

# RuntimeSplitPane

## Extends

[RuntimeComponent](#)

## Property Summary

|                         |                                                                                                                                                                                    |
|-------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <a href="#">String</a>  | <a href="#">bgcolor</a><br>Gets or sets the background color of a field.                                                                                                           |
| <a href="#">String</a>  | <a href="#">border</a><br>Gets or sets the border attribute(s) of a specified element.                                                                                             |
| <a href="#">Boolean</a> | <a href="#">continuousLayout</a><br>Gets or sets if the components should continuously be redrawn as the divider changes position.                                                 |
| <a href="#">Number</a>  | <a href="#">dividerLocation</a><br>Gets or sets divider location.                                                                                                                  |
| <a href="#">Number</a>  | <a href="#">dividerSize</a><br>Gets or sets divider size in pixels.                                                                                                                |
| <a href="#">Boolean</a> | <a href="#">enabled</a><br>Gets or sets the enabled state of a specified field, also known as "grayed".                                                                            |
| <a href="#">String</a>  | <a href="#">fgcolor</a><br>Gets or sets the foreground color of a field.                                                                                                           |
| <a href="#">String</a>  | <a href="#">font</a><br>Gets or sets the font name, style, and size of an element.                                                                                                 |
| <a href="#">Number</a>  | <a href="#">leftFormMinSize</a><br>Gets or sets left form minimum size in pixels.                                                                                                  |
| <a href="#">Boolean</a> | <a href="#">readOnly</a><br>Gets or sets the editable/read-only state of a field; true - read-only; false - editable; ! - the editable/read-only state is inverted (the opposite). |
| <a href="#">Number</a>  | <a href="#">resizeWeight</a><br>Specifies how to distribute extra space when the size of the split pane changes.                                                                   |
| <a href="#">Number</a>  | <a href="#">rightFormMinSize</a><br>Gets or sets right form minimum size in pixels.                                                                                                |
| <a href="#">String</a>  | <a href="#">toolTipText</a><br>Gets or sets the tool tip text of an element; text displays when the mouse cursor hovers over an element.                                           |
| <a href="#">Boolean</a> | <a href="#">transparent</a><br>Gets or sets the transparency of an element; true - transparent; false - not transparent.                                                           |
| <a href="#">Boolean</a> | <a href="#">visible</a><br>Gets or sets the visibility of an element; true - visible; false - not visible; ! - the visibility state is inverted (the opposite).                    |

## Method Summary

|                             |                                                                                                              |
|-----------------------------|--------------------------------------------------------------------------------------------------------------|
| <a href="#">Number</a>      | <a href="#">getAbsoluteFormLocationY()</a><br>Returns the absolute form (designed) Y location.               |
| <a href="#">Object</a>      | <a href="#">getClientProperty(key)</a><br>Gets the specified client property for the element based on a key. |
| <a href="#">Object</a>      | <a href="#">getDesignTimeProperty()</a><br>Get a design-time property of an element.                         |
| <a href="#">String</a>      | <a href="#">getElementType()</a><br>Returns the type of a specified element.                                 |
| <a href="#">Number</a>      | <a href="#">getHeight()</a><br>Returns the height of the current element.                                    |
| <a href="#">RuntimeForm</a> | <a href="#">getLeftForm()</a><br>Returns the left form of the split pane.                                    |
| <a href="#">Number</a>      | <a href="#">getLocationX()</a><br>Returns the x location of the current element.                             |
| <a href="#">Number</a>      | <a href="#">getLocationY()</a><br>Returns the y location of the current element.                             |
| <a href="#">String</a>      | <a href="#">getMnemonicAt(i)</a><br>Returns the mnemonic for a specified tab of a tabpanel.                  |
| <a href="#">String</a>      | <a href="#">getName()</a><br>Returns the name of an element.                                                 |
| <a href="#">RuntimeForm</a> | <a href="#">getRightForm()</a><br>Returns the right form of the split pane.                                  |
| <a href="#">Number</a>      | <a href="#">getWidth()</a><br>Returns the width of the current element.                                      |

