 listView = new ui.ListView(document.getElementById("listView"));
 listView.width = listView.height = ui.Unit.percentage(100);
 listView.theme = theme;
 listView.itemSize = ui.Unit.pixel(100);
 listView.itemDoubleClick.addEventListener(handleItemDoubleClick);
 ....................................
 ....................................
 listView.render();
}
```

## See Also

Control Members
Control Class
MindFusion.Common Namespace

# Control.left Property

Gets or sets the X-coordinate of the location of this control.

**Namespace:** MindFusion.Common
**File:** Control.js

## Syntax

**JavaScript**

```
get left() {}
```

### Property Value

Unit. The X-coordinate of the control's location.

## Example

The following code uses the left property to specify the location of a **Window** instance:

**JavaScript**

```
window = new ui.Window();
window.top = ui.Unit.percentage(25);
window.left = ui.Unit.percentage(25);
window.width = ui.Unit.pixel(400);
window.height = ui.Unit.pixel(300);
```

## See Also

Control Members
Control Class
MindFusion.Common Namespace

# Control.licenseLocation Property

Gets or sets the URL of the control's license file.

**Namespace:** MindFusion.Common
**File:** Control.js

## Syntax

**JavaScript**

```
get licenseLocation() {}
```

### Property Value

Type: String
String. A valid URL string that specifies the location of the license key.

## Example

The following code uses the licenseLocation property to specify the path to the license file of a Gauge control:

**JavaScript**

```
var scale = new OvalScale(clock);
scale.setMinValue(0);
scale.setMaxValue(timeLimit);
scale.setStartAngle(-90);
scale.setEndAngle(270);
scale.setFill('Transparent');
scale.setStroke('Transparent');
scale.licenseLocation = "https://mywebsite.biz/license_file.txt";
```

## See Also

Control Members
Control Class
MindFusion.Common Namespace

# Control.loaded Property

Gets a value indicating whether this control is loaded and ready for interaction.

**Namespace:** MindFusion.Common
**File:** Control.js

## Syntax

**JavaScript**

```
get loaded() {}
```

### Property Value

Type: Boolean
Boolean. True if the control is loaded, otherwise false.

## Example

The following code disables drag of items in a **ToolStrip** if the **menu** is visible e.g. loaded:

**JavaScript**

```
// This is the handler function of the toolstrip.itemDragStart event.
// Drag operations will be cancelled if the item to be dragged is the templated item
// or if the menu is currently visible.
function itemDragStart(sender, args)
{
 if (sender.type == ui.ToolStripItemType.Default)
```

```
  args.cancel = true;
 if (menu.loaded)
  args.cancel = true;
}
```

## See Also

Control Members
Control Class
MindFusion.Common Namespace

# Control.rect Property

Gets the bounding rect of this control.

**Namespace:** MindFusion.Common
**File:** Control.js

## Syntax

**JavaScript**

```
get rect() {}
```

### Property Value

Rect. The bounding rect.

## Example

The following code checks if the rect of a Menu instance is wider than 100:

**JavaScript**

```
menu.theme = sender.theme;
menu.left = ui.Unit.pixel(sender.bounds.x + args.item.bounds.x);
menu.top = ui.Unit.pixel(sender.bounds.y + args.item.bounds.height);

if(menu.rect.width > 100)
{
 //do something
}
```

## See Also

Control Members
Control Class
MindFusion.Common Namespace

# Control.theme Property

Gets or sets the current theme of the control.

**Namespace:** MindFusion.Common
**File:** Control.js

## Syntax

**JavaScript**

```
get theme() {}
```

### Property Value

Type: String
String. The current theme.

## Example

The following code assigns the "gray" theme to a **ListView**:

**JavaScript**

```
var ui = MindFusion.UI;

var listView;
var host;
var theme = "gray";

document.addEventListener("DOMContentLoaded", function ()
{
listView = new ui.ListView(document.getElementById("listView"));
listView.width = listView.height = ui.Unit.percentage(100);
listView.theme = theme;
listView.itemSize = ui.Unit.pixel(100);
listView.itemDoubleClick.addEventListener(handleItemDoubleClick);

.....................................
.....................................
}
```

## See Also

Control Members
Control Class
MindFusion.Common Namespace

# Control.top Property

Gets or sets the Y-coordinate of the location of this control.

**Namespace:** MindFusion.Common
**File:** Control.js

## Syntax

**JavaScript**

```
get top() {}
```

### Property Value

Unit. The Y-coordinate of the control's location.

## Example

The following code uses the top property to specify the location of a **Window** instance:

**JavaScript**

```
window = new ui.Window();
window.top = ui.Unit.percentage(25);
window.left = ui.Unit.percentage(25);
window.width = ui.Unit.pixel(400);
window.height = ui.Unit.pixel(300);
```

## See Also

Control Members
Control Class
MindFusion.Common Namespace

# Control.visible Property

Gets or sets the visibility of this control.

**Namespace:** MindFusion.Common
**File:** Control.js

## Syntax

**JavaScript**

```
get visible() {}
```

### Property Value

Boolean. True if the control is visible, otherwise false.

## Example

The following code opens a **Window** instance if it is not already opened e.g. visible.

**JavaScript**

```
var window = host.windows.where(function (window) { return window.data == args.item
}).first();
if (window)
{
 if (!window.visible)
  window.open();
 if (window.windowState != ui.WindowState.Normal)
  window.restore();
 host.bringToFront(window);
}
```

## See Also

Control Members
Control Class
MindFusion.Common Namespace

# Control.width Property

Gets or sets the width of this control.

**Namespace:** MindFusion.Common
**File:** Control.js

## Syntax

```
JavaScript
get width() {}
```

### Property Value

Unit. The width.

## Example

The following code creates a new instance of the **ListView** control and sets its width and height to 100% using the Unit class.

```
JavaScript
var ui = MindFusion.UI;

var listView;
var host;
var theme = "gray";

document.addEventListener("DOMContentLoaded", function ()
{
 listView = new ui.ListView(document.getElementById("listView"));
 listView.width = listView.height = ui.Unit.percentage(100);
 listView.theme = theme;
 listView.itemSize = ui.Unit.pixel(100);
 listView.itemDoubleClick.addEventListener(handleItemDoubleClick);
 ....................................
 ....................................
 listView.render();
}
```

## See Also

Control Members
Control Class
MindFusion.Common Namespace

# Control Events

# Control.controlLoad Event

Raised when the control is loaded.

**Namespace:** MindFusion.Common
**File:** Control.js

## Syntax

```
JavaScript
EventDispatcher controlLoad
```

## Event Data

The event handler method receives the following arguments:
*sender*

A Control instance, which is the source of the event. This object will be passed to the handler function as the first argument.
*args*
An EventArgs instance, which contains event data. This object will be passed to the handler function as the second argument.

## Example

The following code handles the **controlLoad** event for a **Window** instance.

**JavaScript**

```
// Handle the controlLoad event.
window1.controlLoad.addEventListener(handleWindowLoad);

// This is the handler function of the wndow1.controlLoad event.
// The sender argument is a reference to the Window.
function handleWindowLoad(sender) {
 // Hide the window header.
 sender.header.style.display = "none";

 // Let the window calculate its size, depending on the size of the its contents.
 sender.autoSize();

 // Center the window in the bounds of its parent element.
 sender.center();

 // Display the associated data.
 sender.element.querySelector("#data").value = sender.data;
 ....................................
 ....................................
}
```

## See Also

Control Members
Control Class
MindFusion.Common Namespace

# Control.controlUnload Event

Raised when the control is unloaded.

**Namespace:** MindFusion.Common
**File:** Control.js

## Syntax

**JavaScript**

```
EventDispatcher controlUnload
```

## Event Data

The event handler method receives the following arguments:
*sender*
A Control instance, which is the source of the event. This object will be passed to the handler function as the first argument.
*args*
An EventArgs instance, which contains event data. This object will be passed to the handler function as the second argument.

## Example

The following code uses the controlUnloaded event to dispose a **Window** instance.

**JavaScript**

```
// Handle the controlUnload event.
window1.controlUnload.addEventListener(handleWindowUnload);

// This is the handler function of the wndow1.controlUnload event.
// The sender argument is a reference to the Window.
function handleWindowUnload(sender) {
 // Hide the window header.
 sender.dispose();

}
```

## See Also

Control Members
Control Class
MindFusion.Common Namespace

# Control.focus Event

Raised when the control is focused.

**Namespace:** MindFusion.Common
**File:** Control.js

## Syntax

**JavaScript**

```
EventDispatcher focus
```

## Event Data

The event handler method receives the following arguments:
*sender*
A Control instance, which is the source of the event. This object will be passed to the handler function as the first argument.
*args*
An EventArgs instance, which contains event data. This object will be passed to the handler function as the second argument.

## Example

The following code handles the focus event for a **Window** instance.

**JavaScript**

```
// Handle the focus event.
window1.focus.addEventListener(handleFocusEvent);

// This is the handler function of the wndow1 focus event.
// The sender argument is a reference to the Window.
function handleFocusEvent(sender) {

    if(sender.id == "window1")
    {
```

```
    //do something
  }
}
```

## See Also

Control Members
Control Class
MindFusion.Common Namespace

# DateFormats Class

Defines format string for dates and time.

## Syntax

**JavaScript**

```
// class
DateFormats.prototype = {}
```

## Example

The following code sets the **titleFormat** property of a **timetableSettings** in calendar to **dateFormats**.shortDate.

**JavaScript**

```
calendar.timetableSettings.titleFormat = calendar.formatInfo.dateFormats.shortDate;
```

## Inheritance Hierarchy

**MindFusion.Common.DateFormats**

## See Also

DateFormats Members
MindFusion.Common Namespace

# DateFormats Members

Defines format string for dates and time.

The following tables list the members exposed by the DateFormats type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | DateFormats | Initializes a new instance of the **DateFormats** class. |

## Public Properties

| | Name | Description |
|---|---|---|

| | | |
|---|---|---|
| | dayMonth | Gets or sets the day-month format string. |
| | longDate | Gets or sets the long date format string. |
| | longDateTime | Gets or sets the long date-time format string. |
| | longTime | Gets or sets the long time format string. |
| | shortDate | Gets or sets the short date format string. |
| | shortDateTime | Gets or sets the short date-time format string. |
| | shortTime | Gets or sets the short time format string. |
| | yearMonth | Gets or sets the year-month format string. |

## See Also

DateFormats Class
MindFusion.Common Namespace

# DateFormats Constructor

Initializes a new instance of the DateFormats class.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

**JavaScript**

```
function DateFormats ()
```

## Example

The following code creates a new instance of the DateFormats class and assigns a custom shortTime format to it.

**JavaScript**

```
var m = MindFusion.Scheduling;

var dateFormats = new m.DateFormats();
dateFormats.shortTime = "hh:mm";
```

## See Also

DateFormats Class
MindFusion.Common Namespace

# DateFormats Properties

# DateFormats.dayMonth Property

Gets or sets the day-month format string.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

```
JavaScript
get dayMonth() {}
set dayMonth(value) {}
```

### Property Value

Type: String
*String*

The day-month format string.

## Example

The following code creates a new Locale instance that uses the DateFormats property to initialize the desired date and time formats.

```
JavaScript
var locale = new Locale("us-2");

locale.dateSettings.dateFormats.shortDate = "MM/d/yyyy";
locale.dateSettings.dateFormats.shortTime = "h:mm tt";
locale.dateSettings.dateFormats.shortDateTime = "MM/d/yyyy h:mm tt";
locale.dateSettings.dateFormats.longDate = "dddd, MMMM dd, yyyy";
locale.dateSettings.dateFormats.longTime = "h:mm:ss tt";
locale.dateSettings.dateFormats.longDateTime = "dddd, MMMM dd, yyyy h:mm:ss tt";
locale.dateSettings.dateFormats.dayMonth = "d/MM";
locale.dateSettings.dateFormats.yearMonth = "MMMM, yyyy";

locale.dateSettings.generate();

calendar.locale = locale;
```

## See Also

DateFormats Members
DateFormats Class
MindFusion.Common Namespace

# DateFormats.longDate Property

Gets or sets the long date format string.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

```
JavaScript
get longDate() {}
set longDate(value) {}
```

### Property Value

Type: String
*String*

The long date format string.

## Example

The following code creates a new Locale instance that uses the DateFormats property to initialize the desired date and time formats.

**JavaScript**

```
var locale = new Locale("us-2");

locale.dateSettings.dateFormats.shortDate = "MM/d/yyyy";
locale.dateSettings.dateFormats.shortTime = "h:mm tt";
locale.dateSettings.dateFormats.shortDateTime = "MM/d/yyyy h:mm tt";
locale.dateSettings.dateFormats.longDate = "dddd, MMMM dd, yyyy";
locale.dateSettings.dateFormats.longTime = "h:mm:ss tt";
locale.dateSettings.dateFormats.longDateTime = "dddd, MMMM dd, yyyy h:mm:ss tt";
locale.dateSettings.dateFormats.dayMonth = "d/MM";
locale.dateSettings.dateFormats.yearMonth = "MMMM, yyyy";

locale.dateSettings.generate();

calendar.locale = locale;
```

## See Also

DateFormats Members
DateFormats Class
MindFusion.Common Namespace

# DateFormats.longDateTime Property

Gets or sets the long date-time format string.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

**JavaScript**

```
get longDateTime() {}
set longDateTime(value) {}
```

### Property Value

Type: String
*String*

The long date-time format string.

## See Also

DateFormats Members
DateFormats Class
MindFusion.Common Namespace

# DateFormats.longTime Property

Gets or sets the long time format string.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

**JavaScript**

```
get longTime() {}
set longTime(value) {}
```

### Property Value

Type: String
*String*

The long time format string.

## Example

The following code creates a new Locale instance that uses the DateFormats property to initialize the desired date and time formats.

**JavaScript**

```
var locale = new Locale("us-2");

locale.dateSettings.dateFormats.shortDate = "MM/d/yyyy";
locale.dateSettings.dateFormats.shortTime = "h:mm tt";
locale.dateSettings.dateFormats.shortDateTime = "MM/d/yyyy h:mm tt";
locale.dateSettings.dateFormats.longDate = "dddd, MMMM dd, yyyy";
locale.dateSettings.dateFormats.longTime = "h:mm:ss tt";
locale.dateSettings.dateFormats.longDateTime = "dddd, MMMM dd, yyyy h:mm:ss tt";
locale.dateSettings.dateFormats.dayMonth = "d/MM";
locale.dateSettings.dateFormats.yearMonth = "MMMM, yyyy";

locale.dateSettings.generate();

calendar.locale = locale;
```

## See Also

DateFormats Members
DateFormats Class
MindFusion.Common Namespace

# DateFormats.shortDate Property

Gets or sets the short date format string.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

**JavaScript**

```
get shortDate() {}
set shortDate(value) {}
```

### Property Value

Type: String
*String*

The short date format string.

## Example

The following code creates a new Locale instance that uses the DateFormats property to initialize the desired date and time formats.

**JavaScript**

```
var locale = new Locale("us-2");

locale.dateSettings.dateFormats.shortDate = "MM/d/yyyy";
locale.dateSettings.dateFormats.shortTime = "h:mm tt";
locale.dateSettings.dateFormats.shortDateTime = "MM/d/yyyy h:mm tt";
locale.dateSettings.dateFormats.longDate = "dddd, MMMM dd, yyyy";
locale.dateSettings.dateFormats.longTime = "h:mm:ss tt";
locale.dateSettings.dateFormats.longDateTime = "dddd, MMMM dd, yyyy h:mm:ss tt";
locale.dateSettings.dateFormats.dayMonth = "d/MM";
locale.dateSettings.dateFormats.yearMonth = "MMMM, yyyy";

locale.dateSettings.generate();

calendar.locale = locale;
```

## See Also

DateFormats Members
DateFormats Class
MindFusion.Common Namespace

# DateFormats.shortDateTime Property

Gets or sets the short date-time format string.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

**JavaScript**

```
get shortDateTime() {}
set shortDateTime(value) {}
```

### Property Value

Type: String
*String*

The short date-time format string.

## Example

The following code creates a new Locale instance that uses the DateFormats property to initialize the desired date and time formats.

**JavaScript**

```
var locale = new Locale("us-2");

locale.dateSettings.dateFormats.shortDate = "MM/d/yyyy";
locale.dateSettings.dateFormats.shortTime = "h:mm tt";
locale.dateSettings.dateFormats.shortDateTime = "MM/d/yyyy h:mm tt";
locale.dateSettings.dateFormats.longDate = "dddd, MMMM dd, yyyy";
locale.dateSettings.dateFormats.longTime = "h:mm:ss tt";
locale.dateSettings.dateFormats.longDateTime = "dddd, MMMM dd, yyyy h:mm:ss tt";
locale.dateSettings.dateFormats.dayMonth = "d/MM";
locale.dateSettings.dateFormats.yearMonth = "MMMM, yyyy";

locale.dateSettings.generate();

calendar.locale = locale;
```

## See Also

DateFormats Members
DateFormats Class
MindFusion.Common Namespace

# DateFormats.shortTime Property

Gets or sets the short time format string.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

**JavaScript**

```
get shortTime() {}
set shortTime(value) {}
```

### Property Value

Type: String
*String*

The short time format string.

## Example

The following code creates a new Locale instance that uses the DateFormats property to initialize the desired date and time formats.

**JavaScript**

```
var locale = new Locale("us-2");

locale.dateSettings.dateFormats.shortDate = "MM/d/yyyy";
locale.dateSettings.dateFormats.shortTime = "h:mm tt";
locale.dateSettings.dateFormats.shortDateTime = "MM/d/yyyy h:mm tt";
locale.dateSettings.dateFormats.longDate = "dddd, MMMM dd, yyyy";
locale.dateSettings.dateFormats.longTime = "h:mm:ss tt";
locale.dateSettings.dateFormats.longDateTime = "dddd, MMMM dd, yyyy h:mm:ss tt";
locale.dateSettings.dateFormats.dayMonth = "d/MM";
locale.dateSettings.dateFormats.yearMonth = "MMMM, yyyy";

locale.dateSettings.generate();

calendar.locale = locale;
```

## See Also

DateFormats Members
DateFormats Class
MindFusion.Common Namespace

## DateFormats.yearMonth Property

Gets or sets the year-month format string.

**Namespace:** MindFusion.Common
**File:** Locale.js

### Syntax

**JavaScript**

```
get yearMonth() {}
set yearMonth(value) {}
```

#### Property Value

Type: String
*String*

The long year-month format string.

### Example

The following code creates a new Locale instance that uses the DateFormats property to initialize the desired date and time formats.

**JavaScript**

```
var locale = new Locale("us-2");

locale.dateSettings.dateFormats.shortDate = "MM/d/yyyy";
locale.dateSettings.dateFormats.shortTime = "h:mm tt";
locale.dateSettings.dateFormats.shortDateTime = "MM/d/yyyy h:mm tt";
locale.dateSettings.dateFormats.longDate = "dddd, MMMM dd, yyyy";
locale.dateSettings.dateFormats.longTime = "h:mm:ss tt";
locale.dateSettings.dateFormats.longDateTime = "dddd, MMMM dd, yyyy h:mm:ss tt";
locale.dateSettings.dateFormats.dayMonth = "d/MM";
locale.dateSettings.dateFormats.yearMonth = "MMMM, yyyy";

locale.dateSettings.generate();

calendar.locale = locale;
```

### See Also

DateFormats Members
DateFormats Class
MindFusion.Common Namespace

# DateSettings Class

Defines properties that allow customization of date-specific information.

### Syntax

**JavaScript**

```
// class
DateSettings.prototype = {}
```

## Example

The following code creates a new Locale instance that uses the **dateSettings** property to specify the desired date and time formats.

**JavaScript**

```
var locale = new Locale("us-2");

locale.dateSettings.firstDayOfWeek = 0;
locale.dateSettings.dateSeparator = "/";
locale.dateSettings.timeSeparator = ":";
locale.dateSettings.dayPeriodAM = "AM";
locale.dateSettings.dayPeriodPM = "PM";

locale.dateSettings.dateFormats.shortDate = "MM/d/yyyy";
locale.dateSettings.dateFormats.shortTime = "h:mm tt";
locale.dateSettings.dateFormats.shortDateTime = "MM/d/yyyy h:mm tt";
locale.dateSettings.dateFormats.longDate = "dddd, MMMM dd, yyyy";
locale.dateSettings.dateFormats.longTime = "h:mm:ss tt";
locale.dateSettings.dateFormats.longDateTime = "dddd, MMMM dd, yyyy h:mm:ss tt";
locale.dateSettings.dateFormats.dayMonth = "d/MM";
locale.dateSettings.dateFormats.yearMonth = "MMMM, yyyy";

locale.dateSettings.generate();

calendar.locale = locale;
```

## Inheritance Hierarchy

**MindFusion.Common.DateSettings**

## See Also

DateSettings Members
MindFusion.Common Namespace

# DateSettings Members

Defines properties that allow customization of date-specific information.

The following tables list the members exposed by the DateSettings type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | DateSettings | Initializes a new instance of the **DateSettings** class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | dateFormats | Gets the DateFormats object used to hold date and time format strings. |
|  | dateSeparator | Gets or sets the date separator character. |

| | | |
|---|---|---|
| | dayPeriodAM | Gets or sets the before noon time designator. |
| | dayPeriodPM | Gets or sets the after noon time designator. |
| | dayPeriods | Gets an object containing the time designators. |
| | days | Gets an object containing lists of day names. |
| | firstDayOfWeek | Gets or sets a value indicating the first day of the week to use. |
| | longDays | Gets or sets a list of long day names. |
| | longMonths | Gets or sets a list of long month names. |
| | months | Gets an object containing lists of month names. |
| | narrowDays | Gets or sets a list of narrow day names. |
| | narrowMonths | Gets or sets a list of narrow month names. |
| | shortDays | Gets or sets a list of short month names. |
| | shortMonths | Gets or sets a list of short month names. |
| | timeSeparator | Gets or sets the time separator character. |

## Public Methods

| | Name | Description |
|---|---|---|
| | fromJson | Deserializes the settings from a JSON string. |

## See Also

DateSettings Class
MindFusion.Common Namespace

# DateSettings Constructor

Initializes a new instance of the DateSettings class.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

**JavaScript**

```
function DateSettings (localeId)
```

### Parameters

*localeId*
Type: String
String. The string identifier of the locale.

## Example

The following code creates a new Locale instance that uses the dateSettings property to specify the desired date and time formats.

**JavaScript**

```
var locale = new Locale("us-2");

locale.dateSettings.firstDayOfWeek = 0;
locale.dateSettings.dateSeparator = "/";
locale.dateSettings.timeSeparator = ":";
locale.dateSettings.dayPeriodAM = "AM";
locale.dateSettings.dayPeriodPM = "PM";

locale.dateSettings.dateFormats.shortDate = "MM/d/yyyy";
locale.dateSettings.dateFormats.shortTime = "h:mm tt";
locale.dateSettings.dateFormats.shortDateTime = "MM/d/yyyy h:mm tt";
locale.dateSettings.dateFormats.longDate = "dddd, MMMM dd, yyyy";
locale.dateSettings.dateFormats.longTime = "h:mm:ss tt";
locale.dateSettings.dateFormats.longDateTime = "dddd, MMMM dd, yyyy h:mm:ss tt";
locale.dateSettings.dateFormats.dayMonth = "d/MM";
locale.dateSettings.dateFormats.yearMonth = "MMMM, yyyy";

locale.dateSettings.generate();

calendar.locale = locale;
```

## See Also

DateSettings Class
MindFusion.Common Namespace

# DateSettings Methods

## DateSettings.fromJson Method

Deserializes the settings from a JSON string.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

**JavaScript**

```
function fromJson (json)
```

### Parameters

*json*
Type: String
String. A string containing data for the settings.

## Example

The following code creates a new Locale instance and loads the date and time format settings from a json file.

**JavaScript**

```
var locale = new Locale("us-2");
locale.dateSettings.fromJson("locale.json");
locale.dateSettings.generate();
calendar.locale = locale;
```

## See Also

DateSettings Members
DateSettings Class
MindFusion.Common Namespace

# DateSettings Properties

## DateSettings.dateFormats Property

Gets the DateFormats object used to hold date and time format strings.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

**JavaScript**

```
get dateFormats() {}
set dateFormats(value) {}
```

### Property Value

An instance of the DateFormats class.

## Example

The following code creates a new Locale instance that uses the dateSettings property to specify the desired date and time formats.

**JavaScript**

```
var locale = new Locale("us-2");

locale.dateSettings.firstDayOfWeek = 0;
locale.dateSettings.dateSeparator = "/";
locale.dateSettings.timeSeparator = ":";
locale.dateSettings.dayPeriodAM = "AM";
locale.dateSettings.dayPeriodPM = "PM";

locale.dateSettings.dateFormats.shortDate = "MM/d/yyyy";
locale.dateSettings.dateFormats.shortTime = "h:mm tt";
locale.dateSettings.dateFormats.shortDateTime = "MM/d/yyyy h:mm tt";
locale.dateSettings.dateFormats.longDate = "dddd, MMMM dd, yyyy";
locale.dateSettings.dateFormats.longTime = "h:mm:ss tt";
locale.dateSettings.dateFormats.longDateTime = "dddd, MMMM dd, yyyy h:mm:ss tt";
locale.dateSettings.dateFormats.dayMonth = "d/MM";
locale.dateSettings.dateFormats.yearMonth = "MMMM, yyyy";

locale.dateSettings.generate();

calendar.locale = locale;
```

## See Also

DateSettings Members
DateSettings Class
MindFusion.Common Namespace

# DateSettings.dateSeparator Property

Gets or sets the date separator character.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

**JavaScript**

```
get dateSeparator() {}
set dateSeparator(value) {}
```

### Property Value

Type: String
*String*

The date separator.

## Example

The following code creates a new Locale instance that uses the dateSettings property to specify the desired date and time formats.

**JavaScript**

```
var locale = new Locale("us-2");

locale.dateSettings.firstDayOfWeek = 0;
locale.dateSettings.dateSeparator = "/";
locale.dateSettings.timeSeparator = ":";
locale.dateSettings.dayPeriodAM = "AM";
locale.dateSettings.dayPeriodPM = "PM";

locale.dateSettings.dateFormats.shortDate = "MM/d/yyyy";
locale.dateSettings.dateFormats.shortTime = "h:mm tt";
locale.dateSettings.dateFormats.shortDateTime = "MM/d/yyyy h:mm tt";
locale.dateSettings.dateFormats.longDate = "dddd, MMMM dd, yyyy";
locale.dateSettings.dateFormats.longTime = "h:mm:ss tt";
locale.dateSettings.dateFormats.longDateTime = "dddd, MMMM dd, yyyy h:mm:ss tt";
locale.dateSettings.dateFormats.dayMonth = "d/MM";
locale.dateSettings.dateFormats.yearMonth = "MMMM, yyyy";

locale.dateSettings.generate();

calendar.locale = locale;
```

## See Also

DateSettings Members
DateSettings Class
MindFusion.Common Namespace

# DateSettings.dayPeriodAM Property

Gets or sets the before noon time designator.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

JavaScript

```
get dayPeriodAM() {}
set dayPeriodAM(value) {}
```

### Property Value

Type: String
*String*

The AM time designator.

## Example

The following code creates a new Locale instance that uses the dateSettings property to specify the desired date and time formats.

JavaScript

```
var locale = new Locale("us-2");

locale.dateSettings.firstDayOfWeek = 0;
locale.dateSettings.dateSeparator = "/";
locale.dateSettings.timeSeparator = ":";
locale.dateSettings.dayPeriodAM = "AM";
locale.dateSettings.dayPeriodPM = "PM";

locale.dateSettings.dateFormats.shortDate = "MM/d/yyyy";
locale.dateSettings.dateFormats.shortTime = "h:mm tt";
locale.dateSettings.dateFormats.shortDateTime = "MM/d/yyyy h:mm tt";
locale.dateSettings.dateFormats.longDate = "dddd, MMMM dd, yyyy";
locale.dateSettings.dateFormats.longTime = "h:mm:ss tt";
locale.dateSettings.dateFormats.longDateTime = "dddd, MMMM dd, yyyy h:mm:ss tt";
locale.dateSettings.dateFormats.dayMonth = "d/MM";
locale.dateSettings.dateFormats.yearMonth = "MMMM, yyyy";

locale.dateSettings.generate();

calendar.locale = locale;
```

## See Also

DateSettings Members
DateSettings Class
MindFusion.Common Namespace

# DateSettings.dayPeriodPM Property

Gets or sets the after noon time designator.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

JavaScript

```
get dayPeriodPM() {}
set dayPeriodPM(value) {}
```

### Property Value

Type: String
*String*

The PM time designator.

## Example

The following code creates a new Locale instance that uses the dateSettings property to specify the desired date and time formats.

**JavaScript**

```
var locale = new Locale("us-2");

locale.dateSettings.firstDayOfWeek = 0;
locale.dateSettings.dateSeparator = "/";
locale.dateSettings.timeSeparator = ":";
locale.dateSettings.dayPeriodAM = "AM";
locale.dateSettings.dayPeriodPM = "PM";

locale.dateSettings.dateFormats.shortDate = "MM/d/yyyy";
locale.dateSettings.dateFormats.shortTime = "h:mm tt";
locale.dateSettings.dateFormats.shortDateTime = "MM/d/yyyy h:mm tt";
locale.dateSettings.dateFormats.longDate = "dddd, MMMM dd, yyyy";
locale.dateSettings.dateFormats.longTime = "h:mm:ss tt";
locale.dateSettings.dateFormats.longDateTime = "dddd, MMMM dd, yyyy h:mm:ss tt";
locale.dateSettings.dateFormats.dayMonth = "d/MM";
locale.dateSettings.dateFormats.yearMonth = "MMMM, yyyy";

locale.dateSettings.generate();

calendar.locale = locale;
```

## See Also

DateSettings Members
DateSettings Class
MindFusion.Common Namespace

# DateSettings.dayPeriods Property

Gets an object containing the time designators.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

**JavaScript**

```
get dayPeriods() {}
set dayPeriods(value) {}
```

### Property Value

Type: Object
*Object*

An object containing the time designators.

## Example

The following code creates a new Locale instance that uses the dateSettings property to specify the desired date and time formats.

**JavaScript**

```
var locale = new Locale("us-2");

locale.dateSettings.firstDayOfWeek = 0;
locale.dateSettings.dateSeparator = "/";
locale.dateSettings.timeSeparator = ":";
locale.dateSettings.dayPeriodAM = "AM";
locale.dateSettings.dayPeriodPM = "PM";

locale.dateSettings.dateFormats.shortDate = "MM/d/yyyy";
locale.dateSettings.dateFormats.shortTime = "h:mm tt";
locale.dateSettings.dateFormats.shortDateTime = "MM/d/yyyy h:mm tt";
locale.dateSettings.dateFormats.longDate = "dddd, MMMM dd, yyyy";
locale.dateSettings.dateFormats.longTime = "h:mm:ss tt";
locale.dateSettings.dateFormats.longDateTime = "dddd, MMMM dd, yyyy h:mm:ss tt";
locale.dateSettings.dateFormats.dayMonth = "d/MM";
locale.dateSettings.dateFormats.yearMonth = "MMMM, yyyy";

locale.dateSettings.generate();

calendar.locale = locale;
```

## See Also

DateSettings Members
DateSettings Class
MindFusion.Common Namespace

# DateSettings.days Property

Gets an object containing lists of day names.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

**JavaScript**

```
get days() {}
set days(value) {}
```

### Property Value

Type: Object
*Object*

An object containing lists of day names.

## Example

The following code creates a new Locale instance that uses the dateSettings property to specify the desired date and time formats.

**JavaScript**

```
var locale = new Locale("us-2");

locale.dateSettings.firstDayOfWeek = 0;
locale.dateSettings.dateSeparator = "/";
locale.dateSettings.timeSeparator = ":";
locale.dateSettings.dayPeriodAM = "AM";
locale.dateSettings.dayPeriodPM = "PM";

locale.dateSettings.dateFormats.shortDate = "MM/d/yyyy";
locale.dateSettings.dateFormats.shortTime = "h:mm tt";
locale.dateSettings.dateFormats.shortDateTime = "MM/d/yyyy h:mm tt";
```

```
locale.dateSettings.dateFormats.longDate = "dddd, MMMM dd, yyyy";
locale.dateSettings.dateFormats.longTime = "h:mm:ss tt";
locale.dateSettings.dateFormats.longDateTime = "dddd, MMMM dd, yyyy h:mm:ss tt";
locale.dateSettings.dateFormats.dayMonth = "d/MM";
locale.dateSettings.dateFormats.yearMonth = "MMMM, yyyy";

locale.dateSettings.generate();

calendar.locale = locale;
```

## See Also

DateSettings Members
DateSettings Class
MindFusion.Common Namespace

# DateSettings.firstDayOfWeek Property

Gets or sets a value indicating the first day of the week to use.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

**JavaScript**

```
get firstDayOfWeek() {}
set firstDayOfWeek(value) {}
```

### Property Value

Type: Number
*Number*

The index of the first day of the week.

## Remarks

0 is Sunday, 6 is Saturday.

## Example

The following code creates a new Locale instance that uses the dateSettings property to specify the desired date and time formats.

**JavaScript**

```
var locale = new Locale("us-2");

locale.dateSettings.firstDayOfWeek = 0;
locale.dateSettings.dateSeparator = "/";
locale.dateSettings.timeSeparator = ":";
locale.dateSettings.dayPeriodAM = "AM";
locale.dateSettings.dayPeriodPM = "PM";

locale.dateSettings.dateFormats.shortDate = "MM/d/yyyy";
locale.dateSettings.dateFormats.shortTime = "h:mm tt";
locale.dateSettings.dateFormats.shortDateTime = "MM/d/yyyy h:mm tt";
locale.dateSettings.dateFormats.longDate = "dddd, MMMM dd, yyyy";
locale.dateSettings.dateFormats.longTime = "h:mm:ss tt";
locale.dateSettings.dateFormats.longDateTime = "dddd, MMMM dd, yyyy h:mm:ss tt";
locale.dateSettings.dateFormats.dayMonth = "d/MM";
locale.dateSettings.dateFormats.yearMonth = "MMMM, yyyy";
```

```
locale.dateSettings.generate();

calendar.locale = locale;
```

## See Also

DateSettings Members
DateSettings Class
MindFusion.Common Namespace

# DateSettings.longDays Property

Gets or sets a list of long day names.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

**JavaScript**

```
get longDays() {}
set longDays(value) {}
```

### Property Value

Type: Array
*Array*

An array of long day names.

## Example

The following code creates a new Locale instance that uses the dateSettings property to specify the desired date and time formats.

**JavaScript**

```
var locale = new Locale("us-2");

locale.dateSettings.firstDayOfWeek = 0;
locale.dateSettings.dateSeparator = "/";
locale.dateSettings.timeSeparator = ":";
locale.dateSettings.dayPeriodAM = "AM";
locale.dateSettings.dayPeriodPM = "PM";

locale.dateSettings.dateFormats.shortDate = "MM/d/yyyy";
locale.dateSettings.dateFormats.shortTime = "h:mm tt";
locale.dateSettings.dateFormats.shortDateTime = "MM/d/yyyy h:mm tt";
locale.dateSettings.dateFormats.longDate = "dddd, MMMM dd, yyyy";
locale.dateSettings.dateFormats.longTime = "h:mm:ss tt";
locale.dateSettings.dateFormats.longDateTime = "dddd, MMMM dd, yyyy h:mm:ss tt";
locale.dateSettings.dateFormats.longDays = ["Monday", "Tuesday", "Wednesday", "Thursday",
"Friday", "Saturday", "Sunday"];
locale.dateSettings.dateFormats.dayMonth = "d/MM";
locale.dateSettings.dateFormats.yearMonth = "MMMM, yyyy";

locale.dateSettings.generate();

calendar.locale = locale;
```

## See Also

# DateSettings.longMonths Property

Gets or sets a list of long month names.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

**JavaScript**

```
get longMonths() {}
set longMonths(value) {}
```

### Property Value

Type: Array
*Array*

An array of long month names.

## Example

The following code creates a new Locale instance that uses the dateSettings property to specify the desired date and time formats.

**JavaScript**

```
var locale = new Locale("us-2");

locale.dateSettings.firstDayOfWeek = 0;
locale.dateSettings.dateSeparator = "/";
locale.dateSettings.timeSeparator = ":";
locale.dateSettings.dayPeriodAM = "AM";
locale.dateSettings.dayPeriodPM = "PM";

locale.dateSettings.dateFormats.shortDate = "MM/d/yyyy";
locale.dateSettings.dateFormats.shortTime = "h:mm tt";
locale.dateSettings.dateFormats.shortDateTime = "MM/d/yyyy h:mm tt";
locale.dateSettings.dateFormats.longDate = "dddd, MMMM dd, yyyy";
locale.dateSettings.dateFormats.longTime = "h:mm:ss tt";
locale.dateSettings.dateFormats.longDateTime = "dddd, MMMM dd, yyyy h:mm:ss tt";
locale.dateSettings.dateFormats.longDays = ["Monday", "Tuesday", "Wednesday", "Thursday",
"Friday", "Saturday", "Sunday"];
locale.dateSettings.dateFormats.longMonths = ["January", "February", "March", "April", "May",
"June", "July", "August", "September", "October", "November", "December"];
locale.dateSettings.dateFormats.dayMonth = "d/MM";
locale.dateSettings.dateFormats.yearMonth = "MMMM, yyyy";

locale.dateSettings.generate();

calendar.locale = locale;
```

## See Also

# DateSettings.months Property

Gets an object containing lists of month names.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

**JavaScript**

```
get months() {}
set months(value) {}
```

### Property Value

Type: Object
*Object*

An object containing lists of month names.

## Example

The following code creates a new Locale instance that uses the dateSettings property to specify the desired date and time formats.

**JavaScript**

```
var locale = new Locale("us-2");

locale.dateSettings.firstDayOfWeek = 0;
locale.dateSettings.dateSeparator = "/";
locale.dateSettings.timeSeparator = ":";
locale.dateSettings.dayPeriodAM = "AM";
locale.dateSettings.dayPeriodPM = "PM";

locale.dateSettings.dateFormats.shortDate = "MM/d/yyyy";
locale.dateSettings.dateFormats.shortTime = "h:mm tt";
locale.dateSettings.dateFormats.shortDateTime = "MM/d/yyyy h:mm tt";
locale.dateSettings.dateFormats.longDate = "dddd, MMMM dd, yyyy";
locale.dateSettings.dateFormats.longTime = "h:mm:ss tt";
locale.dateSettings.dateFormats.longDateTime = "dddd, MMMM dd, yyyy h:mm:ss tt";
locale.dateSettings.dateFormats.longDays = ["Monday", "Tuesday", "Wednesday", "Thursday",
"Friday", "Saturday", "Sunday"];
locale.dateSettings.dateFormats.months = ["January", "February", "March", "April", "May",
"June", "July", "August", "September", "October", "November", "December"];
locale.dateSettings.dateFormats.dayMonth = "d/MM";
locale.dateSettings.dateFormats.yearMonth = "MMMM, yyyy";

locale.dateSettings.generate();

calendar.locale = locale;
```

## See Also

DateSettings Members
DateSettings Class
MindFusion.Common Namespace

# DateSettings.narrowDays Property

Gets or sets a list of narrow day names.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

**JavaScript**

```
get narrowDays() {}
set narrowDays(value) {}
```

### Property Value

Type: Array
*Array*

An array of narrow day names.

## Example

The following code creates a new Locale instance that uses the dateSettings property to specify the desired date and time formats.

**JavaScript**

```
var locale = new Locale("us-2");

locale.dateSettings.firstDayOfWeek = 0;
locale.dateSettings.dateSeparator = "/";
locale.dateSettings.timeSeparator = ":";
locale.dateSettings.dayPeriodAM = "AM";
locale.dateSettings.dayPeriodPM = "PM";

locale.dateSettings.dateFormats.shortDate = "MM/d/yyyy";
locale.dateSettings.dateFormats.shortTime = "h:mm tt";
locale.dateSettings.dateFormats.shortDateTime = "MM/d/yyyy h:mm tt";
locale.dateSettings.dateFormats.longDate = "dddd, MMMM dd, yyyy";
locale.dateSettings.dateFormats.longTime = "h:mm:ss tt";
locale.dateSettings.dateFormats.longDateTime = "dddd, MMMM dd, yyyy h:mm:ss tt";
locale.dateSettings.dateFormats.narrowDays = ["Mon", "Tue", "Wed", "Thur", "Fr", "Sat",
"Sun"];
locale.dateSettings.dateFormats.months = ["January", "February", "March", "April", "May",
"June", "July", "August", "September", "October", "November", "December"];
locale.dateSettings.dateFormats.dayMonth = "d/MM";
locale.dateSettings.dateFormats.yearMonth = "MMMM, yyyy";

locale.dateSettings.generate();

calendar.locale = locale;
```

## See Also

DateSettings Members
DateSettings Class
MindFusion.Common Namespace

# DateSettings.narrowMonths Property

Gets or sets a list of narrow month names.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

**JavaScript**

```
get narrowMonths() {}
set narrowMonths(value) {}
```

### Property Value

Type: Array
*Array*

An array of narrow month names.

## Example

The following code creates a new Locale instance that uses the dateSettings property to specify the desired date and time formats.

**JavaScript**

```
var locale = new Locale("us-2");

locale.dateSettings.firstDayOfWeek = 0;
locale.dateSettings.dateSeparator = "/";
locale.dateSettings.timeSeparator = ":";
locale.dateSettings.dayPeriodAM = "AM";
locale.dateSettings.dayPeriodPM = "PM";

locale.dateSettings.dateFormats.shortDate = "MM/d/yyyy";
locale.dateSettings.dateFormats.shortTime = "h:mm tt";
locale.dateSettings.dateFormats.shortDateTime = "MM/d/yyyy h:mm tt";
locale.dateSettings.dateFormats.longDate = "dddd, MMMM dd, yyyy";
locale.dateSettings.dateFormats.longTime = "h:mm:ss tt";
locale.dateSettings.dateFormats.longDateTime = "dddd, MMMM dd, yyyy h:mm:ss tt";
locale.dateSettings.dateFormats.narrowDays = ["Mon", "Tue", "Wed", "Thur", "Fr", "Sat",
"Sun"];
locale.dateSettings.dateFormats.narrowMonths = ["Jan", "Feb", "Mar", "Apr", "May", "Jun",
"Jul", "Aug", "Sept", "Oct", "Nov", "Dec"];
locale.dateSettings.dateFormats.dayMonth = "d/MM";
locale.dateSettings.dateFormats.yearMonth = "MMMM, yyyy";

locale.dateSettings.generate();

calendar.locale = locale;
```

## See Also

DateSettings Members
DateSettings Class
MindFusion.Common Namespace

# DateSettings.shortDays Property

Gets or sets a list of short month names.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

**JavaScript**

```
get shortDays() {}
set shortDays(value) {}
```

### Property Value

Type: Array
*Array*

An array of short month names.

## Example

The following code creates a new Locale instance that uses the dateSettings property to specify the desired date and time formats.

**JavaScript**

```
var locale = new Locale("us-2");

locale.dateSettings.firstDayOfWeek = 0;
locale.dateSettings.dateSeparator = "/";
locale.dateSettings.timeSeparator = ":";
locale.dateSettings.dayPeriodAM = "AM";
locale.dateSettings.dayPeriodPM = "PM";

locale.dateSettings.dateFormats.shortDate = "MM/d/yyyy";
locale.dateSettings.dateFormats.shortTime = "h:mm tt";
locale.dateSettings.dateFormats.shortDateTime = "MM/d/yyyy h:mm tt";
locale.dateSettings.dateFormats.longDate = "dddd, MMMM dd, yyyy";
locale.dateSettings.dateFormats.longTime = "h:mm:ss tt";
locale.dateSettings.dateFormats.longDateTime = "dddd, MMMM dd, yyyy h:mm:ss tt";
locale.dateSettings.dateFormats.shortDays = ["M", "Tu", "W", "Th", "F", "St", "Sn"];
locale.dateSettings.dateFormats.narrowMonths = ["J", "F", "Mr", "Ap", "My", "Ju", "Jl", "Ag",
"St", "O", "N", "D"];
locale.dateSettings.dateFormats.dayMonth = "d/MM";
locale.dateSettings.dateFormats.yearMonth = "MMMM, yyyy";

locale.dateSettings.generate();

calendar.locale = locale;
```

## See Also

DateSettings Members
DateSettings Class
MindFusion.Common Namespace

# DateSettings.shortMonths Property

Gets or sets a list of short month names.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

**JavaScript**

```
get shortMonths() {}
set shortMonths(value) {}
```

### Property Value

Type: Array
*Array*

An array of short month names.

## Example

The following code creates a new Locale instance that uses the dateSettings property to specify the desired date and time formats.

**JavaScript**

```
var locale = new Locale("us-2");

locale.dateSettings.firstDayOfWeek = 0;
locale.dateSettings.dateSeparator = "/";
locale.dateSettings.timeSeparator = ":";
locale.dateSettings.dayPeriodAM = "AM";
locale.dateSettings.dayPeriodPM = "PM";

locale.dateSettings.dateFormats.shortDate = "MM/d/yyyy";
locale.dateSettings.dateFormats.shortTime = "h:mm tt";
locale.dateSettings.dateFormats.shortDateTime = "MM/d/yyyy h:mm tt";
locale.dateSettings.dateFormats.longDate = "dddd, MMMM dd, yyyy";
locale.dateSettings.dateFormats.longTime = "h:mm:ss tt";
locale.dateSettings.dateFormats.longDateTime = "dddd, MMMM dd, yyyy h:mm:ss tt";
locale.dateSettings.dateFormats.shortDays = ["M", "Tu", "W", "Th", "F", "St", "Sn"];
locale.dateSettings.dateFormats.narrowMonths = ["J", "F", "Mr", "Ap", "My", "Ju", "Jl", "Ag",
"St", "O", "N", "D"];
locale.dateSettings.dateFormats.dayMonth = "d/MM";
locale.dateSettings.dateFormats.yearMonth = "MMMM, yyyy";

locale.dateSettings.generate();

calendar.locale = locale;
```

## See Also

DateSettings Members
DateSettings Class
MindFusion.Common Namespace

# DateSettings.timeSeparator Property

Gets or sets the time separator character.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

**JavaScript**

```
get timeSeparator() {}
set timeSeparator(value) {}
```

### Property Value

Type: String
*String*

The time separator.

## Example

The following code creates a new Locale instance that uses the dateSettings property to specify the desired date and time formats.

**JavaScript**

```
var locale = new Locale("us-2");
```

```
locale.dateSettings.firstDayOfWeek = 0;
locale.dateSettings.dateSeparator = "/";
locale.dateSettings.timeSeparator = ":";
locale.dateSettings.dayPeriodAM = "AM";
locale.dateSettings.dayPeriodPM = "PM";

locale.dateSettings.dateFormats.shortDate = "MM/d/yyyy";
locale.dateSettings.dateFormats.shortTime = "h:mm tt";
locale.dateSettings.dateFormats.shortDateTime = "MM/d/yyyy h:mm tt";
locale.dateSettings.dateFormats.longDate = "dddd, MMMM dd, yyyy";
locale.dateSettings.dateFormats.longTime = "h:mm:ss tt";
locale.dateSettings.dateFormats.longDateTime = "dddd, MMMM dd, yyyy h:mm:ss tt";
locale.dateSettings.dateFormats.dayMonth = "d/MM";
locale.dateSettings.dateFormats.yearMonth = "MMMM, yyyy";

locale.dateSettings.generate();

calendar.locale = locale;
```

## See Also

DateSettings Members
DateSettings Class
MindFusion.Common Namespace

# EventDispatcher Class

Represents a dispatcher for an event.

## Syntax

**JavaScript**

```
// class
EventDispatcher.prototype = {}
```

## Remarks

This is the base class for all events raised by UI controls.

## Example

The following code handles the controlUnload method of the **Window** control.

**JavaScript**

```
// Handle the controlUnload event.
window1.controlUnload.addEventListener(handleWindowUnload);

// This is the handler function of the wndow1.controlUnload event.
// The sender argument is a reference to the Window.
function handleWindowUnload(sender) {
 // Hide the window header.
 sender.dispose();

}
```

## Inheritance Hierarchy

**MindFusion.Common.EventDispatcher**

## See Also

# EventDispatcher Members

Represents a dispatcher for an event.

The following tables list the members exposed by the EventDispatcher type.

## Public Methods

| | Name | Description |
|---|---|---|
|  | addEventListener | Subcribes an event listener to this event. |
|  | raiseEvent | Raises this event. |
|  | removeEventListener | Removes an event listener from this event. |

## See Also

EventDispatcher Class
MindFusion.Common Namespace

# EventDispatcher Methods

# EventDispatcher.addEventListener Method

Subcribes an event listener to this event.

**Namespace:** MindFusion.Common
**File:** EventDispatcher.js

## Syntax

**JavaScript**

```
function addEventListener (handler)
```

### Parameters

*handler*
Type: function
function. The handler function.

## Example

The following code handles the controlUnload event of the **Window** class.

**JavaScript**

```
// Handle the controlUnload event.
window1.controlUnload.addEventListener(handleWindowUnload);

// This is the handler function of the wndow1.controlUnload event.
// The sender argument is a reference to the Window.
function handleWindowUnload(sender) {
```

```
    // Hide the window header.
    sender.dispose();
}
```

## See Also

EventDispatcher Members
EventDispatcher Class
MindFusion.Common Namespace

# EventDispatcher.raiseEvent Method

Raises this event.

**Namespace:** MindFusion.Common
**File:** EventDispatcher.js

## Syntax

**JavaScript**

```
function raiseEvent (sender, args)
```

### Parameters

*sender*
Type: Object
Object. The source of the event.

*args*
Type: Object
Object. An object, containing event data.

## See Also

EventDispatcher Members
EventDispatcher Class
MindFusion.Common Namespace

# EventDispatcher.removeEventListener Method

Removes an event listener from this event.

**Namespace:** MindFusion.Common
**File:** EventDispatcher.js

## Syntax

**JavaScript**

```
function removeEventListener (handler)
```

### Parameters

*handler*
Type: function
function. The handler function.

## Example

This is how you remove the nodeClicked event of a **diagram** instance.

**JavaScript**

```
diagram.removeEventListener(Events.nodeClicked, onNodeClicked);
```

## See Also

EventDispatcher Members
EventDispatcher Class
MindFusion.Common Namespace

# IdGenerator Class

Generates unique ids.

## Syntax

**JavaScript**

```
// class
IdGenerator.prototype = {}
```

## Example

This code generates a random id starting with the prefix "APP".

**JavaScript**

```
var id = IdGenerator.generate('APP');
```

## Inheritance Hierarchy

**MindFusion.Common.IdGenerator**

## See Also

IdGenerator Members
MindFusion.Common Namespace

# IdGenerator Members

Generates unique ids.

The following tables list the members exposed by the IdGenerator type.

## Public Methods

| | Name | Description |
|---|---|---|
|  | generate | Generates a new id that starts with the specified prefix. |

## See Also

IdGenerator Class
MindFusion.Common Namespace

## IdGenerator Methods

## IdGenerator.generate Method

Generates a new id that starts with the specified prefix.

**Namespace:** MindFusion.Common
**File:** IdGenerator.js

### Syntax

```
JavaScript
function generate (prefix)
```

#### Parameters

*prefix*
Type: String
String. A prefix to insert at the beginning of the identifier.

#### Return Value

Type: String
String. A string containing the generated identifier.

### Example

This code generates a random id starting with the prefix "APP".

