

Binding

Method Summary

`void` [#setCallBackInfo\(\)](#)
Set callback method for node selection and double click

`void` [#setCheckBoxValueDataprovider\(dataprovider\)](#)
Set checkbox value dataprovider.

`void` [#setChildSortDataprovider\(dataprovider\)](#)
Set the dataprovider name to retrieve column name and sort order for the child nodes.

`void` [#setConfigurationDataprovider\(dataprovider\)](#)
Set configuration dataprovider.

`void` [#setFontTypeDataprovider\(dataprovider\)](#)
Set the dataprovider name to retrieve the node font from

`void` [#setHasCheckBoxDataprovider\(dataprovider\)](#)
Set has checkbox flag dataprovider.

`void` [#setImageMediaDataprovider\(dataprovider\)](#)
Set the dataprovider name to retrieve the node image from (blob column)

`void` [#setImageURLDataprovider\(dataprovider\)](#)
Set the dataprovider name to retrieve the node image from (via url)

`void` [#setMRelationDataprovider\(dataprovider\)](#)
Set m-relation dataprovider.

`void` [#setMRelationName\(relation\)](#)
Set m-relation name

`void` [#setMethodToCallOnCheckBoxChange\(function, returnDataprovider\)](#)
Set method to call on check box status change

`void` [#setMethodToCallOnClick\(function, returnDataprovider\)](#)
Set method to call on click.

`void` [#setMethodToCallOnDoubleClick\(function, returnDataprovider\)](#)
Set method to call on double click.

`void` [#setMethodToCallOnRightClick\(function, returnDataprovider\)](#)
Set method to call on right click.

`void` [#setNRelationDataprovider\(dataprovider\)](#)
Set n-relation dataprovider.

`void` [#setNRelationInfos\(\)](#)
Set n-relation infos (array of RelationInfo objects created using tree.

`void` [#setNRelationName\(relation\)](#)
Set n-relation name

`void` [#setTextDataprovider\(dataprovider\)](#)
Set text dataprovider

`void` [#setToolTipTextDataprovider\(dataprovider\)](#)
Set text dataprovider

Method Details

`setCallBackInfo`

`void` **setCallBackInfo()**

Set callback method for node selection and double click

Returns

`void`

Sample

```
binding.setCallBackInfo(callbackFunction, 'returnDataprovider');
```

`setCheckBoxValueDataprovider`

`void` **setCheckBoxValueDataprovider(dataprovider)**

Set checkbox value dataprovider. Dataprovider returns INTEGER (0 or 1) or STRING (false or true)

Parameters

`dataprovider`

Returns

`void`

Sample

```
binding.setCheckBoxValueDataprovider('dataprovider');
```

setChildSortDataprovider

void **setChildSortDataprovider**(dataprovider)

Set the dataprovider name to retrieve column name and sort order for the child nodes.

The provided data must be a string of form : column_name_used_for_sort sort_order(asc or desc)

Parameters

dataprovider

Returns

void

Sample

```
//Set the dataprovider name to retrieve column name and sort order for the child nodes.  
The provided data must be a string of form : column_name_used_for_sort sort_order(asc or desc)  
binding.setChildSortDataprovider('dataprovider');
```

setConfigurationDataprovider

void **setConfigurationDataprovider**(dataprovider)

Set configuration dataprovider. Dataprovider must be MEDIA type and returns a configuration object

Ex.:

```
var config = new Object();  
config.text = 'my_text';  
config.nRelation = 'my_n_relation';  
config.mRelation = 'my_m_relation';  
config.childSort = 'my_sort_field';  
config.fontType = 'my_font';  
config.toolTipText = 'my_tooltip';  
config.hasCheckBox = 'true';  
config.checkBoxValue = 'true';
```

Parameters

dataprovider

Returns

void

Sample

```
//Set configuration dataprovider. Dataprovider must be MEDIA type and returns a configuration object  
Ex.:  
var config = new Object();  
config.text = 'my_text';  
config.nRelation = 'my_n_relation';  
config.mRelation = 'my_m_relation';  
config.childSort = 'my_sort_field';  
config.fontType = 'my_font';  
config.toolTipText = 'my_tooltip';  
config.hasCheckBox = 'true';  
config.checkBoxValue = 'true';  
  
binding.setConfigurationDataprovider('dataprovider');
```

setFontTypeDataprovider

void **setFontTypeDataprovider**(dataprovider)

Set the dataprovider name to retrieve the node font from

Parameters

dataprovider

Returns

void

Sample

```
binding.setFontTypeDataprovider('dataprovider');
```

setHasCheckBoxDataprovider

void **setHasCheckBoxDataprovider**(dataprovider)