|         |                                                                                                                 |
|---------|-----------------------------------------------------------------------------------------------------------------|
| void    | <a href="#">putClientProperty</a> (key, value)<br>Sets the value for the specified element client property key. |
| Boolean | <a href="#">setLeftForm</a> (form)<br>Set a relationless or related form as left panel.                         |
| Boolean | <a href="#">setLeftForm</a> (form, relation)<br>Set a relationless or related form as left panel.               |
| void    | <a href="#">setLocation</a> (x, y)<br>Sets the location of an element.                                          |
| void    | <a href="#">setMnemonicAt</a> (index, text)<br>Sets the mnemonic for a specified tab in a tabpanel.             |
| Boolean | <a href="#">setRightForm</a> (form)<br>Set a relationless or related form as right panel.                       |
| Boolean | <a href="#">setRightForm</a> (form, relation)<br>Set a relationless or related form as right panel.             |
| void    | <a href="#">setSize</a> (width, height)<br>Sets the size of an element.                                         |

## Property Details

### bgcolor

Gets or sets the background color of a field. The color has to be set using the hexadecimal RGB value as used in HTML.  
It only returns it's correct value if it was explicitly set.

#### Returns

[String](#)

#### Sample

```
//sets the background color of the field
forms.customer.elements.customer_id.bgcolor = "#FFFFFF";
//gets the background color of the field
var c = forms.customer.elements.customer_id.bgcolor;
```

### border

Gets or sets the border attribute(s) of a specified element.

The border attributes:

borderType - EmptyBorder, EtchedBorder, BevelBorder, LineBorder, TitleBorder, MatteBorder, SpecialMatteBorder.

size - (numeric value) for: bottom, left, right, top.

color - (hexadecimal value) for: bottom, left, right, top.

dash pattern - (numeric value) for selected side(s).

rounding radius - (numeric value) for selected side(s).

It only returns it's correct value if it was explicitly set.

NOTE: Use the same value(s) and order of attribute(s) from the element design time property "borderType".

#### Returns

[String](#)

#### Sample

```
//sets the border type to "LineBorder"
//sets a 1 px line width for the bottom and left side of the border
//sets the hexadecimal color of the border to "#ccffcc"
forms.customer.elements.customer_id.border = 'LineBorder,1,#ccffcc';
```

### continuousLayout

Gets or sets if the components should continuously be redrawn as the divider changes position.

#### Returns

[Boolean](#)

#### Sample

```
forms.customer.elements.customer_id.continuousLayout = true;
```

---

**dividerLocation**

Gets or sets divider location.

If location is less than 1 then the location will be considered at (location \* 100) percent of the split pane from left, otherwise it will represent the pixels from left.

**Returns**

Number

**Sample**

```
forms.customer.elements.customer_id.dividerLocation = 0.75;
```

**dividerSize**

Gets or sets divider size in pixels.

**Returns**

Number

**Sample**

```
forms.customer.elements.customer_id.dividerSize = 10;
```

**enabled**

Gets or sets the enabled state of a specified field, also known as "grayed".

true - enabled; false - not enabled; ! - the enabled state is inverted (the opposite).

NOTE: A disabled element cannot be selected by clicking the element (or by pressing the TAB key even if this option is supported by the operating system).

NOTE: A label or button element will not disable if the "displayType" design time property for a field is set to HTML\_AREA.

NOTE: The disabled "grayed" color is dependent on the LAF set in the Servoy Client Application Preferences. For more information see Preferences: Look And Feel in the Servoy Developer User's Guide.

**Returns**

Boolean

**Sample**

```
//gets the enabled state of the field
var currState = forms.customer.elements.customer_id.enabled;

//sets the enabled state of the field
forms.customer.elements.customer_id.enabled = !currentState;
```

**fgcolor**

Gets or sets the foreground color of a field. The color has to be set using the hexadecimal RGB value as used in HTML.

It only returns it's correct value if it was explicitly set.

**Returns**

String

**Sample**

```
//sets the foreground color of the field
forms.customer.elements.customer_id.fgcolor = "#000000";

//gets the foreground color of the field
var c = forms.customer.elements.customer_id.fgcolor;
```

**font**

Gets or sets the font name, style, and size of an element.

font name - the name of the font family.

style - the type of the font. (plain = 0; bold = 1; italic = 2; bold-italic = 3).

size - the size of the font (in points).