```
JavaScript
var id = IdGenerator.generate('APP');
```

### See Also

IdGenerator Members
IdGenerator Class
MindFusion.Common Namespace

# Locale Class

Provides culture-specific information.

### Syntax

```
JavaScript
// class
Locale.prototype = {}
```

### Example

The following code creates a new **Locale** instance that is assigned to the **locale** property of a **Calendar** to specify the desired date and time formats.

**JavaScript**

```
var locale = new Locale("us-2");

locale.dateSettings.dateFormats.shortDate = "MM/d/yyyy";
locale.dateSettings.dateFormats.shortTime = "h:mm tt";
locale.dateSettings.dateFormats.shortDateTime = "MM/d/yyyy h:mm tt";
locale.dateSettings.dateFormats.longDate = "dddd, MMMM dd, yyyy";
locale.dateSettings.dateFormats.longTime = "h:mm:ss tt";
locale.dateSettings.dateFormats.longDateTime = "dddd, MMMM dd, yyyy h:mm:ss tt";
locale.dateSettings.dateFormats.dayMonth = "d/MM";
locale.dateSettings.dateFormats.yearMonth = "MMMM, yyyy";

locale.dateSettings.generate();

calendar.locale = locale;
```

## Inheritance Hierarchy

**MindFusion.Common.Locale**

## See Also

Locale Members
MindFusion.Common Namespace

# Locale Members

Provides culture-specific information.

The following tables list the members exposed by the Locale type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | Locale | Initializes a new instance of the Locale class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | dateFormats | Gets the DateFormats object used to hold date and time format strings. |
| | dateSettings | Gets the DateSettings object used to hold date-specific information. |
| | id | Gets the string identifier of the locale. |
| | strings | Gets or sets the dictionary of custom strings. |

## Public Methods

| | Name | Description |
|---|---|---|
| S | default | Returns a default Locale object. |

| | | |
|---|---|---|
| | fromJson | Deserializes the locale from a JSON string. |

## See Also

Locale Class
MindFusion.Common Namespace

## Locale Constructor

Initializes a new instance of the Locale class.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

**JavaScript**

```
function Locale ([id])
```

### Parameters

*id*
Optional.
Type: String
String. The string identifier of the locale.

## Example

The following code creates a new Locale instance that is assigned to the **locale** property of a **Calendar** to specify the desired date and time formats.

**JavaScript**

```
var locale = new Locale("us-2");

locale.dateSettings.dateFormats.shortDate = "MM/d/yyyy";
locale.dateSettings.dateFormats.shortTime = "h:mm tt";
locale.dateSettings.dateFormats.shortDateTime = "MM/d/yyyy h:mm tt";
locale.dateSettings.dateFormats.longDate = "dddd, MMMM dd, yyyy";
locale.dateSettings.dateFormats.longTime = "h:mm:ss tt";
locale.dateSettings.dateFormats.longDateTime = "dddd, MMMM dd, yyyy h:mm:ss tt";
locale.dateSettings.dateFormats.dayMonth = "d/MM";
locale.dateSettings.dateFormats.yearMonth = "MMMM, yyyy";

locale.dateSettings.generate();

calendar.locale = locale;
```

## See Also

Locale Class
MindFusion.Common Namespace

## Locale Methods

# Locale.default Method

Returns a default Locale object.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

```
JavaScript
function default ()
```

### Return Value

Type: Locale
Locale. A locale object with default settings.

## Example

The following code assigns the default Locale to the **locale** property of a **Calendar** instance.

```
JavaScript
calendar.locale = Locale.default();
```

## See Also

Locale Members
Locale Class
MindFusion.Common Namespace

# Locale.fromJson Method

Deserializes the locale from a JSON string.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

```
JavaScript
function fromJson (dateSettingsJson)
```

### Parameters

*dateSettingsJson*
Type: String
String. A string containing data for the date settings.

## Example

The following code creates a new Locale instance and loads its settings from a json file.

```
JavaScript
calendar.locale = Locale.fromJson("locale.json");
```

## See Also

Locale Members
Locale Class
MindFusion.Common Namespace

# Locale Properties

## Locale.dateFormats Property

Gets the DateFormats object used to hold date and time format strings.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

**JavaScript**

```
get dateFormats() {}
set dateFormats(value) {}
```

### Property Value

DateFormats. The **DateFormats** object.

## Example

The following code creates a new Locale instance that uses the dateSettings property to specify the desired date and time formats.

**JavaScript**

```
var locale = new Locale("us-2");

locale.dateSettings.firstDayOfWeek = 0;
locale.dateSettings.dateSeparator = "/";
locale.dateSettings.timeSeparator = ":";
locale.dateSettings.dayPeriodAM = "AM";
locale.dateSettings.dayPeriodPM = "PM";

locale.dateSettings.dateFormats.shortDate = "MM/d/yyyy";
locale.dateSettings.dateFormats.shortTime = "h:mm tt";
locale.dateSettings.dateFormats.shortDateTime = "MM/d/yyyy h:mm tt";
locale.dateSettings.dateFormats.longDate = "dddd, MMMM dd, yyyy";
locale.dateSettings.dateFormats.longTime = "h:mm:ss tt";
locale.dateSettings.dateFormats.longDateTime = "dddd, MMMM dd, yyyy h:mm:ss tt";
locale.dateSettings.dateFormats.dayMonth = "d/MM";
locale.dateSettings.dateFormats.yearMonth = "MMMM, yyyy";

locale.dateSettings.generate();

calendar.locale = locale;
```

## See Also

Locale Members
Locale Class
MindFusion.Common Namespace

# Locale.dateSettings Property

Gets the DateSettings object used to hold date-specific information.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

**JavaScript**

```
get dateSettings() {}
set dateSettings(value) {}
```

### Property Value

An instance of the DateSettings class.

## Example

The following code creates a new Locale instance that uses the **dateSettings** property to specify the desired date and time formats.

**JavaScript**

```
var locale = new Locale("us-2");

locale.dateSettings.firstDayOfWeek = 0;
locale.dateSettings.dateSeparator = "/";
locale.dateSettings.timeSeparator = ":";
locale.dateSettings.dayPeriodAM = "AM";
locale.dateSettings.dayPeriodPM = "PM";

locale.dateSettings.dateFormats.shortDate = "MM/d/yyyy";
locale.dateSettings.dateFormats.shortTime = "h:mm tt";
locale.dateSettings.dateFormats.shortDateTime = "MM/d/yyyy h:mm tt";
locale.dateSettings.dateFormats.longDate = "dddd, MMMM dd, yyyy";
locale.dateSettings.dateFormats.longTime = "h:mm:ss tt";
locale.dateSettings.dateFormats.longDateTime = "dddd, MMMM dd, yyyy h:mm:ss tt";
locale.dateSettings.dateFormats.dayMonth = "d/MM";
locale.dateSettings.dateFormats.yearMonth = "MMMM, yyyy";

locale.dateSettings.generate();

calendar.locale = locale;
```

## See Also

Locale Members
Locale Class
MindFusion.Common Namespace

# Locale.id Property

Gets the string identifier of the locale.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

**JavaScript**

```
get id() {}
set id(value) {}
```

### Property Value

Type: String
*String*

The id.

## Example

The following code checks the id of the **locale** property of a **calendar**:

**JavaScript**

```
if(calendar.locale.id == "us-2")
{
     //do something
}
```

## See Also

Locale Members
Locale Class
MindFusion.Common Namespace

# Locale.strings Property

Gets or sets the dictionary of custom strings.

**Namespace:** MindFusion.Common
**File:** Locale.js

## Syntax

**JavaScript**

```
get strings() {}
set strings(value) {}
```

### Property Value

Type: Object
*Object*

The dictionary of custom strings.

## See Also

Locale Members
Locale Class
MindFusion.Common Namespace

# NotifyCollectionChangedAction Class

Specifies the type of the collection changed action.

## Syntax

JavaScript

```
// class
NotifyCollectionChangedAction.prototype = {}
```

## Example

The following code handles the collectionChanging event of the **decorations** property of a **DecorationLayer** object in a **MapView** instance.

JavaScript

```
var m = MindFusion.Mapping;

markers.decorations.collectionChanging.addEventListener(handleCollectionChanging);


function (handleCollectionChanging) (sender, args )
{
     if(args.action == m.NotifyCollectionChangedAction.Add)
     {
          args.cancel = true;
     }
}
```

## Inheritance Hierarchy

**MindFusion.Common.NotifyCollectionChangedAction**

## See Also

NotifyCollectionChangedAction Members
MindFusion.Common Namespace

# NotifyCollectionChangedAction Members

Specifies the type of the collection changed action.

The following tables list the members exposed by the NotifyCollectionChangedAction type.

## See Also

NotifyCollectionChangedAction Class
MindFusion.Common Namespace

# NotifyCollectionChangedEventArgs Class

Provides data for CollectionChanged events.

## Syntax

JavaScript

```
// class
NotifyCollectionChangedEventArgs.prototype = {}
```

## Example

The following code handles the collectionChanged event of the **decorations** property of a **DecorationLayer** object in a **Map**.

**JavaScript**

```
var m = MindFusion.Mapping;

var markers = new m.DecorationLayer("Images");

.................
//add some Marker-s to the decorations property
.....................

markers.decorations.collectionChanged.addEventListener(handleCollectionChanged);


function (handleCollectionChanged) (sender, args )
{
     if(args.newItems.contains(mark))
     {
          //do something
     }
}
```

## Inheritance Hierarchy

MindFusion.Common.EventArgs
  **MindFusion.Common.NotifyCollectionChangedEventArgs**

## See Also

NotifyCollectionChangedEventArgs Members
MindFusion.Common Namespace

# NotifyCollectionChangedEventArgs Members

Provides data for CollectionChanged events.

The following tables list the members exposed by the NotifyCollectionChangedEventArgs type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | NotifyCollectionChangedEventArgs | Initializes a new instance of the **NotifyCollectionChangedEventArgs** class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | action | Gets the action that caused the event. |
|  | index | Gets the index where the change occurred. |
|  | newItems | Gets the list of new items involved in the change. |
|  | oldItems | Gets the list of the items affected by a Remove action. |

## See Also

NotifyCollectionChangedEventArgs Class
MindFusion.Common Namespace

# NotifyCollectionChangedEventArgs Constructor

Initializes a new instance of the NotifyCollectionChangedEventArgs class.

**Namespace:** MindFusion.Common
**File:** Events.js

## Syntax

**JavaScript**

```
function NotifyCollectionChangedEventArgs (action, [changes, [index]])
```

### Parameters

*action*
NotifyCollectionChangedAction. The action that caused the event.

*changes*
Optional.
Type: Array
Array. The items affected by the change.

*index*
Optional.
Type: Number
Number. The index where the change occurred.

## Example

The following code handles the collectionChanged event of the **decorations** property of a **DecorationLayer** object in a **Map**.

**JavaScript**

```
var m = MindFusion.Mapping;

var markers = new m.DecorationLayer("Images");

...................
//add some Marker-s to the decorations property
.....................

markers.decorations.collectionChanged.addEventListener(handleCollectionChanged);


function (handleCollectionChanged) (sender, args )
{
     if(args.newItems.contains(mark))
     {
          //do something
     }
}
```

## See Also

# NotifyCollectionChangedEventArgs Properties

## NotifyCollectionChangedEventArgs.action Property

Gets the action that caused the event.

**Namespace:** MindFusion.Common
**File:** Events.js

### Syntax

```
JavaScript
get action() {}
```

#### Property Value

One of the NotifyCollectionChangedAction enumeration values. The action that caused the event.

### Example

The following code handles the collectionChanged event of the **decorations** property of a **DecorationLayer** object in a **map**.

```
JavaScript
var m = MindFusion.Mapping;

var markers = new m.DecorationLayer("Images");

...................
//add some Marker-s to the decorations property
.....................
markers.decorations.collectionChanged.addEventListener(handleCollectionChanged);


function (handleCollectionChanged) (sender, args )
{
     if(args.action === NotifyCollectionChangedAction.Add)
     {
          //do something
     }
}
```

### See Also

NotifyCollectionChangedEventArgs Members
NotifyCollectionChangedEventArgs Class
MindFusion.Common Namespace

## NotifyCollectionChangedEventArgs.index Property

Gets the index where the change occurred.

**Namespace:** MindFusion.Common
**File:** Events.js

## Syntax

JavaScript

```
get index() {}
```

### Property Value

Type: Number
Number. The index.

## Example

The following code handles the collectionChanged event of the **decorations** property of a **DecorationLayer** object in a **map**.

JavaScript

```
var m = MindFusion.Mapping;

var markers = new m.DecorationLayer("Images");

...................
//add some Marker-s to the decorations property
.....................

markers.decorations.collectionChanged.addEventListener(handleCollectionChanged);


function (handleCollectionChanged) (sender, args )
{    //the change has occurred on the first element
     if(args.index === 0)
     {
          //do something
     }
}
```

## See Also

NotifyCollectionChangedEventArgs Members
NotifyCollectionChangedEventArgs Class
MindFusion.Common Namespace

# NotifyCollectionChangedEventArgs.newItems Property

Gets the list of new items involved in the change.

**Namespace:** MindFusion.Common
**File:** Events.js

## Syntax

JavaScript

```
get newItems() {}
```

### Property Value

Type: Array
Array. The list of involved items.

## Example

The following code handles the collectionChanged event of the **decorations** property of a **DecorationLayer** object.

**JavaScript**

```
var m = MindFusion.Mapping;

var markers = new m.DecorationLayer("Images");

...................
//add some Marker-s to the decorations property
.....................

markers.decorations.collectionChanged.addEventListener(handleCollectionChanged);


function (handleCollectionChanged) (sender, args )
{
     if(args.newItems.contains(mark))
     {
          //do something
     }
}
```

## See Also

NotifyCollectionChangedEventArgs Members
NotifyCollectionChangedEventArgs Class
MindFusion.Common Namespace

# NotifyCollectionChangedEventArgs.oldItems Property

Gets the list of the items affected by a Remove action.

**Namespace:** MindFusion.Common
**File:** Events.js

## Syntax

**JavaScript**

```
get oldItems() {}
```

### Property Value

Type: Array
Array. The list of affected items.

## Example

The following code handles the collectionChanged event of the **decorations** property of a **DecorationLayer** object.

**JavaScript**

```
var m = MindFusion.Mapping;

var markers = new m.DecorationLayer("Images");

...................
//add some Marker-s to the decorations property
.....................

markers.decorations.collectionChanged.addEventListener(handleCollectionChanged);


function (handleCollectionChanged) (sender, args )
```

```
{
     if(args.oldItems.contains(mark))
     {
          //do something
     }
}
```

## See Also

NotifyCollectionChangedEventArgs Members
NotifyCollectionChangedEventArgs Class
MindFusion.Common Namespace

# NotifyCollectionChangingEventArgs Class

Provides data for **CollectionChanging** events.

## Syntax

**JavaScript**

```
// class
NotifyCollectionChangingEventArgs.prototype = {}
```

## Example

The following code handles the collectionChanging event of the **decorations** property of a **DecorationLayer** object in a **map**.

**JavaScript**

```
var m = MindFusion.Mapping;

var markers = new m.DecorationLayer("Images");

...................
//add some Marker-s to the decorations property
.....................

markers.decorations.collectionChanging.addEventListener(handleCollectionChanging);


function (handleCollectionChanging) (sender, args )
{
     if(args.action == m.NotifyCollectionChangedAction.Add)
     {
          args.cancel = true;
     }
}
```

## Inheritance Hierarchy

MindFusion.Common.EventArgs
  MindFusion.Common.CancelEventArgs
    **MindFusion.Common.NotifyCollectionChangingEventArgs**

## See Also

NotifyCollectionChangingEventArgs Members
MindFusion.Common Namespace

# NotifyCollectionChangingEventArgs Members

Provides data for **CollectionChanging** events.

The following tables list the members exposed by the NotifyCollectionChangingEventArgs type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | NotifyCollectionChangingEventArgs | Initializes a new instance of the **NotifyCollectionChangingEventArgs** class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | action | Gets the action that caused the event. |
|  | newItems | Gets the list of new items involved in the change. |
|  | oldItems | Gets the list of the items affected by a Remove action. |

## See Also

NotifyCollectionChangingEventArgs Class
MindFusion.Common Namespace

# NotifyCollectionChangingEventArgs Constructor

Initializes a new instance of the **NotifyCollectionChangingEventArgs** class.

**Namespace:** MindFusion.Common
**File:** Events.js

## Syntax

**JavaScript**

```
function NotifyCollectionChangingEventArgs (action, [changes, [index]])
```

### Parameters

*action*
NotifyCollectionChangedAction. The action that caused the event.

*changes*
Optional.
Type: Array
Array. The items affected by the change.

*index*
Optional.
Type: Number
Number. The index where the change occurred.

## Example

The following code handles the collectionChanging event of the **decorations** property of a **DecorationLayer** object in a **map**.

**JavaScript**

```
var m = MindFusion.Mapping;

var markers = new m.DecorationLayer("Images");

...................
//add some Marker-s to the decorations property
.....................

markers.decorations.collectionChanging.addEventListener(handleCollectionChanging);


function (handleCollectionChanging) (sender, args )
{
     if(args.action == m.NotifyCollectionChangedAction.Add)
     {
          args.cancel = true;
     }
}
```

## See Also

NotifyCollectionChangingEventArgs Class
MindFusion.Common Namespace

# NotifyCollectionChangingEventArgs Properties

# NotifyCollectionChangingEventArgs.action Property

Gets the action that caused the event.

**Namespace:** MindFusion.Common
**File:** Events.js

## Syntax

**JavaScript**

```
get action() {}
```

### Property Value

NotifyCollectionChangedAction. The action that caused the event.

## Example

The following code handles the collectionChanging event of the **decorations** property of a **DecorationLayer** object in a **map**.

**JavaScript**

```
var m = MindFusion.Mapping;

var markers = new m.DecorationLayer("Images");

...................
//add some Marker-s to the decorations property
.....................

markers.decorations.collectionChanging.addEventListener(handleCollectionChanging);
```

```
function (handleCollectionChanging) (sender, args )
{
     if(args.action == m.NotifyCollectionChangedAction.Add)
     {
          args.cancel = true;
     }
}
```

## See Also

NotifyCollectionChangingEventArgs Members
NotifyCollectionChangingEventArgs Class
MindFusion.Common Namespace

# NotifyCollectionChangingEventArgs.newItems Property

Gets the list of new items involved in the change.

**Namespace:** MindFusion.Common
**File:** Events.js

## Syntax

**JavaScript**

```
get newItems() {}
```

### Property Value

Type: Array
Array. The list of involved items.

## Example

The following code handles the collectionChanging event of the **decorations** property of a **DecorationLayer** object in a map.

**JavaScript**

```
var m = MindFusion.Mapping;

var markers = new m.DecorationLayer("Images");

.................
//add some Marker-s to the decorations property
.....................
markers.decorations.collectionChanging.addEventListener(handleCollectionChanging);


function (handleCollectionChanging) (sender, args )
{
      if(args.newItems.contains(marker1))
      {
          //do something
      }
}
```

## See Also

NotifyCollectionChangingEventArgs Members
NotifyCollectionChangingEventArgs Class

# NotifyCollectionChangingEventArgs.oldItems Property

Gets the list of the items affected by a Remove action.

**Namespace:** MindFusion.Common
**File:** Events.js

## Syntax

**JavaScript**

```
get oldItems() {}
```

### Property Value

Type: Array
Array. The list of affected items.

## Example

The following code handles the collectionChanged event of the **decorations** property of a **DecorationLayer** object in a **map**.

**JavaScript**

```
var m = MindFusion.Mapping;

var markers = new m.DecorationLayer("Images");

.................
//add some Marker-s to the decorations property
.....................

markers.decorations.collectionChanged.addEventListener(handleCollectionChanged);


function (handleCollectionChanged) (sender, args )
{
      if(args.oldItems.contains(marker1))
      {
          //do something
      }
}
```

## See Also

NotifyCollectionChangingEventArgs Members
NotifyCollectionChangingEventArgs Class
MindFusion.Common Namespace

# PropertyEventArgs Class

Provides data for PropertyValueChanged events.

## Syntax

**JavaScript**

```
// class
PropertyEventArgs.prototype = {}
```

## Example

The following code handles the propertyValueChanged event for a **Calendar** instance.

**JavaScript**

```
calendar.propertyValueChanged.addEventListener(handlePropertyChanged);

function handlePropertyChanged (sender, args )
{
     if(args.propertyName === "theme")
     {
          if(args.newValue == "gray")
          {
             //do something
   }
     }
}
```

## Inheritance Hierarchy

MindFusion.Common.EventArgs
  **MindFusion.Common.PropertyEventArgs**

## See Also

PropertyEventArgs Members
MindFusion.Common Namespace

# PropertyEventArgs Members

Provides data for PropertyValueChanged events.

The following tables list the members exposed by the PropertyEventArgs type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | PropertyEventArgs | Initlializes a new instance of the PropertyEventArgs class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | newValue | Gets the current value of the property. |
|  | oldValue | Gets the value of the property before the change. |
|  | propertyName | Gets the name of the property that changed. |

## See Also

PropertyEventArgs Class
MindFusion.Common Namespace

# PropertyEventArgs Constructor

Initlializes a new instance of the **PropertyEventArgs** class.

**Namespace:** MindFusion.Common
**File:** Events.js

## Syntax

**JavaScript**

```
function PropertyEventArgs (propertyName, oldValue, newValue)
```

### Parameters

*propertyName*
Type: String
String. The name of the property that changed.

*oldValue*
Type: Object
Object. The old value of the property.

*newValue*
Type: Object
Object. The new value of the property.

## Example

The following code handles the propertyValueChanged event for a **Calendar** instance.

**JavaScript**

```
calendar.propertyValueChanged.addEventListener(handlePropertyChanged);

function handlePropertyChanged (sender, args )
{
     if(args.propertyName === "theme")
     {
          if(args.newValue == "gray")
          {
             //do something
   }
     }
}
```

## See Also

PropertyEventArgs Class
MindFusion.Common Namespace

# PropertyEventArgs Properties

# PropertyEventArgs.newValue Property

Gets the current value of the property.

**Namespace:** MindFusion.Common
**File:** Events.js

## Syntax

JavaScript

```
get newValue() {}
```

### Property Value

Type: Object
Object. The new property value.

## Example

The following code handles the propertyValueChanged event for a **Calendar** instance.

JavaScript

```
calendar.propertyValueChanged.addEventListener(handlePropertyChanged);

function handlePropertyChanged (sender, args )
{
     if(args.propertyName === "theme")
     {
          if(args.newValue == "gray")
          {
             //do something
   }
     }
}
```

## See Also

PropertyEventArgs Members
PropertyEventArgs Class
MindFusion.Common Namespace

# PropertyEventArgs.oldValue Property

Gets the value of the property before the change.

**Namespace:** MindFusion.Common
**File:** Events.js

## Syntax

JavaScript

```
get oldValue() {}
```

### Property Value

Type: Object
Object. The old property value.

## Example

The following code handles the propertyValueChanged event for a **Calendar** instance.

JavaScript

```
calendar.propertyValueChanged.addEventListener(handlePropertyChanged);

function handlePropertyChanged (sender, args )
{
```

```
     if(args.propertyName === "theme")
     {
          if(args.oldValue == "gray")
          {
             //do something
   }
     }
}
```

## See Also

PropertyEventArgs Members
PropertyEventArgs Class
MindFusion.Common Namespace

# PropertyEventArgs.propertyName Property

Gets the name of the property that changed.

**Namespace:** MindFusion.Common
**File:** Events.js

## Syntax

**JavaScript**

```
get propertyName() {}
```

### Property Value

Type: String
String. The name of the property.

## Example

The following code handles the propertyValueChanged event for a **Calendar** instance, which uses data of type PropertyEventArgs.

**JavaScript**

```
calendar.propertyValueChanged.addEventListener(handlePropertyChanged);

function handlePropertyChanged (sender, args )
{
     if(args.propertyName === "theme")
     {
          if(args.oldValue == "gray")
          {
             //do something
   }
     }
}
```

## See Also

PropertyEventArgs Members
PropertyEventArgs Class
MindFusion.Common Namespace

# UIControl Class

Initializes a new instance of the Control class.

## Syntax

**JavaScript**

```
// class
UIControl.prototype = {}
```

## Example

The following code creates a **ToolTip** instance, which is a class that derives from **UIControl**. The tooltip is created using the DOM element of a diagram canvas.

**JavaScript**

```
var tooltip = new ui.Tooltip(diagram._element);
tooltip.trigger = ui.TooltipTrigger.None;
tooltip.follow = false;
tooltip.theme = "peach";
```

## Inheritance Hierarchy

**MindFusion.Common.UIControl**

## See Also

UIControl Members
MindFusion.Common Namespace

# UIControl Members

Initializes a new instance of the Control class.

The following tables list the members exposed by the UIControl type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | UIControl | Initializes a new instance of the **UIControl** class. |

## Public Methods

| | Name | Description |
|---|---|---|
|  | draw | Draws the **UIControl**. |

## See Also

UIControl Class
MindFusion.Common Namespace

# UIControl Constructor

Initializes a new instance of the UIControl class.

**Namespace:** MindFusion.Common
**File:** UIControl.js

## Syntax

**JavaScript**

```
function UIControl ([element])
```

### Parameters

*element*
Optional.
Type: HTMLElement
HTMLElement. The control's associated Dom element.

## Example

The following code creates a **ToolTip** instance, which is a class that derives from UIControl. The tooltip is created using the DOM element of a **diagram** canvas.

**JavaScript**

```
var tooltip = new ui.Tooltip(diagram._element);
tooltip.trigger = ui.TooltipTrigger.None;
tooltip.follow = false;
tooltip.theme = "peach";
```

## See Also

UIControl Class
MindFusion.Common Namespace

# UIControl Methods

# UIControl.draw Method

Draws the UIControl.

**Namespace:** MindFusion.Common
**File:** UIControl.js

## Syntax

**JavaScript**

```
function draw ()
```

### Return Value

Type: HTMLElement
HTMLElement. The control DOM element.

## Example

The following code uses the draw method in a method that creates a **ToolStrip** programmatically:

**JavaScript**

```
function loadChatWindow(sender)
{
 var tools = new ui.ToolStrip();
 tools.theme = theme;
 tools.scrollable = false;
 tools.allowDrag = false;
 tools.itemClick.addEventListener(onToolClick);

 tools.items.add(new ui.ToolStripItem(ui.ToolStripItemType.Separator));

 sender.element.querySelector("#toolstrip").appendChild(tools.draw());
 tools.attach();
}
```

## See Also

UIControl Members
UIControl Class
MindFusion.Common Namespace

# Unit Class

Represents a length measurement.

## Syntax

**JavaScript**

```
// class
Unit.prototype = {}
```

## Example

The following code creates a new **ListView** and specifies its size in **Unit**.percentage.

**JavaScript**

```
var ui = MindFusion.UI;

var listView;
var host;
var theme = "gray";

document.addEventListener("DOMContentLoaded", function ()
{
   listView = new ui.ListView(document.getElementById("listView"));
   listView.width = listView.height = ui.Unit.percentage(100);
   listView.theme = theme;
   listView.itemSize = ui.Unit.pixel(100);
   listView.itemDoubleClick.addEventListener(handleItemDoubleClick);

   ........................................
   ........................................
}
```

## Inheritance Hierarchy

**MindFusion.Common.Unit**

## See Also

Unit Members
MindFusion.Common Namespace

# Unit Members

Represents a length measurement.

The following tables list the members exposed by the Unit type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | Unit | Initializes a new instance of the **Unit** class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | isEmpty | Gets a value representing whether this Unit instance has a set value. |
| | type | The unit type. |
| | value | The unit value. |

## Public Methods

| | Name | Description |
|---|---|---|
| | empty | Creates an empty **Unit** instance. |
| | parse | Creates a **Unit** instance from a string representation. |
| | percentage | Creates a **Unit** instance with the specified value and UnitType.Percent. |
| | pixel | Creates a Unit instance with the specified value and **UnitType**.Pixel. |
| | toString | Returns a string representation of this Unit. |

## See Also

Unit Class
MindFusion.Common Namespace

# Unit Constructor

Initializes a new instance of the Unit class.

**Namespace:** MindFusion.Common
**File:** Unit.js

## Syntax

**JavaScript**

```
function Unit ([value, [type]])
```

### Parameters

*value*
Optional.
Type: Number
Number. The value of the unit.

*type*
Optional.
Type: UnitType
UnitType. The type of the unit.

## Example

The following code creates a new **ListView** and specifies its size in Unit.percentage.

**JavaScript**

```
var ui = MindFusion.UI;

var listView;
var host;
var theme = "gray";

document.addEventListener("DOMContentLoaded", function ()
{
   listView = new ui.ListView(document.getElementById("listView"));
   listView.width = listView.height = ui.Unit.percentage(100);
   listView.theme = theme;
   listView.itemSize = ui.Unit.pixel(100);
   listView.itemDoubleClick.addEventListener(handleItemDoubleClick);

   ......................................
   ......................................
}
```

## See Also

Unit Class
MindFusion.Common Namespace

# Unit Methods

## Unit.empty Method

Creates an empty Unit instance.

**Namespace:** MindFusion.Common
**File:** Unit.js

## Syntax

**JavaScript**

```
function empty ()
```

### Return Value

Unit. The new **Unit**.

## Example

The following code creates a new **ListView** and sets its **minWidth** to an empty Unit.

**JavaScript**

```
var ui = MindFusion.UI;

var listView;
var host;
var theme = "gray";

document.addEventListener("DOMContentLoaded", function ()
{
   listView = new ui.ListView(document.getElementById("listView"));
   listView.minWidth = ui.Unit.empty();
   listView.width = listView.height = ui.Unit.percentage(100);
   listView.theme = theme;
   listView.itemSize = ui.Unit.pixel(100);
   listView.itemDoubleClick.addEventListener(handleItemDoubleClick);

   ......................................
   ......................................
}
```

## See Also

Unit Members
Unit Class
MindFusion.Common Namespace

# Unit.parse Method

Creates a Unit instance from a string representation.

**Namespace:** MindFusion.Common
**File:** Unit.js

## Syntax

**JavaScript**

```
function parse (value)
```

### Parameters

*value*
Type: String
String. The string representation of the unit.

### Return Value

Unit. The new **Unit**, or null if the string is invalid.

## Example

The following code creates a new Unit using the parse method.

JavaScript

```
var myMeasure = "80";
var unit = ui.Unit.parse(myMeasure);
```

## See Also

Unit Members
Unit Class
MindFusion.Common Namespace

# Unit.percentage Method

Creates a Unit instance with the specified value and UnitType.Percent.

**Namespace:** MindFusion.Common
**File:** Unit.js

## Syntax

JavaScript

```
function percentage (value)
```

### Parameters

*value*
Type: Number
Number. The value of the unit.

### Return Value

Type: Unit
Unit. The new Unit.

## Example

The following code uses the UnitType enumeration to check the type of a Unit value:

JavaScript

```
var m = MindFusion.Mapping;

var unit = new m.Unit();

if(unit.type == m.UnitType.Percent)
{
   var length = unit.value;
}
```

## See Also

Unit Members
Unit Class
MindFusion.Common Namespace

# Unit.pixel Method

Creates a Unit instance with the specified value and UnitType.Pixel.

**Namespace:** MindFusion.Common
**File:** Unit.js

## Syntax

**JavaScript**

```
function pixel (value)
```

### Parameters

*value*
Type: Number
Number. The value of the unit.

### Return Value

Unit. The new Unit.

## Example

The following code creates a new **ListView** and specifies its size in Unit.pixel.

**JavaScript**

```
var ui = MindFusion.UI;

var listView;
var host;
var theme = "gray";

document.addEventListener("DOMContentLoaded", function ()
{
   listView = new ui.ListView(document.getElementById("listView"));
   listView.minWidth = ui.Unit.empty();
   listView.width = listView.height = ui.Unit.pixel(100);
   listView.theme = theme;
   listView.itemSize = ui.Unit.pixel(100);
   listView.itemDoubleClick.addEventListener(handleItemDoubleClick);

   ......................................
   ......................................
}
```

## See Also

Unit Members
Unit Class
MindFusion.Common Namespace

# Unit.toString Method

Returns a string representation of this Unit.

**Namespace:** MindFusion.Common
**File:** Unit.js

## Syntax

**JavaScript**

```
function toString ()
```

### Return Value

Type: String
String. The string representation.

## Example

The following code converts the width of a **listView**, which is of type Unit, to string.

```
JavaScript
var size = listView.width.toString();
```

## See Also

Unit Members
Unit Class
MindFusion.Common Namespace

# Unit Properties

# Unit.isEmpty Property

Gets a value representing whether this Unit instance has a set value.

**Namespace:** MindFusion.Common
**File:** Unit.js

## Syntax

```
JavaScript
get isEmpty() {}
```

### Property Value

Type: Boolean
Boolean. true if the Unit's value is null, otherwise false.

## Example

The following code checks if the **minWidth** of a **Window** is empty.

```
JavaScript
if(window.minWidth.isEmpty)
{
    window.minWidth = ui.Unit.pixel(100);
}
```

## See Also

Unit Members
Unit Class
MindFusion.Common Namespace

# Unit.type Property

The unit type.

**Namespace:** MindFusion.Common
**File:** Unit.js

## Syntax

```
JavaScript
get type() {}
```

### Property Value

One of the UnitType enumeration values.

## Example

The following code checks the type of the **minWidth** of a **Window**.

```
JavaScript
if(window.minWidth.type == ui.UnitType.Percent)
{
    window.minWidth = ui.Unit.percent(50);
}
```

## See Also

Unit Members
Unit Class
MindFusion.Common Namespace

# Unit.value Property

The unit value.

**Namespace:** MindFusion.Common
**File:** Unit.js

## Syntax

```
JavaScript
get value() {}
```

### Property Value

Type: Number
Number. The value.

## Example

The following code checks the value of the **minWidth** of a **Window** and increases it, if necessary.

```
JavaScript
if(window.minWidth.value < 30 )
{
```

```
    window.minWidth = ui.Unit.percent(50);
}
```

## See Also

Unit Members
Unit Class
MindFusion.Common Namespace

# UnitType Enumeration

Specifies a unit of measurement.

**Namespace:** MindFusion.Common
**File:** Unit.js

## Syntax

**JavaScript**

```
// enum
UnitType = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **Percent** | The measurement is a percentage relative to the parent element. |
| | **Pixel** | The measurement is in pixels. |

## Example

The following code checks the type of the **minWidth** of a **Window**.

**JavaScript**

```
if(window.minWidth.type === ui.UnitType.Percent )
{
    window.minWidth = ui.Unit.percent(50);
}
```

## See Also

MindFusion.Common Namespace

# MindFusion.Common.Collections Namespace

## IEnumerable Class

Represents an array of arbitrary objects.

## Syntax

```
JavaScript
// class
IEnumerable.prototype = {}
```

## Remarks

You are not expected to create directly IEnumerable instances. Instead, derive from it or initialize one of the derived calsses: List, ObservableCollection etc.

## Example

The following sample creates a new instance of the **Menu** class and adss one **MenuItem** to it. The **items** collection of the **Menu** class derives from **IEnumerable**.

```
JavaScript
var menu = new ui.Menu(document.getElementById("links"));
menu.loadOnDemand = true;

var menuItem = new ui.MenuItem("Start", "index.html");
menuItem.target = "_blank";
menu.items.add(menuItem);
```

## Inheritance Hierarchy

**MindFusion.Common.Collections.IEnumerable**
MindFusion.Common.Collections.List

## See Also

IEnumerable Members
MindFusion.Common.Collections Namespace

# IEnumerable Members

Represents an array of arbitrary objects.

The following tables list the members exposed by the IEnumerable type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | IEnumerable | Initializes a new instance of the **IEnumerable** class. |

## Public Methods

| | Name | Description |
|---|---|---|
| | add | Adds an object to the end of the collection. |
| | addRange | Adds a range of elements to the end of the collection. |
| | clear | Clears the collection. |
| | clone | Creates a copy of the collection. |
| | contains | Checks if the given element is present in the collection. |

| | | |
|---|---|---|
| | copyTo | Copies a range of elements from this collection to a destination collection. |
| | count | Gets the number of elements. |
| | first | Returns the first element in the collection. |
| | forEach | Executes a provided function once for each element. |
| | indexOfItem | Gets the index of a given object in a collection. |
| | insert | Adds an element to the collection at the specified index. |
| | item | Gets the element at the given index. |
| | items | Gets the collection as an array. |
| | last | Returns the last element in the collection. |
| | max | Invokes a transform function on each item and returns the maximum value in a sequence of numbers. |
| | min | Invokes a transform function on each item and returns theminimum value in a sequence of numbers. |
| | remove | Removes an element from the collection. |
| | removeAt | Removes the element at the given index. |
| | removeRange | Removes a range of elements starting from the given index. |
| | reverse | Gets the collection as an array in reverse order. |
| | select | Projects each element of a sequence into a new form. |
| | sort | Sorts the underlying array. |
| | sum | Computes the sum of the sequence of number values that are obtained by invoking a transform function on each element. |
| | toArray | Returns a new Object array, containing the contents of the collection. |
| | where | Filters a sequence of values based on a predicate. |

## See Also

IEnumerable Class
MindFusion.Common.Collections Namespace

# IEnumerable Constructor

Initializes a new instance of the IEnumerable class.

**Namespace:** MindFusion.Common.Collections
**File:** IEnumerable.js

## Syntax

**JavaScript**

```
function IEnumerable ([items])
```

### Parameters

*items*
Optional.
Type: Array
Array. The underlying array data structure of the collection.

## Remarks

You are not expected to call directly the IEnumerable constructor. Instead, derive from it or initialize one of the derived calsses: List, ObservableCollection etc.

## Example

The following sample creates a new instance of the **Menu** class and adss one **MenuItem** to it. The items collection of the **Menu** class derives from **IEnumerable**.

**JavaScript**

```
var menu = new ui.Menu(document.getElementById("links"));
menu.loadOnDemand = true;

var menuItem = new ui.MenuItem("Start", "index.html");
menuItem.target = "_blank";
menu.items.add(menuItem);
```

## See Also

IEnumerable Class
MindFusion.Common.Collections Namespace

# IEnumerable Methods

# IEnumerable.add Method

Adds an object to the end of the collection.

**Namespace:** MindFusion.Common.Collections
**File:** IEnumerable.js

## Syntax

**JavaScript**

```
function add (item)
```

### Parameters

*item*
Type: Object
Object. The object to add.

## Example

The following sample creates a new instance of the **Menu** class and adss one **MenuItem** to it. The items collection of the **Menu** class derives from IEnumerable.

JavaScript

```
var menu = new ui.Menu(document.getElementById("links"));
menu.loadOnDemand = true;

var menuItem = new ui.MenuItem("Start", "index.html");
menuItem.target = "_blank";
menu.items.add(menuItem);
```

## See Also

IEnumerable Members
IEnumerable Class
MindFusion.Common.Collections Namespace

# IEnumerable.addRange Method

Adds a range of elements to the end of the collection.

**Namespace:** MindFusion.Common.Collections
**File:** IEnumerable.js

## Syntax

JavaScript

```
function addRange (range)
```

### Parameters

*range*
Type: Array
Array. The range to add.

## Example

The following code uses the **addRange** method to add an array of **TreeNode**-s to the root element of a **TreeView**.

JavaScript

```
// create a tree node that will be a root in our hierarchy
var root = new ui.TreeNode("body");
root.expanded = true;

var span = new ui.TreeNode("span");
var div = new ui.TreeNode("div");
var p = new ui.TreeNode("p");

var elements = [span, div, p];
root.items.addRange(elements);
```

## See Also

IEnumerable Members
IEnumerable Class
MindFusion.Common.Collections Namespace

# IEnumerable.clear Method

Clears the collection.

**Namespace:** MindFusion.Common.Collections
**File:** IEnumerable.js

## Syntax

```
JavaScript
function clear ()
```

## Example

The following example clears all **items** from a **TreeNode** element. The items property of **TreeNode** is of type IEnumerable.

```
JavaScript
rootElement.items.clear();
```

## See Also

IEnumerable Members
IEnumerable Class
MindFusion.Common.Collections Namespace

# IEnumerable.clone Method

Creates a copy of the collection.

**Namespace:** MindFusion.Common.Collections
**File:** IEnumerable.js

## Syntax

```
JavaScript
function clone ()
```

### Return Value

IEnumerable. A copy of this collection.

## Example

The following code copies the collection of **items** of a **TreeNode**, which is of type IEnumerable:

```
JavaScript
var treeView = new ui.TreeView(document.getElementById("treeView"));
treeView.width = treeView.height = ui.Unit.percentage(100);
treeView.theme = "standard";
treeView.loadOnDemand = true;

// create a tree node that will be a root in our hierarchy
var root = new ui.TreeNode("body");
root.expanded = true;
```

```
var span = new ui.TreeNode("span");
var div = new ui.TreeNode("div");
var p = new ui.TreeNode("p");

var elements = [span, div, p];
root.items.addRange(elements);

var copyCollection = root.items.clone();
```

## See Also

IEnumerable Members
IEnumerable Class
MindFusion.Common.Collections Namespace

# IEnumerable.contains Method

Checks if the given element is present in the collection.

**Namespace:** MindFusion.Common.Collections
**File:** IEnumerable.js

## Syntax

**JavaScript**

```
function contains (item)
```

### Parameters

*item*
Type: Object
Object. The object to check for.

### Return Value

Type: Boolean
Boolean. True if the element is found, otherwise false.

## Example

The following code checks if a **TreeNode** element called "div" is contained within the collection of **items** of a **TreeView** instance.

**JavaScript**

```
// create a tree node that will be a root in our hierarchy
var root = new ui.TreeNode("body");
root.expanded = true;

var span = new ui.TreeNode("span");
var div = new ui.TreeNode("div");
var p = new ui.TreeNode("p");

var elements = [span, div, p];
root.items.addRange(elements);
.....................

if (root.items.contains(div))
{
 //do something
}
```

## See Also

IEnumerable Members
IEnumerable Class
MindFusion.Common.Collections Namespace

# IEnumerable.copyTo Method

Copies a range of elements from this collection to a destination collection.

**Namespace:** MindFusion.Common.Collections
**File:** IEnumerable.js

## Syntax

**JavaScript**

```
function copyTo (destination, length, [sourceIndex, [destinationIndex]])
```

### Parameters

*destination*
IEnumerable. The destination collection.

*length*
Type: Number
Number. The length of the range to copy.

*sourceIndex*
Optional.
Type: Number
Number. The starting index of the range to copy.

*destinationIndex*
Optional.
Type: Number
Number. The index at which the range should be copied.

## Example

The following example copies all items from a collection called janEvents at the end of a collection called febEvents:

**JavaScript**

```
janEvents.items.copyTo(febEvents, janEvents.count(), 0, febEvents.count() - 1);
```

## See Also

IEnumerable Members
IEnumerable Class
MindFusion.Common.Collections Namespace

# IEnumerable.count Method

Gets the number of elements.

**Namespace:** MindFusion.Common.Collections
**File:** IEnumerable.js

## Syntax

```
JavaScript
function count ()
```

### Return Value

Type: Number
Number. The number of elements.

## Example

The following code checks if the count of **items** in a **TreeNode** is greater than 10:

```
JavaScript
// create a tree node that will be a root in our hierarchy
var root = new ui.TreeNode("body");
root.expanded = true;

var span = new ui.TreeNode("span");
var div = new ui.TreeNode("div");
var p = new ui.TreeNode("p");

var elements = [span, div, p];
root.items.addRange(elements);
.....................

if (root.items.count() > 10)
{
 //do something
}
```

## See Also

IEnumerable Members
IEnumerable Class
MindFusion.Common.Collections Namespace

# IEnumerable.first Method

Returns the first element in the collection.

**Namespace:** MindFusion.Common.Collections
**File:** IEnumerable.js

## Syntax

```
JavaScript
function first ()
```

### Return Value

Type: Object
Object. The first element in the collection.

## Example

The following code checks if the first item in the collection of **items** in a **TreeNode** is span:

JavaScript

```
// create a tree node that will be a root in our hierarchy
var root = new ui.TreeNode("body");
root.expanded = true;

var span = new ui.TreeNode("span");
var div = new ui.TreeNode("div");
var p = new ui.TreeNode("p");

var elements = [span, div, p];
root.items.addRange(elements);
.....................

if (root.items.first() == span)
{
 //do something
}
```

## See Also

IEnumerable Members
IEnumerable Class
MindFusion.Common.Collections Namespace

# IEnumerable.forEach Method

Executes a provided function once for each element.

**Namespace:** MindFusion.Common.Collections
**File:** IEnumerable.js

## Syntax

JavaScript

```
function forEach (callback, context)
```

### Parameters

*callback*
Type: function
function. A function to execute for each element.

*context*
Type: Object
Object. The invokation context.

## Example

The following code uses the forEach method in the event handler of the **selectionChanged** event of a **ListView** to change the background of **items**:

JavaScript

```
listView.selectionChanged.addEventListener(listSelectionChanged);

function listSelectionChanged(sender, args)
{
 if (args.oldItems)
 {
  args.oldItems.forEach(function (item)
  {
   item.data.style.backgroundColor = "";
```

```
  });
 }
 else
 {
  args.newItems.forEach(function (item)
  {
   item.data.style.backgroundColor = "#9caac6";
  });
 }
}
```

## See Also

IEnumerable Members
IEnumerable Class
MindFusion.Common.Collections Namespace

# IEnumerable.indexOfItem Method

Gets the index of a given object in a collection.

**Namespace:** MindFusion.Common.Collections
**File:** IEnumerable.js

## Syntax

**JavaScript**

```
function indexOfItem (obj, [fromIndex])
```

### Parameters

*obj*
Type: Object
Object. The object to look for.

*fromIndex*
Optional.
Type: Number
Number. The starting index to search from.

### Return Value

Type: Number
Number. The index of the object, or -1 if the object is not present in the collection.

## Example

The following code checks if the old items of a **ListView** instance, which have changed contain an item called item1:

**JavaScript**

```
listView.selectionChanged.addEventListener(listSelectionChanged);


function listSelectionChanged(sender, args)
{
     if( args.oldItems.contains(item1))
     {
     }
}
```

## See Also

IEnumerable Members
IEnumerable Class
MindFusion.Common.Collections Namespace

# IEnumerable.insert Method

Adds an element to the collection at the specified index.

**Namespace:** MindFusion.Common.Collections
**File:** IEnumerable.js

## Syntax

**JavaScript**

```
function insert (index, item)
```

### Parameters

*index*
Type: Number
Number. The index.

*item*
Type: Object
Object. The object to add.

