

# Portal



Apr 07, 2024 13:20

## Supported Clients

SmartClient WebClient NGClient

## Property Summary

Number	<a href="#">anchors</a>	Enables a component to stick to a specific side of form and/or to grow or shrink when a window is resized.
String	<a href="#">background</a>	The background color of the component.
String	<a href="#">borderType</a>	The type, color and style of border of the component.
CSSPosition	<a href="#">cssPosition</a>	The css position of the component.
Object	<a href="#">designTimeProperties</a>	Property to get and add design-time properties for a component.
Boolean	<a href="#">enabled</a>	The enable state of the component, default true.
Number	<a href="#">formIndex</a>	Form index returns the z-index of the form element.
String	<a href="#">initialSort</a>	The default sort order for the rows displayed in the portal.
String	<a href="#">intercellSpacing</a>	The additional spacing between cell rows.
String	<a href="#">location</a>	The x and y position of the component, in pixels, separated by a comma.
Boolean	<a href="#">multiLine</a>	When set, portal rows can have a custom layout of fields, buttons, etc.
String	<a href="#">name</a>	The name of the component.
Boolean	<a href="#">ngReadOnlyMode</a>	
Number	<a href="#">printSliding</a>	Enables an element to resize based on its content and/or move when printing.
Boolean	<a href="#">printable</a>	Flag that tells if the component should be printed or not when the form is printed.
String	<a href="#">relationName</a>	The name of the relationship between the table related to the currently active form and the table you want to show data from in the portal.
Boolean	<a href="#">reorderable</a>	When set, the portal rows can be re-ordered by dragging the column headers.
Boolean	<a href="#">resizable</a>	When set the portal rows can be resized by users.
String	<a href="#">rowBGColorCalculation</a>	The calculation data provider (such as <code>servoy_row_bgcolor</code> ) used to add background color and highlight selected or alternate portal rows.
Number	<a href="#">rowHeight</a>	The height of each row in pixels.
Number	<a href="#">scrollbars</a>	Scrollbar options for the vertical and horizontal scrollbars.
Boolean	<a href="#">showHorizontalLines</a>	When set, the portal displays horizontal lines between the rows.
Boolean	<a href="#">showVerticalLines</a>	When set the portal displays vertical lines between the columns.
String	<a href="#">size</a>	The width and height (in pixels), separated by a comma.
Boolean	<a href="#">sortable</a>	When set, users can sort the contents of the portal by clicking on the column headings.
String	<a href="#">styleClass</a>	The name of the style class that should be applied to this component.
Number	<a href="#">tabSeq</a>	An index that specifies the position of the component in the tab sequence.
Boolean	<a href="#">transparent</a>	Flag that tells if the component is transparent or not.
Boolean	<a href="#">visible</a>	The visible property of the component, default true.

## Event Summary

void	<a href="#">onDrag()</a>	The method that is triggered when (non Design Mode) dragging occurs.
void	<a href="#">onDragEnd()</a>	The method that is triggered when (non Design Mode) dragging end occurs.
void	<a href="#">onDragOver()</a>	The method that is triggered when (non Design Mode) dragging over a component occurs.
void	<a href="#">onDrop()</a>	The method that is triggered when (non Design Mode) dropping occurs.
void	<a href="#">onRender()</a>	The method that is executed when the component is rendered.

## Property Details

### anchors

Enables a component to stick to a specific side of form and/or to grow or shrink when a window is resized.

If opposite anchors are activated then the component will grow or shrink with the window. For example if Top and Bottom are activated, then the component will grow/shrink when the window is vertically resized. If Left and Right are activated then the component will grow/shrink when the window is horizontally resized.

If opposite anchors are not activated, then the component will keep a constant distance from the sides of the window which correspond to the activated anchors.

### Returns

Number

---

**Supported Clients**

SmartClient,WebClient,NGClient

**Sample****background**

The background color of the component.

**Returns**

[String](#)

**Supported Clients**

SmartClient,WebClient,NGClient

**Sample****borderType**

The type, color and style of border of the component.