It only returns it's correct value if it was explicitly set.

---

**Returns**[String](#)**Sample**

```
forms.customer.elements.customer_id.font = 'Tahoma,1,11';
```

**leftFormMinSize**

Gets or sets left form minimum size in pixels.

**Returns**[Number](#)**Sample**

```
forms.customer.elements.customer_id.leftFormMinSize = 100;
```

**readOnly**

Gets or sets the editable/read-only state of a field; true - read-only; false - editable; ! - the editable/read-only state is inverted (the opposite).

NOTE: A field set as read-only can be selected by clicking (or pressing the TAB key if this option is supported by the operating system) and the field data can be copied.

**Returns**[Boolean](#)**Sample**

```
//gets the editable/read-only state of the field
var currentState = forms.customer.elements.customer_id.readOnly;

//sets the editable/read-only state of the field
forms.customer.elements.customer_id.readOnly = !currentState;
```

**resizeWeight**

Specifies how to distribute extra space when the size of the split pane changes.

A value of 0, the default, indicates the right/bottom component gets all the extra space (the left/top component acts fixed), where as a value of 1 specifies the left/top component gets all the extra space (the right/bottom component acts fixed).

Specifically, the left/top component gets (weight \* diff) extra space and the right/bottom component gets (1 - weight) \* diff extra space

**Returns**[Number](#)**Sample**

```
forms.customer.elements.customer_id.resizeWeight = 0.5;
```

**rightFormMinSize**

Gets or sets right form minimum size in pixels.

**Returns**[Number](#)**Sample**

```
forms.customer.elements.customer_id.rightFormMinSize = 100;
```

**toolTipText**

Gets or sets the tool tip text of an element; text displays when the mouse cursor hovers over an element.

NOTE: HTML should be used for multi-line tooltips; you can also use any valid HTML tags to format tooltip text.

**Returns**[String](#)

**Sample**

```
//gets the tooltip text of the element
var tooltip = forms.customer.elements.customer_id.tooltipText;

//sets the tooltip text of the element
forms.customer.elements.customer_id.tooltipText = "New tip";
forms.customer.elements.customer_id.tooltipText = "<html>This includes <b>bolded text</b> and <font
color='blue'>BLUE</font> text as well.";
```

**transparent**

Gets or sets the transparency of an element; true - transparent; false - not transparent.

NOTE: transparency can be inverted using ! operator: elements.elementName.transparent = !elements.elementName.transparent;

NOTE: transparency will be mostly used for background color, a transparent element will receive the background of the element "beneath" it, a non transparent one will use its own background color

**Returns**

[Boolean](#)

**Sample**

```
//gets the transparency of the element
var currentState = forms.customer.elements.customer_id.transparent;

//sets the transparency of the element
forms.customer.elements.customer_id.transparent = !currentState;
```

**visible**

Gets or sets the visibility of an element; true - visible; false - not visible; ! - the visibility state is inverted (the opposite).

NOTE: The visibility of an element is not persistent; the state of visibility only applies to the current user in his/her current session.

**Returns**

[Boolean](#)

**Sample**

```
//sets the element as visible
forms.company.elements.faxBtn.visible = true;

//gets the visibility of the element
var currentState = forms.company.elements.faxBtn.visible;

//sets the element as not visible when the current state is visible
forms.company.elements.faxBtn.visible = !currentState;
```

**Method Details****getAbsoluteFormLocationY**

[Number](#) **getAbsoluteFormLocationY** ()

Returns the absolute form (designed) Y location.

**Returns**

[Number](#) - The y location of the form in pixels.

**Sample**

```
var absolute_y = forms.customer.elements.customer_id.getAbsoluteFormLocationY();
```

**getClientProperty**

[Object](#) **getClientProperty** (key)

---

Gets the specified client property for the element based on a key.

NOTE: Depending on the operating system, a user interface property name may be available.

#### Parameters

{Object} key - user interface key (depends on operating system)

#### Returns

Object - The value of the property for specified key.