## Example

The following code inserts a **TreeNode** at the second position among the **items** of a **TreeView** root:

**JavaScript**

```
var treeView = new ui.TreeView(document.getElementById("treeView"));
treeView.width = treeView.height = ui.Unit.percentage(100);
treeView.theme = "standard";
treeView.loadOnDemand = true;

// create a tree node that will be a root in our hierarchy
var root = new ui.TreeNode("body");
root.expanded = true;

var span = new ui.TreeNode("span");
var div = new ui.TreeNode("div");
var p = new ui.TreeNode("p");

var elements = [span, div, p];
root.items.addRange(elements);

var block = new ui.TreeNode("block");
root.items.insert(block, 1);
```

## See Also

IEnumerable Members
IEnumerable Class
MindFusion.Common.Collections Namespace

# IEnumerable.item Method

Gets the element at the given index.

**Namespace:** MindFusion.Common.Collections
**File:** IEnumerable.js

## Syntax

**JavaScript**

```
function item (index)
```

### Parameters

*index*
Type: Number
Number. The index.

### Return Value

Type: Object
Object. The element at the given index.

## Example

The following code checks if the old items of a **ListView** instance, which have changed have an item called item1 at the beginning of the collection:

**JavaScript**

```
listView.selectionChanged.addEventListener(listSelectionChanged);

function listSelectionChanged(sender, args)
{
     if( args.oldItems[0] == item1)
     {
             //do something
     }
}
```

## See Also

IEnumerable Members
IEnumerable Class
MindFusion.Common.Collections Namespace

# IEnumerable.items Method

Gets the collection as an array.

**Namespace:** MindFusion.Common.Collections
**File:** IEnumerable.js

## Syntax

```
JavaScript
function items ()
```

### Return Value

Type: Array
Array. The underlying array data structure of the collection.

## Example

The following code checks if the **items** of a ListView before the collection was changed are more than 5:

```
JavaScript
function listSelectionChanged(sender, args)
{
     if( args.oldItems.items().count() > 5)
     {
             //do something
     }
}
```

## See Also

IEnumerable Members
IEnumerable Class
MindFusion.Common.Collections Namespace

# IEnumerable.last Method

Returns the last element in the collection.

**Namespace:** MindFusion.Common.Collections
**File:** IEnumerable.js

## Syntax

```
JavaScript
function last ()
```

### Return Value

Type: Object
Object. The last element in the collection.

## Example

The following code checks if the last item in the collection of **items** of a **ListView** before the collection was changed is an object called _item:

```
JavaScript
```

```
listView.selectionChanged.addEventListener(listSelectionChanged);

function listSelectionChanged(sender, args)
{
     if( args.oldItems.items().last() == _item)
     {
             //do something
     }
}
```

## See Also

IEnumerable Members
IEnumerable Class
MindFusion.Common.Collections Namespace

# IEnumerable.max Method

Invokes a transform function on each item and returns the maximum value in a sequence of numbers.

**Namespace:** MindFusion.Common.Collections
**File:** IEnumerable.js

## Syntax

**JavaScript**

```
function max (selector)
```

### Parameters

*selector*
Type: function
function. A transform function to invoke on each element.

### Return Value

Type: Number
Number. The maximum number in the sequence.

## Example

The following code handles the **selectionChanged** event of a listView to check if the max selected item is greater than 1000:

**JavaScript**

```
listView.selectionChanged.addEventListener(listSelectionChanged);

function listSelectionChanged(sender, args)
{
     if( args.newItems.max() > 1000)
     {
             //do something
     }
}
```

## See Also

IEnumerable Members
IEnumerable Class
MindFusion.Common.Collections Namespace

# IEnumerable.min Method

Invokes a transform function on each item and returns theminimum value in a sequence of numbers.

**Namespace:** MindFusion.Common.Collections
**File:** IEnumerable.js

## Syntax

**JavaScript**

```
function min (selector)
```

### Parameters

*selector*
Type: function
function. A transform function to invoke on each element.

### Return Value

Type: Number
Number. The minimum number in the sequence.

## Example

The following code handles the **selectionChanged** event of a **ListView** to check if the min selected item is greater than 10:

**JavaScript**

```
listView.selectionChanged.addEventListener(listSelectionChanged);

function listSelectionChanged(sender, args)
{
     if( args.newItems.min() > 10)
     {
             //do something
     }
}
```

## See Also

IEnumerable Members
IEnumerable Class
MindFusion.Common.Collections Namespace

# IEnumerable.remove Method

Removes an element from the collection.

**Namespace:** MindFusion.Common.Collections
**File:** IEnumerable.js

## Syntax

JavaScript

```
function remove (item)
```

### Parameters

*item*
Type: Object
Object. The object to remove.

## Example

The following code checks if an item called _item is contained within the collection of **items** of a **ListView** after it has changed and if yes, removes it.

JavaScript

```
istView.selectionChanged.addEventListener(listSelectionChanged);

function listSelectionChanged(sender, args)
{
     if( args.newItems.items().contains(_item))
     {
             args.newItems.items().remove(_item);
     }
}
```

## See Also

IEnumerable Members
IEnumerable Class
MindFusion.Common.Collections Namespace

# IEnumerable.removeAt Method

Removes the element at the given index.

**Namespace:** MindFusion.Common.Collections
**File:** IEnumerable.js

## Syntax

JavaScript

```
function removeAt (index)
```

### Parameters

*index*
Type: Number
Number. The index.

## Example

The following code checks if the count of **items** in a **ListView** is more than 100 and if yes, removes the first one:

JavaScript

```
if (listView.items().count()>100)
   listView.items().removeAt(0);
```

## See Also

IEnumerable Members
IEnumerable Class
MindFusion.Common.Collections Namespace

# IEnumerable.removeRange Method

Removes a range of elements starting from the given index.

**Namespace:** MindFusion.Common.Collections
**File:** IEnumerable.js

## Syntax

**JavaScript**

```
function removeRange (index, count)
```

### Parameters

*index*
Type: Number
Number. The starting index of the range.

*count*
Type: Number
Number. The length of the range.

## Example

The following code checks if the count of **items** in a **ListView** is more than 100 and if yes, removes the first 50 elements:

**JavaScript**

```
if (listView.items().count()>100)
   listView.items().removeRange(0, 50);
```

## See Also

IEnumerable Members
IEnumerable Class
MindFusion.Common.Collections Namespace

# IEnumerable.reverse Method

Gets the collection as an array in reverse order.

**Namespace:** MindFusion.Common.Collections
**File:** IEnumerable.js

## Syntax

```
JavaScript
function reverse ()
```

### Return Value

Type: Array
Array. The underlying array data structure of the collection in reverse order.

## Example

The following code checks if the first item in the list of **items** in a **ListView** starts with 'z' and if yes, reverses the order of the items:

```
JavaScript
if (listView.items().item(0).startsWith("z", 0))
   listView.items().reverse();
```

## See Also

IEnumerable Members
IEnumerable Class
MindFusion.Common.Collections Namespace

# IEnumerable.select Method

Projects each element of a sequence into a new form.

**Namespace:** MindFusion.Common.Collections
**File:** IEnumerable.js

## Syntax

```
JavaScript
function select (selector)
```

### Parameters

*selector*
Type: function
function. A transform function to invoke on each element.

### Return Value

Type: IEnumerable
IEnumerable. An collection whose elements are the resultof invoking the transform function on each element.

## See Also

IEnumerable Members
IEnumerable Class
MindFusion.Common.Collections Namespace

# IEnumerable.sort Method

Sorts the underlying array.

**Namespace:** MindFusion.Common.Collections
**File:** IEnumerable.js

## Syntax

**JavaScript**

```
function sort ([compareFn])
```

### Parameters

*compareFn*
Optional.
Type: function
function. The comparing function.

## Example

The following code sorts a List with product instances based on the price of each one:

**JavaScript**

```
var product1 = new Object();
product1.title = "Ford";
product1.price = 25345;

var product2 = new Object();
product1.title = "BMW";
product1.price = 75300;

productList.addRange([product1, product2]);


productList.items().sort(function (a, b)
{
      return a.price - b.price.valueOf();
});
```

## See Also

IEnumerable Members
IEnumerable Class
MindFusion.Common.Collections Namespace

# IEnumerable.sum Method

Computes the sum of the sequence of number values that are obtained by invoking a transform function on each element.

**Namespace:** MindFusion.Common.Collections
**File:** IEnumerable.js

## Syntax

**JavaScript**

```
function sum (selector)
```

### Parameters

*selector*
Type: function
function. A transform function to invoke on each element.

### Return Value

Type: Number
Number. The sum of the number values in the sequence.

## Example

The following code implements sum on a list with product instances:

**JavaScript**

```
var product1 = new Object();
product1.title = "Ford";
product1.price = 25345;

var product2 = new Object();
product1.title = "BMW";
product1.price = 75300;

productList.addRange([product1, product2]);


productList.items().sum(function (a, b)
{
      return a.price + b.price.valueOf();
});
```

## See Also

IEnumerable Members
IEnumerable Class
MindFusion.Common.Collections Namespace

# IEnumerable.toArray Method

Returns a new Object array, containing the contents of the collection.

**Namespace:** MindFusion.Common.Collections
**File:** IEnumerable.js

## Syntax

**JavaScript**

```
function toArray ()
```

### Return Value

Type: Array
Array. The array.

## Example

The following code creates a List with products and gets its underlying array:

**JavaScript**

```
var product1 = new Object();
product1.title = "Ford";
product1.price = 25345;

var product2 = new Object();
product1.title = "BMW";
product1.price = 75300;
......
.....

productList.addRange([product1, product2, ...]);

var products = productList.toArray();
```

## See Also

IEnumerable Members
IEnumerable Class
MindFusion.Common.Collections Namespace

# IEnumerable.where Method

Filters a sequence of values based on a predicate.

**Namespace:** MindFusion.Common.Collections
**File:** IEnumerable.js

## Syntax

**JavaScript**

```
function where (selector)
```

### Parameters

*selector*
Type: function
function. A function to test each element for a condition.

### Return Value

IEnumerable. An collection that contains elements from the input sequence that satisfy the condition.

## Example

The following code uses the **where** function to search for an item in the collection of **items** in a **TreeView**:

**JavaScript**

```
var treeNode = treeView.flatItems.where(function (item) { if (item.data == d) return true;
}).first();
```

## See Also

IEnumerable Members
IEnumerable Class
MindFusion.Common.Collections Namespace

# List Class

Represents an array of arbitrary objects.

## Syntax

**JavaScript**

```
// class
List.prototype = {}
```

## Example

The following code creates a **List** with products:

**JavaScript**

```
var product1 = new Object();
product1.title = "Ford";
product1.price = 25345;

var product2 = new Object();
product1.title = "BMW";
product1.price = 75300;
......
.....

List products = new List([product1, product2, ...]);
```

## Inheritance Hierarchy

MindFusion.Common.Collections.IEnumerable
  **MindFusion.Common.Collections.List**
    MindFusion.Common.Collections.ObservableCollection

## See Also

List Members
MindFusion.Common.Collections Namespace

# List Members

Represents an array of arbitrary objects.

The following tables list the members exposed by the List type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | List | Initializes a new instance of the **List** class. |

## Public Methods

| | Name | Description |
|---|---|---|
|  | add | Adds an object to the end of the collection. (Inherited from IEnumerable.) |

| | | |
|---|---|---|
|  | addRange | Adds a range of elements to the end of the collection. (Inherited from IEnumerable.) |
|  | clear | Clears the collection. (Inherited from IEnumerable.) |
|  | clone | Creates a copy of the collection. (Inherited from IEnumerable.) |
|  | contains | Checks if the given element is present in the collection. (Inherited from IEnumerable.) |
|  | copyTo | Copies a range of elements from this collection to a destination collection. (Inherited from IEnumerable.) |
|  | count | Gets the number of elements. (Inherited from IEnumerable.) |
|  | first | Returns the first element in the collection. (Inherited from IEnumerable.) |
|  | forEach | Executes a provided function once for each element. (Inherited from IEnumerable.) |
|  | indexOfItem | Gets the index of a given object in a collection. (Inherited from IEnumerable.) |
|  | insert | Adds an element to the collection at the specified index. (Inherited from IEnumerable.) |
|  | item | Gets the element at the given index. (Inherited from IEnumerable.) |
|  | items | Gets the collection as an array. (Inherited from IEnumerable.) |
|  | last | Returns the last element in the collection. (Inherited from IEnumerable.) |
|  | max | Invokes a transform function on each item and returns the maximum value in a sequence of numbers. (Inherited from IEnumerable.) |
|  | min | Invokes a transform function on each item and returns theminimum value in a sequence of numbers. (Inherited from IEnumerable.) |
|  | remove | Removes an element from the collection. (Inherited from IEnumerable.) |
|  | removeAt | Removes the element at the given index. (Inherited from IEnumerable.) |
|  | removeRange | Removes a range of elements starting from the given index. (Inherited from IEnumerable.) |
|  | reverse | Gets the collection as an array in reverse order. (Inherited from IEnumerable.) |
|  | select | Projects each element of a sequence into a new form. (Inherited from IEnumerable.) |
|  | sort | Sorts the underlying array. (Inherited from IEnumerable.) |
|  | sum | Computes the sum of the sequence of number values that are obtained by invoking a transform function on each element. (Inherited from IEnumerable.) |

| | | |
|---|---|---|
| | toArray | Returns a new Object array, containing the contents of the collection. (Inherited from IEnumerable.) |
| | where | Filters a sequence of values based on a predicate. (Inherited from IEnumerable.) |

## See Also

List Class
MindFusion.Common.Collections Namespace

### List Constructor

Initializes a new instance of the List class.

**Namespace:** MindFusion.Common.Collections
**File:** List.js

## Syntax

**JavaScript**

```
function List ([items])
```

#### Parameters

*items*
Optional.
Type: Array
Array. The underlying array data structure of the list.

## Example

The following code creates a List with products:

**JavaScript**

```
var product1 = new Object();
product1.title = "Ford";
product1.price = 25345;

var product2 = new Object();
product1.title = "BMW";
product1.price = 75300;
......
.....

List products = new List([product1, product2, ...]);
```

## See Also

List Class
MindFusion.Common.Collections Namespace

# ObservableCollection Class

Represents a collection of arbitrary objects.

## Syntax

**JavaScript**

```
// class
ObservableCollection.prototype = {}
```

## Example

The following code creates an ObservableCollection with products.

**JavaScript**

```
var product1 = new Object();
product1.title = "Ford";
product1.price = 25345;

var product2 = new Object();
product1.title = "BMW";
product1.price = 75300;
......
.....

List products = new List([product1, product2, ...]);
```

## Inheritance Hierarchy

MindFusion.Common.Collections.IEnumerable
  MindFusion.Common.Collections.List
    **MindFusion.Common.Collections.ObservableCollection**

## See Also

ObservableCollection Members
MindFusion.Common.Collections Namespace

# ObservableCollection Members

Represents a collection of arbitrary objects.

The following tables list the members exposed by the ObservableCollection type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | ObservableCollection | Initializes a new instance of the **ObservableCollection** class. |

## Public Methods

| | Name | Description |
|---|---|---|
| | add | Adds an item to the collection. |
| | addRange | Adds a range of elements to the end of the collection. (Inherited from IEnumerable.) |
| | clear | Clears the collection. |

| | | |
|---|---|---|
| | clone | Creates a copy of the collection. (Inherited from IEnumerable.) |
| | contains | Checks if the given element is present in the collection. (Inherited from IEnumerable.) |
| | copyTo | Copies a range of elements from this collection to a destination collection. (Inherited from IEnumerable.) |
| | count | Gets the number of elements. (Inherited from IEnumerable.) |
| | first | Returns the first element in the collection. (Inherited from IEnumerable.) |
| | forEach | Executes a provided function once for each element. (Inherited from IEnumerable.) |
| | indexOfItem | Gets the index of a given object in a collection. (Inherited from IEnumerable.) |
| | insert | Adds an item to the collection at the specified index. |
| | item | Gets the element at the given index. (Inherited from IEnumerable.) |
| | items | Gets the collection as an array. (Inherited from IEnumerable.) |
| | last | Returns the last element in the collection. (Inherited from IEnumerable.) |
| | max | Invokes a transform function on each item and returns the maximum value in a sequence of numbers. (Inherited from IEnumerable.) |
| | min | Invokes a transform function on each item and returns theminimum value in a sequence of numbers. (Inherited from IEnumerable.) |
| | remove | Deletes an item from the collection. |
| | removeAt | Deletes the item at the specified index. |
| | removeRange | Deletes a range of items from the collection. |
| | reverse | Gets the collection as an array in reverse order. (Inherited from IEnumerable.) |
| | select | Projects each element of a sequence into a new form. (Inherited from IEnumerable.) |
| | sort | Sorts the underlying array. (Inherited from IEnumerable.) |
| | sum | Computes the sum of the sequence of number values that are obtained by invoking a transform function on each element. (Inherited from IEnumerable.) |
| | toArray | Returns a new Object array, containing the contents of the collection. (Inherited from IEnumerable.) |
| | where | Filters a sequence of values based on a predicate. (Inherited from IEnumerable.) |

## Public Events

| | Name | Description |
|---|---|---|
| | collectionChanged | Occurs when an item is added, removed, changed, moved, or the entire list is refreshed. |
| | collectionChanging | Occurs just before an item is added, removed, changed, moved, or the entire list is refreshed. |

## See Also

ObservableCollection Class
MindFusion.Common.Collections Namespace

# ObservableCollection Constructor

Initializes a new instance of the ObservableCollection class.

**Namespace:** MindFusion.Common.Collections
**File:** ObservableCollection.js

## Syntax

**JavaScript**

```
function ObservableCollection ([items])
```

### Parameters

*items*
Optional.
Type: Array
Array. The underlying array data structure of the collection.

## Example

The following example creates chart series stored in an ObservableCollection instance:

**JavaScript**

```
var series = new Collections.ObservableCollection();
series.add(new Charting.BarSeries(
      new Collections.List([20, 60, 40, 55]),
      new Collections.List(["Acer", "Biostar", "Foxconn", "Supermicro"]),
      new Collections.List(["", "", "", ""])));
```

## See Also

ObservableCollection Class
MindFusion.Common.Collections Namespace

# ObservableCollection Methods

# ObservableCollection.add Method

Adds an item to the collection.

**Namespace:** MindFusion.Common.Collections
**File:** ObservableCollection.js

## Syntax

**JavaScript**

```
function add (item)
```

### Parameters

*item*
Type: Object
Object. The item to add.

## Example

The following code creates a new List, which derives from ObservableCollection and adds several items to it:

**JavaScript**

```
JavaScript  Copy Code
var product1 = new Object();
product1.title = "Ford";
product1.price = 25345;

var product2 = new Object();
product1.title = "BMW";
product1.price = 75300;
......
.....

List products = new List();
products.add(product1);
products.add(product2);
```

## See Also

ObservableCollection Members
ObservableCollection Class
MindFusion.Common.Collections Namespace

# ObservableCollection.clear Method

Clears the collection.

**Namespace:** MindFusion.Common.Collections
**File:** ObservableCollection.js

## Syntax

**JavaScript**

```
function clear ()
```

## Example

The following code checks if the count of items in the collection is more than 100 and if so, clears the collection:

JavaScript

```
var products = new ObservableCollection([product1, product2, ...]);
.....
.....

if(products.count() > 100)
   products.clear();
```

## See Also

ObservableCollection Members
ObservableCollection Class
MindFusion.Common.Collections Namespace

# ObservableCollection.insert Method

Adds an item to the collection at the specified index.

**Namespace:** MindFusion.Common.Collections
**File:** ObservableCollection.js

## Syntax

JavaScript

```
function insert (index, item)
```

### Parameters

*index*

Type: Number
Number. The index.

*item*

Type: Object
Object. The object to add.

## Example

The following code creates a List with products. Then it adds several products to it and inserts one at the beginning of the **List**. List derives from ObservableCollection.

JavaScript

```
var product1 = new Object();
product1.title = "Ford";
product1.price = 25345;

var product2 = new Object();
product1.title = "BMW";
product1.price = 75300;
......
.....

List products = new List();
products.add(product1);
products.add(product2);


List products = new List([product1, product2, ...]);
.....
.....
```

```
if(products.count() > 100)
   products.insert(0, newProduct);
```

## See Also

ObservableCollection Members
ObservableCollection Class
MindFusion.Common.Collections Namespace

# ObservableCollection.remove Method

Deletes an item from the collection.

**Namespace:** MindFusion.Common.Collections
**File:** ObservableCollection.js

## Syntax

**JavaScript**

```
function remove (item)
```

### Parameters

*item*
Type: Object
Object. The item to remove.

## Example

The following code creates a List with products. Then it adds several products to it and checks if a product called productX is present and if yes - removes it. **List** derives from ObservableCollection.

**JavaScript**

```
var product1 = new Object();
product1.title = "Ford";
product1.price = 25345;

var product2 = new Object();
product1.title = "BMW";
product1.price = 75300;
......
.....

List products = new List();
products.add(product1);
products.add(product2);


List products = new List([product1, product2, ...]);
.....
.....

if(products.contains(productX))
   products.remove(productX);
```

## See Also

ObservableCollection Members
ObservableCollection Class
MindFusion.Common.Collections Namespace

# ObservableCollection.removeAt Method

Deletes the item at the specified index.

**Namespace:** MindFusion.Common.Collections
**File:** ObservableCollection.js

## Syntax

```
JavaScript
function removeAt (itemIndex)
```

### Parameters

*itemIndex*
Type: Number
Number. The index to remove at.

## Example

The following code checks if the List with **items** in a **ListView** is more than 100 and if yes - removes the first element.

```
JavaScript
if(listView.items().count > 100)
   products.removeAt(0);
```

## See Also

ObservableCollection Members
ObservableCollection Class
MindFusion.Common.Collections Namespace

# ObservableCollection.removeRange Method

Deletes a range of items from the collection.

**Namespace:** MindFusion.Common.Collections
**File:** ObservableCollection.js

## Syntax

```
JavaScript
function removeRange (index, count)
```

### Parameters

*index*
Type: Number
Number. The starting index of the range to remove.

*count*
Type: Number
Number. The length of the range to remove.

## Example

The following code checks if the List with **items** in a **ListView** is more than 100 and if yes - leaves only 50 of them.

**JavaScript**

```
if(listView.items().count > 100)
   products.removeRange(50, listView.items().count() - 1);
```

## See Also

ObservableCollection Members
ObservableCollection Class
MindFusion.Common.Collections Namespace

# ObservableCollection Events

## ObservableCollection.collectionChanged Event

Occurs when an item is added, removed, changed, moved, or the entire list is refreshed.

**Namespace:** MindFusion.Common.Collections
**File:** ObservableCollection.js

## Syntax

**JavaScript**

```
EventDispatcher collectionChanged
```

## Event Data

The event handler method receives the following arguments:
*sender*
An ObservableCollection instance, which is the source of the event. This object will be passed to the handler function as the first argument.
args
A NotifyCollectionChangedEventArgs instance, which contains event data. This object will be passed to the handler function as the second argument.

## Example

The following code handles the **collectionChanged** event on the collection of **items** of a **ListView** object that is of type **ObservableCollection**:

**JavaScript**

```
var nodeList = new ui.ListView(document.getElementById("nodeList"));

nodeList.items.collectionChanged.addEventListener(itemsCollectionChangedEventHandler);


function itemsCollectionChangedEventHandler(sender, args)
{
 if(args.action == ui.NotifyCollectionChangedAction.Add)
 {
       //do something
 }
}
```

## See Also

ObservableCollection Members
ObservableCollection Class
MindFusion.Common.Collections Namespace

# ObservableCollection.collectionChanging Event

Occurs just before an item is added, removed, changed, moved, or the entire list is refreshed.

**Namespace:** MindFusion.Common.Collections
**File:** ObservableCollection.js

## Syntax

**JavaScript**

```
EventDispatcher collectionChanging
```

## Event Data

The event handler method receives the following arguments:
*sender*
An ObservableCollection instance, which is the source of the event. This object will be passed to the handler function as the first argument.
args
A NotifyCollectionChangingEventArgs instance, which contains event data. This object will be passed to the handler function as the second argument.

## Remarks

The cancel property of **NotifyCollectionChangingEventArgs** allows you to stop the action.

## Example

The following code handles the **collectionChanging** event on the **items** collection of a **ListView** object that is of type **ObservableCollection**:

**JavaScript**

```
var nodeList = new ui.ListView(document.getElementById("nodeList"));

nodeList.items.collectionChanging.addEventListener(itemsCollectionChangingEventHandler);


function itemsCollectionChangingEventHandler(sender, args)
{
 if(args.action == ui.NotifyCollectionChangedAction.Add)
 {
     //do something
 }
}
```

## See Also

ObservableCollection Members
ObservableCollection Class
MindFusion.Common.Collections Namespace

# MindFusion.Controls Namespace

## Alignment Enumeration

Specifies the alignment of text relative to its layout rectangle.

**Namespace:** MindFusion.Controls
**File:** Enum.js

### Syntax

```
// enum
Alignment = {}
```

### Members

| | Member name | Description |
|---|---|---|
| | **Center** | The text is drawn in the center of the layout rectangle. |
| | **Far** | The text is drawn in the far corner of the layout rectangle. |
| | **Near** | The text is drawn in the near corner of the layout rectangle. |

### See Also

MindFusion.Controls Namespace

## CancelEventArgs Class

Provides a value to use with cancellable events.

### Syntax

```
// class
CancelEventArgs.prototype = {}
```

### Inheritance Hierarchy

MindFusion.Controls.EventArgs
  **MindFusion.Controls.CancelEventArgs**

### See Also

CancelEventArgs Members
MindFusion.Controls Namespace

# CancelEventArgs Members

Provides a value to use with cancellable events.

The following tables list the members exposed by the CancelEventArgs type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | CancelEventArgs | Initializes a new instance of the CancelEventArgs class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | cancel | Gets or sets a value indicating whether to allow the current operation. |
|  | handled | Gets or sets a value indicating whether the event has been handled. (Inherited from EventArgs.) |

## See Also

CancelEventArgs Class
MindFusion.Controls Namespace

# CancelEventArgs Constructor

Initializes a new instance of the CancelEventArgs class.

**Namespace:** MindFusion.Controls
**File:** EventArgs.js

## Syntax

**JavaScript**

```
function CancelEventArgs ()
```

## See Also

CancelEventArgs Class
MindFusion.Controls Namespace

# CancelEventArgs Properties

# CancelEventArgs.cancel Property

Gets or sets a value indicating whether to allow the current operation.

**Namespace:** MindFusion.Controls
**File:** EventArgs.js

## Syntax

| JavaScript |
| --- |
| `get cancel() {}`<br>`set cancel(value) {}` |

### Property Value

*Boolean*

**true** to cancel the operation, or **false** otherwise. The default value is **false**.

## See Also

CancelEventArgs Members
CancelEventArgs Class
MindFusion.Controls Namespace

# Canvas Class

Handles drawing on a CanvasRenderingContext2D.

## Syntax

| JavaScript |
| --- |
| `// class`<br>`Canvas.prototype = {}` |

## Inheritance Hierarchy

MindFusion.Controls.Disposable
  **MindFusion.Controls.Canvas**

## See Also

Canvas Members
MindFusion.Controls Namespace

# Canvas Members

Handles drawing on a CanvasRenderingContext2D.

The following tables list the members exposed by the Canvas type.

## Public Constructors

| | Name | Description |
| --- | --- | --- |
|  | Canvas | Initializes a new instance of the Canvas class. |

## Public Properties

| | Name | Description |
| --- | --- | --- |
|  | bounds | Gets the underlying **Canvas** element's logical bounds. |

| | | |
|---|---|---|
| | element | Returns a reference to the control's DOM element. (Inherited from Disposable.) |
| | enabled | Gets or sets whether mouse events are enabled. (Inherited from Disposable.) |
| | measureUnit | Gets or sets the unit of measure used for logical coordinates. |
| | minVisibleFontSize | Gets or sets a threshold value that hides text if scaled font sizes become smaller. |
| | scale | Gets the current scale of this **Canvas**. |

## Public Methods

| | Name | Description |
|---|---|---|
| | addEventListener | Registers a single event listener on the instance. (Inherited from Disposable.) |
| | dispose | Overrides Disposable.dispose |
| | invalidate | Invalidates the canvas or a region of the canvas, causing it to be repainted. |
| | raiseEvent | Raises an event. (Inherited from Disposable.) |
| | removeEventListener | Removes a single event listener attached to the instance. (Inherited from Disposable.) |
| | repaint | Repaints the canvas. |

## See Also

Canvas Class
MindFusion.Controls Namespace

# Canvas Constructor

Initializes a new instance of the Canvas class.

**Namespace:** MindFusion.Controls
**File:** Canvas.js

## Syntax

**JavaScript**

```
function Canvas ()
```

## See Also

Canvas Class
MindFusion.Controls Namespace

# Canvas Methods

## Canvas.dispose Method

Overrides Disposable.dispose

**Namespace:** MindFusion.Controls
**File:** Canvas.js

### Syntax

```
JavaScript
function dispose ()
```

### See Also

Canvas Members
Canvas Class
MindFusion.Controls Namespace

## Canvas.invalidate Method

Invalidates the canvas or a region of the canvas, causing it to be repainted.

**Namespace:** MindFusion.Controls
**File:** Canvas.js

### Syntax

```
JavaScript
function invalidate ([rect, [force]])
```

#### Parameters

*rect*
Optional.
Rect. A Rect instance specifying the region that should be repainted. If not specified, the whole diagram will be repainted.

*force*
Optional.
Boolean. **true** to force redraw even if currently painting, or **false** otherwise.

### See Also

Canvas Members
Canvas Class
MindFusion.Controls Namespace

## Canvas.repaint Method

Repaints the canvas.

**Namespace:** MindFusion.Controls
**File:** Canvas.js

## Syntax

```
JavaScript
function repaint ([printOptions])
```

### Parameters

*printOptions*
Optional.
Object. For internal use.

## See Also

Canvas Members
Canvas Class
MindFusion.Controls Namespace

# Canvas Properties

# Canvas.bounds Property

Gets the underlying Canvas element's logical bounds.

**Namespace:** MindFusion.Controls
**File:** Canvas.js

## Syntax

```
JavaScript
get bounds() {}
set bounds(value) {}
```

### Property Value

*Rect*

A Rect instance specifying canvas bounds in current measureUnit.

## See Also

Canvas Members
Canvas Class
MindFusion.Controls Namespace
**Diagram.enableSpatialIndex Property**

# Canvas.measureUnit Property

Gets or sets the unit of measure used for logical coordinates.

**Namespace:** MindFusion.Controls
**File:** Canvas.js

## Syntax

JavaScript

```
get measureUnit() {}
set measureUnit(value) {}
```

### Property Value

*GraphicsUnit*

A member of the GraphicsUnit enumeration.

## See Also

Canvas Members
Canvas Class
MindFusion.Controls Namespace

# Canvas.minVisibleFontSize Property

Gets or sets a threshold value that hides text if scaled font sizes become smaller.

**Namespace:** MindFusion.Controls
**File:** Canvas.js

## Syntax

JavaScript

```
get minVisibleFontSize() {}
set minVisibleFontSize(value) {}
```

### Property Value

*Number*

A value specifying the smallest visible font size.

## See Also

Canvas Members
Canvas Class
MindFusion.Controls Namespace

# Canvas.scale Property

Gets the current scale of this Canvas.

**Namespace:** MindFusion.Controls
**File:** Canvas.js

## Syntax

JavaScript

```
get scale() {}
set scale(value) {}
```

### Property Value

*Number*

The canvas scale.

## See Also

Canvas Members
Canvas Class
MindFusion.Controls Namespace

# CanvasControl Class

The CanvasControl class represents a wrapper class for the HTML5 Canvas element.

## Syntax

| JavaScript |
| --- |
| `// class`<br>`CanvasControl.prototype = {}` |

## Inheritance Hierarchy

MindFusion.Controls.Disposable
  MindFusion.Controls.Control
    **MindFusion.Controls.CanvasControl**
      MindFusion.Controls.ZoomControl

## See Also

CanvasControl Members
MindFusion.Controls Namespace

## CanvasControl Members

The CanvasControl class represents a wrapper class for the HTML5 Canvas element.

The following tables list the members exposed by the CanvasControl type.

## Public Constructors

| | Name | Description |
| --- | --- | --- |
|  | CanvasControl | Initializes a new instance of the CanvasControl class. |

## Public Properties

| | Name | Description |
| --- | --- | --- |
|  | element | Returns a reference to the control's DOM element. (Inherited from Control.) |
|  | enabled | Gets or sets whether mouse events are enabled. (Inherited from Control.) |

| | | |
|---|---|---|
| | measureUnit | Gets the unit of measure used for logical coordinates. |

## Public Methods

| | Name | Description |
|---|---|---|
| | addEventListener | Registers a single event listener on the instance. (Inherited from Disposable.) |
| S | addHandlers | Adds an event listener to a DOM element. (Inherited from Control.) |
| S | clearHandlers | Removes all event listeners from a DOM element. (Inherited from Control.) |
| | clientToDoc | Transforms a point from client to document coordinates. |
| | docToClient | Transforms a point from document to client coordinates. |
| S | fromJson | Constructs a JavaScript object from a JSON string. (Inherited from Control.) |
| | raiseEvent | Raises an event. (Inherited from Disposable.) |
| | registerForSubmit | Registers an onsubmit handler for the Control's parent html form to flush postback data. (Inherited from Control.) |
| | removeEventListener | Removes a single event listener attached to the instance. (Inherited from Disposable.) |
| S | toJson | Converts a JavaScript object to a JSON string. (Inherited from Control.) |

## See Also

CanvasControl Class
MindFusion.Controls Namespace

### CanvasControl Constructor

Initializes a new instance of the CanvasControl class.

**Namespace:** MindFusion.Controls
**File:** CanvasControl.js

## Syntax

**JavaScript**

```
function CanvasControl (element, [canvas])
```

#### Parameters

*element*
Type: HTMLCanvasElement
HTMLCanvasElement. The Canvas DOM Element this CanvasControl is associated with.

*canvas*
Optional.

Type: Canvas
Canvas. The Canvas instance this CanvasControl is associated with.

## See Also

CanvasControl Class
MindFusion.Controls Namespace

# CanvasControl Methods

## CanvasControl.clientToDoc Method

Transforms a point from client to document coordinates.

**Namespace:** MindFusion.Controls
**File:** CanvasControl.js

## Syntax

**JavaScript**

```
function clientToDoc (point)
```

### Parameters

*point*
Type: Point
Point. The point to transform.

### Return Value

Type: Point
Point. The transformed point.

## See Also

CanvasControl Members
CanvasControl Class
MindFusion.Controls Namespace

## CanvasControl.docToClient Method

Transforms a point from document to client coordinates.

**Namespace:** MindFusion.Controls
**File:** CanvasControl.js

## Syntax

**JavaScript**

```
function docToClient (point)
```

### Parameters

*point*

Type: Point
Point. The point to transform.

### Return Value

Type: Point
Point. The transformed point.

## See Also

CanvasControl Members
CanvasControl Class
MindFusion.Controls Namespace

# CanvasControl Properties

## CanvasControl.measureUnit Property

Gets the unit of measure used for logical coordinates.

**Namespace:** MindFusion.Controls
**File:** CanvasControl.js

## Syntax

**JavaScript**

```
get measureUnit() {}
set measureUnit(value) {}
```

### Property Value

*GraphicsUnit*

An instance of the GraphicsUnit class.

## See Also

CanvasControl Members
CanvasControl Class
MindFusion.Controls Namespace

# Control Class

A base class for MindFusion controls.

## Syntax

**JavaScript**

```
// class
Control.prototype = {}
```

## Inheritance Hierarchy

MindFusion.Controls.Disposable
**MindFusion.Controls.Control**
MindFusion.Controls.CanvasControl

## See Also

Control Members
MindFusion.Controls Namespace

# Control Members

A base class for MindFusion controls.

The following tables list the members exposed by the Control type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | Control | Initializes a new instance of the Control class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | element | Returns a reference to the control's DOM element. |
| | enabled | Gets or sets whether mouse events are enabled. |

## Public Methods

| | Name | Description |
|---|---|---|
| | addEventListener | Registers a single event listener on the instance. (Inherited from Disposable.) |
| S | addHandlers | Adds an event listener to a DOM element. |
| S | clearHandlers | Removes all event listeners from a DOM element. |
| S | fromJson | Constructs a JavaScript object from a JSON string. |
| | raiseEvent | Raises an event. (Inherited from Disposable.) |
| | registerForSubmit | Registers an onsubmit handler for the Control's parent html form to flush postback data. |
| | removeEventListener | Removes a single event listener attached to the instance. (Inherited from Disposable.) |
| S | toJson | Converts a JavaScript object to a JSON string. |

## See Also

Control Class
MindFusion.Controls Namespace

## Control Constructor

Initializes a new instance of the Control class.

**Namespace:** MindFusion.Controls
**File:** Control.js

### Syntax

**JavaScript**

```
function Control (element)
```

#### Parameters

*element*
Type: HTMLElement
HTMLElement. The DOM Element this Control is associated with.

### See Also

Control Class
MindFusion.Controls Namespace

## Control Methods

## Control.addHandlers Method

Adds an event listener to a DOM element.

**Namespace:** MindFusion.Controls
**File:** Control.js

### Syntax

**JavaScript**

```
function addHandlers ()
```

### See Also

Control Members
Control Class
MindFusion.Controls Namespace

## Control.clearHandlers Method

Removes all event listeners from a DOM element.

**Namespace:** MindFusion.Controls
**File:** Control.js

### Syntax

**JavaScript**

```
function clearHandlers ()
```

## See Also

Control Members
Control Class
MindFusion.Controls Namespace

# Control.fromJson Method

Constructs a JavaScript object from a JSON string.

**Namespace:** MindFusion.Controls
**File:** Control.js

## Syntax

**JavaScript**

```
function fromJson ()
```

### Return Value

Type: Object
Object. object The corresponding to the given JSON text.

## See Also

Control Members
Control Class
MindFusion.Controls Namespace

# Control.registerForSubmit Method

Registers an onsubmit handler for the Control's parent html form to flush postback data.

**Namespace:** MindFusion.Controls
**File:** Control.js

## Syntax

**JavaScript**

```
function registerForSubmit (id)
```

### Parameters

*id*
Type: String
String. The id of the hidden field to flush the data to.

## See Also

Control Members
Control Class
MindFusion.Controls Namespace

## Control.toJson Method

Converts a JavaScript object to a JSON string.

**Namespace:** MindFusion.Controls
**File:** Control.js

### Syntax

```
JavaScript
function toJson (object)
```

#### Parameters

*object*
Type: Object
Object. The object to stringify.

#### Return Value

Type: String
String. The JSON string.

### See Also

Control Members
Control Class
MindFusion.Controls Namespace

## Control Properties

## Control.element Property

Returns a reference to the control's DOM element.

**Namespace:** MindFusion.Controls
**File:** Control.js

### Syntax

```
JavaScript
get element() {}
set element(value) {}
```

#### Property Value

Type: HTMLElement
*HTMLElement*

The HTML element wrapped by this control.

## See Also

Control Members
Control Class
MindFusion.Controls Namespace

### Control.enabled Property

Gets or sets whether mouse events are enabled.

**Namespace:** MindFusion.Controls
**File:** Control.js

## Syntax

**JavaScript**

```
get enabled() {}
set enabled(value) {}
```

#### Property Value

Type: Boolean
*Boolean*

true if the control should respond to mouse events, or false otherwise.

## See Also

Control Members
Control Class
MindFusion.Controls Namespace

# Disposable Class

Disposable abstract class.

## Syntax

**JavaScript**

```
// class
Disposable.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Controls.Disposable**
MindFusion.Controls.Canvas
MindFusion.Controls.Control

## See Also

Disposable Members
MindFusion.Controls Namespace

# Disposable Members

Disposable abstract class.

The following tables list the members exposed by the Disposable type.

## Public Properties

| | Name | Description |
|---|---|---|
| | element | Returns a reference to the control's DOM element. |
| | enabled | Gets or sets whether mouse events are enabled. |

## Public Methods

| | Name | Description |
|---|---|---|
| | addEventListener | Registers a single event listener on the instance. |
| | raiseEvent | Raises an event. |
| | removeEventListener | Removes a single event listener attached to the instance. |

## See Also

Disposable Class
MindFusion.Controls Namespace

# Disposable Methods

## Disposable.addEventListener Method

Registers a single event listener on the instance.

**Namespace:** MindFusion.Controls
**File:** Control.js

## Syntax

**JavaScript**

```
function addEventListener (eventName, handler, [element])
```

### Parameters

*eventName*
Type: String
String. The name of the event.

*handler*
Type: Method
Method. Represents the method that will handle the event specified with eventName.

*element*
Optional.
Type: Object
Object. For internal use.

## See Also

Disposable Members
Disposable Class
MindFusion.Controls Namespace
removeEventListener Method

# Disposable.raiseEvent Method

Raises an event.

**Namespace:** MindFusion.Controls
**File:** Control.js

## Syntax

**JavaScript**

```
function raiseEvent (eventName, args, [element])
```

### Parameters

*eventName*
Type: String
String. The name of the event to raise.

*args*
Type: EventArgs
EventArgs. An instance of type EventArgs that holds data for the event specified with eventName.

*element*
Optional.
Type: Object
Object. For internal use.

## See Also

Disposable Members
Disposable Class
MindFusion.Controls Namespace

# Disposable.removeEventListener Method

Removes a single event listener attached to the instance.

**Namespace:** MindFusion.Controls
**File:** Control.js

## Syntax

**JavaScript**

```
function removeEventListener (eventName, handler, [element])
```

### Parameters

*eventName*

Type: String
String. The name of the event.

*handler*
Type: Method
Method. Represents the method that handles the event specified with eventName.

*element*
Optional.
Type: Object
Object. For internal use.

## See Also

Disposable Members
Disposable Class
MindFusion.Controls Namespace
addEventListener Method

# Disposable Properties

## Disposable.element Property

Returns a reference to the control's DOM element.

**Namespace:** MindFusion.Controls
**File:** Control.js

## Syntax

**JavaScript**

```
get element() {}
set element(value) {}
```

### Property Value

Type: HTMLElement
*HTMLElement*

The HTML element wrapped by this control.

## See Also

Disposable Members
Disposable Class
MindFusion.Controls Namespace

## Disposable.enabled Property

Gets or sets whether mouse events are enabled.

**Namespace:** MindFusion.Controls
**File:** Control.js

## Syntax

```
JavaScript
get enabled() {}
set enabled(value) {}
```

### Property Value

Type: Boolean
*Boolean*

true if the control should respond to mouse events, or false otherwise.

## See Also

Disposable Members
Disposable Class
MindFusion.Controls Namespace

# DomUtils Class

Contains DOM-related helper functions.

## Syntax

```
JavaScript
// class
DomUtils.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Controls.DomUtils**

## See Also

DomUtils Members
MindFusion.Controls Namespace

# DomUtils Members

Contains DOM-related helper functions.

The following tables list the members exposed by the DomUtils type.

## Public Methods

| | Name | Description |
|---|---|---|
|  | formatString | Formats the specified string according to the specified parameters. |
|  | getBounds | Gets the bounds of the specified element. |
|  | getElementAttributes | Gets an object containing element attributes that have values. |

| | | |
|---|---|---|
| | HTMLtoXHTML | Converts HTML to XHMTL. |
| | loadImageList | Loads the images from the specified list of image url-s. |
| | setBounds | Sets the bounds of the specified element. |
| | setSize | Sets the size of the specified element. |
| | toDataUrl | Converts an image to base64 string. |

## See Also

DomUtils Class
MindFusion.Controls Namespace

# DomUtils Methods

# DomUtils.formatString Method

Formats the specified string according to the specified parameters.

**Namespace:** MindFusion.Controls
**File:** DomUtils.js

## Syntax

**JavaScript**

```
function formatString ()
```

## See Also

DomUtils Members
DomUtils Class
MindFusion.Controls Namespace

# DomUtils.getBounds Method

Gets the bounds of the specified element.

**Namespace:** MindFusion.Controls
**File:** DomUtils.js

## Syntax

**JavaScript**

```
function getBounds (element, [parent, [includeScroll]])
```

### Parameters

*element*
Type: HTMLElement
HTMLElement. The element to check.

*parent*

Optional.
Type: HTMLElement
HTMLElement. The parent of the element. If supplied, the returned bounds will be relative to the parent element bounds.

*includeScroll*
Optional.
Type: Boolean
Boolean. For internal use.

### Return Value

Type: Rect
Rect. The bounding rectangle.

## See Also

DomUtils Members
DomUtils Class
MindFusion.Controls Namespace

# DomUtils.getElementAttributes Method

Gets an object containing element attributes that have values.

**Namespace:** MindFusion.Controls
**File:** DomUtils.js

## Syntax

**JavaScript**

```
function getElementAttributes ()
```

## See Also

DomUtils Members
DomUtils Class
MindFusion.Controls Namespace

# DomUtils.HTMLtoXHTML Method

Converts HTML to XHMTL.

**Namespace:** MindFusion.Controls
**File:** DomUtils.js

## Syntax

**JavaScript**

```
function HTMLtoXHTML (html)
```

### Parameters

*html*

Type: String
String. The HTML to convert.

### Return Value

Type: String
String. The resulting XHTML.

## See Also

DomUtils Members
DomUtils Class
MindFusion.Controls Namespace

# DomUtils.loadImageList Method

Loads the images from the specified list of image url-s.

**Namespace:** MindFusion.Controls
**File:** DomUtils.js

## Syntax

**JavaScript**

```
function loadImageList (images)
```

### Parameters

*images*
Type: Array
Array. An array of image url-s.

### Return Value

Type: Promise
Promise. Promise object contaning an array of images in base64 format.

## See Also

DomUtils Members
DomUtils Class
MindFusion.Controls Namespace

# DomUtils.setBounds Method

Sets the bounds of the specified element.

**Namespace:** MindFusion.Controls
**File:** DomUtils.js

## Syntax

**JavaScript**

```
function setBounds (element, bounds)
```

### Parameters

*element*
Type: HTMLElement
HTMLElement. The element.

*bounds*
Type: Rect
Rect. The bounding rectangle.

## See Also

DomUtils Members
DomUtils Class
MindFusion.Controls Namespace

# DomUtils.setSize Method

Sets the size of the specified element.

**Namespace:** MindFusion.Controls
**File:** DomUtils.js

## Syntax

**JavaScript**

```
function setSize (element, size)
```

### Parameters

*element*
Type: HTMLElement
HTMLElement. The element.

*size*
Type: Size
Size. The new size.

## See Also

DomUtils Members
DomUtils Class
MindFusion.Controls Namespace

# DomUtils.toDataUrl Method

Converts an image to base64 string.

**Namespace:** MindFusion.Controls
**File:** DomUtils.js

## Syntax

**JavaScript**

```
function toDataUrl (url, callback)
```

### Parameters

*url*
Type: String
String. The url of the image.

*callback*
Type: function
function. The callback function.

### Return Value

Type: String
String. The base64-encoded string, representing the image.

## See Also

DomUtils Members
DomUtils Class
MindFusion.Controls Namespace

# EventArgs Class

The base type of classes that define arguments passed to event handler functions.

## Syntax

**JavaScript**

```
// class
EventArgs.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Controls.EventArgs**
MindFusion.Controls.CancelEventArgs

## See Also

EventArgs Members
MindFusion.Controls Namespace

# EventArgs Members

The base type of classes that define arguments passed to event handler functions.

The following tables list the members exposed by the EventArgs type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | EventArgs | Initializes a new instance of the EventArgs class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | handled | Gets or sets a value indicating whether the event has been handled. |

## See Also

EventArgs Class
MindFusion.Controls Namespace

# EventArgs Constructor

Initializes a new instance of the EventArgs class.

**Namespace:** MindFusion.Controls
**File:** EventArgs.js

## Syntax

**JavaScript**

```
function EventArgs ()
```

## See Also

EventArgs Class
MindFusion.Controls Namespace

# EventArgs Properties

# EventArgs.handled Property

Gets or sets a value indicating whether the event has been handled.

**Namespace:** MindFusion.Controls
**File:** EventArgs.js

## Syntax

**JavaScript**

```
get handled() {}
set handled(value) {}
```

### Property Value

Type: Boolean
*Boolean*

true to indicate that the event has been handled, or false otherwise.

## See Also

EventArgs Members
EventArgs Class
MindFusion.Controls Namespace

# MouseButton Enumeration

Specifies which button was pressed to trigger the event.

**Namespace:** MindFusion.Controls
**File:** Enum.js

## Syntax

**JavaScript**

```
// enum
MouseButton = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **Auxiliary** | Auxiliary button pressed, usually the wheel button or the middle button. |
| | **Fifth** | Fifth button, typically the Browser Forward button. |
| | **Fourth** | Fourth button, typically the Browser Back button. |
| | **Main** | Main button pressed, usually the left button or the un-initialized state. |
| | **Secondary** | Secondary button pressed, usually the right button. |

## See Also

MindFusion.Controls Namespace

# MouseCursors Enumeration

Specifies the type of the mouse cursor.

**Namespace:** MindFusion.Controls
**File:** Enum.js

## Syntax

**JavaScript**

```
// enum
MouseCursors = {}
```

## See Also

MindFusion.Controls Namespace

# TickPosition Enumeration

Specifies the position of trackbar ticks.