**Returns**

[String](#)

**Supported Clients**

SmartClient,WebClient,NGClient

**Sample****cssPosition**

Thecss position of the component.

**Returns**

[CSSPosition](#)

**Supported Clients**

SmartClient,WebClient,NGClient

**Sample****designTimeProperties**

Property to get and add design-time properties for a component.

**Returns**

[Object](#) map of the design-time properties

**Supported Clients**

SmartClient,WebClient,NGClient

**Sample****enabled**

The enable state of the component, default true.

**Returns**

[Boolean](#) enabled state

**Supported Clients**

SmartClient,WebClient,NGClient

**Sample****formIndex**

---

Form index returns the z-index of the form element. If two elements overlap, then the element with higher Z index is displayed above the form element with lower Z index. In case of form inheritance, form index is evaluated in the context (form) where property is defined. Elements which have child form as context will always be on top of elements which have parent form. If element is overridden in child form but form index property is still inherited its context is still parent form.

**Returns**[Number](#)**Supported Clients**

SmartClient,WebClient,NGClient

**Sample****initialSort**

The default sort order for the rows displayed in the portal.

**Returns**[String](#)**Supported Clients**

SmartClient,WebClient,NGClient

**Sample****intercellSpacing**

The additional spacing between cell rows. Is composed from the horizontal spacing and the vertical spacing.

**Returns**[String](#)**Supported Clients**

SmartClient,WebClient,NGClient

**Sample****location**

The x and y position of the component, in pixels, separated by a comma.

**Returns**[String](#)**Supported Clients**

SmartClient,WebClient,NGClient

**Sample****multiLine**

When set, portal rows can have a custom layout of fields, buttons, etc. displayed for each matching row (rather than the default "grid").

**Returns**[Boolean](#)**Supported Clients**

SmartClient,WebClient,NGClient

**Sample****name**

The name of the component. Through this name it can also accessed in methods. Must be a valid javascript name. (no - in the name or start with number)

**Returns**[String](#)

---

**Supported Clients**

SmartClient,WebClient,NGClient

**Sample**  
**ngReadOnlyMode****Returns**

[Boolean](#)

**Supported Clients**

SmartClient,WebClient,NGClient

**Sample**  
**printSliding**

Enables an element to resize based on its content and/or move when printing. The component can move horizontally or vertically and can grow or shrink in height and width, based on its content and the content of neighboring components.

**Returns**

[Number](#)

**Supported Clients**

SmartClient,WebClient

**Sample**  
**printable**

Flag that tells if the component should be printed or not when the form is printed.

By default components are printable.

**Returns**

[Boolean](#)

**Supported Clients**

SmartClient,WebClient

**Sample**  
**relationName**

The name of the relationship between the table related to the currently active form and the table you want to show data from in the portal.

**Returns**

[String](#)

**Supported Clients**

SmartClient,WebClient,NGClient

**Sample**  
**reorderable**

When set, the portal rows can be re-ordered by dragging the column headers.

**Returns**

[Boolean](#)

**Supported Clients**

SmartClient,WebClient,NGClient

**Sample**  
**resizable**

---

When set the portal rows can be resized by users.

**Returns**

[Boolean](#)

**Supported Clients**

SmartClient,WebClient,NGClient

**Sample****rowBGColorCalculation**

The calculation dataprovider (such as `servoy_row_bgcolor`) used to add background color and highlight selected or alternate portal rows.

NOTE: This property has been deprecated and is kept visible for legacy purposes. Use CSS Row Styling & onRender event instead.

**Returns**

[String](#)

**Supported Clients**

SmartClient,WebClient,NGClient

**Sample****rowHeight**

The height of each row in pixels. If 0 or not set, the height defaults to 10.

**Returns**

[Number](#)

**Supported Clients**

SmartClient,WebClient,NGClient

**Sample****scrollbars**

Scrollbar options for the vertical and horizontal scrollbars. Each of the vertical and horizontal scrollbars can be configured to display all the time, to display only when needed or to never display.