#### Sample

```
var property = forms.customer.elements.customer_id.getClientProperty('ToolTipText');
```

### getDesignTimeProperty

Object getDesignTimeProperty ()

Get a design-time property of an element.

#### Returns

Object

#### Sample

```
var prop = forms.orders.elements.mylabel.getDesignTimeProperty('myprop')
```

### getElementType

String getElementType ()

Returns the type of a specified element.

#### Returns

String - The display type of the element as String.

#### Sample

```
var et = forms.customer.elements.customer_id.getElementType();
```

### getHeight

Number getHeight ()

Returns the height of the current element.

NOTE: getHeight() can be used with getWidth() to set the size of an element using the setSize function. For example:

```
//returns the width (w) and height (h)
var w = forms.company.elements.faxBtn.getWidth();
var h = forms.company.elements.faxBtn.getHeight();

//sets the new size
forms.company.elements.faxBtn.setSize(w,h);

//sets the new size and adds 1 px to both the width and height
forms.company.elements.faxBtn.setSize(w+1,h+1);
```

#### Returns

Number - The height of the element in pixels.

#### Sample

```
var ht = forms.customer.elements.customer_id.getHeight();
```

### getLeftForm

RuntimeForm getLeftForm ()

Returns the left form of the split pane.

#### Returns

RuntimeForm - left form of the split pane

#### Sample

```
var leftForm = forms.customer.elements.customer_id.getLeftForm();
```

### getLocationX

**Number** getLocationX ()

Returns the x location of the current element.

NOTE: getLocationX() can be used with getLocationY() to set the location of an element using the setLocation function. For Example:

```
//returns the X and Y coordinates
var x = forms.company.elements.faxBtn.getLocationX();
var y = forms.company.elements.faxBtn.getLocationY();

//sets the new location 10 px to the right; 10 px down from the current location
forms.company.elements.faxBtn.setLocation(x+10,y+10);
```

#### Returns

**Number** - The x location of the element in pixels.

#### Sample

```
var x = forms.customer.elements.customer_id.getLocationX();
```

### getLocationY

**Number** getLocationY ()

Returns the y location of the current element.

NOTE: getLocationY() can be used with getLocationX() to set the location of an element using the setLocation function. For Example:

```
//returns the X and Y coordinates
var x = forms.company.elements.faxBtn.getLocationX();
var y = forms.company.elements.faxBtn.getLocationY();

//sets the new location 10 px to the right; 10 px down from the current location
forms.company.elements.faxBtn.setLocation(x+10,y+10);
```

#### Returns

**Number** - The y location of the element in pixels.

#### Sample

```
var y = forms.customer.elements.customer_id.getLocationY();
```

### getMnemonicAt

**String** getMnemonicAt (i)

Returns the mnemonic for a specified tab of a tabpanel.

#### Parameters

{**Number**} i - The number of the specified tab.

#### Returns

**String** - The mnemonic.

#### Sample

```
var mnemonic = forms.customer.elements.customer_id.getMnemonicAt(3);
```

### getName

**String** getName ()

Returns the name of an element. (may be null as well)

#### Returns

**String** - The name of the element.

#### Sample

```
var name = forms.customer.elements.customer_id.getName();
```

### getRightForm

**RuntimeForm** getRightForm ()

Returns the right form of the split pane.

**Returns**

[RuntimeForm](#) - right form of the split pane

**Sample**

```
var rightForm = forms.customer.elements.customer_id.getRightForm();
```

**getWidth**

[Number](#) **getWidth ()**

Returns the width of the current element.

NOTE: getWidth() can be used with getHeight() to set the size of an element using the setSize function. For Example:

```
//returns the width (w) and height (h)
var w = forms.company.elements.faxBtn.getWidth();
var h = forms.company.elements.faxBtn.getHeight();

//sets the new size
forms.company.elements.faxBtn.setSize(w,h);

//sets the new size and adds 1 px to both the width and height
forms.company.elements.faxBtn.setSize(w+1,h+1);
```

**Returns**

[Number](#) - The width of the element in pixels.

**Sample**

```
var w = forms.customer.elements.customer_id.getWidth();
```

**putClientProperty**

void **putClientProperty** (key, value)

Sets the value for the specified element client property key.