**Namespace:** MindFusion.Controls
**File:** Enum.js

## Syntax

**JavaScript**

```
// enum
TickPosition = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **Both** | Ticks are rendered on both sides of the trackbar. |
| | **Left** | Ticks are rendered on the left side of the trackbar. |
| | **None** | There are no ticks rendered. |
| | **Right** | Ticks are rendered on the right side of the trackbar. |

## Remarks

Members of this enumeration can be assigned to the tickPosition property of ZoomControl.

## See Also

MindFusion.Controls Namespace

# ZoomControl Class

The **ZoomControl** lets users zoom and pan a target view control interactively.

## Syntax

**JavaScript**

```
// class
ZoomControl.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Dom.Control**
  **MindFusion.Controls.ZoomControl**

## See Also

ZoomControl Members
MindFusion.Controls Namespace

# ZoomControl Members

The ZoomControl lets users zoom and pan a target view control interactively.

The following tables list the members exposed by the **ZoomControl** type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | ZoomControl | Creates an instance of **ZoomControl**. |

## Public Properties

| | Name | Description |
|---|---|---|
| | activeColor | Gets or sets the color used to render depressed buttons. |
| | autoPostBack | Gets or sets a value indicating whether the control will post back to the server when the control's value has changed. |
| | backColor | Gets or sets the background color of the control. |
| | borderColor | Gets or sets the color of **ZoomControl** elements' borders. |
| | cornerRadius | Gets or sets the corner radius of rounded child elements. |
| | element | Returns a reference to the control's DOM element. (Inherited from Control.) |
| | enabled | Gets or sets whether mouse events are enabled. |
| | fill | Gets or sets the color used to fill the **ZoomControl** elements. |
| | innerColor | Gets or sets the color of plus, minus and arrow icons. |
| | maxZoomFactor | Gets or sets the maximum zoom level allowed to set through this control. |
| | measureUnit | Gets the unit of measure used for logical coordinates. (Inherited from CanvasControl.) |
| | minZoomFactor | Gets or sets the minimum zoom level allowed to set through this control. |
| | padding | Gets or sets the padding of the control's contents. |
| | scrollStep | Gets or sets the scroll offset added when users click the pan arrows. |
| | shadowColor | Gets or sets the color of the control elements' shadow. |
| | showLabel | Gets or sets a value indicating whether the label that shows the current zoom level should be visible. |
| | snapToZoomStep | Gets or sets a value indicating whether the trackbar should snap to zoomStep values when dragged. |
| | target | Gets or sets the control modified by this **ZoomControl**. |
| | textColor | Gets or sets the color of the the label that shows the current zoom level. |

| | | |
|---|---|---|
| | tickPosition | Gets or sets the current tick position of the trackbar. |
| | zoomFactor | Gets or sets the zoom factor. |
| | zoomStep | Gets or sets the amount by which to change zoom level when + and - buttons are clicked. |

## Public Methods

| | Name | Description |
|---|---|---|
| | addEventListener | Registers a single event listener on the instance. (Inherited from Disposable.) |
| S | addHandlers | Adds an event listener to a DOM element. (Inherited from Control.) |
| S | clearHandlers | Removes all event listeners from a DOM element. (Inherited from Control.) |
| | clientToDoc | Transforms a point from client to document coordinates. (Inherited from CanvasControl.) |
| S | create | Creates and initializes a new **ZoomControl** from the specified element. This method is static and can be called without creating an instance of the class. |
| | docToClient | Transforms a point from document to client coordinates. (Inherited from CanvasControl.) |
| S | find | Returns the specified **ZoomControl** object. This member is static and can be invoked without creating an instance of the class. |
| S | fromJson | Constructs a JavaScript object from a JSON string. (Inherited from Control.) |
| | raiseEvent | Raises an event. (Inherited from Disposable.) |
| | registerForSubmit | Registers an onsubmit handler for the Control's parent html form to flush postback data. (Inherited from Control.) |
| | removeEventListener | Removes a single event listener attached to the instance. (Inherited from Disposable.) |
| S | toJson | Converts a JavaScript object to a JSON string. (Inherited from Control.) |

## See Also

ZoomControl Class
MindFusion.Controls Namespace

### ZoomControl Constructor

Creates an instance of ZoomControl.

**Namespace:** MindFusion.Controls
**File:** ZoomControl.js

## Syntax

```
JavaScript
function ZoomControl (element)
```

### Parameters

*element*
HTMLCanvasElement. The Canvas DOM Element this ZoomControl is associated with.

## See Also

ZoomControl Class
MindFusion.Controls Namespace

# ZoomControl Methods

## ZoomControl.create Method

Creates and initializes a new ZoomControl from the specified element. This method is static and can be called without creating an instance of the class.

**Namespace:** MindFusion.Controls
**File:** ZoomControl.js

## Syntax

```
JavaScript
function create (element)
```

### Parameters

*element*
Type: HTMLCanvasElement
HTMLCanvasElement. The DOM element that the zoomControl should be attached to.

### Return Value

A ZoomControl object that represents the newly created zoom control.

## See Also

ZoomControl Members
ZoomControl Class
MindFusion.Controls Namespace

## ZoomControl.find Method

Returns the specified ZoomControl object. This member is static and can be invoked without creating an instance of the class.

**Namespace:** MindFusion.Controls
**File:** ZoomControl.js

## Syntax

```
JavaScript
function find (id)
```

### Parameters

*id*
String. The id of the ZoomControl's DOM element.

### Return Value

A **ZoomControl** object with the specified id, if found, or **null** otherwise.

## See Also

ZoomControl Members
ZoomControl Class
MindFusion.Controls Namespace

# ZoomControl Properties

# ZoomControl.activeColor Property

Gets or sets the color used to render depressed buttons.

**Namespace:** MindFusion.Controls
**File:** ZoomControl.js

## Syntax

```
JavaScript
get activeColor() {}
set activeColor(value) {}
```

### Property Value

*String*

The color used to render depressed buttons.

## See Also

ZoomControl Members
ZoomControl Class
MindFusion.Controls Namespace

# ZoomControl.autoPostBack Property

Gets or sets a value indicating whether the control will post back to the server when the control's value has changed.

**Namespace:** MindFusion.Controls
**File:** ZoomControl.js

## Syntax

```
JavaScript
get autoPostBack() {}
set autoPostBack(value) {}
```

### Property Value

*Boolean*

**true** if the control should post back, or **false** otherwise.

## Remarks

Used to implement auto-post back mode in MindFusion ASP.NET controls.

## See Also

ZoomControl Members
ZoomControl Class
MindFusion.Controls Namespace

# ZoomControl.backColor Property

Gets or sets the background color of the control.

**Namespace:** MindFusion.Controls
**File:** ZoomControl.js

## Syntax

```
JavaScript
get backColor() {}
set backColor(value) {}
```

### Property Value

*String*

The color of the control background.

## See Also

ZoomControl Members
ZoomControl Class
MindFusion.Controls Namespace

# ZoomControl.borderColor Property

Gets or sets the color of ZoomControl elements' borders.

**Namespace:** MindFusion.Controls
**File:** ZoomControl.js

## Syntax

JavaScript

```
get borderColor() {}
set borderColor(value) {}
```

### Property Value

*String*

The color of ZoomControl elements' borders.

## See Also

ZoomControl Members
ZoomControl Class
MindFusion.Controls Namespace

# ZoomControl.cornerRadius Property

Gets or sets the corner radius of rounded child elements.

**Namespace:** MindFusion.Controls
**File:** ZoomControl.js

## Syntax

JavaScript

```
get cornerRadius() {}
set cornerRadius(value) {}
```

### Property Value

*Number*

The corner radius of rounded child elements.

## See Also

ZoomControl Members
ZoomControl Class
MindFusion.Controls Namespace

# ZoomControl.enabled Property

Gets or sets whether mouse events are enabled.

**Namespace:** MindFusion.Controls
**File:** ZoomControl.js

## Syntax

JavaScript

```
get enabled() {}
set enabled(value) {}
```

### Property Value

*Boolean*

**true** to respond to mouse events, or **false** otherwise.

## See Also

ZoomControl Members
ZoomControl Class
MindFusion.Controls Namespace

# ZoomControl.fill Property

Gets or sets the color used to fill the ZoomControl elements.

**Namespace:** MindFusion.Controls
**File:** ZoomControl.js

## Syntax

**JavaScript**

```
get fill() {}
set fill(value) {}
```

### Property Value

*String*

The color used to fill the ZoomControl elements.

## See Also

ZoomControl Members
ZoomControl Class
MindFusion.Controls Namespace

# ZoomControl.innerColor Property

Gets or sets the color of plus, minus and arrow icons.

**Namespace:** MindFusion.Controls
**File:** ZoomControl.js

## Syntax

**JavaScript**

```
get innerColor() {}
set innerColor(value) {}
```

### Property Value

*String*

The color of plus, minus and arrow icons.

## See Also

ZoomControl Members
ZoomControl Class
MindFusion.Controls Namespace

# ZoomControl.maxZoomFactor Property

Gets or sets the maximum zoom level allowed to set through this control.

**Namespace:** MindFusion.Controls
**File:** ZoomControl.js

## Syntax

**JavaScript**

```
get maxZoomFactor() {}
set maxZoomFactor(value) {}
```

### Property Value

*Number*

The maximum zoom level, specified as percentage. The default value is 200.

## See Also

ZoomControl Members
ZoomControl Class
MindFusion.Controls Namespace

# ZoomControl.minZoomFactor Property

Gets or sets the minimum zoom level allowed to set through this control.

**Namespace:** MindFusion.Controls
**File:** ZoomControl.js

## Syntax

**JavaScript**

```
get minZoomFactor() {}
set minZoomFactor(value) {}
```

### Property Value

*Number*

The minimum zoom level.

## See Also

ZoomControl Members
ZoomControl Class
MindFusion.Controls Namespace

# ZoomControl.padding Property

Gets or sets the padding of the control's contents.

**Namespace:** MindFusion.Controls
**File:** ZoomControl.js

## Syntax

```
JavaScript
get padding() {}
set padding(value) {}
```

### Property Value

*Number*

The padding of the control's contents.

## See Also

ZoomControl Members
ZoomControl Class
MindFusion.Controls Namespace

# ZoomControl.scrollStep Property

Gets or sets the scroll offset added when users click the pan arrows.

**Namespace:** MindFusion.Controls
**File:** ZoomControl.js

## Syntax

```
JavaScript
get scrollStep() {}
set scrollStep(value) {}
```

### Property Value

*Number*

The scroll step.

## See Also

ZoomControl Members
ZoomControl Class
MindFusion.Controls Namespace

# ZoomControl.shadowColor Property

Gets or sets the color of the control elements' shadow.

**Namespace:** MindFusion.Controls
**File:** ZoomControl.js

## Syntax

```
JavaScript
get shadowColor() {}
set shadowColor(value) {}
```

### Property Value

*String*

The color of the control elements' shadow.

## See Also

ZoomControl Members
ZoomControl Class
MindFusion.Controls Namespace

# ZoomControl.showLabel Property

Gets or sets a value indicating whether the label that shows the current zoom level should be visible.

**Namespace:** MindFusion.Controls
**File:** ZoomControl.js

## Syntax

```
JavaScript
get showLabel() {}
set showLabel(value) {}
```

### Property Value

*Boolean*

**true** if the label is visible, or **false** otherwise.

## See Also

ZoomControl Members
ZoomControl Class
MindFusion.Controls Namespace

# ZoomControl.snapToZoomStep Property

Gets or sets a value indicating whether the trackbar should snap to zoomStep values when dragged.

**Namespace:** MindFusion.Controls
**File:** ZoomControl.js

## Syntax

```
JavaScript
get snapToZoomStep() {}
set snapToZoomStep(value) {}
```

### Property Value

*Boolean*

**true** if snapping is enabled, or **false** otherwise.

## See Also

ZoomControl Members
ZoomControl Class
MindFusion.Controls Namespace

# ZoomControl.target Property

Gets or sets the control modified by this ZoomControl.

**Namespace:** MindFusion.Controls
**File:** ZoomControl.js

## Syntax

**JavaScript**

```
get target() {}
set target(value) {}
```

### Property Value

*Control*

The control modified by this ZoomControl.

## See Also

ZoomControl Members
ZoomControl Class
MindFusion.Controls Namespace

# ZoomControl.textColor Property

Gets or sets the color of the the label that shows the current zoom level.

**Namespace:** MindFusion.Controls
**File:** ZoomControl.js

## Syntax

**JavaScript**

```
get textColor() {}
set textColor(value) {}
```

### Property Value

*String*

The color of the the label that shows the current zoom level.

## See Also

ZoomControl Members
ZoomControl Class
MindFusion.Controls Namespace

# ZoomControl.tickPosition Property

Gets or sets the current tick position of the trackbar.

**Namespace:** MindFusion.Controls
**File:** ZoomControl.js

## Syntax

**JavaScript**

```
get tickPosition() {}
set tickPosition(value) {}
```

### Property Value

*TickPosition*

A member of the TickPosition enumeration.

## See Also

ZoomControl Members
ZoomControl Class
MindFusion.Controls Namespace

# ZoomControl.zoomFactor Property

Gets or sets the zoom factor.

**Namespace:** MindFusion.Controls
**File:** ZoomControl.js

## Syntax

**JavaScript**

```
get zoomFactor() {}
set zoomFactor(value) {}
```

### Property Value

*Number*

The current zoom factor, specified as percentage.

## See Also

ZoomControl Members
ZoomControl Class
MindFusion.Controls Namespace

## ZoomControl.zoomStep Property

Gets or sets the amount by which to change zoom level when + and - buttons are clicked.

**Namespace:** MindFusion.Controls
**File:** ZoomControl.js

### Syntax

**JavaScript**

```
get zoomStep() {}
set zoomStep(value) {}
```

#### Property Value

*Number*

The zoom step. The default value is 10.

### See Also

ZoomControl Members
ZoomControl Class
MindFusion.Controls Namespace

# MindFusion.Drawing Namespace

## Border3D Class

Represents a 3D border.

### Syntax

**JavaScript**

```
// class
Border3D.prototype = {}
```

### Inheritance Hierarchy

**MindFusion.Drawing.Border3D**

### See Also

Border3D Members
MindFusion.Drawing Namespace

## Border3D Members

Represents a 3D border.

The following tables list the members exposed by the Border3D type.

### Public Constructors

| | Name | Description |
|---|---|---|
|  | Border3D | Initializes a new instance of the Border3D class. |

## See Also

Border3D Class
MindFusion.Drawing Namespace

# Border3D Constructor

Initializes a new instance of the Border3D class.

**Namespace:** MindFusion.Drawing
**File:** Border3D.js

## Syntax

**JavaScript**

```
function Border3D (rect)
```

### Parameters

*rect*
Type: Rect
Rect. A Rect instance containing the border coordinates.

## See Also

Border3D Class
MindFusion.Drawing Namespace

# Brush Class

Represents a brush.

## Syntax

**JavaScript**

```
// class
Brush.prototype = {}
```

## Inheritance Hierarchy

## See Also

Brush Members
MindFusion.Drawing Namespace

# Brush Members

Represents a brush.

The following tables list the members exposed by the Brush type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | Brush | Overloaded. |

## Public Properties

| | Name | Description |
|---|---|---|
| | color | Gets or sets the color of this Brush object. |

## Public Methods

| | Name | Description |
|---|---|---|
| | apply | Applies the brush properties to a CanvasRenderingContext2D drawing object. |
| | clone | Creates a copy of this brush. |
| | equals | Determines whether the specified object is equal to this Brush. |
| S | fromObject | Returns a Brush from a JSON object or string describing this brush. |
| | toObject | Returns a JSON object describing this brush. |

## See Also

Brush Class
MindFusion.Drawing Namespace

# Brush Constructor

## Overload List

| Name | Description |
|---|---|
| Brush | Initializes a new instance of the Brush class. |
| Brush | Initializes a new instance of the Brush class. |

## See Also

Brush Class
MindFusion.Drawing Namespace

# Brush Constructor

Initializes a new instance of the Brush class.

**Namespace:** MindFusion.Drawing
**File:** Brush.js

## Syntax

**JavaScript**

```
function Brush (color)
```

### Parameters

*color*
Type: Color | string
The color of this brush.

## Example

The following code creates a new Brush and assigns it as a background of a ToolTip:

**JavaScript**

```
ToolTip.brush = new Drawing.Brush("#fafafa");
```

## See Also

Brush Constructor Overload List
Brush Class
MindFusion.Drawing Namespace

# Brush Constructor

Initializes a new instance of the Brush class.

**Namespace:** MindFusion.Drawing
**File:** Brush.js

## Syntax

**JavaScript**

```
function Brush (color)
```

### Parameters

*color*
Type: Color, string
The color of this brush.

## Example

The following code creates a new Brush and assigns it as a background of a ToolTip. The Brush is semi-transparent gray:

**JavaScript**

```
ToolTip.brush = new Drawing.Brush(Drawing.Color.fromArgb(125, 100, 100, 100));
```

## See Also

Brush Constructor Overload List
Brush Class
MindFusion.Drawing Namespace

# Brush Methods

# Brush.apply Method

Applies the brush properties to a CanvasRenderingContext2D drawing object.

**Namespace:** MindFusion.Drawing
**File:** Brush.js

## Syntax

**JavaScript**

```
function apply (context)
```

### Parameters

*context*
Type: CanvasRenderingContext2D
The CanvasRenderingContext2D drawing object.

## See Also

Brush Members
Brush Class
MindFusion.Drawing Namespace

# Brush.clone Method

Creates a copy of this brush.

**Namespace:** MindFusion.Drawing
**File:** Brush.js

## Syntax

**JavaScript**

```
function clone ()
```

### Return Value

Type: Brush
The copy of this brush.

## See Also

Brush Members
Brush Class
MindFusion.Drawing Namespace

# Brush.equals Method

Determines whether the specified object is equal to this Brush.

**Namespace:** MindFusion.Drawing
**File:** Brush.js

## Syntax

**JavaScript**

```
function equals (The)
```

### Parameters

*The*
Type: Object
object to compare with.

### Return Value

Type: Boolean
true if the specified object is equal to this Brush; otherwise, false.

## See Also

Brush Members
Brush Class
MindFusion.Drawing Namespace

# Brush.fromObject Method

Returns a Brush from a JSON object or string describing this brush.

**Namespace:** MindFusion.Drawing
**File:** Brush.js

## Syntax

**JavaScript**

```
function fromObject (obj)
```

### Parameters

*obj*
Type: void

### Return Value

Type: Brush
The brush object.

## Remarks

Used for backward compatibility with the JSLib class.

## See Also

Brush Members
Brush Class
MindFusion.Drawing Namespace

# Brush.toObject Method

Returns a JSON object describing this brush.

**Namespace:** MindFusion.Drawing
**File:** Brush.js

## Syntax

**JavaScript**

```
function toObject ()
```

### Return Value

Type: Object
The object describing this brush.

## Remarks

Used for backward compatibility with the JSLib class.

## See Also

Brush Members
Brush Class
MindFusion.Drawing Namespace

# Brush Properties

# Brush.color Property

Gets or sets the color of this Brush object.

**Namespace:** MindFusion.Drawing
**File:** Brush.js

## Syntax

**JavaScript**

```
get color() {}
```

### Property Value

Type: Color

## See Also

Brush Members
Brush Class
MindFusion.Drawing Namespace

# Color Class

Represents a color.

## Syntax

| JavaScript |
| --- |
| `// class`<br>`Color.prototype = {}` |

## Inheritance Hierarchy

## See Also

Color Members
MindFusion.Drawing Namespace

# Color Members

Represents a color.

The following tables list the members exposed by the Color type.

## Public Constructors

| | Name | Description |
| --- | --- | --- |
| | Color | Initializes a new instance of the Color class. |

## Public Fields

| | Name | Description |
| --- | --- | --- |
| | knownColors | The list of color names as defined by the CSS specification. |

## Public Methods

| | Name | Description |
| --- | --- | --- |
| | fromArgb | Creates a Color instance from alpha, red, green and blue values. |

## See Also

Color Class
MindFusion.Drawing Namespace

# Color Fields

## Color.knownColors Field

The list of color names as defined by the CSS specification.

**Namespace:** MindFusion.Drawing
**File:** Color.js

### Syntax

```
JavaScript
var knownColors
```

### See Also

Color Members
Color Class
MindFusion.Drawing Namespace

# Color Constructor

Initializes a new instance of the Color class.

**Namespace:** MindFusion.Drawing
**File:** Color.js

### Syntax

```
JavaScript
function Color (value)
```

#### Parameters

*value*
Type: String
A string representation of the color.

### See Also

Color Class
MindFusion.Drawing Namespace

# Color Methods

## Color.fromArgb Method

Creates a Color instance from alpha, red, green and blue values.

**Namespace:** MindFusion.Drawing
**File:** Color.js

## Syntax

```
JavaScript
function fromArgb (alpha, red, green, blue)
```

### Parameters

*alpha*
Type: Number
The alpha value.

*red*
Type: Number
The red value.

*green*
Type: Number
The green value.

*blue*
Type: Number
The blue value.

### Return Value

Type: Color
The new Color instance.

## See Also

Color Members
Color Class
MindFusion.Drawing Namespace

# Color.knownColors Class

The list of color names as defined by the CSS specification.

## Syntax

```
JavaScript
// class
Color.knownColors.prototype = {}
```

## Inheritance Hierarchy

## See Also

Color.knownColors Members
MindFusion.Drawing Namespace

## Color.knownColors Members

The list of color names as defined by the CSS specification.

The following tables list the members exposed by the Color.knownColors type.

## See Also

Color.knownColors Class
MindFusion.Drawing Namespace

# Component Class

Represents components in **CompositeNode** visual tree.

## Syntax

| JavaScript |
| --- |
| `// class`<br>`Component.prototype = {}` |

## Inheritance Hierarchy

MindFusion.Drawing.ComponentBase
**MindFusion.Drawing.Component**
MindFusion.Drawing.Container

## See Also

Component Members
MindFusion.Drawing Namespace

## Component Members

Represents components in **CompositeNode** visual tree.

The following tables list the members exposed by the Component type.

## Public Constructors

| | Name | Description |
| --- | --- | --- |
|  | Component | Initializes a new instance of the **Component** class. |

## Public Properties

| | Name | Description |
| --- | --- | --- |
|  | visibility | Gets or sets the visibility of this component. (Inherited from ComponentBase.) |

## See Also

Component Class
MindFusion.Drawing Namespace

## Component Constructor

Initializes a new instance of the Component class.

**Namespace:** MindFusion.Drawing
**File:** Component.js

### Syntax

```
JavaScript
function Component ()
```

### See Also

Component Class
MindFusion.Drawing Namespace

# ComponentBase Class

A base class for components in **CompositeNode** visual tree.

### Syntax

```
JavaScript
// class
ComponentBase.prototype = {}
```

### Inheritance Hierarchy

**MindFusion.Drawing.ComponentBase**
MindFusion.Drawing.Component
MindFusion.Drawing.Ellipse
MindFusion.Drawing.Image
MindFusion.Drawing.Path
MindFusion.Drawing.Rect
MindFusion.Drawing.Text
MindFusion.Drawing.Video

### See Also

ComponentBase Members
MindFusion.Drawing Namespace

## ComponentBase Members

A base class for components in **CompositeNode** visual tree.

The following tables list the members exposed by the ComponentBase type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | ComponentBase | Initializes a new instance of the **ComponentBase** class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | visibility | Gets or sets the visibility of this component. |

## See Also

ComponentBase Class
MindFusion.Drawing Namespace

### ComponentBase Constructor

Initializes a new instance of the ComponentBase class.

**Namespace:** MindFusion.Drawing
**File:** ComponentBase.js

## Syntax

**JavaScript**

```
function ComponentBase ()
```

## See Also

ComponentBase Class
MindFusion.Drawing Namespace

### ComponentBase Methods

### ComponentBase Properties

### ComponentBase.visibility Property

Gets or sets the visibility of this component.

**Namespace:** MindFusion.Drawing
**File:** ComponentBase.js

## Syntax

**JavaScript**

```
get visibility() {}
set visibility(value) {}
```

### Property Value

*Visibility*

A member of the Visibility enumeration.

## See Also

ComponentBase Members
ComponentBase Class
MindFusion.Drawing Namespace

# Container Class

Represents container components in drawing tree.

## Syntax

**JavaScript**

```
// class
Container.prototype = {}
```

## Inheritance Hierarchy

MindFusion.Drawing.ComponentBase
  MindFusion.Drawing.Component
    **MindFusion.Drawing.Container**

## See Also

Container Members
MindFusion.Drawing Namespace

# Container Members

Represents container components in drawing tree.

The following tables list the members exposed by the Container type.

## Public Properties

| | Name | Description |
|---|---|---|
|  | visibility | Gets or sets the visibility of this component. (Inherited from ComponentBase.) |

## See Also

Container Class
MindFusion.Drawing Namespace

# DashStyle Enumeration

Specifies the dash pattern of lines.

**Namespace:** MindFusion.Drawing
**File:** Enum.js

## Syntax

**JavaScript**

```
// enum
DashStyle = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **Custom** | Custom pattern. |
| | **Dash** | Draw dashed line. |
| | **DashDot** | Draw dash-dot pattern. |
| | **DashDotDot** | Draw dash-dot-dot pattern. |
| | **Dot** | Draw dotted line. |
| | **Solid** | Draw solid line. |

## See Also

MindFusion.Drawing Namespace

# DrawingUtils Class

Contains drawing-related helper functions.

## Syntax

**JavaScript**

```
// class
DrawingUtils.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Drawing.DrawingUtils**

## See Also

DrawingUtils Members
MindFusion.Drawing Namespace

# DrawingUtils Members

Contains drawing-related helper functions.

The following tables list the members exposed by the DrawingUtils type.

## Public Methods

| | Name | Description |
|---|---|---|
| | approximateBezier | Approximates a Bezier curve with the given quality and by a series of line segments. |
| | checkIntersect | Checks if the specified rectangle and circle intersect. |
| | degrees | Converts the specified value from radians to degrees. |
| | distToPolyline | Calculates the shortest distance from the specified point to the specified polyline, also returning the index of the segment the point is closest to. |
| | distToRectPoint | Returns the point that lies on the outline of a rectangle and is nearest to the given point. |
| | getCenter | Returns the center of the specified rectangle. |
| | getPointsBounds | Calculates the bounding rect of the specified polyline. |
| | getSegmentIntersection | Checks whether the segments defined by the specified point pairs intersect and returns the intersection point. |
| | inflate | Creates and returns an enlarged copy of a Rect instance. |
| | intersect | Determines whether the specified line segments intersect. |
| | minDistToRect | Calculates the minimum distance between a given point and a given rectangle. |
| | pointInCircle | Checks whether the specified circle contains the specified point. |
| | pointInEllipse | Checks whether the specified ellipse contains the specified point. |
| | pointInPolygon | Checks whether the specified rectangle contains the specified point. |
| | radians | Converts the specified value from degrees to radians. |
| | unionRects | Returns the smallest possible rectangle containing both of the specified rectangles. |

## See Also

DrawingUtils Class
MindFusion.Drawing Namespace

# DrawingUtils Methods

# DrawingUtils.approximateBezier Method

Approximates a Bezier curve with the given quality and by a series of line segments.

**Namespace:** MindFusion.Drawing
**File:** DrawingUtils.js

## Syntax

**JavaScript**

```
function approximateBezier (points, quality, start)
```

### Parameters

*points*
Type: Array
Array. A list of Bezier control points.

*quality*
Type: Number
Number. Specifies the approximation quality.

*start*
Type: Number
Number. Specifies the start point index.

### Return Value

Type: Array
Array. A list of points that approximate the curve as a series of straight line segments.

## See Also

DrawingUtils Members
DrawingUtils Class
MindFusion.Drawing Namespace

# DrawingUtils.checkIntersect Method

Checks if the specified rectangle and circle intersect.

**Namespace:** MindFusion.Drawing
**File:** DrawingUtils.js

## Syntax

**JavaScript**

```
function checkIntersect (point, rect, rad)
```

### Parameters

*point*
Type: Point
Point. The center of the circle.

*rect*
Type: Rect
Rect. The rectangle.

*rad*
Type: Number
Number. The radius of the circle.

### Return Value

Type: Boolean
Boolean. true if the rectangle and the circle intersect; otherwise, false.

## See Also

DrawingUtils Members
DrawingUtils Class
MindFusion.Drawing Namespace

# DrawingUtils.degrees Method

Converts the specified value from radians to degrees.

**Namespace:** MindFusion.Drawing
**File:** DrawingUtils.js

## Syntax

**JavaScript**

```
function degrees (radians)
```

### Parameters

*radians*
Type: Number
Number. The radians to convert.

### Return Value

Type: Number
Number. The converted angle.

## See Also

DrawingUtils Members
DrawingUtils Class
MindFusion.Drawing Namespace

# DrawingUtils.distToPolyline Method

Calculates the shortest distance from the specified point to the specified polyline, also returning the index of the segment the point is closest to.

**Namespace:** MindFusion.Drawing
**File:** DrawingUtils.js

## Syntax

**JavaScript**

```
function distToPolyline ()
```

## See Also

DrawingUtils Members
DrawingUtils Class
MindFusion.Drawing Namespace

# DrawingUtils.distToRectPoint Method

Returns the point that lies on the outline of a rectangle and is nearest to the given point.

**Namespace:** MindFusion.Drawing
**File:** DrawingUtils.js

## Syntax

**JavaScript**

```
function distToRectPoint (point, rect)
```

### Parameters

*point*
Type: Point
Point. The point to consider.

*rect*
Type: Rect
Rect. The rectangle to consider.

### Return Value

Type: Number
Number. The nearest point.

## See Also

DrawingUtils Members
DrawingUtils Class
MindFusion.Drawing Namespace

# DrawingUtils.getCenter Method

Returns the center of the specified rectangle.

**Namespace:** MindFusion.Drawing
**File:** DrawingUtils.js

## Syntax

**JavaScript**

```
function getCenter (rect)
```

### Parameters

*rect*
Type: Rect
Rect. The rectangle.

### Return Value

Type: Point
Point. The center.

## See Also

DrawingUtils Members
DrawingUtils Class
MindFusion.Drawing Namespace

# DrawingUtils.getPointsBounds Method

Calculates the bounding rect of the specified polyline.

**Namespace:** MindFusion.Drawing
**File:** DrawingUtils.js

## Syntax

**JavaScript**

```
function getPointsBounds (points)
```

### Parameters

*points*
Type: Array
Array. An array of points.

### Return Value

Type: Rect
Rect. The bounding rect.

## See Also

DrawingUtils Members
DrawingUtils Class
MindFusion.Drawing Namespace

# DrawingUtils.getSegmentIntersection Method

Checks whether the segments defined by the specified point pairs intersect and returns the intersection point.

**Namespace:** MindFusion.Drawing
**File:** DrawingUtils.js

## Syntax

**JavaScript**

```
function getSegmentIntersection (p1, p2, p3, p4)
```

### Parameters

*p1*
Type: Point
Point. The start of the first segment.

*p2*
Type: Point
Point. The end of the first segment.

*p3*
Type: Point
Point. The start of the second segment.

*p4*
Type: Point
Point. The end of the second segment.

### Return Value

Type: Boolean
Boolean. true if the segments intersect, false if not.

## See Also

DrawingUtils Members
DrawingUtils Class
MindFusion.Drawing Namespace

# DrawingUtils.inflate Method

Creates and returns an enlarged copy of a Rect instance.

**Namespace:** MindFusion.Drawing
**File:** DrawingUtils.js

## Syntax

**JavaScript**

```
function inflate (rect, x, y)
```

### Parameters

*rect*
Type: Rect
Rect. The Rect instance to inflate.

*x*
Type: Number
Number. The amount to inflate horizontally.

*y*
Type: Number
Number. The amount to inflate vertically.

### Return Value

Type: Rect
Rect. The inflated Rect.

## See Also

DrawingUtils Members
DrawingUtils Class
MindFusion.Drawing Namespace

# DrawingUtils.intersect Method

Determines whether the specified line segments intersect.

**Namespace:** MindFusion.Drawing
**File:** DrawingUtils.js

## Syntax

**JavaScript**

```
function intersect (p1, p2, p3, p4)
```

### Parameters

*p1*
Type: Point
Point. The start of the first line segment.

*p2*
Type: Point
Point. The end of the first line segment.

*p3*
Type: Point
Point. The start of the second line segment.

*p4*
Type: Point
Point. The end of the second line segment.

### Return Value

Type: Boolean
Boolean. true if the segments intersect, false if not.

## See Also

DrawingUtils Members
DrawingUtils Class
MindFusion.Drawing Namespace

# DrawingUtils.minDistToRect Method

Calculates the minimum distance between a given point and a given rectangle.

**Namespace:** MindFusion.Drawing
**File:** DrawingUtils.js

## Syntax

JavaScript

```
function minDistToRect (point, rect)
```

### Parameters

*point*
Type: Point
Point. The point.

*rect*
Type: Rectabgle
Rectabgle. The rectangle.

### Return Value

Type: Number
Number. The minimum distance.

## See Also

DrawingUtils Members
DrawingUtils Class
MindFusion.Drawing Namespace

# DrawingUtils.pointInCircle Method

Checks whether the specified circle contains the specified point.

**Namespace:** MindFusion.Drawing
**File:** DrawingUtils.js

## Syntax

JavaScript

```
function pointInCircle (point, center, radius)
```

### Parameters

*point*
Type: Point
Point. The point to check.

*center*
Type: Point
Point. The center of the circle.

*radius*
Type: Number
Number. The radius of the circle.

### Return Value

Type: Boolean
Boolean. true if the circle contains the specified point, false if not.

## See Also

DrawingUtils Members
DrawingUtils Class
MindFusion.Drawing Namespace

# DrawingUtils.pointInEllipse Method

Checks whether the specified ellipse contains the specified point.

**Namespace:** MindFusion.Drawing
**File:** DrawingUtils.js

## Syntax

**JavaScript**

```
function pointInEllipse (point, rect)
```

### Parameters

*point*
Type: Point
Point. The point to check.

*rect*
Type: Rect
Rect. The bounds of the ellipse.

### Return Value

Type: Boolean
Boolean. true if the ellipse contains the specified point, false if not.

## See Also

DrawingUtils Members
DrawingUtils Class
MindFusion.Drawing Namespace

# DrawingUtils.pointInPolygon Method

Checks whether the specified rectangle contains the specified point.

**Namespace:** MindFusion.Drawing
**File:** DrawingUtils.js

## Syntax

**JavaScript**

```
function pointInPolygon (point, polygon)
```

### Parameters

*point*
Type: Point
Point. The point to check.

*polygon*

Type: Array
Array. An array with the points that define the polygon.

### Return Value

Type: Boolean
Boolean. true if the polygon contains the specified point, false if not.

## See Also

DrawingUtils Members
DrawingUtils Class
MindFusion.Drawing Namespace

# DrawingUtils.radians Method

Converts the specified value from degrees to radians.

**Namespace:** MindFusion.Drawing
**File:** DrawingUtils.js

## Syntax

**JavaScript**

```
function radians (degrees)
```

### Parameters

*degrees*
Type: Number
Number. The degrees to convert.

### Return Value

Type: Number
Number. The converted angle.

## See Also

DrawingUtils Members
DrawingUtils Class
MindFusion.Drawing Namespace

# DrawingUtils.unionRects Method

Returns the smallest possible rectangle containing both of the specified rectangles.

**Namespace:** MindFusion.Drawing
**File:** DrawingUtils.js

## Syntax

**JavaScript**

```
function unionRects (rect1, rect2)
```

### Parameters

*rect1*
Type: Rect
Rect. The first rectangle.

*rect2*
Type: Rect
Rect. The second rectangle.

### Return Value

Type: Rect
Rect. A Rect instance that represents the union of the specified arguments.

## See Also

DrawingUtils Members
DrawingUtils Class
MindFusion.Drawing Namespace

# Ellipse Class

Represents an Ellipse.

## Syntax

| JavaScript |
| --- |
| `// class`<br>`Ellipse.prototype = {}` |

## Inheritance Hierarchy

MindFusion.Drawing.ComponentBase
**MindFusion.Drawing.Ellipse**

## See Also

Ellipse Members
MindFusion.Drawing Namespace

# Ellipse Members

Represents an Ellipse.

The following tables list the members exposed by the Ellipse type.

## Public Constructors

| | Name | Description |
| --- | --- | --- |
|  | Ellipse | Initializes a new instance of the Ellipse class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | height | Gets or sets the height of the ellipse. |
| | visibility | Gets or sets the visibility of this component. (Inherited from ComponentBase.) |
| | width | Gets or sets the width of the ellipse. |
| | x | Gets or sets the x-coordinate of the upper-left corner of the ellipse. |
| | y | Gets or sets the y-coordinate of the upper-left corner of the ellipse. |

## Public Methods

| | Name | Description |
|---|---|---|
| | clone | Creates a **Ellipse** object identical to the current object. |

## See Also

Ellipse Class
MindFusion.Drawing Namespace

# Ellipse Constructor

Initializes a new instance of the Ellipse class.

**Namespace:** MindFusion.Drawing
**File:** Ellipse.js

## Syntax

**JavaScript**

```
function Ellipse (x, y, width, height)
```

### Parameters

*x*
Type: Number
Number. The X-coordinate of the top left corner of the Ellipse.

*y*
Type: Number
Number. The Y-coordinate of the top left corner of the Ellipse.

*width*
Type: Number
Number. The width of the Ellipse.

*height*
Type: Number
Number. The height of the Ellipse.

## See Also

Ellipse Class
MindFusion.Drawing Namespace

# Ellipse Methods

## Ellipse.clone Method

Creates a Ellipse object identical to the current object.

**Namespace:** MindFusion.Drawing
**File:** Ellipse.js

### Syntax

```
JavaScript
function clone ()
```

#### Return Value

The newly created Ellipse object.

### See Also

Ellipse Members
Ellipse Class
MindFusion.Drawing Namespace

# Ellipse Properties

## Ellipse.height Property

Gets or sets the height of the ellipse.

**Namespace:** MindFusion.Drawing
**File:** Ellipse.js

### Syntax

```
JavaScript
get height() {}
set height(value) {}
```

#### Property Value

Type: Number
*Number*

The height of the ellipse.

### See Also

Ellipse Members
Ellipse Class
MindFusion.Drawing Namespace

# Ellipse.width Property

Gets or sets the width of the ellipse.

**Namespace:** MindFusion.Drawing
**File:** Ellipse.js

## Syntax

```
JavaScript
get width() {}
set width(value) {}
```

### Property Value

Type: Number
*Number*

The width of the ellipse.

## See Also

Ellipse Members
Ellipse Class
MindFusion.Drawing Namespace

# Ellipse.x Property

Gets or sets the x-coordinate of the upper-left corner of the ellipse.

**Namespace:** MindFusion.Drawing
**File:** Ellipse.js

## Syntax

```
JavaScript
get x() {}
set x(value) {}
```

### Property Value

Type: Number
*Number*

The x-coordinate of the upper-left corner of the ellipse.

## See Also

Ellipse Members
Ellipse Class
MindFusion.Drawing Namespace

## Ellipse.y Property

Gets or sets the y-coordinate of the upper-left corner of the ellipse.

**Namespace:** MindFusion.Drawing
**File:** Ellipse.js

### Syntax

**JavaScript**

```
get y() {}
set y(value) {}
```

#### Property Value

Type: Number
*Number*

The x-coordinate of the upper-left corner of the ellipse.

### See Also

Ellipse Members
Ellipse Class
MindFusion.Drawing Namespace

# Font Class

Represents a font.

### Syntax

**JavaScript**

```
// class
Font.prototype = {}
```

### Inheritance Hierarchy

**MindFusion.Drawing.Font**

### See Also

Font Members
MindFusion.Drawing Namespace

## Font Members

Represents a font.

The following tables list the members exposed by the Font type.

### Public Constructors

| | Name | Description |
|---|---|---|

| | | |
|---|---|---|
| | Font | Initializes a new instance of the Font class with the specified attributes. |

## Public Properties

| | Name | Description |
|---|---|---|
| | bold | Gets whether the font is bold. |
| | fontStyle | Gets or sets the style of this font. |
| | italic | Gets whether the font is italic. |
| | name | Gets or sets the name of this font. |
| | size | Gets or sets the size of this font. |
| | underline | Gets whether the font is underlined. |

## Public Methods

| | Name | Description |
|---|---|---|
| | apply | Applies the font properties to a CanvasRenderingContext2D drawing object. |
| S | fromObject | Returns a Font from a JSON object or string describing this font. |
| | toObject | Returns a JSON object describing this font. |

## See Also

Font Class
MindFusion.Drawing Namespace

# Font Constructor

Initializes a new instance of the Font class with the specified attributes.

**Namespace:** MindFusion.Drawing
**File:** Font.js

## Syntax

**JavaScript**

```
function Font (name, [size, [bold, [italic, [underline]]]])
```

### Parameters

*name*
Type: String
String. The font name.

*size*
Optional.

Type: Number
Number. The font size.

*bold*
Optional.
Type: Boolean | FontStyle
Boolean | FontStyle. true if this font is bold, or false otherwise.

*italic*
Optional.
Type: Boolean
Boolean. true if this font is italic, or false otherwise.

*underline*
Optional.
Type: Boolean
Boolean. true if this font is underlined, or false otherwise.

## See Also

Font Class
MindFusion.Drawing Namespace

# Font Methods

# Font.apply Method

Applies the font properties to a CanvasRenderingContext2D drawing object.

**Namespace:** MindFusion.Drawing
**File:** Font.js

## Syntax

**JavaScript**

```
function apply (context)
```

### Parameters

*context*
Type: CanvasRenderingContext2D
CanvasRenderingContext2D. The CanvasRenderingContext2D drawing object.

## See Also

Font Members
Font Class
MindFusion.Drawing Namespace

# Font.fromObject Method

Returns a Font from a JSON object or string describing this font.

**Namespace:** MindFusion.Drawing
**File:** Font.js

## Syntax

```
JavaScript
function fromObject ()
```

### Return Value

Type: Font
Font. The font object.

## See Also

Font Members
Font Class
MindFusion.Drawing Namespace

# Font.toObject Method

Returns a JSON object describing this font.

**Namespace:** MindFusion.Drawing
**File:** Font.js

## Syntax

```
JavaScript
function toObject ()
```

### Return Value

Type: Object
Object. The object describing this font.

## See Also

Font Members
Font Class
MindFusion.Drawing Namespace

# Font Properties

# Font.bold Property

Gets whether the font is bold.

**Namespace:** MindFusion.Drawing
**File:** Font.js

## Syntax

```
JavaScript
get bold() {}
```

### Property Value

*Boolean*

**true** if Bold flag is set in fontStyle, or **false** otherwise.

## See Also

Font Members
Font Class
MindFusion.Drawing Namespace

# Font.fontStyle Property

Gets or sets the style of this font.

**Namespace:** MindFusion.Drawing
**File:** Font.js

## Syntax

**JavaScript**

```
get fontStyle() {}
set fontStyle(value) {}
```

### Property Value

*FontStyle*

A combination of FontStyle enumeration flags.

## See Also

Font Members
Font Class
MindFusion.Drawing Namespace

# Font.italic Property

Gets whether the font is italic.

**Namespace:** MindFusion.Drawing
**File:** Font.js

## Syntax

**JavaScript**

```
get italic() {}
```

### Property Value

*Boolean*

**true** if Italic flag is set in fontStyle, or **false** otherwise.

## See Also

Font Members
Font Class

# Font.name Property

Gets or sets the name of this font.

**Namespace:** MindFusion.Drawing
**File:** Font.js

## Syntax

```
JavaScript
get name() {}
set name(value) {}
```

### Property Value

*String*

A string containing font name.

## See Also

Font Members
Font Class
MindFusion.Drawing Namespace

# Font.size Property

Gets or sets the size of this font.

**Namespace:** MindFusion.Drawing
**File:** Font.js

## Syntax

```
JavaScript
get size() {}
set size(value) {}
```

### Property Value

*Number*

The font size.

## See Also

Font Members
Font Class
MindFusion.Drawing Namespace

# Font.underline Property

Gets whether the font is underlined.

**Namespace:** MindFusion.Drawing
**File:** Font.js

## Syntax

```
JavaScript
get underline() {}
```

### Property Value

*Boolean*

**true** if Underline flag is set in fontStyle, or **false** otherwise.

## See Also

Font Members
Font Class
MindFusion.Drawing Namespace

# FontStyle Enumeration

Specifies font style attributes.