**Returns**

[Number](#)

**Supported Clients**

SmartClient,WebClient,NGClient

**Sample****showHorizontalLines**

When set, the portal displays horizontal lines between the rows.

NOTE:

In a multi-line portal, a horizontal line is only displayed in the selected row; to display a horizontal line in all rows, add a line to the portal.

**Returns**

[Boolean](#)

**Supported Clients**

SmartClient,WebClient,NGClient

**Sample****showVerticalLines**

---

When set the portal displays vertical lines between the columns.

**NOTE:**

In a multi-line portal, a vertical line is only displayed in the selected row; to display a vertical line in all rows, add a line to the portal.

**Returns**

[Boolean](#)

**Supported Clients**

SmartClient,WebClient,NGClient

**Sample****size**

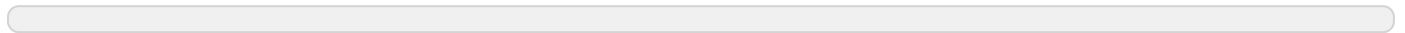
The width and height (in pixels), separated by a comma.

**Returns**

[String](#)

**Supported Clients**

SmartClient,WebClient,NGClient

**Sample****sortable**

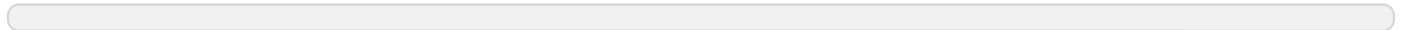
When set, users can sort the contents of the portal by clicking on the column headings.

**Returns**

[Boolean](#)

**Supported Clients**

SmartClient,WebClient,NGClient

**Sample****styleClass**

The name of the style class that should be applied to this component.

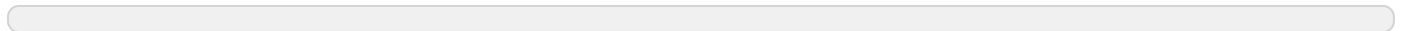
When defining style classes for specific component types, their names must be prefixed according to the type of the component. For example in order to define a class names 'fancy' for fields, in the style definition the class must be named 'field.fancy'. If it would be intended for labels, then it would be named 'label.fancy'. When specifying the class name for a component, the prefix is dropped however. Thus the field or the label will have its styleClass property set to 'fancy' only.

**Returns**

[String](#)

**Supported Clients**

SmartClient,WebClient,NGClient

**Sample****tabSeq**

An index that specifies the position of the component in the tab sequence. The components are put into the tab sequence in increasing order of this property. A value of 0 means to use the default mechanism of building the tab sequence (based on their location on the form). A value of -2 means to remove the component from the tab sequence.

**Returns**

[Number](#)

**Supported Clients**

SmartClient,WebClient,NGClient

**Sample**

### transparent

Flag that tells if the component is transparent or not.

The default value is "false", that is the components are not transparent.

#### Returns

[Boolean](#)

#### Supported Clients

SmartClient,WebClient,NGClient

#### Sample

### visible

The visible property of the component, default true.

#### Returns

[Boolean](#) visible property

#### Supported Clients

SmartClient,WebClient,NGClient

#### Sample

## Event Details

### onDrag()

The method that is triggered when (non Design Mode) dragging occurs.

#### Supported Clients

SmartClient,WebClient

#### Sample

### onDragEnd()

The method that is triggered when (non Design Mode) dragging end occurs.

#### Supported Clients

SmartClient,WebClient

#### Sample

### onDragOver()

The method that is triggered when (non Design Mode) dragging over a component occurs.

#### Supported Clients

SmartClient,WebClient

#### Sample

### onDrop()

The method that is triggered when (non Design Mode) dropping occurs.

#### Supported Clients

SmartClient,WebClient

#### Sample

### onRender()

The method that is executed when the component is rendered.

Supported Clients

SmartClient, WebClient

Sample