NOTE: Depending on the operating system, a user interface property name may be available.

**Parameters**

[Object](#) key - user interface key (depends on operating system)  
[Object](#) value - a predefined value for the key

**Returns**

void

**Sample**

```
forms.customer.elements.customer_id.putClientProperty('ToolTipText','some text');
```

**setLeftForm**

[Boolean](#) **setLeftForm** (form)

Set a relationless or related form as left panel.

**Parameters**

[Object](#) form - the specified form or form name you wish to add as left panel

**Returns**

[Boolean](#) - a boolean value indicating if tab was successfully added

**Sample**

```
forms.customer.elements.customer_id.setLeftForm(forms.orders);
```

**setLeftForm**

[Boolean](#) **setLeftForm** (form, relation)

Set a relationless or related form as left panel.

**Parameters**

[Object](#) form - the specified form or form name you wish to add as left panel  
[Object](#) relation - the relation name or a related foundset or null for relationless



**Returns**

**Boolean** - a boolean value indicating if tab was successfully added

**Sample**

```
forms.customer.elements.customer_id.setLeftForm(forms.orders, 'orders_to_order_details');
```

**setLocation**

void **setLocation** (x, y)

Sets the location of an element. It takes as input the X (horizontal) and Y (vertical) coordinates - starting from the TOP LEFT side of the screen. Please note that location should not be altered at runtime when an element is anchored. Use the solutionModel in such a situation.

NOTE: getLocationX() can be used with getLocationY() to return the current location of an element; then use the X and Y coordinates with the setLocation function to set a new location. For Example:

```
//returns the X and Y coordinates
var x = forms.company.elements.faxBtn.getLocationX();
var y = forms.company.elements.faxBtn.getLocationY();

//sets the new location 10 px to the right; 10 px down from the current location
forms.company.elements.faxBtn.setLocation(x+10,y+10);
```

**Parameters**

**{Number}** x - the X coordinate of the element in pixels.  
**{Number}** y - the Y coordinate of the element in pixels.

**Returns**

void

**Sample**

```
forms.customer.elements.customer_id.setLocation(200,200);
```

**setMnemonicAt**

void **setMnemonicAt** (index, text)

Sets the mnemonic for a specified tab in a tabpanel.

**Parameters**

index - the number of the specified tab  
 text - the text to be set for the specified tab

**Returns**

void

**Sample**

```
forms.customer.elements.customer_id.setMnemonicAt(3, 'a');
```

**setRightForm**

**Boolean** **setRightForm** (form)

Set a relationless or related form as right panel.

**Parameters**

**{Object}** form - the specified form or form name you wish to add as right panel

**Returns**

**Boolean** - a boolean value indicating if tab was successfully added

**Sample**

```
forms.customer.elements.customer_id.setRightForm(forms.orders);
```

**setRightForm**

**Boolean** **setRightForm** (form, relation)

Set a relationless or related form as right panel.

**Parameters**

**{Object}** form - the specified form or form name you wish to add as right panel  
**{Object}** relation - the relation name or a related foundset or null for relationless

---

**Returns**

[Boolean](#) - a boolean value indicating if tab was successfully added

**Sample**

```
forms.customer.elements.customer_id.setRightForm(forms.orders, 'orders_to_order_details');
```

**setSize**

void **setSize** (width, height)

Sets the size of an element. It takes as input the width and the height.

Please note that size should not be altered at runtime when an element is anchored. Use the solutionModel in such a situation.

NOTE: getWidth() can be used with getHeight() to set the size of an element using the setSize function. For Example:

```
//returns the width (w) and height (h)
```

```
var w = forms.company.elements.faxBtn.getWidth();
```

```
var h = forms.company.elements.faxBtn.getHeight();
```

```
//sets the new size
```

```
forms.company.elements.faxBtn.setSize(w,h);
```

```
//sets the new size and adds 1 px to both the width and height
```

```
forms.company.elements.faxBtn.setSize(w+1,h+1);
```

**Parameters**

[{Number}](#) width - the width of the element in pixels.

[{Number}](#) height - the height of the element in pixels.

**Returns**

void

**Sample**

```
forms.customer.elements.customer_id.setSize(20,30);
```