**Namespace:** MindFusion.Drawing
**File:** Enum.js

## Syntax

```
JavaScript
// enum
FontStyle = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **Bold** | Bold text. |
| | **Italic** | Italic text. |
| | **Regular** | Normal text. |
| | **Strikeout** | Striked text. |
| | **Underline** | Underlined text. |

## See Also

MindFusion.Drawing Namespace

# Graphics Class

Contains helper functions for drawing on a Canvas element.

## Syntax

```
// class
Graphics.prototype = {}
```

## Inheritance Hierarchy

## See Also

Graphics Members
MindFusion.Drawing Namespace

# Graphics Members

Contains helper functions for drawing on a Canvas element.

The following tables list the members exposed by the Graphics type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | Graphics | Initializes a new instance of the Graphics class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | context | Gets or sets the underlying CanvasRenderingContext2D drawing object. |

## Public Methods

| | Name | Description |
|---|---|---|
| | beginFigure | Starts a new figure. |
| | clear | Clears the canvas. |
| | clip | Sets the clip rectangle. |
| | closeFigure | Closes the current figure. |
| | drawCurve | Draws a spline curve. |
| | drawEllipse | Draws an ellipse. |
| | drawImage | Draws an image. |
| | drawLines | Draws a series of straight lines. |
| | drawPie | Draws a pie. |
| | drawPolygon | Draws a polygon. |
| | drawRectangle | Draws a rectangle. |

| | | |
|---|---|---|
| | drawString | Overloaded. |
| | fillEllipse | Fills an ellipse. |
| | fillPie | Fills a pie. |
| | fillPolygon | Fills a polygon. |
| | fillRectangle | Fills a rectangle. |
| | measureString | Gets the size of the specified string. |
| | pointInPie | Checks if the given point is within the bounds of the given pie. |
| | pointInPoly | Checks if the given point is within the bounds of the given polygon. |
| | restore | Restores to a previously saved graphics state. |
| | save | Saves the current graphics state. |
| | setTransform | Sets a transform. |
| | translate | Sets a translate transform. |

## See Also

Graphics Class
MindFusion.Drawing Namespace

# Graphics Constructor

Initializes a new instance of the Graphics class.

**Namespace:** MindFusion.Drawing
**File:** Graphics.js

## Syntax

**JavaScript**

```
function Graphics (context)
```

### Parameters

*context*
Type: CanvasRenderingContext2D
The CanvasRenderingContext2D drawing object.

## See Also

Graphics Class
MindFusion.Drawing Namespace

# Graphics Methods

# Graphics.beginFigure Method

Starts a new figure.

**Namespace:** MindFusion.Drawing
**File:** Graphics.js

## Syntax

**JavaScript**

```
function beginFigure ([pen, [brush, [bounds]]])
```

### Parameters

*pen*
Optional.
Type: Pen
The pen to stroke the figure with.

*brush*
Optional.
Type: Brush
The brush to fill the figure with.

*bounds*
Optional.
Type: Rect
The bounds to which the brush should be applied.

## See Also

Graphics Members
Graphics Class
MindFusion.Drawing Namespace

# Graphics.clear Method

Clears the canvas.

**Namespace:** MindFusion.Drawing
**File:** Graphics.js

## Syntax

**JavaScript**

```
function clear ()
```

## See Also

Graphics Members
Graphics Class
MindFusion.Drawing Namespace

# Graphics.clip Method

Sets the clip rectangle.

**Namespace:** MindFusion.Drawing
**File:** Graphics.js

## Syntax

```
JavaScript
function clip (rect)
```

### Parameters

*rect*
Type: Rect
The clip rectangle

## See Also

Graphics Members
Graphics Class
MindFusion.Drawing Namespace

# Graphics.closeFigure Method

Closes the current figure.

**Namespace:** MindFusion.Drawing
**File:** Graphics.js

## Syntax

```
JavaScript
function closeFigure ([pen, [brush, [bounds]]])
```

### Parameters

*pen*
Optional.
Type: Pen
The pen to stroke the figure with.

*brush*
Optional.
Type: Brush
The brush to fill the figure with.

*bounds*
Optional.
Type: Rect
The bounds to which the brush should be applied.

## See Also

Graphics Members
Graphics Class
MindFusion.Drawing Namespace

# Graphics.drawCurve Method

Draws a spline curve.

**Namespace:** MindFusion.Drawing
**File:** Graphics.js

## Syntax

```
JavaScript
function drawCurve (pen, points)
```

### Parameters

*pen*
Type: Pen
The pen to stroke the curve with.

*points*

## See Also

Graphics Members
Graphics Class
MindFusion.Drawing Namespace

# Graphics.drawEllipse Method

Draws an ellipse.

**Namespace:** MindFusion.Drawing
**File:** Graphics.js

## Syntax

```
JavaScript
function drawEllipse (pen, rect)
```

### Parameters

*pen*
Type: Pen
The pen to stroke the ellipse with.

*rect*
Type: Rect
The rectangle describing the ellipse.

## See Also

Graphics Members
Graphics Class
MindFusion.Drawing Namespace

# Graphics.drawImage Method

Draws an image.

**Namespace:** MindFusion.Drawing
**File:** Graphics.js

## Syntax

**JavaScript**

```
function drawImage (image, rect, align)
```

### Parameters

*image*
Type: HTMLImageElement
The image to draw.

*rect*
Type: Rect
The bounds of the image.

*align*
Type: ImageAlign
The align to use.

## See Also

Graphics Members
Graphics Class
MindFusion.Drawing Namespace

# Graphics.drawLines Method

Draws a series of straight lines.

**Namespace:** MindFusion.Drawing
**File:** Graphics.js

## Syntax

**JavaScript**

```
function drawLines (pen, points)
```

### Parameters

*pen*
Type: Pen
The pen to draw the lines with.

*points*

## See Also

Graphics Members
Graphics Class
MindFusion.Drawing Namespace

# Graphics.drawPie Method

Draws a pie.

**Namespace:** MindFusion.Drawing
**File:** Graphics.js

## Syntax

**JavaScript**

```
function drawPie (pen, rect, radius, startAngle, endAngle, [doughnut])
```

### Parameters

*pen*
Type: Pen
The pen to stroke the pie with.

*rect*
Type: Rect
The bounds of the pie.

*radius*
Type: Number
The radius of the pie.

*startAngle*
Type: Number
The startAngle of the pie.

*endAngle*
Type: Number
The endAngle of the pie.

*doughnut*
Optional.
Type: Boolean
True if the pie should be a doughnut, otherwise false.

## See Also

Graphics Members
Graphics Class
MindFusion.Drawing Namespace

# Graphics.drawPolygon Method

Draws a polygon.

**Namespace:** MindFusion.Drawing
**File:** Graphics.js

## Syntax

**JavaScript**

```
function drawPolygon (pen, points)
```

### Parameters

*pen*
Type: Pen
The pen to draw the polygon with.

*points*

## See Also

Graphics Members
Graphics Class
MindFusion.Drawing Namespace

# Graphics.drawRectangle Method

Draws a rectangle.

**Namespace:** MindFusion.Drawing
**File:** Graphics.js

## Syntax

| JavaScript |
| --- |
| `function drawRectangle (pen, rect)` |

### Parameters

*pen*
Type: Pen
The pen to stroke the rectangle with.

*rect*
Type: Rect
The rectangle to draw.

## See Also

Graphics Members
Graphics Class
MindFusion.Drawing Namespace

# Graphics.drawString Method

## Overload List

| Name | Description |
| --- | --- |
| Graphics.drawString | Draws the specified string with the specified font and brush at the specified location. |
| Graphics.drawString | Draws the specified string with the specified font and brush at the speciied location. |

## See Also

Graphics Members
Graphics Class
MindFusion.Drawing Namespace

# Graphics.drawString Method

Draws the specified string with the specified font and brush at the specified location.

**Namespace:** MindFusion.Drawing
**File:** Graphics.js

## Syntax

```
JavaScript
function drawString (text, font, brush, location, [format])
```

### Parameters

*text*
Type: String
The string to draw.

*font*
Type: Font
The font to use.

*brush*
Type: Brush
The brush to use.

*location*
Type: Point, Rect
The position at which to draw.

*format*
Optional.
Type: StringFormat
The format to use.

## See Also

drawString Method Overload List
Graphics Members
Graphics Class
MindFusion.Drawing Namespace

# Graphics.drawString Method

Draws the specified string with the specified font and brush at the speciied location.

**Namespace:** MindFusion.Drawing
**File:** Graphics.js

## Syntax

```
JavaScript
function drawString (text, font, brush, location, [format])
```

### Parameters

*text*
Type: String
The string to draw.

*font*
Type: Font
The font to use.

*brush*
Type: Brush
The brush to use.

*location*
Type: Point|Rect
The position at which to draw.

*format*
Optional.
Type: StringFormat
The format to use.

## See Also

drawString Method Overload List
Graphics Members
Graphics Class
MindFusion.Drawing Namespace

# Graphics.fillEllipse Method

Fills an ellipse.

**Namespace:** MindFusion.Drawing
**File:** Graphics.js

## Syntax

**JavaScript**

```
function fillEllipse (brush, rect)
```

### Parameters

*brush*
Type: Brush
The brush to fill the ellipse with.

*rect*
Type: Rect
The rectangle describing the ellipse.

## See Also

Graphics Members
Graphics Class
MindFusion.Drawing Namespace

# Graphics.fillPie Method

Fills a pie.

**Namespace:** MindFusion.Drawing
**File:** Graphics.js

## Syntax

**JavaScript**

```
function fillPie (brush, rect, radius, startAngle, endAngle, [doughnut])
```

### Parameters

*brush*
Type: Brush
The brush to fill the pie with.

*rect*
Type: Rect
The bounds of the pie.

*radius*
Type: Number
The radius of the pie.

*startAngle*
Type: Number
The startAngle of the pie.

*endAngle*
Type: Number
The endAngle of the pie.

*doughnut*
Optional.
Type: Boolean
True if the pie should be a doughnut, otherwise false.

## See Also

Graphics Members
Graphics Class
MindFusion.Drawing Namespace

# Graphics.fillPolygon Method

Fills a polygon.

**Namespace:** MindFusion.Drawing
**File:** Graphics.js

## Syntax

**JavaScript**

```
function fillPolygon (brush, points)
```

### Parameters

*brush*
Type: Brush
The brush to fill the polygon with.

*points*

## See Also

Graphics Members
Graphics Class
MindFusion.Drawing Namespace

# Graphics.fillRectangle Method

Fills a rectangle.

**Namespace:** MindFusion.Drawing
**File:** Graphics.js

## Syntax

**JavaScript**

```
function fillRectangle (brush, rect)
```

### Parameters

*brush*
Type: Brush
The pen to fill the rectangle with.

*rect*
Type: Rect
The rectangle to fill.

## See Also

Graphics Members
Graphics Class
MindFusion.Drawing Namespace

# Graphics.measureString Method

Gets the size of the specified string.

**Namespace:** MindFusion.Drawing
**File:** Graphics.js

## Syntax

**JavaScript**

```
function measureString (text, font, [width])
```

### Parameters

*text*

Type: String
The string to measure.

*font*
Type: Font
The font to use.

*width*
Optional.
Type: Number
The maximum width.

### Return Value

Type: Size
The size of the given string when drawn with the given font.

## See Also

Graphics Members
Graphics Class
MindFusion.Drawing Namespace

# Graphics.pointInPie Method

Checks if the given point is within the bounds of the given pie.

**Namespace:** MindFusion.Drawing
**File:** Graphics.js

## Syntax

**JavaScript**

```
function pointInPie (point, rect, radius, startAngle, endAngle, [doughnut])
```

### Parameters

*point*
Type: Point
The point to check.

*rect*
Type: Rect
The bounds of the pie.

*radius*
Type: Number
The radius of the pie.

*startAngle*
Type: Number
The startAngle of the pie.

*endAngle*
Type: Number
The endAngle of the pie.

*doughnut*
Optional.
Type: Boolean
True if the pie is a doughnut, otherwise false.

### Return Value

Type: Boolean
True if the point lies within pie bounds, otherwise false.

## See Also

Graphics Members
Graphics Class
MindFusion.Drawing Namespace

# Graphics.pointInPoly Method

Checks if the given point is within the bounds of the given polygon.

**Namespace:** MindFusion.Drawing
**File:** Graphics.js

## Syntax

**JavaScript**

```
function pointInPoly (point, poly)
```

### Parameters

*point*
Type: Point
The point to check.

*poly*

## See Also

Graphics Members
Graphics Class
MindFusion.Drawing Namespace

# Graphics.restore Method

Restores to a previously saved graphics state.

**Namespace:** MindFusion.Drawing
**File:** Graphics.js

## Syntax

**JavaScript**

```
function restore ()
```

## See Also

Graphics Members
Graphics Class
MindFusion.Drawing Namespace

# Graphics.save Method

Saves the current graphics state.

**Namespace:** MindFusion.Drawing
**File:** Graphics.js

## Syntax

**JavaScript**

```
function save ()
```

## See Also

Graphics Members
Graphics Class
MindFusion.Drawing Namespace

# Graphics.setTransform Method

Sets a transform.

**Namespace:** MindFusion.Drawing
**File:** Graphics.js

## Syntax

**JavaScript**

```
function setTransform (transform)
```

### Parameters

*transform*
Type: Matrix
The transform matrix.

## See Also

Graphics Members
Graphics Class
MindFusion.Drawing Namespace

# Graphics.translate Method

Sets a translate transform.

**Namespace:** MindFusion.Drawing
**File:** Graphics.js

## Syntax

**JavaScript**

```
function translate (x, y)
```

### Parameters

*x*
Type: Number
The distance to translate along the x-axis.

*y*
Type: Number
The distance to translate along the y-axis.

## See Also

Graphics Members
Graphics Class
MindFusion.Drawing Namespace

# Graphics Properties

## Graphics.context Property

Gets or sets the underlying CanvasRenderingContext2D drawing object.

**Namespace:** MindFusion.Drawing
**File:** Graphics.js

## Syntax

```
JavaScript
get context() {}
```

### Property Value

Type: CanvasRenderingContext2D

## See Also

Graphics Members
Graphics Class
MindFusion.Drawing Namespace

# GraphicsUnit Class

Specifies the unit of measure for length and size properties.

## Syntax

```
JavaScript
// class
GraphicsUnit.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Drawing.GraphicsUnit**

## See Also

GraphicsUnit Members
MindFusion.Drawing Namespace

# GraphicsUnit Members

Specifies the unit of measure for length and size properties.

The following tables list the members exposed by the GraphicsUnit type.

## Public Properties

| | Name | Description |
|---|---|---|
| | Centimeter | Specifies the centimeter as the unit of measure. |
| | Display | Specifies the display unit as the unit of measure. |
| | Document | Specifies the document unit as the unit of measure. |
| | Inch | Specifies the inch as the unit of measure. |
| | Millimeter | Specifies the millimeter as the unit of measure. |
| | Percent | Specifies the world coordinate system unit as the unit of measure. |
| | Pixel | Specifies the pixel unit as the unit of measure. |
| | Point | Specifies a printer's point as the unit of measure. |
| | World | Specifies the world coordinate system unit as the unit of measure. |
| | WpfPoint | Specifies the WpfPoint unit as the unit of measure. |

## See Also

GraphicsUnit Class
MindFusion.Drawing Namespace

# GraphicsUnit Properties

# GraphicsUnit.Centimeter Property

Specifies the centimeter as the unit of measure.

**Namespace:** MindFusion.Drawing
**File:** GraphicsUnit.js

## Syntax

**JavaScript**

```
get Centimeter() {}
set Centimeter(value) {}
```

### Property Value

Type: Number
*Number*

Defines 2.54 DPI resolution.

## See Also

GraphicsUnit Members
GraphicsUnit Class
MindFusion.Drawing Namespace

# GraphicsUnit.Display Property

Specifies the display unit as the unit of measure.

**Namespace:** MindFusion.Drawing
**File:** GraphicsUnit.js

## Syntax

**JavaScript**

```
get Display() {}
set Display(value) {}
```

### Property Value

Type: Number
*Number*

Defines 100 DPI resolution.

## See Also

GraphicsUnit Members
GraphicsUnit Class
MindFusion.Drawing Namespace

# GraphicsUnit.Document Property

Specifies the document unit as the unit of measure.

**Namespace:** MindFusion.Drawing
**File:** GraphicsUnit.js

## Syntax

**JavaScript**

```
get Document() {}
set Document(value) {}
```

### Property Value

Type: Number
*Number*

Defines 300 DPI resolution.

## See Also

GraphicsUnit Members
GraphicsUnit Class
MindFusion.Drawing Namespace

# GraphicsUnit.Inch Property

Specifies the inch as the unit of measure.

**Namespace:** MindFusion.Drawing
**File:** GraphicsUnit.js

## Syntax

**JavaScript**

```
get Inch() {}
set Inch(value) {}
```

### Property Value

Type: Number
*Number*

Defines 1 DPI resolution.

## See Also

GraphicsUnit Members
GraphicsUnit Class
MindFusion.Drawing Namespace

# GraphicsUnit.Millimeter Property

Specifies the millimeter as the unit of measure.

**Namespace:** MindFusion.Drawing
**File:** GraphicsUnit.js

## Syntax

**JavaScript**

```
get Millimeter() {}
set Millimeter(value) {}
```

### Property Value

Type: Number
*Number*

Defines 25.4 DPI resolution.

## See Also

GraphicsUnit Members
GraphicsUnit Class
MindFusion.Drawing Namespace

## GraphicsUnit.Percent Property

Specifies the world coordinate system unit as the unit of measure.

**Namespace:** MindFusion.Drawing
**File:** GraphicsUnit.js

### Syntax

**JavaScript**

```
get Percent() {}
set Percent(value) {}
```

#### Property Value

Type: Number
*Number*

For internal use.

### See Also

GraphicsUnit Members
GraphicsUnit Class
MindFusion.Drawing Namespace

## GraphicsUnit.Pixel Property

Specifies the pixel unit as the unit of measure.

**Namespace:** MindFusion.Drawing
**File:** GraphicsUnit.js

### Syntax

**JavaScript**

```
get Pixel() {}
set Pixel(value) {}
```

#### Property Value

Type: Number
*Number*

Defines 96 DPI resolution.

### See Also

GraphicsUnit Members
GraphicsUnit Class
MindFusion.Drawing Namespace

## GraphicsUnit.Point Property

Specifies a printer's point as the unit of measure.

**Namespace:** MindFusion.Drawing
**File:** GraphicsUnit.js

### Syntax

**JavaScript**

```
get Point() {}
set Point(value) {}
```

#### Property Value

Type: Number
*Number*

Defines 72 DPI resolution.

### See Also

GraphicsUnit Members
GraphicsUnit Class
MindFusion.Drawing Namespace

## GraphicsUnit.World Property

Specifies the world coordinate system unit as the unit of measure.

**Namespace:** MindFusion.Drawing
**File:** GraphicsUnit.js

### Syntax

**JavaScript**

```
get World() {}
set World(value) {}
```

#### Property Value

Type: Number
*Number*

For internal use.

### See Also

GraphicsUnit Members
GraphicsUnit Class
MindFusion.Drawing Namespace

## GraphicsUnit.WpfPoint Property

Specifies the WpfPoint unit as the unit of measure.

**Namespace:** MindFusion.Drawing
**File:** GraphicsUnit.js

## Syntax

```
JavaScript
get WpfPoint() {}
set WpfPoint(value) {}
```

### Property Value

Type: Number
*Number*

Defines 96 DPI resolution.

## See Also

GraphicsUnit Members
GraphicsUnit Class
MindFusion.Drawing Namespace

# Image Class

Represents an image.

## Syntax

```
JavaScript
// class
Image.prototype = {}
```

## Inheritance Hierarchy

MindFusion.Drawing.ComponentBase
**MindFusion.Drawing.Image**

## See Also

Image Members
MindFusion.Drawing Namespace

# Image Members

Represents an image.

The following tables list the members exposed by the Image type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | Image | Initializes a new instance of the **Image** class. |

## Public Fields

| | Name | Description |
|---|---|---|
| | bounds | The object's bounds. |
| | image | The image to be displayed. |

## Public Properties

| | Name | Description |
|---|---|---|
| | visibility | Gets or sets the visibility of this component. (Inherited from ComponentBase.) |

## See Also

Image Class
MindFusion.Drawing Namespace

# Image Fields

# Image.bounds Field

The object's bounds.

**Namespace:** MindFusion.Drawing
**File:** Image.js

## Syntax

| JavaScript |
|---|
| `var bounds` |

## See Also

Image Members
Image Class
MindFusion.Drawing Namespace

# Image.image Field

The image to be displayed.

**Namespace:** MindFusion.Drawing
**File:** Image.js

## Syntax

| JavaScript |
|---|
| `var image` |

## See Also

Image Members
Image Class
MindFusion.Drawing Namespace

### Image Constructor

Initializes a new instance of the Image class.

**Namespace:** MindFusion.Drawing
**File:** Image.js

## Syntax

**JavaScript**

```
function Image (bounds)
```

#### Parameters

*bounds*
Type: Rect
Rect. The bounds of the image.

## See Also

Image Class
MindFusion.Drawing Namespace

# ImageAlign Enumeration

Specifies the position and alignment of a picture in a node, or that of the background image.

**Namespace:** MindFusion.Drawing
**File:** Enum.js

## Syntax

**JavaScript**

```
// enum
ImageAlign = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **BottomCenter** | The image is centered horizontally and aligned to the bottom side. |
| | **BottomLeft** | The image is aligned to the bottom left corner of the node or the diagram. |
| | **BottomRight** | The image is aligned to the bottom right corner of the node or the diagram. |

| | | |
|---|---|---|
| | **Center** | The image is centered in the node or diagram. |
| | **Fit** | The image is resized to fit the size of the object or the component's client area. |
| | **FitBottom** | The image is resized to fit the size of the object and aligned to bottom side. |
| | **FitLeft** | The image is resized to fit the size of the object and aligned to left side. |
| | **FitRight** | The image is resized to fit the size of the object and aligned to right side. |
| | **FitTop** | The image is resized to fit the size of the object and aligned to top side. |
| | **MiddleLeft** | The image is centered vertically and aligned to the left-hand side. |
| | **MiddleRight** | The image is centered vertically and aligned to the right-hand side. |
| | **Stretch** | The image is stretched to fill the object or the component's client area. |
| | **Tile** | The image is tiled to cover the node or the diagram. |
| | **TopCenter** | The image is centered horizontally and aligned to the top side. |
| | **TopLeft** | The image is aligned to the top left corner of the node or the diagram. |
| | **TopRight** | The image is aligned to the top right corner of the node or the diagram. |

## See Also

MindFusion.Drawing Namespace

# LayoutAlignment Enumeration

Specifies alignment of components.

**Namespace:** MindFusion.Drawing
**File:** Enum.js

## Syntax

**JavaScript**

```
// enum
LayoutAlignment = {}
```

## Members

| | Member name | Description |
|---|---|---|

| | | |
|---|---|---|
| | **Center** | Center alignment. |
| | **Far** | Right ot bottom alignment |
| | **Near** | Left or top alignment. |
| | **Stretch** | Stretch the component. |

## See Also

MindFusion.Drawing Namespace

# LinearGradientBrush Class

Represents a linear gradient brush.

## Syntax

**JavaScript**

```
// class
LinearGradientBrush.prototype = {}
```

## Inheritance Hierarchy

## See Also

LinearGradientBrush Members
MindFusion.Drawing Namespace

# LinearGradientBrush Members

Represents a linear gradient brush.

The following tables list the members exposed by the LinearGradientBrush type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | LinearGradientBrush | Overloaded. |

## Public Properties

| | Name | Description |
|---|---|---|
| | angle | Gets or sets angle of gradient rotation. |
| | endColor | Gets or sets the end color of the gradient. |
| | startColor | Gets or sets the start color of the gradient. |

## Public Methods

| | Name | Description |
|---|---|---|
| | apply | Applies the brush properties to a CanvasRenderingContext2D drawing object. |
| | clone | Creates a copy of this brush. |
| | equals | Determines whether the specified object is equal to this LinearGradientBrush. |
| S | fromObject | Returns a LinearGradientBrush from a JSON object or string describing this brush. |
| | toObject | Returns a JSON object describing this brush. |

## See Also

LinearGradientBrush Class
MindFusion.Drawing Namespace

# LinearGradientBrush Constructor

## Overload List

| Name | Description |
|---|---|
| LinearGradientBrush | Initializes a new instance of the LinearGradientBrush class. |
| LinearGradientBrush | Initializes a new instance of the **LinearGradientBrush** class. |

## See Also

LinearGradientBrush Class
MindFusion.Drawing Namespace

# LinearGradientBrush Constructor

Initializes a new instance of the LinearGradientBrush class.

**Namespace:** MindFusion.Drawing
**File:** Brush.js

## Syntax

**JavaScript**

```
function LinearGradientBrush (startColor, endColor, [angle, [bounds]])
```

### Parameters

*startColor*
Type: Color | string
The starting color of the gradient.

*endColor*

Type: **Color** | string
The ending color of the gradient.

*angle*
Optional.
Type: Number
The angle of the gradient.

*bounds*
Optional.
Type: Rect
The bounds of the gradient.

## Example

The following code creates three linear brushes to be used by an AreaChart:

**JavaScript**

```
// create area brushes
var firstBrush = new Drawing.Brush("#ce0000");
var secondBrush = new Drawing.LinearGradientBrush("#003466", "#000063");
var thirdBrush = new Drawing.LinearGradientBrush("#e0e9e9", "#669acc");
```

## See Also

LinearGradientBrush Constructor Overload List
LinearGradientBrush Class
MindFusion.Drawing Namespace

# LinearGradientBrush Constructor

Initializes a new instance of the LinearGradientBrush class.

**Namespace:** MindFusion.Drawing
**File:** Brush.js

## Syntax

**JavaScript**

```
function LinearGradientBrush (startColor, endColor, [angle, [bounds]])
```

### Parameters

*startColor*
Type: Color, string
The starting color of the gradient.

*endColor*
Type: **Color**, string
The ending color of the gradient.

*angle*
Optional.
Type: Number
The angle of the gradient.

*bounds*
Optional.
Type: Rect
The bounds of the gradient.

## Example

The following code creates three linear brushes to be used by an AreaChart:

**JavaScript**

```
// create area brushes
var firstBrush = new Drawing.Brush("#ce0000");
var secondBrush = new Drawing.LinearGradientBrush("#003466", "#000063");
var thirdBrush = new Drawing.LinearGradientBrush("#e0e9e9", "#669acc");
```

## See Also

LinearGradientBrush Constructor Overload List
LinearGradientBrush Class
MindFusion.Drawing Namespace

# LinearGradientBrush Methods

# LinearGradientBrush.apply Method

Applies the brush properties to a CanvasRenderingContext2D drawing object.

**Namespace:** MindFusion.Drawing
**File:** Brush.js

## Syntax

**JavaScript**

```
function apply (context, bounds)
```

### Parameters

*context*
Type: CanvasRenderingContext2D
The CanvasRenderingContext2D drawing object.

*bounds*
Type: Rect
The bounds of the gradient.

## See Also

LinearGradientBrush Members
LinearGradientBrush Class
MindFusion.Drawing Namespace

# LinearGradientBrush.clone Method

Creates a copy of this brush.

**Namespace:** MindFusion.Drawing
**File:** Brush.js

## Syntax

JavaScript

```
function clone ()
```

### Return Value

Type: LinearGradientBrush
The copy of this brush.

## See Also

LinearGradientBrush Members
LinearGradientBrush Class
MindFusion.Drawing Namespace

# LinearGradientBrush.equals Method

Determines whether the specified object is equal to this LinearGradientBrush.

**Namespace:** MindFusion.Drawing
**File:** Brush.js

## Syntax

JavaScript

```
function equals (The)
```

### Parameters

*The*
Type: Object
object to compare with.

### Return Value

Type: Boolean
true if the specified object is equal to this LinearGradientBrush; otherwise, false.

## See Also

LinearGradientBrush Members
LinearGradientBrush Class
MindFusion.Drawing Namespace

# LinearGradientBrush.fromObject Method

Returns a LinearGradientBrush from a JSON object or string describing this brush.

**Namespace:** MindFusion.Drawing
**File:** Brush.js

## Syntax

JavaScript

```
function fromObject (obj)
```

### Parameters

*obj*
Type: void

### Return Value

Type: LinearGradientBrush
The brush object.

## Remarks

Used for backward compatibility with the JSLib class.

## See Also

LinearGradientBrush Members
LinearGradientBrush Class
MindFusion.Drawing Namespace

# LinearGradientBrush.toObject Method

Returns a JSON object describing this brush.

**Namespace:** MindFusion.Drawing
**File:** Brush.js

## Syntax

**JavaScript**

```
function toObject ()
```

### Return Value

Type: Object
The object describing this brush.

## Remarks

Used for backward compatibility with the JSLib class.

## See Also

LinearGradientBrush Members
LinearGradientBrush Class
MindFusion.Drawing Namespace

# LinearGradientBrush Properties

# LinearGradientBrush.angle Property

Gets or sets angle of gradient rotation.

**Namespace:** MindFusion.Drawing
**File:** Brush.js

## Syntax

```
JavaScript
get angle() {}
```

### Property Value

Type: Number

## See Also

LinearGradientBrush Members
LinearGradientBrush Class
MindFusion.Drawing Namespace

# LinearGradientBrush.endColor Property

Gets or sets the end color of the gradient.

**Namespace:** MindFusion.Drawing
**File:** Brush.js

## Syntax

```
JavaScript
get endColor() {}
```

### Property Value

Type: Color

## See Also

LinearGradientBrush Members
LinearGradientBrush Class
MindFusion.Drawing Namespace

# LinearGradientBrush.startColor Property

Gets or sets the start color of the gradient.

**Namespace:** MindFusion.Drawing
**File:** Brush.js

## Syntax

```
JavaScript
get startColor() {}
```

### Property Value

Type: Color

## See Also

LinearGradientBrush Members
LinearGradientBrush Class
MindFusion.Drawing Namespace

# LineJoin Enumeration

Specifies how to join consecutive line or curve segments in a figure.

**Namespace:** MindFusion.Drawing
**File:** Enum.js

## Syntax

**JavaScript**

```
// enum
LineJoin = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **Bevel** | Produces a diagonal corner. |
| | **Miter** | Produces a sharp corner or a clipped corner, depending on whether the length of the miter exceeds the miter limit. |
| | **Round** | Produces a smooth, circular arc between the lines. |

## See Also

MindFusion.Drawing Namespace

# Path Class

Represents a path.

## Syntax

**JavaScript**

```
// class
Path.prototype = {}
```

## Inheritance Hierarchy

MindFusion.Drawing.ComponentBase
  **MindFusion.Drawing.Path**

## See Also

# Path Members

Represents a path.

The following tables list the members exposed by the Path type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | Path | Initializes a new instance of the **Path** class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | brush | Gets or sets the brush used to fill a closed path. |
| | pen | Gets or sets the pen used to draw the path. |
| | text | Gets or sets the text displayed inside this path. |
| | visibility | Gets or sets the visibility of this component. (Inherited from ComponentBase.) |

## Public Methods

| | Name | Description |
|---|---|---|
| | addEllipse | Adds an ellipse figure to the path. |
| | addRect | Adds a rectangle figure to the path. |
| | arcTo | Draws an arc. |
| | bezierTo | Draws a cubic bezier curve. |
| | clone | Creates a **Path** object identical to the current object. |
| | close | Closes the path. |
| | done | Ends the path definition. |
| | empty | Gets a value indicating whether this **Path** is empty. |
| | getBounds | Gets the bounding rect of the path. |
| | init | Begins a path or resets the current path. |
| | lineTo | Draws a line from the current point to the specified location. |
| | moveTo | Moves the path to the specified location. |
| | quadraticCurveTo | Draws a quadratic bezier curve. |

## See Also

Path Class
MindFusion.Drawing Namespace

# Path Constructor

Initializes a new instance of the Path class.

**Namespace:** MindFusion.Drawing
**File:** Path.js

## Syntax

**JavaScript**

```
function Path ([pathString])
```

### Parameters

*pathString*
Optional.
Type: String
String. A string representing the path figures.

## See Also

Path Class
MindFusion.Drawing Namespace

# Path Methods

# Path.addEllipse Method

Adds an ellipse figure to the path.

**Namespace:** MindFusion.Drawing
**File:** Path.js

## Syntax

**JavaScript**

```
function addEllipse (x, y, width, height)
```

### Parameters

*x*
Type: Number
Number. The x-coordinate of the center of the ellipse.

*y*
Type: Number
Number. The y-coordinate of the center of the ellipse.

*width*
Type: Number
Number. The width of the ellipse.

*height*

Type: Number
Number. The height of the ellipse.

## See Also

Path Members
Path Class
MindFusion.Drawing Namespace

# Path.addRect Method

Adds a rectangle figure to the path.

**Namespace:** MindFusion.Drawing
**File:** Path.js

## Syntax

**JavaScript**

```
function addRect (x, y, width, height)
```

### Parameters

*x*
Type: Number
Number. The x-coordinate of the rectangle.

*y*
Type: Number
Number. The y-coordinate of the rectangle.

*width*
Type: Number
Number. The width of the rectangle.

*height*
Type: Number
Number. The height of the rectangle.

## See Also

Path Members
Path Class
MindFusion.Drawing Namespace

# Path.arcTo Method

Draws an arc.

**Namespace:** MindFusion.Drawing
**File:** Path.js

## Syntax

**JavaScript**

```
function arcTo (x, y, radius, startAngle, endAngle, anticlockwise)
```

### Parameters

*x*
Type: Number
Number. The x-coordinate of the center of the circle.

*y*
Type: Number
Number. The y-coordinate of the center of the circle.

*radius*
Type: Number
Number. The radius of the circle

*startAngle*
Type: Number
Number. The starting angle in radians.

*endAngle*
Type: Number
Number. The ending angle in radians.

*anticlockwise*
Type: Boolean
Boolean. Specifies whether the drawing should be counterclockwise or clockwise.

## See Also

Path Members
Path Class
MindFusion.Drawing Namespace

# Path.bezierTo Method

Draws a cubic bezier curve.

**Namespace:** MindFusion.Drawing
**File:** Path.js

## Syntax

**JavaScript**

```
function bezierTo (x1, y1, x2, y2, x3, y3)
```

### Parameters

*x1*
Type: Number
Number. The x-coordinate of the first bezier control point.

*y1*
Type: Number
Number. The y-coordinate of the first bezier control point.

*x2*
Type: Number
Number. The x-coordinate of the second bezier control point.

*y2*
Type: Number
Number. The y-coordinate of the second bezier control point.

*x3*
Type: Number
Number. The x-coordinate of the ending point.

*y3*
Type: Number
Number. The y-coordinate of the ending point.

## See Also

Path Members
Path Class
MindFusion.Drawing Namespace

# Path.clone Method

Creates a Path object identical to the current object.

**Namespace:** MindFusion.Drawing
**File:** Path.js

## Syntax

**JavaScript**

```
function clone ()
```

### Return Value

The newly created Path object.

## See Also

Path Members
Path Class
MindFusion.Drawing Namespace

# Path.close Method

Closes the path.

**Namespace:** MindFusion.Drawing
**File:** Path.js

## Syntax

**JavaScript**

```
function close ()
```

## See Also

Path Members
Path Class
MindFusion.Drawing Namespace

## Path.done Method

Ends the path definition.

**Namespace:** MindFusion.Drawing
**File:** Path.js

### Syntax

**JavaScript**

```
function done ()
```

### See Also

Path Members
Path Class
MindFusion.Drawing Namespace

## Path.empty Method

Gets a value indicating whether this Path is empty.

**Namespace:** MindFusion.Drawing
**File:** Path.js

### Syntax

**JavaScript**

```
function empty ()
```

#### Return Value

**true** if the path does not contain any figures; otherwise, **false**.

### See Also

Path Members
Path Class
MindFusion.Drawing Namespace

## Path.getBounds Method

Gets the bounding rect of the path.

**Namespace:** MindFusion.Drawing
**File:** Path.js

### Syntax

**JavaScript**

```
function getBounds ()
```

#### Return Value

Type: Rect
Rect. The bounding rect.

## See Also

Path Members
Path Class
MindFusion.Drawing Namespace

# Path.init Method

Begins a path or resets the current path.

**Namespace:** MindFusion.Drawing
**File:** Path.js

## Syntax

**JavaScript**

```
function init ()
```

## See Also

Path Members
Path Class
MindFusion.Drawing Namespace

# Path.lineTo Method

Draws a line from the current point to the specified location.

**Namespace:** MindFusion.Drawing
**File:** Path.js

## Syntax

**JavaScript**

```
function lineTo (x, y)
```

### Parameters

*x*
Type: Number
Number. The x-coordinate of the point.

*y*
Type: Number
Number. The y-coordinate of the point.

## See Also

Path Members
Path Class
MindFusion.Drawing Namespace

# Path.moveTo Method

Moves the path to the specified location.

**Namespace:** MindFusion.Drawing
**File:** Path.js

## Syntax

**JavaScript**

```
function moveTo (x, y)
```

### Parameters

*x*
Type: Number
Number. The x-coordinate of the point.

*y*
Type: Number
Number. The y-coordinate of the point.

## See Also

Path Members
Path Class
MindFusion.Drawing Namespace

# Path.quadraticCurveTo Method

Draws a quadratic bezier curve.

**Namespace:** MindFusion.Drawing
**File:** Path.js

## Syntax

**JavaScript**

```
function quadraticCurveTo (x1, y1, x, y)
```

### Parameters

*x1*
Type: Number
Number. The x-coordinate of the bezier control point.

*y1*
Type: Number
Number. The y-coordinate of the bezier control point.

*x*
Type: Number
Number. The x-coordinate of the ending point.

*y*
Type: Number
Number. The y-coordinate of the ending point.

## See Also

Path Members
Path Class
MindFusion.Drawing Namespace

# Path Properties

## Path.brush Property

Gets or sets the brush used to fill a closed path.

**Namespace:** MindFusion.Drawing
**File:** Path.js

## Syntax

```
JavaScript
get brush() {}
set brush(value) {}
```

### Property Value

*Object*

A color string or an object literal specifying gradient.

## See Also

Path Members
Path Class
MindFusion.Drawing Namespace

## Path.pen Property

Gets or sets the pen used to draw the path.

**Namespace:** MindFusion.Drawing
**File:** Path.js

## Syntax

```
JavaScript
get pen() {}
set pen(value) {}
```

### Property Value

*Object*

A color string.

## See Also

Path Members
Path Class
MindFusion.Drawing Namespace

### Path.text Property

Gets or sets the text displayed inside this path.

**Namespace:** MindFusion.Drawing
**File:** Path.js

## Syntax

```
JavaScript
get text() {}
set text(value) {}
```

#### Property Value

*String*

A string containing path's text.

## See Also

Path Members
Path Class
MindFusion.Drawing Namespace

# Pen Class

Represents a pen.

## Syntax

```
JavaScript
// class
Pen.prototype = {}
```

## Inheritance Hierarchy

## See Also

Pen Members
MindFusion.Drawing Namespace

### Pen Members

Represents a pen.

The following tables list the members exposed by the Pen type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | Pen | Overloaded. |

## Public Properties

| | Name | Description |
|---|---|---|
| | color | Gets or sets the color of this Pen. |
| | dashStyle | Gets or sets the dash style of this Pen. |
| | width | Gets or sets the width of this Pen. |

## Public Methods

| | Name | Description |
|---|---|---|
| | apply | Applies the pen properties to a CanvasRenderingContext2D drawing object. |
| | equals | Determines whether the specified object is equal to this LinearGradientBrush. |
| S | fromObject | Returns a Pen from a string describing this pen. |
| | toString | Returns a string representation of this pen. |

## See Also

Pen Class
MindFusion.Drawing Namespace

# Pen Constructor

## Overload List

| Name | Description |
|---|---|
| Pen | Initializes a new instance of the Pen class. |
| Pen | Initializes a new instance of the **Pen** class. |

## See Also

Pen Class
MindFusion.Drawing Namespace

# Pen Constructor

Initializes a new instance of the Pen class.

**Namespace:** MindFusion.Drawing
**File:** Pen.js

## Syntax

**JavaScript**

```
function Pen (brush, [width])
```

### Parameters

*brush*
Type: Brush | Color | string
The color of this pen.

*width*
Optional.
Type: Number
The width of this pen.

## Example

The following code customizes the ToolTip-s rendered over chart elements:

**JavaScript**

```
// customize tooltips
ToolTip.position = new Point (-5, -3);
ToolTip.text = "Click on a bar to see detailed chart";
ToolTip.brush = new Drawing.Brush("#fafafa");
ToolTip.pen = new Drawing.Pen("#9caac6");
```

## See Also

Pen Constructor Overload List
Pen Class
MindFusion.Drawing Namespace

# Pen Constructor

Initializes a new instance of the Pen class.

**Namespace:** MindFusion.Drawing
**File:** Pen.js

## Syntax

**JavaScript**

```
function Pen (brush, [width])
```

### Parameters

*brush*
Type: Brush, Color, string
The color of this pen.

*width*
Optional.
Type: Number
The width of this pen.

## Example

The following code customizes the ToolTip-s rendered over chart elements:

**JavaScript**

```
// customize tooltips
ToolTip.position = new Point (-5, -3);
ToolTip.text = "Click on a bar to see detailed chart";
ToolTip.brush = new Drawing.Brush("#fafafa");
ToolTip.pen = new Drawing.Pen("#9caac6", 2);
```

## See Also

Pen Constructor Overload List
Pen Class
MindFusion.Drawing Namespace

# Pen Methods

# Pen.apply Method

Applies the pen properties to a CanvasRenderingContext2D drawing object.

**Namespace:** MindFusion.Drawing
**File:** Pen.js

## Syntax

**JavaScript**

```
function apply (context)
```

### Parameters

*context*
Type: CanvasRenderingContext2D
The CanvasRenderingContext2D drawing object.

## See Also

Pen Members
Pen Class
MindFusion.Drawing Namespace

# Pen.equals Method

Determines whether the specified object is equal to this LinearGradientBrush.

**Namespace:** MindFusion.Drawing
**File:** Pen.js

## Syntax

**JavaScript**

```
function equals (The)
```

### Parameters

*The*
Type: Object
object to compare with.

### Return Value

Type: Boolean
true if the specified object is equal to this LinearGradientBrush; otherwise, false.

## See Also

Pen Members
Pen Class
MindFusion.Drawing Namespace

# Pen.fromObject Method

Returns a Pen from a string describing this pen.

**Namespace:** MindFusion.Drawing
**File:** Pen.js

## Syntax

**JavaScript**

```
function fromObject (obj)
```

### Parameters

*obj*
Type: void

### Return Value

Type: Pen
The pen object.

## Remarks

Used for backward compatibility with the JSLib class.

## See Also

Pen Members
Pen Class
MindFusion.Drawing Namespace

# Pen.toString Method

Returns a string representation of this pen.

**Namespace:** MindFusion.Drawing
**File:** Pen.js

## Syntax

```
JavaScript
function toString ()
```

### Return Value

Type: String
The string representation of this pen.

## Remarks

Used for backward compatibility with the JSLib class.

## See Also

Pen Members
Pen Class
MindFusion.Drawing Namespace

# Pen Properties

# Pen.color Property

Gets or sets the color of this Pen.

**Namespace:** MindFusion.Drawing
**File:** Pen.js

## Syntax

```
JavaScript
get color() {}
```

### Property Value

Type: Color

## See Also

Pen Members
Pen Class
MindFusion.Drawing Namespace

# Pen.dashStyle Property

Gets or sets the dash style of this Pen.

**Namespace:** MindFusion.Drawing
**File:** Pen.js

## Syntax

```
```

JavaScript

```
get dashStyle() {}
```

### Property Value

Type: DashStyle

## See Also

Pen Members
Pen Class
MindFusion.Drawing Namespace

# Pen.width Property

Gets or sets the width of this Pen.

**Namespace:** MindFusion.Drawing
**File:** Pen.js

## Syntax

JavaScript

```
get width() {}
```

### Property Value

Type: Number

## See Also

Pen Members
Pen Class
MindFusion.Drawing Namespace

# Point Class

Represents a point.

## Syntax

JavaScript

```
// class
Point.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Drawing.Point**

## See Also

# Point Members

Represents a point.

The following tables list the members exposed by the Point type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | Point | Initializes a new instance of the **Point** class with the specified X and Y coordinates. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | x | Gets or sets the x-coordinate of the point. |
|  | y | Gets or sets the y-coordinate of the point. |

## Public Methods

| | Name | Description |
|---|---|---|
|  | empty | Gets a value indicating whether this **Point** is empty. |
|  | equals | Specifies whether this **Point** contains the same coordinates as the specified point. |

## See Also

Point Class
MindFusion.Drawing Namespace

# Point Constructor

Initializes a new instance of the Point class with the specified X and Y coordinates.

**Namespace:** MindFusion.Drawing
**File:** Point.js

## Syntax

| JavaScript |
|---|
| `function Point (x, y)` |

### Parameters

*x*
The X-coordinate of the Point.

*y*
The Y-coordinate of the **Point**.

## See Also

Point Class
MindFusion.Drawing Namespace

# Point Methods

## Point.empty Method

Gets a value indicating whether this Point is empty.

**Namespace:** MindFusion.Drawing
**File:** Point.js

## Syntax

```
JavaScript
function empty ()
```

### Return Value

**true** if both x and y are 0; otherwise, **false**.

## See Also

Point Members
Point Class
MindFusion.Drawing Namespace

## Point.equals Method

Specifies whether this Point contains the same coordinates as the specified point.

**Namespace:** MindFusion.Drawing
**File:** Point.js

## Syntax

```
JavaScript
function equals ([point])
```

### Parameters

*point*
Optional. The Point to test.

### Return Value

**true** if *point* is a Point and has the same coordinates as this **Point**.

## See Also

Point Members
Point Class
MindFusion.Drawing Namespace

# Point Properties

## Point.x Property

Gets or sets the x-coordinate of the point.

**Namespace:** MindFusion.Drawing
**File:** Point.js

### Syntax

```
JavaScript
get x() {}
set x(value) {}
```

#### Property Value

Type: Number
*Number*

The x-coordinate of the point.

### See Also

Point Members
Point Class
MindFusion.Drawing Namespace

## Point.y Property

Gets or sets the y-coordinate of the point.

**Namespace:** MindFusion.Drawing
**File:** Point.js

### Syntax

```
JavaScript
get y() {}
set y(value) {}
```

#### Property Value

Type: Number
*Number*

The y-coordinate of the point.

### See Also

Point Members
Point Class

MindFusion.Drawing Namespace

# Rect Class

Represents a rectangle.

## Syntax

```
JavaScript
// class
Rect.prototype = {}
```

## Inheritance Hierarchy

MindFusion.Drawing.ComponentBase
**MindFusion.Drawing.Rect**

## See Also

Rect Members
MindFusion.Drawing Namespace

# Rect Members

Represents a rectangle.

The following tables list the members exposed by the Rect type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | Rect | Overloaded. |

## Public Properties

| | Name | Description |
|---|---|---|
| | height | Gets or sets the height of the rectangle. |
| | visibility | Gets or sets the visibility of this component. (Inherited from ComponentBase.) |
| | width | Gets or sets the width of the rectangle. |
| | x | Gets or sets the x-coordinate of the upper-left corner of the rectangle. |
| | y | Gets or sets the y-coordinate of the upper-left corner of the rectangle. |

## Public Methods

| | Name | Description |
|---|---|---|

| | | |
|---|---|---|
| | bottom | Gets the y-coordinate that is the sum of the y and height values of this **Rect** object. |
| | bottomLeft | Gets the bottom-left edge of this **Rect** object. |
| | bottomRight | Gets the bottom-right edge of this **Rect** object. |
| | center | Gets the center of this **Rect** object. |
| | clone | Creates a **Rect** object identical to the current object. |
| | contains | Determines if the rectangular region represented by *rect* is entirely contained within this **Rect** object. |
| | containsPoint | Determines if the specified point is contained within this **Rect** object. |
| | equals | Specifies whether this Rect contains the same coordinates as the specified rectangle. |
| S | fromLTRB | Creates a Rect object with the specified edge locations. |
| | intersect | Returns a **Rect** object representing the intersection of the current rectangle with the specified rectangle. |
| | intersectsWith | Checks if this **Rect** intersects with the specified rect. |
| | isEmpty | Gets a value indicating whether this Rect is empty. |
| | left | Gets the x-coordinate of the left edge of this **Rect** object. |
| | right | Gets the x-coordinate that is the sum of x and width values of this **Rect**. |
| | setCenter | Sets the center of this **Rect** object. |
| | setLocation | Sets the top-left edge of this **Rect** object. |
| | top | Gets the y-coordinate of the top edge of this **Rect** object. |
| | topLeft | Gets the top-left edge of this **Rect**. |
| | topMiddle | Gets the top-middle of this **Rect** object. |
| | topRight | Gets the top-right edge of this **Rect** object. |
| | union | Returns a **Rect** object representing the union of the current rectangle with the specified rectangle. |

## See Also

Rect Class
MindFusion.Drawing Namespace

# Rect Constructor

## Overload List

| Name | Description |
|---|---|
| | |

| | |
|---|---|
| Rect | Initializes a new instance of the Rect class. |
| Rect | Initializes a new instance of the Rect class. |

## See Also

Rect Class
MindFusion.Drawing Namespace

# Rect Constructor

Initializes a new instance of the Rect class.

**Namespace:** MindFusion.Drawing
**File:** Rect.js

## Syntax

**JavaScript**

```
function Rect (x, y, [width, [height]])
```

### Parameters

*x*
Type: Number | Point
Number | Point. The X-coordinate of the top left corner of the rectangle.

*y*
Type: Number | Size
Number | Size. The Y-coordinate of the top left corner of the rectangle.

*width*
Optional.
Type: Number
Number. The width of the rectangle.

*height*
Optional.
Type: Number
Number. The height of the rectangle.

## See Also

Rect Constructor Overload List
Rect Class
MindFusion.Drawing Namespace

# Rect Constructor

Initializes a new instance of the Rect class.

**Namespace:** MindFusion.Drawing
**File:** Rect.js

## Syntax

JavaScript

```
function Rect (x, y, width, height)
```

### Parameters

*x*
Type: Number
The X-coordinate of the top left corner of the rectangle.

*y*
Type: Number
The Y-coordinate of the top left corner of the rectangle.

*width*
Type: Number
The width of the rectangle.

*height*
Type: Number
The height of the rectangle.

## See Also

Rect Constructor Overload List
Rect Class
MindFusion.Drawing Namespace

# Rect Methods

# Rect.bottom Method

Gets the y-coordinate that is the sum of the y and height values of this Rect object.

**Namespace:** MindFusion.Drawing
**File:** Rect.js

## Syntax

JavaScript

```
function bottom ()
```

### Return Value

The y-coordinate that is the sum of y and height of this Rect.

## See Also

Rect Members
Rect Class
MindFusion.Drawing Namespace

# Rect.bottomLeft Method

Gets the bottom-left edge of this Rect object.

**Namespace:** MindFusion.Drawing
**File:** Rect.js

## Syntax

```
JavaScript
function bottomLeft ()
```

### Return Value

The bottom-left edge, which is a Point with x that is equal to x and with y that is the sum of y and height.

## See Also

Rect Members
Rect Class
MindFusion.Drawing Namespace

# Rect.bottomRight Method

Gets the bottom-right edge of this Rect object.

**Namespace:** MindFusion.Drawing
**File:** Rect.js

## Syntax

```
JavaScript
function bottomRight ()
```

### Return Value

The bottom-right edge, which is a Point with x that is the sum of x and width and with y that is the sum of y and height.

## See Also

Rect Members
Rect Class
MindFusion.Drawing Namespace

# Rect.center Method

Gets the center of this Rect object.

**Namespace:** MindFusion.Drawing
**File:** Rect.js

## Syntax

```
JavaScript
function center ()
```

### Return Value

The center, which is a Point with x that is the sum of x and half of width and with y that is the sum of y and half of height.

## See Also

Rect Members
Rect Class
MindFusion.Drawing Namespace

# Rect.clone Method

Creates a Rect object identical to the current object.

**Namespace:** MindFusion.Drawing
**File:** Rect.js

## Syntax

**JavaScript**

```
function clone ()
```

### Return Value

Type: Rect
Rect. The newly created Rect object.

## See Also

Rect Members
Rect Class
MindFusion.Drawing Namespace

# Rect.contains Method

Determines if the rectangular region represented by *rect* is entirely contained within this Rect object.

**Namespace:** MindFusion.Drawing
**File:** Rect.js

## Syntax

**JavaScript**

```
function contains (rect)
```

### Parameters

*rect*
The Rect to test.

### Return Value

**true** if the rectangular region represented by *rect* is entirely contained within this **Rect**; otherwise, **false**.

## See Also

Rect Members
Rect Class
MindFusion.Drawing Namespace

# Rect.containsPoint Method

Determines if the specified point is contained within this Rect object.

**Namespace:** MindFusion.Drawing
**File:** Rect.js

## Syntax

**JavaScript**

```
function containsPoint (point)
```

### Parameters

*point*
Type: Point
Point. The point to check.

### Return Value

**true** if point is contained within the Rect; otherwise, **false**.

## See Also

Rect Members
Rect Class
MindFusion.Drawing Namespace

# Rect.equals Method

Specifies whether this Rect contains the same coordinates as the specified rectangle.

**Namespace:** MindFusion.Drawing
**File:** Rect.js

## Syntax

**JavaScript**

```
function equals (rect)
```

### Parameters

*rect*
Type: Rect
Rect. The Rect to test.

### Return Value

Type: Boolean
Boolean. true if rect has the same coordinates as this Rect.

## See Also

Rect Members
Rect Class
MindFusion.Drawing Namespace

# Rect.fromLTRB Method

Creates a Rect object with the specified edge locations.

**Namespace:** MindFusion.Drawing
**File:** Rect.js

## Syntax

**JavaScript**

```
function fromLTRB (l, t, r, b)
```

### Parameters

*l*
Type: Number
Number. The x-coordinate of the upper-left corner of this Rect.

*t*
Type: Number
Number. The y-coordinate of the upper-left corner of this Rect.

*r*
Type: Number
Number. The x-coordinate of the lower-right corner of this Rect.

*b*
Type: Number
Number. The y-coordinate of the lower-right corner of this Rect.

### Return Value

Type: Rect
Rect. A rectangle with the specified coordinates.

## See Also

Rect Members
Rect Class
MindFusion.Drawing Namespace

# Rect.intersect Method

Returns a Rect object representing the intersection of the current rectangle with the specified rectangle.

**Namespace:** MindFusion.Drawing
**File:** Rect.js

## Syntax

JavaScript

```
function intersect (rect)
```

### Parameters

*rect*
Type: Rect
Rect. The rectangle to intersect.

### Return Value

A Rect object representing the intersection of the two rectangles.

## See Also

Rect Members
Rect Class
MindFusion.Drawing Namespace

# Rect.intersectsWith Method

Checks if this Rect intersects with the specified rect.

**Namespace:** MindFusion.Drawing
**File:** Rect.js

## Syntax

JavaScript

```
function intersectsWith (rect)
```

### Parameters

*rect*
The other Rect.

### Return Value

**true** if this **Rect** and the specified **Rect** intersect; otherwise, **false**.

## See Also

Rect Members
Rect Class
MindFusion.Drawing Namespace

# Rect.isEmpty Method

Gets a value indicating whether this Rect is empty.

**Namespace:** MindFusion.Drawing
**File:** Rect.js

## Syntax

JavaScript

```
function isEmpty ()
```

### Return Value

**true** if both width and height are 0; otherwise, **false**.

## See Also

Rect Members
Rect Class
MindFusion.Drawing Namespace

# Rect.left Method

Gets the x-coordinate of the left edge of this Rect object.

**Namespace:** MindFusion.Drawing
**File:** Rect.js

## Syntax

JavaScript

```
function left ()
```

### Return Value

The x-coordinate of the left edge of this Rect object.

## See Also

Rect Members
Rect Class
MindFusion.Drawing Namespace

# Rect.right Method

Gets the x-coordinate that is the sum of x and width values of this Rect.

**Namespace:** MindFusion.Drawing
**File:** Rect.js

## Syntax

JavaScript

```
function right ()
```

### Return Value

The x-coordinate that is the sum of x and width of this rectangle.

## See Also

Rect Members
Rect Class
MindFusion.Drawing Namespace

## Rect.setCenter Method

Sets the center of this Rect object.

**Namespace:** MindFusion.Drawing
**File:** Rect.js

### Syntax

```
JavaScript
function setCenter (point)
```

#### Parameters

*point*
The center of this Rect.

### See Also

Rect Members
Rect Class
MindFusion.Drawing Namespace

## Rect.setLocation Method

Sets the top-left edge of this Rect object.

**Namespace:** MindFusion.Drawing
**File:** Rect.js

### Syntax

```
JavaScript
function setLocation (point)
```

#### Parameters

*point*
The top-left edge of the Rect.

### See Also

Rect Members
Rect Class
MindFusion.Drawing Namespace

## Rect.top Method

Gets the y-coordinate of the top edge of this Rect object.

**Namespace:** MindFusion.Drawing
**File:** Rect.js

## Syntax

```
JavaScript
function top ()
```

### Return Value

Type: Number
Number. The y-coordinate of the top edge of this Rect object.

## See Also

Rect Members
Rect Class
MindFusion.Drawing Namespace

# Rect.topLeft Method

Gets the top-left edge of this Rect.

**Namespace:** MindFusion.Drawing
**File:** Rect.js

## Syntax

```
JavaScript
function topLeft ()
```

### Return Value

The top-left edge of this Rect.

## See Also

Rect Members
Rect Class
MindFusion.Drawing Namespace

# Rect.topMiddle Method

Gets the top-middle of this Rect object.

**Namespace:** MindFusion.Drawing
**File:** Rect.js

## Syntax

```
JavaScript
function topMiddle ()
```

### Return Value

The top-middle coordinate, which is a Point with x that is the sum of x and half of width and with y equal to y.

## See Also

Rect Members
Rect Class
MindFusion.Drawing Namespace

# Rect.topRight Method

Gets the top-right edge of this Rect object.

**Namespace:** MindFusion.Drawing
**File:** Rect.js

## Syntax

**JavaScript**

```
function topRight ()
```

### Return Value

The top-right edge, which is a Point with x that is the sum of x and width and with y equal to y.

## See Also

Rect Members
Rect Class
MindFusion.Drawing Namespace

# Rect.union Method

Returns a Rect object representing the union of the current rectangle with the specified rectangle.

**Namespace:** MindFusion.Drawing
**File:** Rect.js

## Syntax

**JavaScript**

```
function union (rect)
```

### Parameters

*rect*
Type: Rect
Rect. The rectangle to union.

### Return Value

A Rect object that bounds the union of the two rectangles.

## See Also

Rect Members
Rect Class
MindFusion.Drawing Namespace

# Rect Properties

# Rect.height Property

Gets or sets the height of the rectangle.

**Namespace:** MindFusion.Drawing
**File:** Rect.js

## Syntax

```
JavaScript
get height() {}
set height(value) {}
```

### Property Value

*Number*

The height of the rectangle.

## See Also

Rect Members
Rect Class
MindFusion.Drawing Namespace

# Rect.width Property

Gets or sets the width of the rectangle.

**Namespace:** MindFusion.Drawing
**File:** Rect.js

## Syntax

```
JavaScript
get width() {}
set width(value) {}
```

### Property Value

*Number*

The width of the rectangle.

## See Also

Rect Members
Rect Class
MindFusion.Drawing Namespace

# Rect.x Property

Gets or sets the x-coordinate of the upper-left corner of the rectangle.

**Namespace:** MindFusion.Drawing
**File:** Rect.js

## Syntax

**JavaScript**

```
get x() {}
set x(value) {}
```

### Property Value

*Number*

The x-coordinate of the upper-left corner of the rectangle.

## See Also

Rect Members
Rect Class
MindFusion.Drawing Namespace

# Rect.y Property

Gets or sets the y-coordinate of the upper-left corner of the rectangle.

**Namespace:** MindFusion.Drawing
**File:** Rect.js

## Syntax

**JavaScript**

```
get y() {}
set y(value) {}
```

### Property Value

*Number*

The x-coordinate of the upper-left corner of the rectangle.

## See Also

Rect Members
Rect Class
MindFusion.Drawing Namespace

# Size Class

Represents the size of 2D object.

## Syntax

```
// class
Size.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Drawing.Size**

## See Also

Size Members
MindFusion.Drawing Namespace

# Size Members

Represents the size of 2D object.