Set has checkbox flag dataprovider. Dataprovider returns INTEGER (0 / 1 / 2) or STRING (false / true / disabled) for (does not have / have / have but disabled)

Parameters

dataprovider

Returns

void

Sample

```
binding.setHasCheckBoxDataprovider('dataprovider');
```

[setImageMediaDataprovider](#)

void **setImageMediaDataprovider**(dataprovider)

Set the dataprovider name to retrieve the node image from (blob column)

Parameters

dataprovider

Returns

void

Sample

```
binding.setImageMediaDataprovider('dataprovider');
```

[setImageURLDataprovider](#)

void **setImageURLDataprovider**(dataprovider)

Set the dataprovider name to retrieve the node image from (via url)

Parameters

dataprovider

Returns

void

Sample

```
binding.setImageURLDataprovider('dataprovider');
```

[setMRelationDataprovider](#)

void **setMRelationDataprovider**(dataprovider)

Set m-relation dataprovider. Dataprovider returns the name of the m-relation

Parameters

dataprovider

Returns

void

Sample

```
binding.setMRelationDataprovider('dataprovider');
```

[setMRelationName](#)

void **setMRelationName**(relation)

Set m-relation name

Parameters

relation

Returns

void

Sample

```
binding.setMRelationName('relation');
```

[setMethodToCallOnCheckBoxChange](#)

void **setMethodToCallOnCheckBoxChange**(function, returnDataprovider)

Set method to call on check box status change

Parameters

function

returnDataprovider

Returns

void

Sample

```
binding.setMethodToCallOnCheckBoxChange(callbackFunction, 'returnDataprovider');
```

setMethodToCallOnClick

void **setMethodToCallOnClick**(function, returnDataprovider)

Set method to call on click.

The callback will be called with the following arguments : returnDataprovider, tableName, mouseX, mouseY

Parameters

function

returnDataprovider

Returns

void

Sample

```
//Set method to call on click.  
The callback will be called with the following arguments : returnDataprovider, tableName, mouseX, mouseY  
binding.setMethodToCallOnClick(callbackFunction, 'returnDataprovider');
```

setMethodToCallOnDoubleClick

void **setMethodToCallOnDoubleClick**(function, returnDataprovider)

Set method to call on double click.

The callback will be called with the following arguments : returnDataprovider, tableName, mouseX, mouseY

Parameters

function

returnDataprovider

Returns

void

Sample

```
//Set method to call on double click.  
The callback will be called with the following arguments : returnDataprovider, tableName, mouseX, mouseY  
binding.setMethodToCallOnDoubleClick(callbackFunction, 'returnDataprovider');
```

setMethodToCallOnRightClick

void **setMethodToCallOnRightClick**(function, returnDataprovider)

Set method to call on right click.

The callback will be called with the following arguments : returnDataprovider, tableName, mouseX, mouseY

Parameters

function

returnDataprovider

Returns

void

Sample

```
//Set method to call on right click.  
The callback will be called with the following arguments : returnDataprovider, tableName, mouseX, mouseY  
binding.setMethodToCallOnRightClick(callbackFunction, 'returnDataprovider');
```

setNRelationDataprovider

void **setNRelationDataprovider**(dataprovider)

Set n-relation dataprovider. Dataprovider returns the name of the n-relation

Parameters

dataprovider

Returns

void

Sample

```
binding.setNRelationDataprovider('dataprovider');
```

setNRelationInfos

void **setNRelationInfos**()

Set n-relation infos (array of RelationInfo objects created using tree.createRelationInfo()) for having multiple child relations for one node)

Returns

void

Sample

```
var company_relations = new Array();
company_relations[0] = tree.createRelationInfo();
company_relations[0].setLabel('Employees');
company_relations[0].setNRelationName('companies_to_employees');
company_relations[1] = tree.createRelationInfo();
company_relations[1].setLabel('Customers');
company_relations[1].setNRelationName('companies_to_customers');
binding.setNRelationInfos(company_relations);
```

setNRelationName

void **setNRelationName**(relation)

Set n-relation name

Parameters

relation

Returns

void

Sample

```
binding.setNRelationName('relation');
```

setTextDataprovider

void **setTextDataprovider**(dataprovder)

Set text dataprovder

Parameters

dataprovder

Returns

void

Sample

```
binding.setTextDataprovider('dataprovder');
```

setToolTipTextDataprovider

void **setToolTipTextDataprovider**(dataprovder)

Set text dataprovder

Parameters

dataprovder

Returns

void

Sample

```
binding.setToolTipTextDataprovider('dataprovder');
```