

controller

Property Summary

Boolean [enabled](#)
Gets or sets the enabled state of a form; also known as "grayed-out".

Method Summary

String [getDataSource\(\)](#)
Get the used datasource.

String [getName\(\)](#)
Get the name of this form.

Number [getSelectedIndex\(\)](#)
Gets the current record index of the current foundset.

void [setSelectedIndex\(index\)](#)
Sets the current record index of the current foundset.

void [show\(\)](#)
Shows the form (makes the form visible)
This function does not affect the form foundset in any way.

void [showRecords\(foundset\)](#)
Load data into the form and shows the form, is a shortcut for the functions 'loadRecords' and 'show'.

Property Details

enabled

Gets or sets the enabled state of a form; also known as "grayed-out".

Notes:

- A disabled element(s) cannot be selected by clicking the form.
- The disabled "grayed" color is dependent on the LAF set in the Servoy Smart Client Application Preferences.

Returns

Boolean

Sample

```
//gets the enabled state of the form
var state = forms.customer.controller.enabled;
//enables the form for input
forms.customer.controller.enabled = true;
```

Method Details

getDataSource

String [getDataSource \(\)](#)

Get the used datasource.

Returns

String - the datasource

Sample

```
var dataSource = forms.customer.controller.getDataSource();
```

getName

`String` **getName** ()

Get the name of this form.

Returns

`String` - the name

Sample

```
var formName = forms.customer.controller.getName();
```

getSelectedIndex

`Number` **getSelectedIndex** ()

Gets the current record index of the current foundset.

Returns

`Number` - the index

Sample

```
//gets the current record index in the current foundset
var current = forms.customer.controller.getSelectedIndex();
//sets the next record in the foundset, will be reflected in UI
forms.customer.controller.setSelectedIndex(current+1);
```

setSelectedIndex

`void` **setSelectedIndex** (index)

Sets the current record index of the current foundset.

Parameters

{`Number`} index - the index to select

Returns

`void`

Sample

```
//gets the current record index in the current foundset
var current = forms.customer.controller.getSelectedIndex();
//sets the next record in the foundset, will be reflected in UI
forms.customer.controller.setSelectedIndex(current+1);
```

show

`void` **show** ()

Shows the form (makes the form visible)

This function does not affect the form foundset in any way.

Returns

`void`

Sample

```
// show the form in the current window/dialog
forms.customer.controller.show();
// show the form in newly created named modal dialog
var w = application.createWindow("mydialog", JSWindow.MODAL_DIALOG);
forms.customer.controller.show(w);
// show the form in an existing window/dialog
var w = application.getWindow("mydialog"); // use null name for main app. window
forms.customer.controller.show(w);
// or forms.customer.controller.show("mydialog");
//show the form in the main window
//forms.customer.controller.show(null);
```

showRecords

void **showRecords** (foundset)

Load data into the form and shows the form, is a shortcut for the functions 'loadRecords' and 'show'.

Parameters

[JSFoundSet](#) foundset - the foundset to load before showing the form.

Returns

void

Sample

```
forms.customer.controller.showRecords(foundset);  
// load foundset & show the form in newly created named modal dialog  
var w = application.createWindow("mydialog", JSWindow.MODAL_DIALOG);  
forms.customer.controller.showRecords(foundset, w);  
// load foundset & show the form in an existing window/dialog  
var w = application.getWindow("mydialog"); // use null name for main app. window  
forms.customer.controller.showRecords(foundset, w);  
//forms.customer.controller.showRecords(foundset, "mydialog");
```