The following tables list the members exposed by the Size type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | Size | Initializes a new instance of the **Size** class with the specified width and height. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | height | Gets or sets the size height. |
|  | width | Gets or sets the size width. |

## Public Methods

| | Name | Description |
|---|---|---|
|  | empty | Gets a value indicating whether this Size is empty. |

## See Also

Size Class
MindFusion.Drawing Namespace

# Size Constructor

Initializes a new instance of the Size class with the specified width and height.

**Namespace:** MindFusion.Drawing
**File:** Size.js

## Syntax

```
JavaScript
function Size (width, height)
```

### Parameters

*width*
Type: Number
Number. Specifies width.

*height*
Type: Number
Number. Specifies height.

## See Also

Size Class
MindFusion.Drawing Namespace

# Size Methods

# Size.empty Method

Gets a value indicating whether this Size is empty.

**Namespace:** MindFusion.Drawing
**File:** Size.js

## Syntax

```
JavaScript
function empty ()
```

### Return Value

Type: Boolean
Boolean. true if both width and height are 0; otherwise, false.

## See Also

Size Members
Size Class
MindFusion.Drawing Namespace

# Size Properties

## Size.height Property

Gets or sets the size height.

**Namespace:** MindFusion.Drawing
**File:** Size.js

### Syntax

```
JavaScript
get height() {}
set height(value) {}
```

#### Property Value

Type: Number
*Number*

The size height.

### See Also

Size Members
Size Class
MindFusion.Drawing Namespace

## Size.width Property

Gets or sets the size width.

**Namespace:** MindFusion.Drawing
**File:** Size.js

### Syntax

```
JavaScript
get width() {}
set width(value) {}
```

#### Property Value

Type: Number
*Number*

The size width.

### See Also

Size Members
Size Class
MindFusion.Drawing Namespace

# StringAlignment Enumeration

Specifies the alignment of a text string relative to its layout rectangle.

**Namespace:** MindFusion.Drawing
**File:** Enum.js

## Syntax

```
JavaScript
// enum
StringAlignment = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **Center** | The text is drawn in the center of the layout rectangle. |
| | **Far** | The text is drawn in the far corner of the layout rectangle.le. |
| | **Near** | The text is drawn in the near corner of the layout rectangle. |

## See Also

MindFusion.Drawing Namespace

# StringFormat Class

Encapsulates text layout information.

## Syntax

```
JavaScript
// class
StringFormat.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Drawing.StringFormat**

## See Also

StringFormat Members
MindFusion.Drawing Namespace

# StringFormat Members

Encapsulates text layout information.

The following tables list the members exposed by the StringFormat type.

## Public Constructors

| | Name | Description |
|---|---|---|

| | | |
|---|---|---|
| | StringFormat | Initializes a new instance of the StringFormat class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | alignment | Gets or sets the horizontal alignment of the text. |
| | lineAlignment | Gets or sets the vertical alignment of the text. |

## See Also

StringFormat Class
MindFusion.Drawing Namespace

# StringFormat Constructor

Initializes a new instance of the StringFormat class.

**Namespace:** MindFusion.Drawing
**File:** Enum.js

## Syntax

**JavaScript**

```
function StringFormat ()
```

## See Also

StringFormat Class
MindFusion.Drawing Namespace

# StringFormat Properties

# StringFormat.alignment Property

Gets or sets the horizontal alignment of the text.

**Namespace:** MindFusion.Drawing
**File:** Enum.js

## Syntax

**JavaScript**

```
get alignment() {}
set alignment(value) {}
```

### Property Value

Type: StringAlignment
StringAlignment

## See Also

StringFormat Members
StringFormat Class
MindFusion.Drawing Namespace

### StringFormat.lineAlignment Property

Gets or sets the vertical alignment of the text.

**Namespace:** MindFusion.Drawing
**File:** Enum.js

## Syntax

**JavaScript**

```
get lineAlignment() {}
set lineAlignment(value) {}
```

#### Property Value

Type: StringAlignment
StringAlignment

## See Also

StringFormat Members
StringFormat Class
MindFusion.Drawing Namespace

# Text Class

Represents a text container.

## Syntax

**JavaScript**

```
// class
Text.prototype = {}
```

## Inheritance Hierarchy

MindFusion.Drawing.ComponentBase
**MindFusion.Drawing.Text**

## See Also

Text Members
MindFusion.Drawing Namespace

# Text Members

Represents a text container.

The following tables list the members exposed by the Text type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | Text | Initializes a new instance of the **Text** class. |

## Public Fields

| | Name | Description |
|---|---|---|
| | bounds | The object's bounds. |

## Public Properties

| | Name | Description |
|---|---|---|
| | font | Gets or sets the font used to render text. |
| | rotationAngle | Gets a value indicating the rotation of the text container. |
| | text | Gets or sets the text to render. |
| | visibility | Gets or sets the visibility of this component. (Inherited from ComponentBase.) |

## Public Methods

| | Name | Description |
|---|---|---|
| | clone | Creates a **Text** object identical to the current object. |
| | getBounds | Gets a value indicating the bounding rectangle of the text container. |
| | setBounds | Sets a value indicating the bounding rectangle of the text container. |

## See Also

Text Class
MindFusion.Drawing Namespace

# Text Fields

## Text.bounds Field

The object's bounds.

**Namespace:** MindFusion.Drawing
**File:** Text.js

## Syntax

```
JavaScript
var bounds
```

## See Also

Text Members
Text Class
MindFusion.Drawing Namespace

# Text Constructor

Initializes a new instance of the Text class.

**Namespace:** MindFusion.Drawing
**File:** Text.js

## Syntax

```
JavaScript
function Text (text, bounds)
```

### Parameters

*text*

Type: String
String. The text to be displayed in the container.

*bounds*

Type: Rect
Rect. The bounds of the container.

## See Also

Text Class
MindFusion.Drawing Namespace

# Text Methods

# Text.clone Method

Creates a Text object identical to the current object.

**Namespace:** MindFusion.Drawing
**File:** Text.js

## Syntax

```
JavaScript
function clone ()
```

### Return Value

The newly created Text object.

## See Also

Text Members
Text Class
MindFusion.Drawing Namespace

# Text.getBounds Method

Gets a value indicating the bounding rectangle of the text container.

**Namespace:** MindFusion.Drawing
**File:** Text.js

## Syntax

**JavaScript**

```
function getBounds ()
```

### Return Value

Type: Rect
Rect. The bounding rectangle.

## See Also

Text Members
Text Class
MindFusion.Drawing Namespace

# Text.setBounds Method

Sets a value indicating the bounding rectangle of the text container.

**Namespace:** MindFusion.Drawing
**File:** Text.js

## Syntax

**JavaScript**

```
function setBounds (bounds, [angle])
```

### Parameters

*bounds*
Type: Rect
Rect. The bounding rectangle.

*angle*
Optional.
Type: Number
Number. The rotation angle to be applied to the bounding rectangle.

## See Also

Text Members
Text Class
MindFusion.Drawing Namespace

# Text Properties

## Text.font Property

Gets or sets the font used to render text.

**Namespace:** MindFusion.Drawing
**File:** Text.js

### Syntax

```
JavaScript
get font() {}
set font(value) {}
```

#### Property Value

*Font*

The Font of this text.

### See Also

Text Members
Text Class
MindFusion.Drawing Namespace

## Text.rotationAngle Property

Gets a value indicating the rotation of the text container.

**Namespace:** MindFusion.Drawing
**File:** Text.js

### Syntax

```
JavaScript
get rotationAngle() {}
set rotationAngle(value) {}
```

#### Property Value

*Number*

The angle.

### See Also

Text Members
Text Class
MindFusion.Drawing Namespace

## Text.text Property

Gets or sets the text to render.

**Namespace:** MindFusion.Drawing
**File:** Text.js

### Syntax

```
JavaScript
get text() {}
set text(value) {}
```

#### Property Value

*String*

The text.

### See Also

Text Members
Text Class
MindFusion.Drawing Namespace

# Thickness Class

Describes the thickness of a rectangular frame.

### Syntax

```
JavaScript
// class
Thickness.prototype = {}
```

### Inheritance Hierarchy

**MindFusion.Drawing.Thickness**

### See Also

Thickness Members
MindFusion.Drawing Namespace

## Thickness Members

Describes the thickness of a rectangular frame.

The following tables list the members exposed by the Thickness type.

### Public Constructors

| | Name | Description |
|---|---|---|

| | | |
|---|---|---|
| | Thickness | Overloaded. |

## Public Properties

| | Name | Description |
|---|---|---|
| | bottom | Gets or sets the width of the bottom side of the frame. |
| | isRelative | Gets or sets a flag indicating whether the thickness properties are expressed as relative or absolute quantities. |
| | left | Gets or sets the width of the left side of the frame. |
| | right | Gets or sets the width of the right side of the frame. |
| | top | Gets or sets the width of the top side of the frame. |

## Public Methods

| | Name | Description |
|---|---|---|
| | addToRect | Applies the current thickness to the specified rectangle by inflating the rectangle. |
| | applyTo | Applies the current thickness to the specified rectangle by deflating the rectangle. |
| | toAbsolute | Converts the current thickness to absolute value relative to the specified size. |

## See Also

Thickness Class
MindFusion.Drawing Namespace

# Thickness Constructor

## Overload List

| Name | Description |
|---|---|
| Thickness | Initializes a new instance of the Thickness class with the specified widths. |
| Thickness | Initializes a new instance of the Thickness class with the specified widths. |

## See Also

Thickness Class
MindFusion.Drawing Namespace

## Thickness Constructor

Initializes a new instance of the Thickness class with the specified widths.

**Namespace:** MindFusion.Drawing
**File:** Thickness.js

## Syntax

**JavaScript**

```
function Thickness (left, top, right, bottom)
```

### Parameters

*left*
Type: Number
Specifies the width of the left side of the frame.

*top*
Type: Number
Specifies the width of the top side of the frame.

*right*
Type: Number
Specifies the width of the right side of the frame.

*bottom*
Type: Number
Specifies the width of the bottom side of the frame.

## See Also

Thickness Constructor Overload List
Thickness Class
MindFusion.Drawing Namespace

## Thickness Constructor

Initializes a new instance of the Thickness class with the specified widths.

**Namespace:** MindFusion.Drawing
**File:** Thickness.js

## Syntax

**JavaScript**

```
function Thickness (left, [top, [right, [bottom, [isRelative]]]])
```

### Parameters

*left*
Type: Number
Number. Specifies the width of the left side of the frame.

*top*
Optional.
Type: Number
Number. Specifies the width of the top side of the frame.

*right*
Optional.
Type: Number
Number. Specifies the width of the right side of the frame.

*bottom*
Optional.
Type: Number
Number. Specifies the width of the bottom side of the frame.

*isRelative*
Optional.
Type: Boolean
Boolean. A flag indicating whether the thickness properties are expressed as relative or absolute quantities.

## See Also

Thickness Constructor Overload List
Thickness Class
MindFusion.Drawing Namespace

# Thickness Methods

# Thickness.addToRect Method

Applies the current thickness to the specified rectangle by inflating the rectangle.

**Namespace:** MindFusion.Drawing
**File:** Thickness.js

## Syntax

**JavaScript**

```
function addToRect (rect)
```

### Parameters

*rect*
Type: Rect
Rect. The rectangle to inflate.

## See Also

Thickness Members
Thickness Class
MindFusion.Drawing Namespace

# Thickness.applyTo Method

Applies the current thickness to the specified rectangle by deflating the rectangle.

**Namespace:** MindFusion.Drawing
**File:** Thickness.js

## Syntax

JavaScript

```
function applyTo (rect)
```

### Parameters

*rect*
Type: Rect
Rect. The rectangle to deflate.

## See Also

Thickness Members
Thickness Class
MindFusion.Drawing Namespace

# Thickness.toAbsolute Method

Converts the current thickness to absolute value relative to the specified size.

**Namespace:** MindFusion.Drawing
**File:** Thickness.js

## Syntax

JavaScript

```
function toAbsolute ()
```

## See Also

Thickness Members
Thickness Class
MindFusion.Drawing Namespace

# Thickness Properties

# Thickness.bottom Property

Gets or sets the width of the bottom side of the frame.

**Namespace:** MindFusion.Drawing
**File:** Thickness.js

## Syntax

JavaScript

```
get bottom() {}
set bottom(value) {}
```

### Property Value

*Number*

The width of the bottom side of the frame.

## See Also

Thickness Members
Thickness Class
MindFusion.Drawing Namespace

# Thickness.isRelative Property

Gets or sets a flag indicating whether the thickness properties are expressed as relative or absolute quantities.

**Namespace:** MindFusion.Drawing
**File:** Thickness.js

## Syntax

**JavaScript**

```
get isRelative() {}
set isRelative(value) {}
```

### Property Value

*Boolean*

**true** if the thickness is relative, or **false** otherwise.

## See Also

Thickness Members
Thickness Class
MindFusion.Drawing Namespace

# Thickness.left Property

Gets or sets the width of the left side of the frame.

**Namespace:** MindFusion.Drawing
**File:** Thickness.js

## Syntax

**JavaScript**

```
get left() {}
set left(value) {}
```

### Property Value

*Number*

The width of the left side of the frame.

## See Also

Thickness Members
Thickness Class
MindFusion.Drawing Namespace

# Thickness.right Property

Gets or sets the width of the right side of the frame.

**Namespace:** MindFusion.Drawing
**File:** Thickness.js

## Syntax

**JavaScript**

```
get right() {}
set right(value) {}
```

### Property Value

*Number*

The width of the right side of the frame.

## See Also

Thickness Members
Thickness Class
MindFusion.Drawing Namespace

# Thickness.top Property

Gets or sets the width of the top side of the frame.

**Namespace:** MindFusion.Drawing
**File:** Thickness.js

## Syntax

**JavaScript**

```
get top() {}
set top(value) {}
```

### Property Value

*Number*

The width of the top side of the frame.

## See Also

Thickness Members
Thickness Class
MindFusion.Drawing Namespace

# Vector Class

Represents displacement in 2D space.

## Syntax

**JavaScript**

```
// class
Vector.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Drawing.Vector**

## See Also

Vector Members
MindFusion.Drawing Namespace

# Vector Members

Represents displacement in 2D space.

The following tables list the members exposed by the Vector type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | Vector | Initializes a new instance of the Vector class with the specified X and Y coordinates. |

## Public Fields

| | Name | Description |
|---|---|---|
| | x | The x-coordinate of the vector. |
| | y | The y-coordinate of the vector. |

## See Also

Vector Class
MindFusion.Drawing Namespace

# Vector Fields

## Vector.x Field

The x-coordinate of the vector.

**Namespace:** MindFusion.Drawing
**File:** Vector.js

## Syntax

```
JavaScript
var x
```

## See Also

Vector Members
Vector Class
MindFusion.Drawing Namespace

# Vector.y Field

The y-coordinate of the vector.

**Namespace:** MindFusion.Drawing
**File:** Vector.js

## Syntax

```
JavaScript
var y
```

## See Also

Vector Members
Vector Class
MindFusion.Drawing Namespace

# Vector Constructor

Initializes a new instance of the Vector class with the specified X and Y coordinates.

**Namespace:** MindFusion.Drawing
**File:** Vector.js

## Syntax

```
JavaScript
function Vector (x, y)
```

### Parameters

*x*
Type: Number
Number. The X-coordinate of the Vector.

*y*
Type: Number
Number. The Y-coordinate of the Vector.

## See Also

# Video Class

A component that displays video stream.

## Syntax

**JavaScript**

```
// class
Video.prototype = {}
```

## Remarks

The video does not play automatically. To start playing it, call the component's play method in response to user input events.

If autoProperty is enabled for a **Video** component in **CompositeNode** template, the template loader creates getter and setter wrapping the component's MediaLocation property.

## Example

The following template fragment adds a Video component to the node:

**JavaScript**

```
{
    component: "Video",
    name: "Video",
    autoProperty: true,
    mediaLocation: "video.webm",
    margin: "1"
},
```

It automatically adds getVideo and setVideo methods to the node class, which can be used to get or set the video stream URL:

**JavaScript**

```
var VideoNode = CompositeNode.classFromTemplate(
    "VideoNode", templateJson);
var node = new VideoNode();
node.setVideo("video2.webm");
diagram.addItem(node);
```

## Inheritance Hierarchy

MindFusion.Drawing.ComponentBase
  **MindFusion.Drawing.Video**

## See Also

Video Members
MindFusion.Drawing Namespace

# Video Members

A component that displays video stream.

The following tables list the members exposed by the Video type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | Video | Initializes a new instance of the Video class. |

## Public Fields

| | Name | Description |
|---|---|---|
| | bounds | The object's bounds. |

## Public Properties

| | Name | Description |
|---|---|---|
| | mediaLocation | Gets or sets URL of the video stream displayed in this component. |
| | visibility | Gets or sets the visibility of this component. (Inherited from ComponentBase.) |

## Public Methods

| | Name | Description |
|---|---|---|
| | clone | Creates a Video object identical to the current object. |
| | isPlaying | Detects if video is playing. |
| | pause | Pauses the video. |
| | play | Starts playing the video stream. |

## See Also

Video Class
MindFusion.Drawing Namespace

# Video Fields

## Video.bounds Field

The object's bounds.

**Namespace:** MindFusion.Drawing
**File:** Video.js

## Syntax

JavaScript

```
var bounds
```

## See Also

Video Members
Video Class
MindFusion.Drawing Namespace

# Video Constructor

Initializes a new instance of the Video class.

**Namespace:** MindFusion.Drawing
**File:** Video.js

## Syntax

JavaScript

```
function Video (bounds)
```

### Parameters

*bounds*
Type: Rect
Rect. The bounds of the Video.

## See Also

Video Class
MindFusion.Drawing Namespace

# Video Methods

# Video.clone Method

Creates a Video object identical to the current object.

**Namespace:** MindFusion.Drawing
**File:** Video.js

## Syntax

JavaScript

```
function clone ()
```

### Return Value

Type: Video

## See Also

Video Members
Video Class
MindFusion.Drawing Namespace

## Video.isPlaying Method

Detects if video is playing.

**Namespace:** MindFusion.Drawing
**File:** Video.js

### Syntax

```
JavaScript
function isPlaying ()
```

### See Also

Video Members
Video Class
MindFusion.Drawing Namespace

## Video.pause Method

Pauses the video.

**Namespace:** MindFusion.Drawing
**File:** Video.js

### Syntax

```
JavaScript
function pause ()
```

### See Also

Video Members
Video Class
MindFusion.Drawing Namespace

## Video.play Method

Starts playing the video stream.

**Namespace:** MindFusion.Drawing
**File:** Video.js

### Syntax

```
JavaScript
function play ()
```

## See Also

Video Members
Video Class
MindFusion.Drawing Namespace

# Video Properties

## Video.mediaLocation Property

Gets or sets URL of the video stream displayed in this component.

**Namespace:** MindFusion.Drawing
**File:** Video.js

## Syntax

| JavaScript |
| --- |
| `get mediaLocation() {}`<br>`set mediaLocation(value) {}` |

### Property Value

*String*

A URL string that identifies the video to play.

## See Also

Video Members
Video Class
MindFusion.Drawing Namespace

# Visibility Enumeration

Specifies visibility for components.

**Namespace:** MindFusion.Drawing
**File:** Enum.js

## Syntax

| JavaScript |
| --- |
| `// enum`<br>`Visibility = {}` |

## Members

| | Member name | Description |
| --- | --- | --- |
| | **Collapsed** | Specifies that the component is hidden, and the space it occupies in the layout is not preserved. |

| | | |
|---|---|---|
| | **Hidden** | Specifies that the component is hidden, but the space it occupies in the layout is preserved. |
| | **Visible** | Specifies that the component is visible. |

## See Also

MindFusion.Drawing Namespace

# MindFusion.Gauges Namespace

## Alignment Enumeration

Indicates an alignment of an object relative to the scale. In the descriptionsbelow 'outbound' refers to an element's outer outline, 'inbound' refers to anelement's inner outline and 'center' refers to en element's center, that is,the middle of the inbound and outbound.

**Namespace:** MindFusion.Gauges
**File:** Enum.js

## Syntax

**JavaScript**

```
// enum
Alignment = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **CenterInside** | Specifies that the outbound of an object is aligned with the center of the scale. |
| | **CenterOutside** | Specifies that the inbound of an object is aligned with the center of the scale. |
| | **InnerCenter** | Specifies that the center of an object is aligned with the inbound of the scale. |
| | **InnerInside** | Specifies that the outbound of an object is aligned with the inbound of the scale. |
| | **InnerOutside** | Specifies that the inbound of an object is aligned with the inbound of the scale. |
| | **OuterCenter** | Specifies that the center of an object is aligned with the outbound of the scale. |
| | **OuterInside** | Specifies that the outbound of an object is aligned with the outbound of the scale. |
| | **OuterOutside** | Specifies that the inbound of an object is aligned with the outbound of the scale. |
| | **TrueCenter** | Specifies that the center of an object is aligned with the center of the scale. |

## See Also

MindFusion.Gauges Namespace

# ArcArea Class

Represents an ellipse visual.

## Syntax

**JavaScript**

```
// class
ArcArea.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Gauges.ArcArea**

## See Also

ArcArea Members
MindFusion.Gauges Namespace

# ArcArea Members

Represents an ellipse visual.

The following tables list the members exposed by the ArcArea type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | ArcArea | Creates an instance of ArcArea class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | endAngle | Gets or sets the end angle of the arc. |
|  | startAngle | Gets or sets the start angle of the arc. |

## See Also

ArcArea Class
MindFusion.Gauges Namespace

# ArcArea Constructor

Creates an instance of **ArcArea** class.

**Namespace:** MindFusion.Gauges
**File:** ArcArea.js

## Syntax

```
JavaScript
function ArcArea ()
```

## See Also

ArcArea Class
MindFusion.Gauges Namespace

# ArcArea Properties

## ArcArea.endAngle Property

Gets or sets the end angle of the arc.

**Namespace:** MindFusion.Gauges
**File:** ArcArea.js

## Syntax

```
JavaScript
get endAngle() {}
set endAngle(value) {}
```

### Property Value

Type: Number
*Number*

The end angle of the arc.

## See Also

ArcArea Members
ArcArea Class
MindFusion.Gauges Namespace

## ArcArea.startAngle Property

Gets or sets the start angle of the arc.

**Namespace:** MindFusion.Gauges
**File:** ArcArea.js

## Syntax

```
JavaScript
get startAngle() {}
set startAngle(value) {}
```

### Property Value

Type: Number
*Number*

The start angle of the arc.

## See Also

ArcArea Members
ArcArea Class
MindFusion.Gauges Namespace

# ArcSegment Class

Represents an elliptical arc between two points.

## Syntax

| JavaScript |
| --- |
| `// class`<br>`ArcSegment.prototype = {}` |

## Inheritance Hierarchy

**MindFusion.Gauges.ArcSegment**

## See Also

ArcSegment Members
MindFusion.Gauges Namespace

# ArcSegment Members

Represents an elliptical arc between two points.

The following tables list the members exposed by the ArcSegment type.

## Public Constructors

| | Name | Description |
| --- | --- | --- |
|  | ArcSegment | Creates an instance of **ArcSegment** class. |

## Public Properties

| | Name | Description |
| --- | --- | --- |
|  | isCounterclockwise | Gets or sets a value that indicates whether the arc is drawn in counterclockwise or clockwise direction. |
|  | isLargeArc | Gets or sets a value that indicates whether the arc should be greater than 180 degrees. |

| | | |
|---|---|---|
| | point | Gets or sets the endpoint of the elliptical arc. |
| | rotationAngle | Gets or sets the amount (in degrees) by which the ellipse is rotated about the x-axis. |
| | size | Gets or sets the x- and y-radius of the arc as a Size structure. |

## Public Methods

| | Name | Description |
|---|---|---|
| | addTo | Segment.addTo override. |

## See Also

ArcSegment Class
MindFusion.Gauges Namespace

# ArcSegment Constructor

Creates an instance of ArcSegment class.

**Namespace:** MindFusion.Gauges
**File:** Segment.js

## Syntax

| JavaScript |
|---|
| `function ArcSegment ()` |

## See Also

ArcSegment Class
MindFusion.Gauges Namespace

# ArcSegment Methods

# ArcSegment.addTo Method

Segment.addTo override.

**Namespace:** MindFusion.Gauges
**File:** Segment.js

## Syntax

| JavaScript |
|---|
| `function addTo ()` |

## See Also

ArcSegment Members
ArcSegment Class
MindFusion.Gauges Namespace

# ArcSegment Properties

## ArcSegment.isCounterclockwise Property

Gets or sets a value that indicates whether the arc is drawn in counterclockwise or clockwise direction.

**Namespace:** MindFusion.Gauges
**File:** Segment.js

### Syntax

```
JavaScript
get isCounterclockwise() {}
set isCounterclockwise(value) {}
```

#### Property Value

Type: Boolean
*Boolean*

True if the arc is drawn in counterclockwise, otherwise false.

### See Also

ArcSegment Members
ArcSegment Class
MindFusion.Gauges Namespace

## ArcSegment.isLargeArc Property

Gets or sets a value that indicates whether the arc should be greater than 180 degrees.

**Namespace:** MindFusion.Gauges
**File:** Segment.js

### Syntax

```
JavaScript
get isLargeArc() {}
set isLargeArc(value) {}
```

#### Property Value

Type: Boolean
*Boolean*

True if the arc should be greater than 180 degrees, otherwise false.

### See Also

ArcSegment Members
ArcSegment Class

MindFusion.Gauges Namespace

# ArcSegment.point Property

Gets or sets the endpoint of the elliptical arc.

**Namespace:** MindFusion.Gauges
**File:** Segment.js

## Syntax

**JavaScript**

```
get point() {}
set point(value) {}
```

### Property Value

Type: Point
*Point*

The endpoint of the elliptical arc.

## See Also

ArcSegment Members
ArcSegment Class
MindFusion.Gauges Namespace

# ArcSegment.rotationAngle Property

Gets or sets the amount (in degrees) by which the ellipse is rotated about the x-axis.

**Namespace:** MindFusion.Gauges
**File:** Segment.js

## Syntax

**JavaScript**

```
get rotationAngle() {}
set rotationAngle(value) {}
```

### Property Value

Type: Number
*Number*

The amount (in degrees) by which the ellipse is rotated about the x-axis.

## See Also

ArcSegment Members
ArcSegment Class
MindFusion.Gauges Namespace

## ArcSegment.size Property

Gets or sets the x- and y-radius of the arc as a Size structure.

**Namespace:** MindFusion.Gauges
**File:** Segment.js

### Syntax

```
JavaScript

get size() {}
set size(value) {}
```

#### Property Value

Type: Size
*Size*

The x- and y-radius of the arc as a Size structure.

### See Also

ArcSegment Members
ArcSegment Class
MindFusion.Gauges Namespace

# BaseGauge Class

Represents the base class of linear and oval gauges.

### Syntax

```
JavaScript

// class
BaseGauge.prototype = {}
```

### Inheritance Hierarchy

**MindFusion.Gauges.BaseGauge**

### See Also

BaseGauge Members
MindFusion.Gauges Namespace

## BaseGauge Members

Represents the base class of linear and oval gauges.

The following tables list the members exposed by the BaseGauge type.

### Public Constructors

| | Name | Description |
|---|---|---|

| | | |
|---|---|---|
| | BaseGauge | Creates an instance of **BaseGauge**. |

## Public Properties

| | Name | Description |
|---|---|---|
| | autoPostBack | Gets or sets a value indicating whether the control will post back to the server when a pointer's value has changed. |
| | licenseLocation | Gets or sets the URL of the gauge's license file. |
| | scales | Returns the array of all scales in this gauge. |

## Public Methods

| | Name | Description |
|---|---|---|
| | addScale | Adds a scale to the gauge. |
| | fromJson | Deserializes the gauge from a JSON string. |
| | getElementByName | Returns the VisualElement object corresponding to the specified name. |
| | removeScale | Removes a scale from the gauge. |
| | toJson | Serializes the gauge into a JSON string. |

## See Also

BaseGauge Class
MindFusion.Gauges Namespace

# BaseGauge Fields

# BaseGauge Constructor

Creates an instance of BaseGauge.

**Namespace:** MindFusion.Gauges
**File:** BaseGauge.js

## Syntax

**JavaScript**

```
function BaseGauge (element)
```

### Parameters

*element*
Type: Object
Object. The Div DOM element this gauge is associated with.

## See Also

BaseGauge Class
MindFusion.Gauges Namespace

# BaseGauge Methods

## BaseGauge.addScale Method

Adds a scale to the gauge.

**Namespace:** MindFusion.Gauges
**File:** BaseGauge.js

### Syntax

**JavaScript**

```
function addScale (scale)
```

#### Parameters

*scale*
Type: BaseScale
BaseScale. The scale to add.

### See Also

BaseGauge Members
BaseGauge Class
MindFusion.Gauges Namespace

## BaseGauge.fromJson Method

Deserializes the gauge from a JSON string.

**Namespace:** MindFusion.Gauges
**File:** BaseGauge.js

### Syntax

**JavaScript**

```
function fromJson (json)
```

#### Parameters

*json*
Type: String
String. A string created by the toJson method.

### See Also

BaseGauge Members
BaseGauge Class
MindFusion.Gauges Namespace

# BaseGauge.getElementByName Method

Returns the VisualElement object corresponding to the specified name.

**Namespace:** MindFusion.Gauges
**File:** BaseGauge.js

## Syntax

**JavaScript**

```
function getElementByName (name)
```

### Parameters

*name*
Type: String
String. The name of the object to find.

## See Also

BaseGauge Members
BaseGauge Class
MindFusion.Gauges Namespace

# BaseGauge.removeScale Method

Removes a scale from the gauge.

**Namespace:** MindFusion.Gauges
**File:** BaseGauge.js

## Syntax

**JavaScript**

```
function removeScale (scale)
```

### Parameters

*scale*
Type: BaseScale
BaseScale. The scale to remove.

## See Also

BaseGauge Members
BaseGauge Class
MindFusion.Gauges Namespace

# BaseGauge.toJson Method

Serializes the gauge into a JSON string.

**Namespace:** MindFusion.Gauges
**File:** BaseGauge.js

## Syntax

```
JavaScript
function toJson ()
```

### Return Value

Type: String
String. A string containing the view's JSON representation.

## See Also

BaseGauge Members
BaseGauge Class
MindFusion.Gauges Namespace

# BaseGauge Properties

# BaseGauge.autoPostBack Property

Gets or sets a value indicating whether the control will post back to the server when a pointer's value has changed.

**Namespace:** MindFusion.Gauges
**File:** BaseGauge.js

## Syntax

```
JavaScript
get autoPostBack() {}
set autoPostBack(value) {}
```

### Property Value

Type: Boolean
*Boolean*

true if the control should post back, otherwise, false.

## See Also

BaseGauge Members
BaseGauge Class
MindFusion.Gauges Namespace

# BaseGauge.licenseLocation Property

Gets or sets the URL of the gauge's license file.

**Namespace:** MindFusion.Gauges
**File:** BaseGauge.js

## Syntax

JavaScript

```
get licenseLocation() {}
set licenseLocation(value) {}
```

### Property Value

Type: String
*String*

A valid URL string that specifies the location of the license key.

## See Also

BaseGauge Members
BaseGauge Class
MindFusion.Gauges Namespace

# BaseGauge.scales Property

Returns the array of all scales in this gauge.

**Namespace:** MindFusion.Gauges
**File:** BaseGauge.js

## Syntax

JavaScript

```
get scales() {}
set scales(value) {}
```

### Property Value

Type: Array
*Array*

Array of all BaseScale instances in the gauge.

## See Also

BaseGauge Members
BaseGauge Class
MindFusion.Gauges Namespace

# BaseScale Class

Represents a scale within a gauge control.

## Syntax

JavaScript

```
// class
BaseScale.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Gauges.BaseScale**

## See Also

BaseScale Members
MindFusion.Gauges Namespace

# BaseScale Members

Represents a scale within a gauge control.

The following tables list the members exposed by the BaseScale type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | BaseScale | Creates an instance of **BaseScale**. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | customFunction | Gets or sets the function to be used for distribution of values along the scale when functionType is set to Custom. |
|  | functionArgument | Gets or sets the argument passed to custom functions. |
|  | functionType | Gets or sets the type of the function used to calculate the distribution of values along the scale. |
|  | logarithmBase | Gets or sets the logarithm base when FunctionType is set to Logarithmic. |
|  | majorTickSettings | Gets the settings for the major ticks and their associated labels. |
|  | maxValue | Gets or sets the maximal value displayed by the scale. |
|  | middleTickSettings | Gets the settings for the middle ticks and their associated labels. |
|  | minorTickSettings | Gets the settings for the minor ticks and their associated labels. |
|  | minValue | Gets or sets the minimal value displayed by the scale. |
|  | pointers | Returns the array of all pointers in this scale. |
|  | ranges | Returns the array of all ranges in this scale. |
|  | reversedCustomFunction | Gets or sets the reversed version of the function specified through customFunction. |
|  | startWidth | Gets or sets the width of the scale at its beginning. |

## Public Methods

| | Name | Description |
|---|---|---|

| | | |
|---|---|---|
| | addPointer | Adds a Pointer object to the pointers collection. |
| | addRange | Adds a Range object to the ranges collection. |
| | removePointer | Removes a **Pointer** object from the pointers collection. |
| | removeRange | Removes a **Range** object from the ranges collection. |

## See Also

BaseScale Class
MindFusion.Gauges Namespace

# BaseScale Fields

# BaseScale Constructor

Creates an instance of BaseScale.

**Namespace:** MindFusion.Gauges
**File:** BaseScale.js

## Syntax

**JavaScript**

```
function BaseScale (parent)
```

### Parameters

*parent*
Type: BaseGauge
BaseGauge. The gauge control this scale belongs to.

## See Also

BaseScale Class
MindFusion.Gauges Namespace

# BaseScale Methods

# BaseScale.addPointer Method

Adds a Pointer object to the pointers collection.

**Namespace:** MindFusion.Gauges
**File:** BaseScale.js

## Syntax

**JavaScript**

```
function addPointer (pointer)
```

### Parameters

*pointer*
Pointer. The pointer to add.

## See Also

BaseScale Members
BaseScale Class
MindFusion.Gauges Namespace

# BaseScale.addRange Method

Adds a Range object to the ranges collection.

**Namespace:** MindFusion.Gauges
**File:** BaseScale.js

## Syntax

**JavaScript**

```
function addRange (range)
```

### Parameters

*range*
Range. The range to add.

## See Also

BaseScale Members
BaseScale Class
MindFusion.Gauges Namespace

# BaseScale.removePointer Method

Removes a Pointer object from the pointers collection.

**Namespace:** MindFusion.Gauges
**File:** BaseScale.js

## Syntax

**JavaScript**

```
function removePointer (pointer)
```

### Parameters

*pointer*
Pointer. The pointer to remove.

## See Also

BaseScale Members
BaseScale Class
MindFusion.Gauges Namespace

## BaseScale.removeRange Method

Removes a Range object from the ranges collection.

**Namespace:** MindFusion.Gauges
**File:** BaseScale.js

### Syntax

**JavaScript**

```
function removeRange (range)
```

#### Parameters

*range*
Range. The ranges to remove.

### See Also

BaseScale Members
BaseScale Class
MindFusion.Gauges Namespace

## BaseScale Properties

## BaseScale.customFunction Property

Gets or sets the function to be used for distribution of values along the scale when functionType is set to Custom.

**Namespace:** MindFusion.Gauges
**File:** BaseScale.js

### Syntax

**JavaScript**

```
get customFunction() {}
set customFunction(value) {}
```

#### Property Value

Type: FunctionDelegate
*FunctionDelegate*

The function to be used for distribution of values along the scale when functionType is set to Custom.

### See Also

BaseScale Members
BaseScale Class
MindFusion.Gauges Namespace

## BaseScale.functionArgument Property

Gets or sets the argument passed to custom functions.

**Namespace:** MindFusion.Gauges
**File:** BaseScale.js

## Syntax

```
JavaScript
get functionArgument() {}
set functionArgument(value) {}
```

### Property Value

Type: Number
*Number*

The argument passed to custom functions.

## See Also

BaseScale Members
BaseScale Class
MindFusion.Gauges Namespace

# BaseScale.functionType Property

Gets or sets the type of the function used to calculate the distribution of values along the scale.

**Namespace:** MindFusion.Gauges
**File:** BaseScale.js

## Syntax

```
JavaScript
get functionType() {}
set functionType(value) {}
```

### Property Value

Type: FunctionType
*FunctionType*

The type of the function used to calculate the distribution of values along the scale.

## See Also

BaseScale Members
BaseScale Class
MindFusion.Gauges Namespace

# BaseScale.logarithmBase Property

Gets or sets the logarithm base when FunctionType is set to Logarithmic.

**Namespace:** MindFusion.Gauges
**File:** BaseScale.js

## Syntax

```
JavaScript
get logarithmBase() {}
set logarithmBase(value) {}
```

### Property Value

Type: Number
*Number*

The logarithm base when functionType is set to Logarithmic.

## See Also

BaseScale Members
BaseScale Class
MindFusion.Gauges Namespace

# BaseScale.majorTickSettings Property

Gets the settings for the major ticks and their associated labels.

**Namespace:** MindFusion.Gauges
**File:** BaseScale.js

## Syntax

```
JavaScript
get majorTickSettings() {}
set majorTickSettings(value) {}
```

### Property Value

Type: MajorTickSettings
*MajorTickSettings*

The settings for the major ticks and their associated labels.

## See Also

BaseScale Members
BaseScale Class
MindFusion.Gauges Namespace

# BaseScale.maxValue Property

Gets or sets the maximal value displayed by the scale.

**Namespace:** MindFusion.Gauges
**File:** BaseScale.js

## Syntax

```
JavaScript
get maxValue() {}
set maxValue(value) {}
```

### Property Value

Type: Number
*Number*

The maximal value displayed by the scale.

## See Also

BaseScale Members
BaseScale Class
MindFusion.Gauges Namespace

# BaseScale.middleTickSettings Property

Gets the settings for the middle ticks and their associated labels.

**Namespace:** MindFusion.Gauges
**File:** BaseScale.js

## Syntax

**JavaScript**

```
get middleTickSettings() {}
set middleTickSettings(value) {}
```

### Property Value

Type: MiddleTickSettings
*MiddleTickSettings*

The settings for the middle ticks and their associated labels.

## See Also

BaseScale Members
BaseScale Class
MindFusion.Gauges Namespace

# BaseScale.minorTickSettings Property

Gets the settings for the minor ticks and their associated labels.

**Namespace:** MindFusion.Gauges
**File:** BaseScale.js

## Syntax

**JavaScript**

```
get minorTickSettings() {}
set minorTickSettings(value) {}
```

### Property Value

Type: MinorTickSettings
*MinorTickSettings*

The settings for the minor ticks and their associated labels.

## See Also

BaseScale Members
BaseScale Class
MindFusion.Gauges Namespace

# BaseScale.minValue Property

Gets or sets the minimal value displayed by the scale.

**Namespace:** MindFusion.Gauges
**File:** BaseScale.js

## Syntax

**JavaScript**

```
get minValue() {}
set minValue(value) {}
```

### Property Value

Type: Number
*Number*

The minimal value displayed by the scale.

## See Also

BaseScale Members
BaseScale Class
MindFusion.Gauges Namespace

# BaseScale.pointers Property

Returns the array of all pointers in this scale.

**Namespace:** MindFusion.Gauges
**File:** BaseScale.js

## Syntax

**JavaScript**

```
get pointers() {}
set pointers(value) {}
```

### Property Value

Type: Array
*Array*

Array of all Pointer instances in the scale.

## See Also

BaseScale Members
BaseScale Class
MindFusion.Gauges Namespace

# BaseScale.ranges Property

Returns the array of all ranges in this scale.

**Namespace:** MindFusion.Gauges
**File:** BaseScale.js

## Syntax

```
JavaScript
get ranges() {}
set ranges(value) {}
```

### Property Value

Type: Array
*Array*

Array of all Range instances in the scale.

## See Also

BaseScale Members
BaseScale Class
MindFusion.Gauges Namespace

# BaseScale.reversedCustomFunction Property

Gets or sets the reversed version of the function specified through customFunction.

**Namespace:** MindFusion.Gauges
**File:** BaseScale.js

## Syntax

```
JavaScript
get reversedCustomFunction() {}
set reversedCustomFunction(value) {}
```

### Property Value

Type: FunctionDelegate
*FunctionDelegate*

The reversed version of the function specified through CustomFunction.

## See Also

BaseScale Members
BaseScale Class
MindFusion.Gauges Namespace

## BaseScale.startWidth Property

Gets or sets the width of the scale at its beginning.

**Namespace:** MindFusion.Gauges
**File:** BaseScale.js

### Syntax

**JavaScript**

```
get startWidth() {}
set startWidth(value) {}
```

#### Property Value

Type: Length
*Length*

The width of the scale at its beginning.

### See Also

BaseScale Members
BaseScale Class
MindFusion.Gauges Namespace

# CenterPanel Class

Represents a container which centers its children and makes them with square size.

### Syntax

**JavaScript**

```
// class
CenterPanel.prototype = {}
```

### Inheritance Hierarchy

**MindFusion.Gauges.CenterPanel**

### See Also

CenterPanel Members
MindFusion.Gauges Namespace

## CenterPanel Members

Represents a container which centers its children and makes them with square size.

The following tables list the members exposed by the CenterPanel type.

### Public Constructors

| | Name | Description |
|---|---|---|

| | | |
|---|---|---|
|  | CenterPanel | Creates an instance of **CenterPanel** class. |

## See Also

CenterPanel Class
MindFusion.Gauges Namespace

### CenterPanel Constructor

Creates an instance of CenterPanel class.

**Namespace:** MindFusion.Gauges
**File:** VisualElementContainer.js

## Syntax

**JavaScript**

```
function CenterPanel ()
```

## See Also

CenterPanel Class
MindFusion.Gauges Namespace

# CustomInterval Class

Represents an interval with associated custom values for fill and stroke.

## Syntax

**JavaScript**

```
// class
CustomInterval.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Gauges.CustomInterval**

## See Also

CustomInterval Members
MindFusion.Gauges Namespace

### CustomInterval Members

Represents an interval with associated custom values for fill and stroke.

The following tables list the members exposed by the CustomInterval type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | CustomInterval | Creates an instance of **CustomInterval** class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | fill | Gets or sets the brush to apply as a background of the elements within the custom interval. |
| | foreground | Gets or sets the color to apply as a foreground to the elements within this custom interval. |
| | maxValue | Gets or sets the end of this custom interval. |
| | minValue | Gets or sets the start of this custom interval. |
| | stroke | Gets or sets the brush to apply as an outline of the elements within the custom interval. |

## Public Methods

| | Name | Description |
|---|---|---|
| | contains | Checks whether the specified value falls within this interval. |

## See Also

CustomInterval Class
MindFusion.Gauges Namespace

### CustomInterval Constructor

Creates an instance of CustomInterval class.

**Namespace:** MindFusion.Gauges
**File:** CustomInterval.js

## Syntax

| JavaScript |
|---|
| `function CustomInterval ()` |

## See Also

CustomInterval Class
MindFusion.Gauges Namespace

### CustomInterval Methods

### CustomInterval.contains Method

Checks whether the specified value falls within this interval.

**Namespace:** MindFusion.Gauges
**File:** CustomInterval.js

## Syntax

```
JavaScript
function contains ()
```

## See Also

CustomInterval Members
CustomInterval Class
MindFusion.Gauges Namespace

# CustomInterval Properties

# CustomInterval.fill Property

Gets or sets the brush to apply as a background of the elements within the custom interval.

**Namespace:** MindFusion.Gauges
**File:** CustomInterval.js

## Syntax

```
JavaScript
get fill() {}
set fill(value) {}
```

### Property Value

Type: Object
*Object*

The brush to apply as a background of the elements within the custom interval.

## See Also

CustomInterval Members
CustomInterval Class
MindFusion.Gauges Namespace

# CustomInterval.foreground Property

Gets or sets the color to apply as a foreground to the elements within this custom interval.

**Namespace:** MindFusion.Gauges
**File:** CustomInterval.js

## Syntax

```
JavaScript
```

```
get foreground() {}
set foreground(value) {}
```

### Property Value

Type: Object
*Object*

The color to apply as a foreground to the elements within this custom interval.

## See Also

CustomInterval Members
CustomInterval Class
MindFusion.Gauges Namespace

# CustomInterval.maxValue Property

Gets or sets the end of this custom interval.

**Namespace:** MindFusion.Gauges
**File:** CustomInterval.js

## Syntax

**JavaScript**

```
get maxValue() {}
set maxValue(value) {}
```

### Property Value

Type: Number
*Number*

The end of this custom interval.

## See Also

CustomInterval Members
CustomInterval Class
MindFusion.Gauges Namespace

# CustomInterval.minValue Property

Gets or sets the start of this custom interval.

**Namespace:** MindFusion.Gauges
**File:** CustomInterval.js

## Syntax

**JavaScript**

```
get minValue() {}
set minValue(value) {}
```

### Property Value

Type: Number
*Number*

The start of this custom interval.

## See Also

CustomInterval Members
CustomInterval Class
MindFusion.Gauges Namespace

### CustomInterval.stroke Property

Gets or sets the brush to apply as an outline of the elements within the custom interval.

**Namespace:** MindFusion.Gauges
**File:** CustomInterval.js

## Syntax

**JavaScript**

```
get stroke() {}
set stroke(value) {}
```

#### Property Value

Type: Object
*Object*

The brush to apply as an outline of the elements within the custom interval.

## See Also

CustomInterval Members
CustomInterval Class
MindFusion.Gauges Namespace

# DisplayType Enumeration

Specifies a display condition.

**Namespace:** MindFusion.Gauges
**File:** Enum.js

## Syntax

**JavaScript**

```
// enum
DisplayType = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **Always** | Indicates that the the element should always be displayed. |

| | | |
|---|---|---|
| | **Auto** | Indicates that the visibility of an element should be determined automatically. |
| | **Never** | Indicates that the the element should always be hidden. |

## See Also

MindFusion.Gauges Namespace

# Ellipse Class

Represents an ellipse visual.

## Syntax

**JavaScript**

```
// class
Ellipse.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Gauges.Ellipse**

## See Also

Ellipse Members
MindFusion.Gauges Namespace

# Ellipse Members

Represents an ellipse visual.

The following tables list the members exposed by the Ellipse type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | Ellipse | Creates an instance of **Ellipse** class. |

## See Also

Ellipse Class
MindFusion.Gauges Namespace

# Ellipse Constructor

Creates an instance of Ellipse class.

**Namespace:** MindFusion.Gauges
**File:** Ellipse.js

## Syntax

```
JavaScript
function Ellipse ()
```

## See Also

Ellipse Class
MindFusion.Gauges Namespace

# Events Class

Defines all events raised by the Gauge component.

## Syntax

```
JavaScript
// class
Events.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Gauges.Events**

## See Also

Events Members
MindFusion.Gauges Namespace

# Events Members

Defines all events raised by the Gauge component.

The following tables list the members exposed by the Events type.

## Public Properties

| | Name | Description |
|---|---|---|
| | paintBackground | Raised when the gauge background is being painted. |
| | paintForeground | Raised when the gauge foreground is being painted. |
| | paintPointer | Raised when a gauge pointer is being painted. |
| | paintScale | Raised when a gauge scale is being painted. |
| | paintTick | Raised when a gauge tick is being painted. |
| | prepaintBackground | Raised before the gauge background is painted. |
| | prepaintForeground | Raised before the gauge foreground is painted. |
| | prepaintPointer | Raised before a gauge pointer is painted. |

| | | |
|---|---|---|
| | prepaintScale | Raised before a gauge scale is painted. |
| | prepaintTick | Raised before a gauge tick is painted. |
| | valueChanged | Raised when the value of a pointer is being changed interactively. |
| | valueChanging | Raised when the value of a pointer has changed interactively. |

## See Also

Events Class
MindFusion.Gauges Namespace

# Events Properties

## Events.paintBackground Property

Raised when the gauge background is being painted.

**Namespace:** MindFusion.Gauges
**File:** Events.js

## Syntax

**JavaScript**

```
get paintBackground() {}
set paintBackground(value) {}
```

### Property Value

Type: String
*String*

A string containing the event name.

## See Also

Events Members
Events Class
MindFusion.Gauges Namespace

## Events.paintForeground Property

Raised when the gauge foreground is being painted.

**Namespace:** MindFusion.Gauges
**File:** Events.js

## Syntax

**JavaScript**

```
get paintForeground() {}
set paintForeground(value) {}
```

### Property Value

Type: String
*String*

A string containing the event name.

## See Also

Events Members
Events Class
MindFusion.Gauges Namespace

# Events.paintPointer Property

Raised when a gauge pointer is being painted.

**Namespace:** MindFusion.Gauges
**File:** Events.js

## Syntax

```
JavaScript
get paintPointer() {}
set paintPointer(value) {}
```

### Property Value

Type: String
*String*

A string containing the event name.

## See Also

Events Members
Events Class
MindFusion.Gauges Namespace

# Events.paintScale Property

Raised when a gauge scale is being painted.

**Namespace:** MindFusion.Gauges
**File:** Events.js

## Syntax

```
JavaScript
get paintScale() {}
set paintScale(value) {}
```

### Property Value

Type: String
*String*

A string containing the event name.

## See Also

Events Members
Events Class
MindFusion.Gauges Namespace

# Events.paintTick Property

Raised when a gauge tick is being painted.

**Namespace:** MindFusion.Gauges
**File:** Events.js

## Syntax

```
JavaScript
get paintTick() {}
set paintTick(value) {}
```

### Property Value

Type: String
*String*

A string containing the event name.

## See Also

Events Members
Events Class
MindFusion.Gauges Namespace

# Events.prepaintBackground Property

Raised before the gauge background is painted.

**Namespace:** MindFusion.Gauges
**File:** Events.js

## Syntax

```
JavaScript
get prepaintBackground() {}
set prepaintBackground(value) {}
```

### Property Value

Type: String
*String*

A string containing the event name.

## See Also

Events Members
Events Class
MindFusion.Gauges Namespace

## Events.prepaintForeground Property

Raised before the gauge foreground is painted.

**Namespace:** MindFusion.Gauges
**File:** Events.js

### Syntax

```
JavaScript
get prepaintForeground() {}
set prepaintForeground(value) {}
```

#### Property Value

Type: String
*String*

A string containing the event name.

### See Also

Events Members
Events Class
MindFusion.Gauges Namespace

## Events.prepaintPointer Property

Raised before a gauge pointer is painted.

**Namespace:** MindFusion.Gauges
**File:** Events.js

### Syntax

```
JavaScript
get prepaintPointer() {}
set prepaintPointer(value) {}
```

#### Property Value

Type: String
*String*

A string containing the event name.

### See Also

Events Members
Events Class
MindFusion.Gauges Namespace

## Events.prepaintScale Property

Raised before a gauge scale is painted.

**Namespace:** MindFusion.Gauges
**File:** Events.js

### Syntax

**JavaScript**

```
get prepaintScale() {}
set prepaintScale(value) {}
```

#### Property Value

Type: String
*String*

A string containing the event name.

### See Also

Events Members
Events Class
MindFusion.Gauges Namespace

## Events.prepaintTick Property

Raised before a gauge tick is painted.

**Namespace:** MindFusion.Gauges
**File:** Events.js

### Syntax

**JavaScript**

```
get prepaintTick() {}
set prepaintTick(value) {}
```

#### Property Value

Type: String
*String*

A string containing the event name.

### See Also

Events Members
Events Class
MindFusion.Gauges Namespace

## Events.valueChanged Property

Raised when the value of a pointer is being changed interactively.

**Namespace:** MindFusion.Gauges
**File:** Events.js

## Syntax

```
JavaScript
get valueChanged() {}
set valueChanged(value) {}
```

### Property Value

Type: String
*String*

A string containing the event name.

## See Also

Events Members
Events Class
MindFusion.Gauges Namespace

# Events.valueChanging Property

Raised when the value of a pointer has changed interactively.

**Namespace:** MindFusion.Gauges
**File:** Events.js

## Syntax

```
JavaScript
get valueChanging() {}
set valueChanging(value) {}
```

### Property Value

Type: String
*String*

A string containing the event name.

## See Also

Events Members
Events Class
MindFusion.Gauges Namespace

# FunctionType Enumeration

Indicates the type of function that will be used to distribute values along the gauge scale.

**Namespace:** MindFusion.Gauges
**File:** Enum.js

## Syntax

```
JavaScript
// enum
FunctionType = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **Custom** | Specifies a user defined function, which will be supplied by the CustomFunction property. |
| | **Linear** | Specifies that the values will be distributed linearly. |
| | **Logarithmic** | Specifies a logarithmic function with base specified by the LogarithmBase property. |

## See Also

MindFusion.Gauges Namespace

# GeometryFactory Class

A factory class for various geometry objects.

## Syntax

```
JavaScript
// class
GeometryFactory.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Gauges.GeometryFactory**

## See Also

GeometryFactory Members
MindFusion.Gauges Namespace

# GeometryFactory Members

A factory class for various geometry objects.

The following tables list the members exposed by the GeometryFactory type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | GeometryFactory | Creates an instance of **GeometryFactory** class. |

## Public Methods

| | Name | Description |
|---|---|---|
| | createLinearGeometry | Generates a linear bar geometry with the specified parameters. |
| | createRadialGeometry | Generates a radial bar geometry with the specified parameters. |

## See Also

GeometryFactory Class
MindFusion.Gauges Namespace

# GeometryFactory Constructor

Creates an instance of GeometryFactory class.

**Namespace:** MindFusion.Gauges
**File:** GeometryFactory.js

## Syntax

**JavaScript**

```
function GeometryFactory ()
```

## See Also

GeometryFactory Class
MindFusion.Gauges Namespace

# GeometryFactory Methods

# GeometryFactory.createLinearGeometry Method

Generates a linear bar geometry with the specified parameters.

**Namespace:** MindFusion.Gauges
**File:** GeometryFactory.js

## Syntax

**JavaScript**

```
function createLinearGeometry (centerX, centerY, startInnerOffset, startOuterOffset,
endInnerOffset, endOuterOffset, length, isVertical, capStart, capEnd, strokeInner,
strokeOuter)
```

### Parameters

*centerX*

Type: Number
Number. The x-coordinate of the generated geometry.

*centerY*
Type: Number
Number. The y-coordinate of the generated geometry.

*startInnerOffset*
Type: Number
Number. The start inner offset of the generated geometry relative to the specified center.

*startOuterOffset*
Type: Number
Number. The start outer offset of the generated geometry relative to the specified center.

*endInnerOffset*
Type: Number
Number. The end inner offset of the generated geometry relative to the specified center.

*endOuterOffset*
Type: Number
Number. The end outer offset of the generated geometry relative to the specified center.

*length*
Type: Number
Number. The length of the generated geometry.

*isVertical*
Type: Boolean
Boolean. A flag indicating whether the generated geometry is vertical.

*capStart*
Type: Boolean
Boolean. A flag indicating whether to draw a stroke at the start of the bar.

*capEnd*
Type: Boolean
Boolean. A flag indicating whether to draw a stroke at the end of the bar.

*strokeInner*
Type: Boolean
Boolean. A flag indicating whether to draw a stroke at the inner side of the bar.

*strokeOuter*
Type: Boolean
Boolean. A flag indicating whether to draw a stroke at the outer side of the bar.

### Return Value

A PathFigure representing a linear bar with the specified parameters.

## Remarks

The generated geometry is a trapezium.For example, a vertical linear geometry with centerX = 10, centerY = 10,startInnerOffset = 0, startOuterOffset = 5, endInnerOffset = 0, endOuerOffset = 10,length = 100; will be a trapezium with points (10, 110), (15, 110), (20, 10), (10, 10).A horizontal lienar geometry with the above parameters will be a trapeziumwith points (10, 10), (10, 15), (110, 20), (110, 10).

## See Also

GeometryFactory Members
GeometryFactory Class
MindFusion.Gauges Namespace

# GeometryFactory.createRadialGeometry Method

Generates a radial bar geometry with the specified parameters.

**Namespace:** MindFusion.Gauges
**File:** GeometryFactory.js

## Syntax

**JavaScript**

```
function createRadialGeometry (centerX, centerY, startInnerRadius, startOuterRadius,
endInnerRadius, endOuterRadius, startAngle, endAngle, capStart, capEnd, strokeInner,
strokeOuter)
```

### Parameters

*centerX*
Type: Number
Number. The x-coordinate of the generated geometry.

*centerY*
Type: Number
Number. The y-coordinate of the generated geometry.

*startInnerRadius*
Type: Number
Number. The radius of the start point of the inner outline of the bar.

*startOuterRadius*
Type: Number
Number. The radius of the start point of the outer outline of the bar.

*endInnerRadius*
Type: Number
Number. The radius of the end point of the inner outline of the bar.

*endOuterRadius*
Type: Number
Number. The radius of the end point of the outer outline of the bar.

*startAngle*
Type: Number
Number. The start angle of the radial bar.

*endAngle*
Type: Number
Number. The end angle of the radial bar.

*capStart*
Type: Boolean
Boolean. A flag indicating whether to draw a stroke at the start of the bar.

*capEnd*
Type: Boolean
Boolean. A flag indicating whether to draw a stroke at the end of the bar.

*strokeInner*
Type: Boolean
Boolean. A flag indicating whether to draw a stroke at the inner curve of the bar.

*strokeOuter*
Type: Boolean
Boolean. A flag indicating whether to draw a stroke at the outer curve of the bar.

### Return Value

A PathFigure representing a radial bar with the specified parameters.

## See Also

GeometryFactory Members
GeometryFactory Class
MindFusion.Gauges Namespace

# LabelRotation Enumeration

Indicates the rotation of a label within a scale.

**Namespace:** MindFusion.Gauges
**File:** Enum.js

## Syntax

**JavaScript**

```
// enum
LabelRotation = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **Auto** | Specifies that the labels are automatically rotated so that their baseline always points downwards. |
| | **BaselineAwayFromCenter** | Specifies that the labels are rotated so that their baseline points away from the center of the scale. |
| | **BaselineToCenter** | Specifies that the labels are rotated so that their baseline points towards the center of the scale. |
| | **None** | Specifies no rotation. |
| | **Sideways** | Specifies that the labels are aligned sideways to the scale. |

## See Also

MindFusion.Gauges Namespace

# Length Class

Specifies distance expressed either absolutely, through pixels, or relatively, through a percentage of the container's size.

## Syntax

**JavaScript**

```
// class
Length.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Gauges.Length**

## See Also

Length Members
MindFusion.Gauges Namespace

# Length Members

Specifies distance expressed either absolutely, through pixels, or relatively, through a percentage of the container's size.

The following tables list the members exposed by the Length type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | Length | Creates an instance of **Length** class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | type | Gets or sets the type of this length. |
|  | value | Gets or sets the value representing this length. |

## Public Methods

| | Name | Description |
|---|---|---|
|  | getAbsoluteLength | Calculates the absolute length corresponding to the specified argument, relative to the the specified area. |

## See Also

Length Class
MindFusion.Gauges Namespace

# Length Constructor

Creates an instance of Length class.

**Namespace:** MindFusion.Gauges
**File:** Length.js

## Syntax

| JavaScript |
|---|
| `function Length (value, [type])` |

### Parameters

*value*
Type: Number
Number. A number specifying the initial length value.

*type*
Optional.
A member of the LengthType enumeration specifying the length type.

## See Also

Length Class
MindFusion.Gauges Namespace

# Length Methods

## Length.getAbsoluteLength Method

Calculates the absolute length corresponding to the specified argument, relative to the the specified area.

**Namespace:** MindFusion.Gauges
**File:** Length.js

## Syntax

**JavaScript**

```
function getAbsoluteLength ()
```

## See Also

Length Members
Length Class
MindFusion.Gauges Namespace

# Length Properties

## Length.type Property

Gets or sets the type of this length.

**Namespace:** MindFusion.Gauges
**File:** Length.js

## Syntax

**JavaScript**

```
get type() {}
set type(value) {}
```

### Property Value

Type: LengthType
*LengthType*

The type of this length.

## See Also

Length Members
Length Class
MindFusion.Gauges Namespace

# Length.value Property

Gets or sets the value representing this length.

**Namespace:** MindFusion.Gauges
**File:** Length.js

## Syntax

| JavaScript |
| --- |
| `get value() {}`<br>`set value(value) {}` |

### Property Value

Type: Number
*Number*

The value representing this length.

## See Also

Length Members
Length Class
MindFusion.Gauges Namespace

# LengthType Enumeration

Indicates the type of the units of a Length object.

**Namespace:** MindFusion.Gauges
**File:** Enum.js

## Syntax

| JavaScript |
| --- |
| `// enum`<br>`LengthType = {}` |

## Members

| | Member name | Description |
| --- | --- | --- |
| | **Absolute** | Specifies absolute length, expressed in pixels. |
| | **Relative** | Specifies relative length, expressed as a percentage of the container's size. |

## See Also

MindFusion.Gauges Namespace

# LinearGauge Class

Represents a linear gauge control.

## Syntax

| JavaScript |
| --- |
| `// class`<br>`LinearGauge.prototype = {}` |

## Inheritance Hierarchy

**MindFusion.Gauges.LinearGauge**

## See Also

LinearGauge Members
MindFusion.Gauges Namespace

# LinearGauge Members

Represents a linear gauge control.

The following tables list the members exposed by the LinearGauge type.

## Public Constructors

| | Name | Description |
| --- | --- | --- |
|  | LinearGauge | Creates an instance of **LinearGauge**. |

## Public Properties

| | Name | Description |
| --- | --- | --- |
|  | orientation | Gets or sets the orientation of the gauge. |

## Public Methods

| | Name | Description |
| --- | --- | --- |
|  | create | Creates and initializes a new LinearGauge from the specified element. This method is static and can be called without creating an instance of the class. |
|  | find | Returns the specified LinearGauge object. This member is static and can be invoked without creating an instance of the class. |

## See Also

LinearGauge Class
MindFusion.Gauges Namespace

# LinearGauge Constructor

Creates an instance of **LinearGauge**.

**Namespace:** MindFusion.Gauges
**File:** LinearGauge.js

## Syntax

**JavaScript**

```
function LinearGauge (element)
```

### Parameters

*element*
The Div DOM Element this **LinearGauge** is associated with.

## See Also

LinearGauge Class
MindFusion.Gauges Namespace

# LinearGauge Methods

# LinearGauge.create Method

Creates and initializes a new LinearGauge from the specified element. This method is static and can be called without creating an instance of the class.

**Namespace:** MindFusion.Gauges
**File:** LinearGauge.js

## Syntax

**JavaScript**

```
function create (element, [ignoreMouseEvents])
```

### Parameters

*element*
Type: Object
Object. The DOM element that the LinearGauge should be attached to.

*ignoreMouseEvents*
Optional.
Type: Boolean
Boolean. Optional.

### Return Value

Type: LinearGauge
LinearGauge. A LinearGauge object that represents the newly created gauge.

## See Also

LinearGauge Members
LinearGauge Class
MindFusion.Gauges Namespace

# LinearGauge.find Method

Returns the specified LinearGauge object. This member is static and can be invoked without creating an instance of the class.

**Namespace:** MindFusion.Gauges
**File:** LinearGauge.js

## Syntax

**JavaScript**

```
function find (id)
```

### Parameters

*id*
Type: String
String. A string that contains the ID of the LinearGauge to find.

### Return Value

Type: LinearGauge
LinearGauge. A LinearGauge object that contains the gauge requested by id, if found; otherwise, null.

## See Also

LinearGauge Members
LinearGauge Class
MindFusion.Gauges Namespace

# LinearGauge Properties

# LinearGauge.orientation Property

Gets or sets the orientation of the gauge.

**Namespace:** MindFusion.Gauges
**File:** LinearGauge.js

## Syntax

**JavaScript**

```
get orientation() {}
set orientation(value) {}
```

### Property Value

Type: Orientation
*Orientation*

The orientation of the gauge.

## See Also

LinearGauge Members
LinearGauge Class
MindFusion.Gauges Namespace

# LinearScale Class

Represents a linear scale.

## Syntax

**JavaScript**

```
// class
LinearScale.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Gauges.LinearScale**

## See Also

LinearScale Members
MindFusion.Gauges Namespace

# LinearScale Members

Represents a linear scale.

The following tables list the members exposed by the LinearScale type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | LinearScale | Creates an instance of **LinearScale**. |

## Public Properties

| | Name | Description |
|---|---|---|
| | left | Gets or sets the distance between the left of this scale and the left of its container. |
| | orientation | Gets or sets the orientation of the scale. |
| | scaleAlignment | Gets or sets the alignment of the scale. |

| | | |
|---|---|---|
| | scaleLength | Gets or sets the length of the scale. |
| | top | Gets or sets the distance between the top of this scale and the top of its container. |

## Public Methods

| | Name | Description |
|---|---|---|
| | getValueFromPoint | BaseScale.getValueFromPoint override. |

## See Also

LinearScale Class
MindFusion.Gauges Namespace

# LinearScale Constructor

Creates an instance of LinearScale.

**Namespace:** MindFusion.Gauges
**File:** LinearScale.js

## Syntax

**JavaScript**

```
function LinearScale (parent)
```

### Parameters

*parent*
Type: BaseGauge
BaseGauge. The gauge control this scale belongs to.

## See Also

LinearScale Class
MindFusion.Gauges Namespace

# LinearScale Methods

# LinearScale.getValueFromPoint Method

BaseScale.getValueFromPoint override.

**Namespace:** MindFusion.Gauges
**File:** LinearScale.js

## Syntax

**JavaScript**

```
function getValueFromPoint (point)
```

### Parameters

*point*
Type: Point
Point. The point to test.

### Return Value

Type: Number
Number. The value from the scale, corresponding to the point, possibly MinValue orMaxValue if the specified point cannot be associated with a value.

## See Also

LinearScale Members
LinearScale Class
MindFusion.Gauges Namespace

# LinearScale Properties

## LinearScale.left Property

Gets or sets the distance between the left of this scale and the left of its container.

**Namespace:** MindFusion.Gauges
**File:** LinearScale.js

## Syntax

**JavaScript**

```
get left() {}
set left(value) {}
```

### Property Value

Type: Length
*Length*

The distance between the left of this scale and the left of its container.

## See Also

LinearScale Members
LinearScale Class
MindFusion.Gauges Namespace

## LinearScale.orientation Property

Gets or sets the orientation of the scale.

**Namespace:** MindFusion.Gauges
**File:** LinearScale.js

## Syntax

```
JavaScript
get orientation() {}
set orientation(value) {}
```

### Property Value

Type: Orientation
*Orientation*

The orientation of the scale.

## See Also

LinearScale Members
LinearScale Class
MindFusion.Gauges Namespace

# LinearScale.scaleAlignment Property

Gets or sets the alignment of the scale.

**Namespace:** MindFusion.Gauges
**File:** LinearScale.js

## Syntax

```
JavaScript
get scaleAlignment() {}
set scaleAlignment(value) {}
```

### Property Value

Type: Alignment
*Alignment*

The alignment of the scale.

## Remarks

All Inner* values align a vertical scale to the left side and a horizontalscale to the top side of the control. All Outer* values align a verticalscale to the right side and a horizontal scale to the bottom side of thecontrol. All Center* values center the scale along its length.

## See Also

LinearScale Members
LinearScale Class
MindFusion.Gauges Namespace

# LinearScale.scaleLength Property

Gets or sets the length of the scale.

**Namespace:** MindFusion.Gauges
**File:** LinearScale.js

## Syntax

**JavaScript**

```
get scaleLength() {}
set scaleLength(value) {}
```

### Property Value

Type: Length
*Length*

The length of the scale.

## See Also

LinearScale Members
LinearScale Class
MindFusion.Gauges Namespace

# LinearScale.top Property

Gets or sets the distance between the top of this scale and the top of its container.

**Namespace:** MindFusion.Gauges
**File:** LinearScale.js

## Syntax

**JavaScript**

```
get top() {}
set top(value) {}
```

### Property Value

Type: Length
*Length*

The distance between the top of this scale and the top of its container.

## See Also

LinearScale Members
LinearScale Class
MindFusion.Gauges Namespace

# LineSegment Class

Represents a line between two points.

## Syntax

**JavaScript**

```
// class
LineSegment.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Gauges.LineSegment**

## See Also

LineSegment Members
MindFusion.Gauges Namespace

# LineSegment Members

Represents a line between two points.

The following tables list the members exposed by the LineSegment type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | LineSegment | Creates an instance of **LineSegment** class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | point | Gets or sets the end point of the line segment. |

## Public Methods

| | Name | Description |
|---|---|---|
|  | addTo | Segment.addTo override. |

## See Also

LineSegment Class
MindFusion.Gauges Namespace

# LineSegment Constructor

Creates an instance of LineSegment class.

**Namespace:** MindFusion.Gauges
**File:** Segment.js

## Syntax

| JavaScript |
|---|
| `function LineSegment ()` |

## See Also

LineSegment Class
MindFusion.Gauges Namespace

# LineSegment Methods

## LineSegment.addTo Method

Segment.addTo override.

**Namespace:** MindFusion.Gauges
**File:** Segment.js

### Syntax

```
JavaScript
function addTo ()
```

### See Also

LineSegment Members
LineSegment Class
MindFusion.Gauges Namespace

# LineSegment Properties

## LineSegment.point Property

Gets or sets the end point of the line segment.

**Namespace:** MindFusion.Gauges
**File:** Segment.js

### Syntax

```
JavaScript
get point() {}
set point(value) {}
```

#### Property Value

Type: Point
*Point*

The end point of the line segment.

### See Also

LineSegment Members
LineSegment Class
MindFusion.Gauges Namespace

# MajorTickSettings Class

Provides settings for the major ticks and labels in a gauge scale.

## Syntax

**JavaScript**

```
// class
MajorTickSettings.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Gauges.MajorTickSettings**

## See Also

MajorTickSettings Members
MindFusion.Gauges Namespace

# MajorTickSettings Members

Provides settings for the major ticks and labels in a gauge scale.

The following tables list the members exposed by the MajorTickSettings type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | MajorTickSettings | Creates an instance of **MajorTickSettings** class. |

## See Also

MajorTickSettings Class
MindFusion.Gauges Namespace

# MajorTickSettings Constructor

Creates an instance of MajorTickSettings class.

**Namespace:** MindFusion.Gauges
**File:** TickSettings.js

## Syntax

**JavaScript**

```
function MajorTickSettings (parent)
```

### Parameters

*parent*
Type: BaseScale
BaseScale. The scale these settings belong to.

## See Also

MajorTickSettings Class
MindFusion.Gauges Namespace

# MiddleTickSettings Class

Provides settings for the middle ticks and labels in a gauge scale.

## Syntax

**JavaScript**

```
// class
MiddleTickSettings.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Gauges.MiddleTickSettings**

## See Also

MiddleTickSettings Members
MindFusion.Gauges Namespace

# MiddleTickSettings Members

Provides settings for the middle ticks and labels in a gauge scale.

The following tables list the members exposed by the MiddleTickSettings type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | MiddleTickSettings | Creates an instance of **MiddleTickSettings** class. |

## See Also

MiddleTickSettings Class
MindFusion.Gauges Namespace

# MiddleTickSettings Constructor

Creates an instance of MiddleTickSettings class.

**Namespace:** MindFusion.Gauges
**File:** TickSettings.js

## Syntax

**JavaScript**

```
function MiddleTickSettings (parent)
```

### Parameters

*parent*
Type: BaseScale
BaseScale. The scale these settings belong to.

## See Also

MiddleTickSettings Class
MindFusion.Gauges Namespace

# MinorTickSettings Class

Provides settings for the minor ticks and labels in a gauge scale.

## Syntax

**JavaScript**

```
// class
MinorTickSettings.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Gauges.MinorTickSettings**

## See Also

MinorTickSettings Members
MindFusion.Gauges Namespace

# MinorTickSettings Members

Provides settings for the minor ticks and labels in a gauge scale.

The following tables list the members exposed by the MinorTickSettings type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | MinorTickSettings | Creates an instance of **MinorTickSettings** class. |

## See Also

MinorTickSettings Class
MindFusion.Gauges Namespace

# MinorTickSettings Constructor

Creates an instance of MinorTickSettings class.

**Namespace:** MindFusion.Gauges
**File:** TickSettings.js

## Syntax

| JavaScript |
| --- |
| `function MinorTickSettings (parent)` |

### Parameters

*parent*
Type: BaseScale
BaseScale. The scale these settings belong to.

## See Also

MinorTickSettings Class
MindFusion.Gauges Namespace

# MouseAction Enumeration

Specifies an action performed with the mouse.

**Namespace:** MindFusion.Gauges
**File:** Enum.js

## Syntax

| JavaScript |
| --- |
| `// enum`<br>`MouseAction = {}` |

## Members

| | Member name | Description |
| --- | --- | --- |
| | **Down** | Indicates the press of a mouse button. |
| | **Move** | Indicates the move of a mouse. |
| | **Up** | Indicates the release of a mouse button. |

## See Also

MindFusion.Gauges Namespace

# MouseInfo Class

Provides mouse information when processing mouse events in the visual element hierarchy.

## Syntax

```
JavaScript
// class
MouseInfo.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Gauges.MouseInfo**

## See Also

MouseInfo Members
MindFusion.Gauges Namespace

# MouseInfo Members

Provides mouse information when processing mouse events in the visual element hierarchy.

The following tables list the members exposed by the MouseInfo type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | MouseInfo | Creates an instance of **MouseInfo** class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | action | Gets the action that was performed. |
|  | point | Gets the mouse location. |

## See Also

MouseInfo Class
MindFusion.Gauges Namespace

# MouseInfo Constructor

Creates an instance of MouseInfo class.

**Namespace:** MindFusion.Gauges
**File:** MouseInfo.js

## Syntax

```
JavaScript
function MouseInfo ()
```

## See Also

# MouseInfo Properties

## MouseInfo.action Property

Gets the action that was performed.

**Namespace:** MindFusion.Gauges
**File:** MouseInfo.js

### Syntax

```
JavaScript
get action() {}
```

#### Property Value

Type: MouseAction
*MouseAction*

The action that was performed.

### See Also

MouseInfo Members
MouseInfo Class
MindFusion.Gauges Namespace

## MouseInfo.point Property

Gets the mouse location.

**Namespace:** MindFusion.Gauges
**File:** MouseInfo.js

### Syntax

```
JavaScript
get point() {}
```

#### Property Value

Type: Point
*Point*

The mouse location.

### See Also

MouseInfo Members
MouseInfo Class
MindFusion.Gauges Namespace

# Orientation Enumeration

Specifies the orientation of elements.

**Namespace:** MindFusion.Gauges
**File:** Enum.js

## Syntax

**JavaScript**

```
// enum
Orientation = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **Horizontal** | The element is oriented horizontally. |
| | **Vertical** | The element is oriented vertically. |

## See Also

MindFusion.Gauges Namespace

# OvalGauge Class

Represents a circular gauge control.

## Syntax

**JavaScript**

```
// class
OvalGauge.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Gauges.OvalGauge**

## See Also

OvalGauge Members
MindFusion.Gauges Namespace

# OvalGauge Members

Represents a circular gauge control.

The following tables list the members exposed by the OvalGauge type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | OvalGauge | Creates an instance of **OvalGauge**. |

## Public Properties

| | Name | Description |
|---|---|---|
| | style | Gets or sets the visual style of the gauge. |

## Public Methods

| | Name | Description |
|---|---|---|
| | create | Creates and initializes a new OvalGauge from the specified element. This method is static and can be called without creating an instance of the class. |
| | find | Returns the specified OvalGauge object. This member is static and can be invoked without creating an instance of the class. |

## See Also

OvalGauge Class
MindFusion.Gauges Namespace

# OvalGauge Constructor

Creates an instance of **OvalGauge**.

**Namespace:** MindFusion.Gauges
**File:** OvalGauge.js

## Syntax

**JavaScript**

```
function OvalGauge (element)
```

### Parameters

*element*
The Div DOM Element this **OvalGauge** is associated with.

## See Also

OvalGauge Class
MindFusion.Gauges Namespace

# OvalGauge Methods

# OvalGauge.create Method

Creates and initializes a new OvalGauge from the specified element. This method is static and can be called without creating an instance of the class.

**Namespace:** MindFusion.Gauges
**File:** OvalGauge.js

## Syntax

**JavaScript**

```
function create (element, [ignoreMouseEvents])
```

### Parameters

*element*
Type: Object
Object. The DOM element that the OvalGauge should be attached to.

*ignoreMouseEvents*
Optional.
Type: Boolean
Boolean. Optional.

### Return Value

Type: OvalGauge
OvalGauge. A OvalGauge object that represents the newly created gauge.

## See Also

OvalGauge Members
OvalGauge Class
MindFusion.Gauges Namespace

# OvalGauge.find Method

Returns the specified OvalGauge object. This member is static and can be invoked without creating an instance of the class.

**Namespace:** MindFusion.Gauges
**File:** OvalGauge.js

## Syntax

**JavaScript**

```
function find (id)
```

### Parameters

*id*
Type: String
String. A string that contains the ID of the OvalGauge to find.

### Return Value

Type: OvalGauge
OvalGauge. A OvalGauge object that contains the gauge requested by id, if found; otherwise, null.

## See Also

OvalGauge Members
OvalGauge Class
MindFusion.Gauges Namespace

# OvalGauge Properties

## OvalGauge.style Property

Gets or sets the visual style of the gauge.

**Namespace:** MindFusion.Gauges
**File:** OvalGauge.js

## Syntax

**JavaScript**

```
get style() {}
set style(value) {}
```

### Property Value

Type: OvalGaugeStyle
*OvalGaugeStyle*

The visual style of the gauge.

## See Also

OvalGauge Members
OvalGauge Class
MindFusion.Gauges Namespace

# OvalGaugeStyle Enumeration

Specifies the appearance of an oval gauge.

**Namespace:** MindFusion.Gauges
**File:** Enum.js

## Syntax

**JavaScript**

```
// enum
OvalGaugeStyle = {}
```

## Members

| | Member name | Description |
|---|---|---|

| | | |
|---|---|---|
| | **Circular** | Indicates a regular circular appearance. |
| | **QuadraticNE** | Indicates a quadratic appearance with the scales oriented towards North-East. |
| | **QuadraticNW** | Indicates a quadratic appearance with the scales oriented towards North-West. |
| | **QuadraticSE** | Indicates a quadratic appearance with the scales oriented towards South-East. |
| | **QuadraticSW** | Indicates a quadratic appearance with the scales oriented towards South-West. |
| | **SemicircleE** | Indicates a semi-circular appearance with the scales oriented towards East. |
| | **SemicircleN** | Indicates a semi-circular appearance with the scales oriented towards North. |
| | **SemicircleS** | Indicates a semi-circular appearance with the scales oriented towards South. |
| | **SemicircleW** | Indicates a semi-circular appearance with the scales oriented towards West. |

## See Also

MindFusion.Gauges Namespace

# OvalScale Class

Represents a radial scale.

## Syntax

**JavaScript**

```
// class
OvalScale.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Gauges.OvalScale**

## See Also

OvalScale Members
MindFusion.Gauges Namespace

# OvalScale Members

Represents a radial scale.

The following tables list the members exposed by the OvalScale type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | OvalScale | Creates an instance of **OvalScale**. |

## Public Properties

| | Name | Description |
|---|---|---|
| | endAngle | Gets or sets the end angle of the scale. |
| | scaleRelativeCenter | Gets or sets the center of the scale relative to the scale bounds. |
| | scaleRelativeRadius | Gets or sets the radius of the scale relative to the scale bounds. |
| | startAngle | Gets or sets the start angle of the scale. |

## Public Methods

| | Name | Description |
|---|---|---|
| | getValueFromPoint | BaseScale.getValueFromPoint override. |

## See Also

OvalScale Class
MindFusion.Gauges Namespace

# OvalScale Constructor

Creates an instance of OvalScale.

**Namespace:** MindFusion.Gauges
**File:** OvalScale.js

## Syntax

**JavaScript**

```
function OvalScale (parent)
```

### Parameters

*parent*
Type: BaseGauge
BaseGauge. The gauge control this scale belongs to.

## See Also

OvalScale Class
MindFusion.Gauges Namespace

# OvalScale Methods

# OvalScale.getValueFromPoint Method

BaseScale.getValueFromPoint override.

**Namespace:** MindFusion.Gauges
**File:** OvalScale.js

## Syntax

**JavaScript**

```
function getValueFromPoint (point)
```

### Parameters

*point*
Type: Point
Point. The point to test.

### Return Value

Type: Number
Number. The value from the scale, corresponding to the point, possibly MinValue orMaxValue if the specified point cannot be associated with a value.

## See Also

OvalScale Members
OvalScale Class
MindFusion.Gauges Namespace

# OvalScale Properties

# OvalScale.endAngle Property

Gets or sets the end angle of the scale.

**Namespace:** MindFusion.Gauges
**File:** OvalScale.js

## Syntax

**JavaScript**

```
get endAngle() {}
set endAngle(value) {}
```

### Property Value

Type: Number
*Number*

The end angle of the scale.

## See Also

OvalScale Members
OvalScale Class

# OvalScale.scaleRelativeCenter Property

Gets or sets the center of the scale relative to the scale bounds.

**Namespace:** MindFusion.Gauges
**File:** OvalScale.js

## Syntax

**JavaScript**

```
get scaleRelativeCenter() {}
set scaleRelativeCenter(value) {}
```

### Property Value

Type: Point
*Point*

The center of the scale relative to the scale bounds.

## Remarks

A value of (0.5, 0.5) indicates the center of the bounds, a value of (0, 0) indicates the top-left corner.

## See Also

OvalScale Members
OvalScale Class
MindFusion.Gauges Namespace

# OvalScale.scaleRelativeRadius Property

Gets or sets the radius of the scale relative to the scale bounds.

**Namespace:** MindFusion.Gauges
**File:** OvalScale.js

## Syntax

**JavaScript**

```
get scaleRelativeRadius() {}
set scaleRelativeRadius(value) {}
```

### Property Value

Type: Number
*Number*

The radius of the scale relative to the scale bounds.

## Remarks

A value of 0.5 indicates that the radius is half the size of the scale.

## See Also

OvalScale Members
OvalScale Class
MindFusion.Gauges Namespace

# OvalScale.startAngle Property

Gets or sets the start angle of the scale.

**Namespace:** MindFusion.Gauges
**File:** OvalScale.js

## Syntax

**JavaScript**

```
get startAngle() {}
set startAngle(value) {}
```

### Property Value

Type: Number
*Number*

The start angle of the scale.

## See Also

OvalScale Members
OvalScale Class
MindFusion.Gauges Namespace

# PaintEventArgs Class

Provides data for the various custom painting event.

## Syntax

**JavaScript**

```
// class
PaintEventArgs.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Gauges.PaintEventArgs**

## See Also

PaintEventArgs Members
MindFusion.Gauges Namespace

# PaintEventArgs Members

Provides data for the various custom painting event.

The following tables list the members exposed by the PaintEventArgs type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | PaintEventArgs | Creates an instance of **PaintEventArgs** class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | context | Gets the canvas rendering context. |
| | element | Gets or sets the element being custom drawn. |

## Public Methods

| | Name | Description |
|---|---|---|
| | paintVisualElement | Paints the specified visual element to the underlying canvas. |

## See Also

PaintEventArgs Class
MindFusion.Gauges Namespace

# PaintEventArgs Constructor

Creates an instance of PaintEventArgs class.

**Namespace:** MindFusion.Gauges
**File:** EventArgs.js

## Syntax

| JavaScript |
|---|
| `function PaintEventArgs ()` |

## See Also

PaintEventArgs Class
MindFusion.Gauges Namespace

# PaintEventArgs Methods

# PaintEventArgs.paintVisualElement Method

Paints the specified visual element to the underlying canvas.

**Namespace:** MindFusion.Gauges
**File:** EventArgs.js

## Syntax

```
JavaScript
function paintVisualElement (element, constraint)
```

### Parameters

*element*
Type: VisualElement
VisualElement. The VisualElement to paint.

*constraint*
Type: Size
Size. The available size.

## See Also

PaintEventArgs Members
PaintEventArgs Class
MindFusion.Gauges Namespace

# PaintEventArgs Properties

# PaintEventArgs.context Property

Gets the canvas rendering context.

**Namespace:** MindFusion.Gauges
**File:** EventArgs.js

## Syntax

```
JavaScript
get context() {}
set context(value) {}
```

### Property Value

Type: CanvasRenderingContext2D
*CanvasRenderingContext2D*

A rendering context used to draw on the Canvas element.

## See Also

PaintEventArgs Members
PaintEventArgs Class
MindFusion.Gauges Namespace

# PaintEventArgs.element Property

Gets or sets the element being custom drawn.

**Namespace:** MindFusion.Gauges
**File:** EventArgs.js

## Syntax

```
JavaScript
get element() {}
set element(value) {}
```

### Property Value

Type: VisualElement
*VisualElement*

A VisualElement instance.

## See Also

PaintEventArgs Members
PaintEventArgs Class
MindFusion.Gauges Namespace

# PathFigure Class

Represents a connected series of geometric segments.

## Syntax

```
JavaScript
// class
PathFigure.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Gauges.PathFigure**

## See Also

PathFigure Members
MindFusion.Gauges Namespace

## PathFigure Members

Represents a connected series of geometric segments.

The following tables list the members exposed by the PathFigure type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | PathFigure | Creates an instance of **PathFigure** class. |

## See Also

PathFigure Class
MindFusion.Gauges Namespace

### PathFigure Constructor

Creates an instance of PathFigure class.

**Namespace:** MindFusion.Gauges
**File:** PathFigure.js

## Syntax

**JavaScript**

```
function PathFigure (data)
```

#### Parameters

*data*
Type: String
String. The path data string.

## See Also

PathFigure Class
MindFusion.Gauges Namespace

# Pointer Class

Represents a needle (or arrow) within a gauge scale.

## Syntax

**JavaScript**

```
// class
Pointer.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Gauges.Pointer**

## See Also

Pointer Members
MindFusion.Gauges Namespace

### Pointer Members

Represents a needle (or arrow) within a gauge scale.

The following tables list the members exposed by the Pointer type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | Pointer | Creates an instance of **Pointer** class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | alignment | Gets or sets the alignment of the pointer relative to the scale. |
| | customShape | Gets or sets the definition of the pointer shape when Shape is set to Custom. |
| | isDiscrete | Gets or sets a value indicating whether the pointer value can be changed only discretely(the pointer Value can be only integer). |
| | isInteractive | Gets or sets a value indicating whether the pointer position should be affected by user interactions. |
| | pointerHeight | Gets or sets the height of the pointer. |
| | pointerOffset | Gets or sets the offset of the pointer along the direction it points to. |
| | pointerWidth | Gets or sets the width of the pointer. |
| | shape | Gets or sets the shape of the pointer. |
| | value | Gets or sets the value this pointer points to. |

## See Also

Pointer Class
MindFusion.Gauges Namespace

# Pointer Constructor

Creates an instance of Pointer class.

**Namespace:** MindFusion.Gauges
**File:** Pointer.js

## Syntax

| JavaScript |
|---|
| `function Pointer ()` |

## See Also

Pointer Class
MindFusion.Gauges Namespace

# Pointer Properties

## Pointer.alignment Property

Gets or sets the alignment of the pointer relative to the scale.

**Namespace:** MindFusion.Gauges
**File:** Pointer.js

### Syntax

```
JavaScript
get alignment() {}
set alignment(value) {}
```

#### Property Value

Type: Alignment
*Alignment*

The alignment of the pointer relative to the scale.

### Remarks

This property is ignored in oval gauges where the pointer is always pinned to the center of the gauge.

### See Also

Pointer Members
Pointer Class
MindFusion.Gauges Namespace

## Pointer.customShape Property

Gets or sets the definition of the pointer shape when Shape is set to Custom.

**Namespace:** MindFusion.Gauges
**File:** Pointer.js

### Syntax

```
JavaScript
get customShape() {}
set customShape(value) {}
```

#### Property Value

Type: String
*String*

The definition of the pointer shape when Shape is set to Custom.

### See Also

Pointer Members
Pointer Class
MindFusion.Gauges Namespace

## Pointer.isDiscrete Property

Gets or sets a value indicating whether the pointer value can be changed only discretely(the pointer Value can be only integer).

**Namespace:** MindFusion.Gauges
**File:** Pointer.js

### Syntax

**JavaScript**

```
get isDiscrete() {}
set isDiscrete(value) {}
```

#### Property Value

Type: Boolean
*Boolean*

True if the pointer is discrete, otherwise false.

### See Also

Pointer Members
Pointer Class
MindFusion.Gauges Namespace

## Pointer.isInteractive Property

Gets or sets a value indicating whether the pointer position should be affected by user interactions.

**Namespace:** MindFusion.Gauges
**File:** Pointer.js

### Syntax

**JavaScript**

```
get isInteractive() {}
set isInteractive(value) {}
```

#### Property Value

Type: Boolean
*Boolean*

True if the pointer is interactive, otherwise false.

### See Also

Pointer Members
Pointer Class
MindFusion.Gauges Namespace

## Pointer.pointerHeight Property

Gets or sets the height of the pointer.

**Namespace:** MindFusion.Gauges
**File:** Pointer.js

### Syntax

```
JavaScript
get pointerHeight() {}
set pointerHeight(value) {}
```

#### Property Value

Type: Length
*Length*

The height of the pointer.

### See Also

Pointer Members
Pointer Class
MindFusion.Gauges Namespace

## Pointer.pointerOffset Property

Gets or sets the offset of the pointer along the direction it points to.

**Namespace:** MindFusion.Gauges
**File:** Pointer.js

### Syntax

```
JavaScript
get pointerOffset() {}
set pointerOffset(value) {}
```

#### Property Value

Type: Length
*Length*

The offset of the pointer along the direction it points to.

### See Also

Pointer Members
Pointer Class
MindFusion.Gauges Namespace

## Pointer.pointerWidth Property

Gets or sets the width of the pointer.

**Namespace:** MindFusion.Gauges
**File:** Pointer.js

### Syntax

```
JavaScript
get pointerWidth() {}
set pointerWidth(value) {}
```

#### Property Value

Type: Length
*Length*

The width of the pointer.

### See Also

Pointer Members
Pointer Class
MindFusion.Gauges Namespace

## Pointer.shape Property

Gets or sets the shape of the pointer.

**Namespace:** MindFusion.Gauges
**File:** Pointer.js

### Syntax

```
JavaScript
get shape() {}
set shape(value) {}
```

#### Property Value

Type: PointerShape
*PointerShape*

The shape of the pointer.

### See Also

Pointer Members
Pointer Class
MindFusion.Gauges Namespace

## Pointer.value Property

Gets or sets the value this pointer points to.

**Namespace:** MindFusion.Gauges
**File:** Pointer.js

## Syntax

```
JavaScript
get value() {}
set value(value) {}
```

### Property Value

Type: Number
*Number*

The value this pointer points to.

## See Also

Pointer Members
Pointer Class
MindFusion.Gauges Namespace

# PointerShape Enumeration

Indicates the shape of a pointer.

**Namespace:** MindFusion.Gauges
**File:** Enum.js

## Syntax

```
JavaScript
// enum
PointerShape = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **Custom** | Specifies a custom shape. |
| | **Ellipse** | Specifies an elliptical shape. |
| | **Needle** | Specifies a sharp pointer with a circle at its base. |
| | **Needle2** | Specifies a sharp pointer with a circle at its base. |
| | **None** | Specifies an empty shape, that is, the pointer is not displayed. |
| | **Rectangle** | Specifies a rectangular shape. |

## See Also

MindFusion.Gauges Namespace

# PrepaintEventArgs Class

Provides data for various pre-paint events.

## Syntax

**JavaScript**

```
// class
PrepaintEventArgs.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Gauges.PrepaintEventArgs**

## See Also

PrepaintEventArgs Members
MindFusion.Gauges Namespace

# PrepaintEventArgs Members

Provides data for various pre-paint events.

The following tables list the members exposed by the PrepaintEventArgs type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | PrepaintEventArgs | Creates an instance of **PrepaintEventArgs** class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | cancelDefaultPainting | Gets or sets a value indicating whether the default painting of this element should be performed. |

## See Also

PrepaintEventArgs Class
MindFusion.Gauges Namespace

# PrepaintEventArgs Constructor

Creates an instance of PrepaintEventArgs class.

**Namespace:** MindFusion.Gauges
**File:** EventArgs.js

## Syntax

**JavaScript**

```
function PrepaintEventArgs ()
```

## See Also

PrepaintEventArgs Class
MindFusion.Gauges Namespace

# PrepaintEventArgs Properties

## PrepaintEventArgs.cancelDefaultPainting Property

Gets or sets a value indicating whether the default painting of this element should be performed.

**Namespace:** MindFusion.Gauges
**File:** EventArgs.js

## Syntax

**JavaScript**

```
get cancelDefaultPainting() {}
set cancelDefaultPainting(value) {}
```

### Property Value

Type: Boolean
*Boolean*

true if the default painting of this element should be performed; otherwise, false.

## See Also

PrepaintEventArgs Members
PrepaintEventArgs Class
MindFusion.Gauges Namespace

# Range Class

Represents a range within a gauge scale.

## Syntax

**JavaScript**

```
// class
Range.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Gauges.Range**

## See Also

Range Members
MindFusion.Gauges Namespace

# Range Members

Represents a range within a gauge scale.

The following tables list the members exposed by the Range type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | Range | Creates an instance of **Range** class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | alignment | Gets or sets the alignment of this range relative to the scale it is associated with. |
|  | autoSize | Gets or sets a value indicating whether the start and end width of the range will be automatically adjusted to match the width of the scale. |
|  | capEnd | Gets or sets a value indicating whether to draw a stroke in the end of this range. |
|  | capStart | Gets or sets a value indicating whether to draw a stroke in the beginning of this range. |
|  | endWidth | Gets or sets the width of this range at its end. |
|  | maxValue | Gets or sets the end value of this range. |
|  | minValue | Gets or sets the start value of this range. |
|  | offset | Gets or sets the offset of the range from the position calculated according to its alignment. |
|  | startWidth | Gets or sets the width of this range at its start. |
|  | strokeInside | Gets or sets a value indicating whether to draw a stroke at the inside of this range. |
|  | strokeOutside | Gets or sets a value indicating whether to draw a stroke at the outside of this range. |

## See Also

Range Class
MindFusion.Gauges Namespace

# Range Constructor

Creates an instance of Range class.

**Namespace:** MindFusion.Gauges
**File:** Range.js

## Syntax

**JavaScript**

```
function Range ()
```

## See Also

Range Class
MindFusion.Gauges Namespace

# Range Properties

## Range.alignment Property

Gets or sets the alignment of this range relative to the scale it is associated with.

**Namespace:** MindFusion.Gauges
**File:** Range.js

## Syntax

**JavaScript**

```
get alignment() {}
set alignment(value) {}
```

### Property Value

Type: Alignment
*Alignment*

The alignment of this range relative to the scale it is associated with.

## See Also

Range Members
Range Class
MindFusion.Gauges Namespace

## Range.autoSize Property

Gets or sets a value indicating whether the start and end width of the range will be automatically adjusted to match the width of the scale.

**Namespace:** MindFusion.Gauges
**File:** Range.js

## Syntax

**JavaScript**

```

```

```
get autoSize() {}
set autoSize(value) {}
```

### Property Value

Type: Boolean
*Boolean*

True if the range is autosized, otherwise false.

## Remarks

If this property is set to true the values of startWidth and endWidth properties are ignored.

## See Also

Range Members
Range Class
MindFusion.Gauges Namespace

# Range.capEnd Property

Gets or sets a value indicating whether to draw a stroke in the end of this range.

**Namespace:** MindFusion.Gauges
**File:** Range.js

## Syntax

**JavaScript**

```
get capEnd() {}
set capEnd(value) {}
```

### Property Value

Type: Boolean
*Boolean*

True if there should be a stroke in the end of this range, otherwise false.

## See Also

Range Members
Range Class
MindFusion.Gauges Namespace

# Range.capStart Property

Gets or sets a value indicating whether to draw a stroke in the beginning of this range.

**Namespace:** MindFusion.Gauges
**File:** Range.js

## Syntax

**JavaScript**

```
get capStart() {}
set capStart(value) {}
```

### Property Value

Type: Boolean
*Boolean*

True if there should be a stroke in the beginning of this range, otherwise false.

## See Also

Range Members
Range Class
MindFusion.Gauges Namespace

# Range.endWidth Property

Gets or sets the width of this range at its end.

**Namespace:** MindFusion.Gauges
**File:** Range.js

## Syntax

**JavaScript**

```
get endWidth() {}
set endWidth(value) {}
```

### Property Value

Type: Length
*Length*

The width of this range at its end.

## See Also

Range Members
Range Class
MindFusion.Gauges Namespace

# Range.maxValue Property

Gets or sets the end value of this range.

**Namespace:** MindFusion.Gauges
**File:** Range.js

## Syntax

**JavaScript**

```
get maxValue() {}
set maxValue(value) {}
```

### Property Value

Type: Number
*Number*

The end value of this range.

## See Also

Range Members
Range Class
MindFusion.Gauges Namespace

# Range.minValue Property

Gets or sets the start value of this range.

**Namespace:** MindFusion.Gauges
**File:** Range.js

## Syntax

**JavaScript**

```
get minValue() {}
set minValue(value) {}
```

### Property Value

Type: Number
*Number*

The start value of this range.

## See Also

Range Members
Range Class
MindFusion.Gauges Namespace

# Range.offset Property

Gets or sets the offset of the range from the position calculated according to its alignment.

**Namespace:** MindFusion.Gauges
**File:** Range.js

## Syntax

**JavaScript**

```
get offset() {}
set offset(value) {}
```

### Property Value

Type: Length
*Length*

The offset of the range from the position calculated according to its alignment.

## See Also

Range Members
Range Class
MindFusion.Gauges Namespace

# Range.startWidth Property

Gets or sets the width of this range at its start.

**Namespace:** MindFusion.Gauges
**File:** Range.js

## Syntax

**JavaScript**

```
get startWidth() {}
set startWidth(value) {}
```

### Property Value

Type: Length
*Length*

The width of this range at its start.

## See Also

Range Members
Range Class
MindFusion.Gauges Namespace

# Range.strokeInside Property

Gets or sets a value indicating whether to draw a stroke at the inside of this range.

**Namespace:** MindFusion.Gauges
**File:** Range.js

## Syntax

**JavaScript**

```
get strokeInside() {}
set strokeInside(value) {}
```

### Property Value

Type: Boolean
*Boolean*

True if there should be a stroke at the inside of this range, otherwise false.

## See Also

Range Members
Range Class

MindFusion.Gauges Namespace

## Range.strokeOutside Property

Gets or sets a value indicating whether to draw a stroke at the outside of this range.

**Namespace:** MindFusion.Gauges
**File:** Range.js

### Syntax

**JavaScript**

```
get strokeOutside() {}
set strokeOutside(value) {}
```

#### Property Value

Type: Boolean
*Boolean*

True if there should be a stroke at the outside of this range, otherwise false.

### See Also

Range Members
Range Class
MindFusion.Gauges Namespace

# RoundRectangle Class

Represents a rounded rectangle.

### Syntax

**JavaScript**

```
// class
RoundRectangle.prototype = {}
```

### Inheritance Hierarchy

**MindFusion.Gauges.RoundRectangle**

### See Also

RoundRectangle Members
MindFusion.Gauges Namespace

## RoundRectangle Members

Represents a rounded rectangle.

The following tables list the members exposed by the RoundRectangle type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | RoundRectangle | Creates an instance of **RoundRectangle** class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | roundness | Gets or sets the relative roundness of this rectangle's corners. |

## See Also

RoundRectangle Class
MindFusion.Gauges Namespace

# RoundRectangle Constructor

Creates an instance of RoundRectangle class.

**Namespace:** MindFusion.Gauges
**File:** RoundRectangle.js

## Syntax

**JavaScript**

```
function RoundRectangle ()
```

## See Also

RoundRectangle Class
MindFusion.Gauges Namespace

# RoundRectangle Properties

# RoundRectangle.roundness Property

Gets or sets the relative roundness of this rectangle's corners.

**Namespace:** MindFusion.Gauges
**File:** RoundRectangle.js

## Syntax

**JavaScript**

```
get roundness() {}
set roundness(value) {}
```

### Property Value

Type: Number
*Number*

The relative roundness of this rectangle's corners.

## See Also

RoundRectangle Members
RoundRectangle Class
MindFusion.Gauges Namespace

# Segment Class

Represents a segment of a PathFigure object.

## Syntax

| JavaScript |
| --- |
| `// class`<br>`Segment.prototype = {}` |

## Inheritance Hierarchy

**MindFusion.Gauges.Segment**

## See Also

Segment Members
MindFusion.Gauges Namespace

# Segment Members

Represents a segment of a PathFigure object.

The following tables list the members exposed by the Segment type.

## Public Constructors

| | Name | Description |
| --- | --- | --- |
|  | Segment | Creates an instance of **Segment** class. |

## Public Properties

| | Name | Description |
| --- | --- | --- |
|  | isStroked | Gets or sets a value that indicates whether the segment is stroked. |

## Public Methods

| | Name | Description |
| --- | --- | --- |

| | | |
|---|---|---|
| | addTo | Adds the segment to the specified graphics path. |

## See Also

Segment Class
MindFusion.Gauges Namespace

# Segment Constructor

Creates an instance of Segment class.

**Namespace:** MindFusion.Gauges
**File:** Segment.js

## Syntax

**JavaScript**

```
function Segment ()
```

## See Also

Segment Class
MindFusion.Gauges Namespace

# Segment Methods

# Segment.addTo Method

Adds the segment to the specified graphics path.

**Namespace:** MindFusion.Gauges
**File:** Segment.js

## Syntax

**JavaScript**

```
function addTo ()
```

## See Also

Segment Members
Segment Class
MindFusion.Gauges Namespace

# Segment Properties

# Segment.isStroked Property

Gets or sets a value that indicates whether the segment is stroked.

**Namespace:** MindFusion.Gauges
**File:** Segment.js

## Syntax

```
JavaScript
get isStroked() {}
set isStroked(value) {}
```

### Property Value

Type: Boolean
*Boolean*

True if the segment is stroked, otherwise false.

## See Also

Segment Members
Segment Class
MindFusion.Gauges Namespace

# Tick Class

Represents a tick mark within a gauge scale.

## Syntax

```
JavaScript
// class
Tick.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Gauges.Tick**

## See Also

Tick Members
MindFusion.Gauges Namespace

## Tick Members

Represents a tick mark within a gauge scale.

The following tables list the members exposed by the Tick type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | Tick | Creates an instance of **Tick** class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | rawValue | Gets the value associated with this tick before it was processed by any value distribution functions. |
|  | settings | Gets the TickSettings object associated with this tick. |
|  | value | Gets the value associated with this tick. |

## See Also

Tick Class
MindFusion.Gauges Namespace

# Tick Constructor

Creates an instance of Tick class.

**Namespace:** MindFusion.Gauges
**File:** Tick.js

## Syntax

**JavaScript**

```
function Tick (settings)
```

### Parameters

*settings*
The TickSettings object associated with this tick.

## See Also

Tick Class
MindFusion.Gauges Namespace

# Tick Properties

# Tick.rawValue Property

Gets the value associated with this tick before it was processed by any value distribution functions.

**Namespace:** MindFusion.Gauges
**File:** Tick.js

## Syntax

**JavaScript**

```
get rawValue() {}
```

### Property Value

Type: Number
*Number*

The value associated with this tick before it was processed by any value distribution functions.

## See Also

Tick Members
Tick Class
MindFusion.Gauges Namespace

# Tick.settings Property

Gets the TickSettings object associated with this tick.

**Namespace:** MindFusion.Gauges
**File:** Tick.js

## Syntax

**JavaScript**

```
get settings() {}
set settings(value) {}
```

### Property Value

Type: TickSettings
*TickSettings*

The TickSettings object associated with this tick.

## See Also

Tick Members
Tick Class
MindFusion.Gauges Namespace

# Tick.value Property

Gets the value associated with this tick.

**Namespace:** MindFusion.Gauges
**File:** Tick.js

## Syntax

**JavaScript**

```
get value() {}
```

### Property Value

Type: Number
*Number*

The value associated with this tick.

## See Also

Tick Members
Tick Class
MindFusion.Gauges Namespace

# TickLabel Class

Represents a tick label within a gauge scale.

## Syntax

**JavaScript**

```
// class
TickLabel.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Gauges.TickLabel**

## See Also

TickLabel Members
MindFusion.Gauges Namespace

# TickLabel Members

Represents a tick label within a gauge scale.

The following tables list the members exposed by the TickLabel type.

## Public Constructors

| | Name | Description |
|---|---|---|
|  | TickLabel | Creates an instance of **TickLabel** class. |

## Public Properties

| | Name | Description |
|---|---|---|
|  | foreground | Gets the foreground color of this label. |
|  | rawValue | Gets the value associated with this tick before it was processed by any value distribution functions. |
|  | settings | Gets the TickSettings object associated with this label. |
|  | value | Gets the value associated with this label. |

## See Also

# TickLabel Constructor

Creates an instance of TickLabel class.

**Namespace:** MindFusion.Gauges
**File:** TickLabel.js

## Syntax

```
JavaScript
function TickLabel (settings)
```

### Parameters

*settings*
Type: TickSettings
TickSettings. The TickSettings object associated with this label.

## See Also

TickLabel Class
MindFusion.Gauges Namespace

# TickLabel Properties

# TickLabel.foreground Property

Gets the foreground color of this label.

**Namespace:** MindFusion.Gauges
**File:** TickLabel.js

## Syntax

```
JavaScript
get foreground() {}
```

### Property Value

Type: Color
*Color*

The foreground color of this label.

## See Also

TickLabel Members
TickLabel Class
MindFusion.Gauges Namespace

# TickLabel.rawValue Property

Gets the value associated with this tick before it was processed by any value distribution functions.

**Namespace:** MindFusion.Gauges
**File:** TickLabel.js

## Syntax

```
JavaScript
get rawValue() {}
```

### Property Value

Type: Number
*Number*

The value associated with this tick before it was processed by any value distribution functions.

## See Also

TickLabel Members
TickLabel Class
MindFusion.Gauges Namespace

# TickLabel.settings Property

Gets the TickSettings object associated with this label.

**Namespace:** MindFusion.Gauges
**File:** TickLabel.js

## Syntax

```
JavaScript
get settings() {}
```

### Property Value

Type: TickSettings
*TickSettings*

The TickSettings object associated with this label.

## See Also

TickLabel Members
TickLabel Class
MindFusion.Gauges Namespace

# TickLabel.value Property

Gets the value associated with this label.

**Namespace:** MindFusion.Gauges
**File:** TickLabel.js

## Syntax

| JavaScript |
| --- |
| `get value() {}` |

### Property Value

Type: Number
*Number*

The value associated with this label.

## See Also

TickLabel Members
TickLabel Class
MindFusion.Gauges Namespace

# TickSettings Class

Provides settings for the ticks and labels in a gauge scale.

## Syntax

| JavaScript |
| --- |
| `// class`<br>`TickSettings.prototype = {}` |

## Inheritance Hierarchy

**MindFusion.Gauges.TickSettings**

## See Also

TickSettings Members
MindFusion.Gauges Namespace

# TickSettings Members

Provides settings for the ticks and labels in a gauge scale.

The following tables list the members exposed by the TickSettings type.

## Public Constructors

| | Name | Description |
| --- | --- | --- |
|  | TickSettings | Creates an instance of **TickSettings** class. |

## Public Properties

| | Name | Description |
| --- | --- | --- |

| | | |
|---|---|---|
| | count | Gets or sets the number of ticks and labels. |
| | customIntervals | Gets an array of CustomInterval objects, which can be used to customize the appearance of the ticks in a particular interval. |
| | fill | Gets or sets the fill brush of the ticks. |
| | fontFamily | Gets or sets the font family of the labels. |
| | fontSize | Gets or sets the font size of the labels. |
| | fontStyle | Gets or sets the font style of the labels. |
| | labelAlignment | Gets or sets the alignment of the labels relative to the scale. |
| | labelForeground | Gets or sets the brush used to paint label texts. |
| | labelOffset | Gets or sets the offset of the labels relative to their calculated position. |
| | labelRotation | Gets or sets the rotation mode of the labels. |
| | numberPrecision | Gets or sets the number recision of the labels. |
| | showLabels | Gets or sets a value indicating whether the labels are visible. |
| | showMaxValueTick | Gets or sets a value indicating whether to display a tick for the maxValue of the scale. |
| | showTicks | Gets or sets a value indicating whether the ticks are visible. |
| | step | Gets or sets the value between adjacent ticks and labels. |
| | stroke | Gets or sets the stroke brush of the ticks. |
| | tickAlignment | Gets or sets the alignment of the ticks relative to the scale. |
| | tickHeight | Gets or sets the height of a tick. |
| | tickOffset | Gets or sets the offset of the ticks relative to their calculated position. |
| | tickShape | Gets or sets the shape of the ticks. |
| | tickWidth | Gets or sets the width of a tick. |

## Public Methods

| | Name | Description |
|---|---|---|
| | addCustomInterval | Adds a CustomInterval object to these settings' customIntervals collection. |
| | removeCustomInterval | Removes a CustomInterval object from these settings' customIntervals collection. |

## See Also

TickSettings Class
MindFusion.Gauges Namespace

# TickSettings Constructor

Creates an instance of TickSettings class.

**Namespace:** MindFusion.Gauges
**File:** TickSettings.js

## Syntax

**JavaScript**

```
function TickSettings (parent)
```

### Parameters

*parent*
Type: BaseScale
BaseScale. The scale these settings belong to.

## See Also

TickSettings Class
MindFusion.Gauges Namespace

# TickSettings Methods

# TickSettings.addCustomInterval Method

Adds a CustomInterval object to these settings' customIntervals collection.

**Namespace:** MindFusion.Gauges
**File:** TickSettings.js

## Syntax

**JavaScript**

```
function addCustomInterval (interval)
```

### Parameters

*interval*
Type: CustomInterval
CustomInterval. The interval to add.

## See Also

TickSettings Members
TickSettings Class
MindFusion.Gauges Namespace

# TickSettings.removeCustomInterval Method

Removes a CustomInterval object from these settings' customIntervals collection.

**Namespace:** MindFusion.Gauges
**File:** TickSettings.js

## Syntax

```
JavaScript
function removeCustomInterval (interval)
```

### Parameters

*interval*
Type: CustomInterval
CustomInterval. The interval to remove.

## See Also

TickSettings Members
TickSettings Class
MindFusion.Gauges Namespace

# TickSettings Properties

# TickSettings.count Property

Gets or sets the number of ticks and labels.

**Namespace:** MindFusion.Gauges
**File:** TickSettings.js

## Syntax

```
JavaScript
get count() {}
set count(value) {}
```

### Property Value

Type: Number
*Number*

The number of ticks and labels.

## Remarks

This property is mutually exclusive with the Step property. If both are specified the Step property will take precedence.

## See Also

TickSettings Members
TickSettings Class
MindFusion.Gauges Namespace

# TickSettings.customIntervals Property

Gets an array of CustomInterval objects, which can be used to customize the appearance of the ticks in a particular interval.

**Namespace:** MindFusion.Gauges
**File:** TickSettings.js

## Syntax

**JavaScript**

```
get customIntervals() {}
set customIntervals(value) {}
```

### Property Value

Type: Array
*Array*

An array of CustomInterval objects, which can be used to customize the appearance of the ticks in a particular interval.

## See Also

TickSettings Members
TickSettings Class
MindFusion.Gauges Namespace

# TickSettings.fill Property

Gets or sets the fill brush of the ticks.

**Namespace:** MindFusion.Gauges
**File:** TickSettings.js

## Syntax

**JavaScript**

```
get fill() {}
set fill(value) {}
```

### Property Value

Type: Object
*Object*

The fill brush of the ticks.

## See Also

TickSettings Members
TickSettings Class
MindFusion.Gauges Namespace

# TickSettings.fontFamily Property

Gets or sets the font family of the labels.

**Namespace:** MindFusion.Gauges
**File:** TickSettings.js

## Syntax

```
JavaScript
get fontFamily() {}
set fontFamily(value) {}
```

### Property Value

Type: String
*String*

The font family of the labels.

## See Also

TickSettings Members
TickSettings Class
MindFusion.Gauges Namespace

# TickSettings.fontSize Property

Gets or sets the font size of the labels.

**Namespace:** MindFusion.Gauges
**File:** TickSettings.js

## Syntax

```
JavaScript
get fontSize() {}
set fontSize(value) {}
```

### Property Value

Type: Length
*Length*

The font size of the labels.

## See Also

TickSettings Members
TickSettings Class
MindFusion.Gauges Namespace

# TickSettings.fontStyle Property

Gets or sets the font style of the labels.

**Namespace:** MindFusion.Gauges
**File:** TickSettings.js

## Syntax

```
JavaScript
get fontStyle() {}
set fontStyle(value) {}
```

### Property Value

Type: String
*String*

The font style of the labels.

## See Also

TickSettings Members
TickSettings Class
MindFusion.Gauges Namespace

# TickSettings.labelAlignment Property

Gets or sets the alignment of the labels relative to the scale.

**Namespace:** MindFusion.Gauges
**File:** TickSettings.js

## Syntax

```
JavaScript
get labelAlignment() {}
set labelAlignment(value) {}
```

### Property Value

Type: Alignment
*Alignment*

The alignment of the labels relative to the scale.

## See Also

TickSettings Members
TickSettings Class
MindFusion.Gauges Namespace

# TickSettings.labelForeground Property

Gets or sets the brush used to paint label texts.

**Namespace:** MindFusion.Gauges
**File:** TickSettings.js

## Syntax

```
JavaScript
get labelForeground() {}
set labelForeground(value) {}
```

### Property Value

Type: Object
*Object*

The brush used to paint label texts.

## See Also

TickSettings Members
TickSettings Class
MindFusion.Gauges Namespace

# TickSettings.labelOffset Property

Gets or sets the offset of the labels relative to their calculated position.

**Namespace:** MindFusion.Gauges
**File:** TickSettings.js

## Syntax

```
JavaScript
get labelOffset() {}
set labelOffset(value) {}
```

### Property Value

Type: Length
*Length*

The offset of the labels relative to their calculated position.

## See Also

TickSettings Members
TickSettings Class
MindFusion.Gauges Namespace

# TickSettings.labelRotation Property

Gets or sets the rotation mode of the labels.

**Namespace:** MindFusion.Gauges
**File:** TickSettings.js

## Syntax

```
JavaScript
get labelRotation() {}
set labelRotation(value) {}
```

### Property Value

Type: LabelRotation
*LabelRotation*

The rotation mode of the labels.

## See Also

TickSettings Members
TickSettings Class
MindFusion.Gauges Namespace

# TickSettings.numberPrecision Property

Gets or sets the number recision of the labels.

**Namespace:** MindFusion.Gauges
**File:** TickSettings.js

## Syntax

**JavaScript**

```
get numberPrecision() {}
set numberPrecision(value) {}
```

### Property Value

Type: Number
*Number*

The number recision of the labels.

## See Also

TickSettings Members
TickSettings Class
MindFusion.Gauges Namespace

# TickSettings.showLabels Property

Gets or sets a value indicating whether the labels are visible.

**Namespace:** MindFusion.Gauges
**File:** TickSettings.js

## Syntax

**JavaScript**

```
get showLabels() {}
set showLabels(value) {}
```

### Property Value

Type: Boolean
*Boolean*

True if labels are visible, otherwise false.

## See Also

TickSettings Members
TickSettings Class
MindFusion.Gauges Namespace

# TickSettings.showMaxValueTick Property

Gets or sets a value indicating whether to display a tick for the maxValue of the scale.

**Namespace:** MindFusion.Gauges
**File:** TickSettings.js

## Syntax

**JavaScript**

```
get showMaxValueTick() {}
set showMaxValueTick(value) {}
```

### Property Value

Type: Boolean
*Boolean*

True to display a tick for the maxValue, otherwise false.

## See Also

TickSettings Members
TickSettings Class
MindFusion.Gauges Namespace

# TickSettings.showTicks Property

Gets or sets a value indicating whether the ticks are visible.

**Namespace:** MindFusion.Gauges
**File:** TickSettings.js

## Syntax

**JavaScript**

```
get showTicks() {}
set showTicks(value) {}
```

### Property Value

Type: Boolean
*Boolean*

True if ticks are visible, otherwise false.

## See Also

TickSettings Members
TickSettings Class
MindFusion.Gauges Namespace

# TickSettings.step Property

Gets or sets the value between adjacent ticks and labels.

**Namespace:** MindFusion.Gauges
**File:** TickSettings.js

## Syntax

```
JavaScript
get step() {}
set step(value) {}
```

### Property Value

Type: Number
*Number*

The value between adjacent ticks and labels.

## Remarks

This property is mutually exclusive with the Count property. If both are specified the Step property will take precedence.

## See Also

TickSettings Members
TickSettings Class
MindFusion.Gauges Namespace

# TickSettings.stroke Property

Gets or sets the stroke brush of the ticks.

**Namespace:** MindFusion.Gauges
**File:** TickSettings.js

## Syntax

```
JavaScript
get stroke() {}
set stroke(value) {}
```

### Property Value

Type: Object
*Object*

The stroke brush of the ticks.

## See Also

TickSettings Members
TickSettings Class
MindFusion.Gauges Namespace

# TickSettings.tickAlignment Property

Gets or sets the alignment of the ticks relative to the scale.

**Namespace:** MindFusion.Gauges
**File:** TickSettings.js

## Syntax

**JavaScript**

```
get tickAlignment() {}
set tickAlignment(value) {}
```

### Property Value

Type: Alignment
*Alignment*

The alignment of the ticks relative to the scale.

## See Also

TickSettings Members
TickSettings Class
MindFusion.Gauges Namespace

# TickSettings.tickHeight Property

Gets or sets the height of a tick.

**Namespace:** MindFusion.Gauges
**File:** TickSettings.js

## Syntax

**JavaScript**

```
get tickHeight() {}
set tickHeight(value) {}
```

### Property Value

Type: Length
*Length*

The height of a tick.

## See Also

TickSettings Members
TickSettings Class
MindFusion.Gauges Namespace

# TickSettings.tickOffset Property

Gets or sets the offset of the ticks relative to their calculated position.

**Namespace:** MindFusion.Gauges
**File:** TickSettings.js

## Syntax

**JavaScript**

```
get tickOffset() {}
set tickOffset(value) {}
```

### Property Value

Type: Length
*Length*

The offset of the ticks relative to their calculated position.

## See Also

TickSettings Members
TickSettings Class
MindFusion.Gauges Namespace

# TickSettings.tickShape Property

Gets or sets the shape of the ticks.

**Namespace:** MindFusion.Gauges
**File:** TickSettings.js

## Syntax

**JavaScript**

```
get tickShape() {}
set tickShape(value) {}
```

### Property Value

Type: TickShape
*TickShape*

The shape of the ticks.

## See Also

TickSettings Members
TickSettings Class
MindFusion.Gauges Namespace

# TickSettings.tickWidth Property

Gets or sets the width of a tick.

**Namespace:** MindFusion.Gauges
**File:** TickSettings.js

## Syntax

```
JavaScript
get tickWidth() {}
set tickWidth(value) {}
```

### Property Value

Type: Length
*Length*

The width of a tick.

## See Also

TickSettings Members
TickSettings Class
MindFusion.Gauges Namespace

# TickShape Enumeration

Indicates the shape of a scale tick.

**Namespace:** MindFusion.Gauges
**File:** Enum.js

## Syntax

```
JavaScript
// enum
TickShape = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **Arrow1** | Specifies an arrow shape. |
| | **Arrow2** | Specifies an arrow shape. |
| | **Arrow3** | Specifies an arrow shape. |
| | **Custom** | Specifies a custom shape. |
| | **Ellipse** | Specifies an elliptical shape. |
| | **Line** | Specifies a line as a shape. |
| | **None** | Specifies an empty shape, that is, the tick is not displayed. |
| | **Rectangle** | Specifies a rectangular shape. |
| | **Rhombus** | Specifies a rhombus as a shape. |

| | | |
|---|---|---|
| | **RoundRect** | Specifies a rounded rectangle as a shape. |
| | **Triangle** | Specifies a triangular shape. |

## See Also

MindFusion.Gauges Namespace

# TickType Enumeration

Indicates the type of a scale tick.

**Namespace:** MindFusion.Gauges
**File:** Enum.js

## Syntax

**JavaScript**

```
// enum
TickType = {}
```

## Members

| | Member name | Description |
|---|---|---|
| | **Major** | Specifies a major tick. |
| | **Middle** | Specifies a middle tick. |
| | **Minor** | Specifies a minor tick. |
| | **None** | Specifies ticks of an unknown type. |

## See Also

MindFusion.Gauges Namespace

# Utils Class

Contains helper functions.

## Syntax

**JavaScript**

```
// class
Utils.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Gauges.Utils**

## See Also

# Utils Members

Contains helper functions.

The following tables list the members exposed by the Utils type.

## Public Methods

| | Name | Description |
|---|---|---|
|  | createLinearGradient | Creates a linear gradient brush with the specified data. |
|  | createRadialGradient | Creates a radial gradient brush with the specified data. |

## See Also

Utils Class
MindFusion.Gauges Namespace

# Utils Methods

# Utils.createLinearGradient Method

Creates a linear gradient brush with the specified data.

**Namespace:** MindFusion.Gauges
**File:** Utils.js

## Syntax

| JavaScript |
|---|
| `function createLinearGradient ()` |

### Return Value

Type: Object
Object. The brush.

## See Also

Utils Members
Utils Class
MindFusion.Gauges Namespace

# Utils.createRadialGradient Method

Creates a radial gradient brush with the specified data.

**Namespace:** MindFusion.Gauges
**File:** Utils.js

## Syntax

| JavaScript |
| --- |
| `function createRadialGradient ()` |

### Return Value

Type: Object
Object. The brush.

## See Also

Utils Members
Utils Class
MindFusion.Gauges Namespace

# ValueChangedEventArgs Class

Contains the arguments passed to value changed notification handlers.

## Syntax

| JavaScript |
| --- |
| `// class`<br>`ValueChangedEventArgs.prototype = {}` |

## Inheritance Hierarchy

**MindFusion.Gauges.ValueChangedEventArgs**

## See Also

ValueChangedEventArgs Members
MindFusion.Gauges Namespace

# ValueChangedEventArgs Members

Contains the arguments passed to value changed notification handlers.

The following tables list the members exposed by the ValueChangedEventArgs type.

## Public Constructors

| | Name | Description |
| --- | --- | --- |
|  | ValueChangedEventArgs | Creates an instance of **ValueChangedEventArgs** class. |

## Public Properties

| | Name | Description |
| --- | --- | --- |
|  | newValue | Gets the new value of the changed property. |

| | | |
|---|---|---|
|  | oldValue | Gets the previous value of the changed property. |

## See Also

ValueChangedEventArgs Class
MindFusion.Gauges Namespace

# ValueChangedEventArgs Constructor

Creates an instance of ValueChangedEventArgs class.

**Namespace:** MindFusion.Gauges
**File:** EventArgs.js

## Syntax

**JavaScript**

```
function ValueChangedEventArgs ()
```

## See Also

ValueChangedEventArgs Class
MindFusion.Gauges Namespace

# ValueChangedEventArgs Properties

# ValueChangedEventArgs.newValue Property

Gets the new value of the changed property.

**Namespace:** MindFusion.Gauges
**File:** EventArgs.js

## Syntax

**JavaScript**

```
get newValue() {}
set newValue(value) {}
```

### Property Value

Type: Object
*Object*

The new value of the changed property.

## See Also

ValueChangedEventArgs Members
ValueChangedEventArgs Class
MindFusion.Gauges Namespace

## ValueChangedEventArgs.oldValue Property

Gets the previous value of the changed property.

**Namespace:** MindFusion.Gauges
**File:** EventArgs.js

### Syntax

**JavaScript**

```
get oldValue() {}
set oldValue(value) {}
```

#### Property Value

Type: Object
*Object*

The previous value of the changed property.

### See Also

ValueChangedEventArgs Members
ValueChangedEventArgs Class
MindFusion.Gauges Namespace

# ValueChangingEventArgs Class

Contains the arguments passed to value changing notification handlers.

### Syntax

**JavaScript**

```
// class
ValueChangingEventArgs.prototype = {}
```

### Inheritance Hierarchy

**MindFusion.Gauges.ValueChangingEventArgs**

### See Also

ValueChangingEventArgs Members
MindFusion.Gauges Namespace

## ValueChangingEventArgs Members

Contains the arguments passed to value changing notification handlers.

The following tables list the members exposed by the ValueChangingEventArgs type.

### Public Constructors

| | Name | Description |
|---|---|---|

| | | |
|---|---|---|
| | ValueChangingEventArgs | Creates an instance of **ValueChangingEventArgs** class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | newValue | Gets the new value of the changing property. |
| | oldValue | Gets the previous value of the changing property. |

## See Also

ValueChangingEventArgs Class
MindFusion.Gauges Namespace

# ValueChangingEventArgs Constructor

Creates an instance of ValueChangingEventArgs class.

**Namespace:** MindFusion.Gauges
**File:** EventArgs.js

## Syntax

**JavaScript**

```
function ValueChangingEventArgs ()
```

## See Also

ValueChangingEventArgs Class
MindFusion.Gauges Namespace

# ValueChangingEventArgs Properties

# ValueChangingEventArgs.newValue Property

Gets the new value of the changing property.

**Namespace:** MindFusion.Gauges
**File:** EventArgs.js

## Syntax

**JavaScript**

```
get newValue() {}
set newValue(value) {}
```

### Property Value

Type: Object
*Object*

The new value of the changing property.

## See Also

ValueChangingEventArgs Members
ValueChangingEventArgs Class
MindFusion.Gauges Namespace

# ValueChangingEventArgs.oldValue Property

Gets the previous value of the changing property.

**Namespace:** MindFusion.Gauges
**File:** EventArgs.js

## Syntax

**JavaScript**

```
get oldValue() {}
set oldValue(value) {}
```

### Property Value

Type: Object
*Object*

The previous value of the changing property.

## See Also

ValueChangingEventArgs Members
ValueChangingEventArgs Class
MindFusion.Gauges Namespace

# VisualElement Class

Represents an object with outline and fill, which can render itself to a canvas.

## Syntax

**JavaScript**

```
// class
VisualElement.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Gauges.VisualElement**

## See Also

VisualElement Members
MindFusion.Gauges Namespace

# VisualElement Members

Represents an object with outline and fill, which can render itself to a canvas.

The following tables list the members exposed by the VisualElement type.

## Public Constructors

| | Name | Description |
|---|---|---|
| | VisualElement | Creates an instance of VisualElement class. |

## Public Properties

| | Name | Description |
|---|---|---|
| | bounds | Gets or sets the bounding rectangle of this element. |
| | fill | Gets or sets the fill of this element. |
| | isVisible | Gets or sets a value indicating whether this element is visible. |
| | margin | Gets or sets the margin of this element. |
| | name | Gets or sets the name of this element. |
| | relativeCoordinates | Gets or sets a value indicating whether the metrics of this element are expressed as relative or absolute quantities. |
| | renderSize | Gets or sets the rendering size of the element. |
| | stroke | Gets or sets the stroke of this element. |
| | x | Gets or sets the normalized x position of this element, relative to its parent. |
| | y | Gets or sets the normalized y position of this element, relative to its parent. |

## Public Methods

| | Name | Description |
|---|---|---|
| | getSize | Gets the size of this element. |

## See Also

VisualElement Class
MindFusion.Gauges Namespace

# VisualElement Constructor

Creates an instance of VisualElement class.

**Namespace:** MindFusion.Gauges
**File:** VisualElement.js

## Syntax

```
JavaScript
function VisualElement ()
```

## See Also

VisualElement Class
MindFusion.Gauges Namespace

# VisualElement Methods

## VisualElement.getSize Method

Gets the size of this element.

**Namespace:** MindFusion.Gauges
**File:** VisualElement.js

## Syntax

```
JavaScript
function getSize ()
```

### Return Value

Type: Size
Size. The size of this element.

## See Also

VisualElement Members
VisualElement Class
MindFusion.Gauges Namespace

# VisualElement Properties

## VisualElement.bounds Property

Gets or sets the bounding rectangle of this element.

**Namespace:** MindFusion.Gauges
**File:** VisualElement.js

## Syntax

```
JavaScript
get bounds() {}
set bounds(value) {}
```

### Property Value

Type: Rect
*Rect*

The bounding rectangle of this element.

## See Also

VisualElement Members
VisualElement Class
MindFusion.Gauges Namespace

# VisualElement.fill Property

Gets or sets the fill of this element.

**Namespace:** MindFusion.Gauges
**File:** VisualElement.js

## Syntax

**JavaScript**

```
get fill() {}
set fill(value) {}
```

### Property Value

Type: Object
*Object*

The fill of this element.

## See Also

VisualElement Members
VisualElement Class
MindFusion.Gauges Namespace

# VisualElement.isVisible Property

Gets or sets a value indicating whether this element is visible.

**Namespace:** MindFusion.Gauges
**File:** VisualElement.js

## Syntax

**JavaScript**

```
get isVisible() {}
set isVisible(value) {}
```

### Property Value

Type: Boolean
*Boolean*

True if the element is visible, otherwise false.

## See Also

VisualElement Members
VisualElement Class
MindFusion.Gauges Namespace

# VisualElement.margin Property

Gets or sets the margin of this element.

**Namespace:** MindFusion.Gauges
**File:** VisualElement.js

## Syntax

**JavaScript**

```
get margin() {}
set margin(value) {}
```

### Property Value

Type: Thickness
*Thickness*

The margin of this element.

## See Also

VisualElement Members
VisualElement Class
MindFusion.Gauges Namespace

# VisualElement.name Property

Gets or sets the name of this element.

**Namespace:** MindFusion.Gauges
**File:** VisualElement.js

## Syntax

**JavaScript**

```
get name() {}
set name(value) {}
```

### Property Value

Type: String
*String*

The name of this element.

## Remarks

The name can be used to obtain a reference to the element through a call to the getElementByName method.

## See Also

VisualElement Members
VisualElement Class
MindFusion.Gauges Namespace

## VisualElement.relativeCoordinates Property

Gets or sets a value indicating whether the metrics of this element are expressed as relative or absolute quantities.

**Namespace:** MindFusion.Gauges
**File:** VisualElement.js

## Syntax

**JavaScript**

```
get relativeCoordinates() {}
set relativeCoordinates(value) {}
```

### Property Value

Type: Boolean
*Boolean*

True if the metrics are relative, otherwise false.

## See Also

VisualElement Members
VisualElement Class
MindFusion.Gauges Namespace

## VisualElement.renderSize Property

Gets or sets the rendering size of the element.

**Namespace:** MindFusion.Gauges
**File:** VisualElement.js

## Syntax

**JavaScript**

```
get renderSize() {}
set renderSize(value) {}
```

### Property Value

Type: Size
*Size*

The rendering size of the element.

## See Also

VisualElement Members
VisualElement Class
MindFusion.Gauges Namespace

# VisualElement.stroke Property

Gets or sets the stroke of this element.

**Namespace:** MindFusion.Gauges
**File:** VisualElement.js

## Syntax

**JavaScript**

```
get stroke() {}
set stroke(value) {}
```

### Property Value

Type: Object
*Object*

The stroke of this element.

## See Also

VisualElement Members
VisualElement Class
MindFusion.Gauges Namespace

# VisualElement.x Property

Gets or sets the normalized x position of this element, relative to its parent.

**Namespace:** MindFusion.Gauges
**File:** VisualElement.js

## Syntax

**JavaScript**

```
get x() {}
set x(value) {}
```

### Property Value

Type: Number
*Number*

The normalized x position of this element, relative to its parent.

## See Also

VisualElement Members
VisualElement Class
MindFusion.Gauges Namespace

## VisualElement.y Property

Gets or sets the normalized y position of this element, relative to its parent.

**Namespace:** MindFusion.Gauges
**File:** VisualElement.js

### Syntax

**JavaScript**

```
get y() {}
set y(value) {}
```

#### Property Value

Type: Number
*Number*

The normalized y position of this element, relative to its parent.

### See Also

VisualElement Members
VisualElement Class
MindFusion.Gauges Namespace

# VisualElementContainer Class

Represents a visual element which contains other visual elements.

### Syntax

**JavaScript**

```
// class
VisualElementContainer.prototype = {}
```

### Inheritance Hierarchy

**MindFusion.Gauges.VisualElementContainer**

### See Also

VisualElementContainer Members
MindFusion.Gauges Namespace

## VisualElementContainer Members

Represents a visual element which contains other visual elements.

The following tables list the members exposed by the VisualElementContainer type.

### Public Constructors

| | Name | Description |
|---|---|---|

| | | |
|---|---|---|
| | VisualElementContainer | Creates an instance of VisualElementContainer class. |

## See Also

VisualElementContainer Class
MindFusion.Gauges Namespace

### VisualElementContainer Constructor

Creates an instance of VisualElementContainer class.

**Namespace:** MindFusion.Gauges
**File:** VisualElementContainer.js

## Syntax

**JavaScript**

```
function VisualElementContainer ()
```

## See Also

VisualElementContainer Class
MindFusion.Gauges Namespace

# WebComponents Class

Implements registration of web components.

## Syntax

**JavaScript**

```
// class
WebComponents.prototype = {}
```

## Inheritance Hierarchy

**MindFusion.Gauges.WebComponents**

## See Also

WebComponents Members
MindFusion.Gauges Namespace

### WebComponents Members

Implements registration of web components.

The following tables list the members exposed by the WebComponents type.

## Public Methods

| | Name | Description |
|---|---|---|
|  | register | Registers the controls in the package as web components. |

## See Also

WebComponents Class
MindFusion.Gauges Namespace

# WebComponents Methods

## WebComponents.register Method

Registers the controls in the package as web components.

**Namespace:** MindFusion.Gauges
**File:** WebComponents.js

## Syntax

| JavaScript |
|---|
| `function register ()` |

## Remarks

Call this method to register the control classes from Gauges package as web components. After registration, the browser instantiates corresponding classes for following tags in HTML code:

| HTML | JS Class |
|---|---|
| <mindfusion-lineargauge> | LinearGauge |
| <mindfusion-ovalgauge> | OvalGauge |

You can get a reference to objects created by the browser by calling the find method of corresponding class with **id** attribute's value as argument.

## See Also

WebComponents Members
WebComponents Class
MindFusion.Gauges Namespace
LinearGauge.create Method
OvalGauge.create Method

# Online Resources

### Video Tutorials

- Getting started with the Js Chart
- A funnel chart in JavaScript
- A JavaScript application for server load monitoring

### Blog Posts

- A funnel chart in JavaScript
- Using MindFusion JavaScript Components With WordPress Elementor Plugin

# Technical Support

Detailed information about everything connected with software products offered by MindFusion - online documentation, demos, information about forthcoming products can be found at https://mindfusion.dev/. The site is updated every day, so make sure you visit it regularly to catch up with the newest from us.

If you need product support, MindFusion employs a dedicated and technically savvy support team. This guarantees that all your questions and inquiries will be answered in quick and professional manner. You can use e-mail support@mindfusion.eu or telephone +359 888 951737.

You can post questions on the MindFusion.Charting for JavaScript discussion board:

https://mindfusion.dev/Forum/YaBB.pl?board=chart_blazor

or visit our help desk:

https://mindfusion.dev/HelpDesk/index.php

We will be happy to hear any suggestions and recommendations you might have regarding our products. We do take in account the opinion of every single person who has shown interest in our software.

# Licensing

For non-commercial use, you can disable evaluation mode by adding an https://mindfusion.dev/ link to your page, and assigning the link's id value to the **linkBackId** property of ChartBase (and its *Chart subclasses). For example:

**HTML**

```
<div class="footer">
    ...
    <a id="mindfusionLink" href="https://mindfusion.dev/">h/t MindFusion</a>
</div>
```

**JavaScript**

```
barchart.linkBackId = "mindfusionLink";
```

If you have a commercial license, copy your license key to a mindfusion_lic.txt file in same folder where the chart html page is located. The file name and location can be changed by setting chart's **licenseLocation** property. Alternatively, the key string can be assigned directly to chart's **licenseKey** property.

MindFusion.Charting licenses are perpetual and royalty-free. When you purchase the component, you can find its license key string in confirmation e-mail, and also listed on Keys & Downloads page at our client portal.

# MindFusion Products

MindFusion offers various programming components for different platforms. Below is a list of our products with links to their sections on our website as well link to download a trial version. Clients that have already purchased a product use an increasing discount for each subsequent one.

| Diagramming | Links |
| --- | --- |
| Diagramming for **WPF** | Product Information<br>Download |
| Diagramming for **WinForms** | Product Information<br>Download |
| Diagramming for **Java Swing** | Product Information<br>Download |
| Diagramming for **JavaScript** | Product Information<br>Download |
| Diagramming for **ASP.NET** | Product Information<br>Download |
| Diagramming for **ASP.NET MVC** | Product Information<br>Download |
| Diagramming for **Xamarin** | Product Information<br>Download |
| Diagramming for **UWP** | Product Information<br>Download |
| Diagramming for **Android** | Product Information<br>Download |
| DiagramViewer for **iOS/Cocoa Touch** | Product Information<br>Download |
| Diagramming for **ActiveX** | Product Information<br>Download |
| Diagramming for **Silverlight** | Product Information<br>Download |

| Charting | Links |
| --- | --- |
| Charting for **WPF** | Product Information<br>Download |
| Charting for **WinForms** | Product Information<br>Download |
| Charting for **Java Swing** | Product Information<br>Download |
| Charting for **JavaScript** | Product Information<br>Download |
| Charting for **ASP.NET** | Product Information<br>Download |
| Charting for **Xamarin** | Product Information<br>Download |
| Charting for **Android** | Product Information<br>Download |
| Charting for **Silverlight** | Product Information<br>Download |

| Scheduling | Links |
| --- | --- |
| Scheduling for **WPF** | Product Information<br>Download |
| Scheduling for **WinForms** | Product Information<br>Download |
| Scheduling for **Java Swing** | Product Information<br>Download |
| Scheduling for **JavaScript** | Product Information<br>Download |
| Scheduling for **ASP.NET** | Product Information<br>Download |
| Scheduling for **Xamarin** | Product Information<br>Download |
| Scheduling for **Silverlight** | Product Information<br>Download |

| Spreadsheet | Links |
| --- | --- |
| Spreadsheet for **WPF** | Product Information<br>Download |
| Spreadsheet for **WinForms** | Product Information<br>Download |
| Spreadsheet for **Java Swing** | Product Information<br>Download |

| Reporting | Links |
| --- | --- |
| Reporting for **WPF** | Product Information<br>Download |
| Reporting for **WinForms** | Product Information<br>Download |
| Reporting for **Silverlight** | Product Information<br>Download |

| Virtual Keyboard | Links |
| --- | --- |
| Virtual Keyboard for **WPF** | Product Information<br>Download |
| Virtual Keyboard for **WinForms** | Product Information<br>Download |
| Virtual Keyboard for **Java Swing** | Product Information<br>Download |
| Virtual Keyboard for **JavaScript** | Product Information<br>Download |

| Mapping | Links |
| --- | --- |
| | |

| | |
|---|---|
| Mapping for **WPF** | Product Information<br>Download |
| Mapping for **WinForms** | Product Information<br>Download |
| Mapping for **ASP.NET** | Product Information<br>Download |
| Mapping for **Silverlight** | Product Information<br>Download |

| Suites | Links |
|---|---|
| Pack for **WPF** | Product Information<br>Download |
| Pack for **WinForms** | Product Information<br>Download |
| Pack for **Java Swing** | Product Information<br>Download |
| Pack for **JavaScript** | Product Information<br>Download |
| Pack for **ASP.NET** | Product Information<br>Download |
| Pack for **Xamarin** | Product Information<br>Download |
| Pack for **Silverlight** | Product Information<br>Download |
| Pack for **Diagramming** | Product Information<br>Download |
| Pack for **Scheduling** | Product Information<br>Download |