TextField

Event Summary

	onAction The method that is executed when the component is clicked.
Boolean	onDataChange Method that is executed when the data in the component is successfully changed.
	onFocusGained The method that is executed when the component gains focus.
	onFocusLost The method that is executed when the component looses focus.
	onRender The method that is executed when the component is rendered.
	onRightClick The method that is executed when the component is right clicked.

Property Summary

Number	anchors Enables a component to stick to a specific side of form and/or to grow or shrink when a window is resized.
String	background The background color of the component.
String	borderType The type, color and style of border of the component.
String	dataProvider The dataprovider of the component.
Object	designTimeProperties Property to get and add design-time properties for a component.
Number	displayType The type of display used by the field.
Boolean	displaysTags Flag that enables or disables merging of data inside components using tags (placeholders).
Boolean	editable Flag that tells if the content of the field can be edited or not.
Boolean	enabled The enable state of the component, default true.
String	fontType The font type of the component.
String	foreground The foreground color of the component.
Number	formIndex
String	format The format that should be applied when displaying the data in the component.
Number	horizontalAlignment Horizontal alignment of the text inside the component.
String	location The x and y position of the component, in pixels, separated by a comma.
String	margin The margins of the component.
String	name The name of the component.
String	placeholderText The text that is displayed in field when the field doesn't have a text value.
Number	printSliding Enables an element to resize based on its content and/or move when printing.
Boolean	printable Flag that tells if the component should be printed or not when the form is printed.
Number	scrollbars Scrollbar options for the vertical and horizontal scrollbars.
Boolean	selectOnEnter Flag that tells if the content of the field should be automatically selected when the field receives focus.
String	size The width and height (in pixels), separated by a comma.
String	styleClass The name of the style class that should be applied to this component.

Number	tabSeq An index that specifies the position of the component in the tab sequence.
String	titleText Header text to component
String	toolTipText The text displayed when hovering over the component with a mouse cursor.
Boolean	transparent Flag that tells if the component is transparent or not.
Number	valuelist The valuelist that is used by this field when displaying data.
Boolean	visible The visible property of the component, default true.

Event Details

onAction

The method that is executed when the component is clicked.

onDataChange

Method that is executed when the data in the component is successfully changed.

Parameters

oldValue - old value
newValue - new value
{JSEvent} event - the event that triggered the action

Returns

Boolean

onFocusGained

The method that is executed when the component gains focus.
NOTE: Do not call methods that will influence the focus itself.

Parameters

{JSEvent} event - the event that triggered the action

onFocusLost

The method that is executed when the component looses focus.

Parameters

{JSEvent} event - the event that triggered the action

onRender

The method that is executed when the component is rendered.

onRightClick

The method that is executed when the component is right clicked.

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 with 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

background

The background color of the component.

Returns

String

borderType

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

Returns

String

dataProvider

The dataprovider of the component.

Returns

String

designTimeProperties

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

Returns

Object - map of the design-time properties

displayType

The type of display used by the field. Can be one of CALENDAR, CHECKS, COMBOBOX, HTML_AREA, IMAGE_MEDIA, PASSWORD, RADIOS, RTF_AREA, TEXT_AREA, TEXT_FIELD, TYPE_AHEAD, LIST_BOX, MULTISELECT_LISTBOX or SPINNER.

Returns

Number

displaysTags

Flag that enables or disables merging of data inside components using tags (placeholders). Tags (or placeholders) are words surrounded by %%on each side. There are data tags and standard tags. Data tags consist in names of dataproviders surrounded by%%. Standard tags are a set of predefined tags that are made available by the system.

See the "Merging data" section for more details about tags.

The default value of this flag is "false", that is merging of data is disabled by default.

Returns

Boolean

editable

Flag that tells if the content of the field can be edited or not.
The default value of this flag is "true", that is the content can be edited.

Returns

Boolean

enabled

The enable state of the component, default true.

Returns

Boolean - enabled state

fontType

The font type of the component.

Returns

String

foreground

The foreground color of the component.

Returns

String

formIndex

Returns

Number

format

The format that should be applied when displaying the data in the component.
There are different options for the different dataprovider types that are assigned to this field.
For Integer fields, there is a display and an edit format, using http://docs.oracle.com/javase/7/docs/api/java/text/DecimalFormat.html and the max (string) length can be set.
For Text/String fields, there are options to force uppercase,lowercase or only numbers. Or a mask can be set that restrict the input the pattern chars can be found here: http://docs.oracle.com/javase/7/docs/api/javax/swing/text/MaskFormatter.html
A mask can have a placehoder (what is shown when there is no data) and if the data must be stored raw (without literals of the mask). A max text length can also be set to force
the max text length input, this doesn't work on mask because that max length is controlled with the mask.
For Date fields a display and edit format can be set by using a pattern from here: http://docs.oracle.com/javase/7/docs/api/java/text/SimpleDateFormat.html, you can also say this must behave like a mask (the edit format)
A mask only works with when the edit format is exactly that mask (1 char is 1 number/char), because for example MM then only 2 numbers are allowed MMM that displays the month as a string is not supported as a mask.
Some examples are "dd-MM-yyyy", "MM-dd-yyyy", etc.
The format property is also used to set the UI Converter, this means that you can convert the value object to something else before it gets set into the field, this can also result in a type change of the data.
So a string in scripting/db is converted to a integer in the ui, then you have to set an integer format.
This property is applicable only for types: TEXT_FIELD, COMBOBOX, TYPE_AHEAD, CALENDAR and SPINNER.

Returns

String

horizontalAlignment

Horizontal alignment of the text inside the component. Can be one of
LEFT, CENTER or RIGHT.

Note that this property does not refer to the horizontal alignment
of the component inside the form.

Returns

Number

location

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

Returns

String

margin

The margins of the component. They are specified in this order,
separated by commas: top, left, bottom, right.

Returns

String

name

The name of the component. Through this name it can also accessed in methods.

Returns

String

placeholderText

The text that is displayed in field when the field doesn't have a text value.

Returns

String

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

printable

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

By default components are printable.

Returns

Boolean

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

selectOnEnter

Flag that tells if the content of the field should be automatically selected when the field receives focus. The default value of this field is "false".

Returns

Boolean

size

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

Returns

String

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

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

titleText

Header text to component

Returns

String

toolTipText

The text displayed when hovering over the component with a mouse cursor.

NOTE:
HTML should be used for multi-line tooltips; you can also use any valid HTML tags to format tooltip text. For example:
<html>This includesbolded text and
BLUE text as well.</html>

Returns

String

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

valuelist

The valuelist that is used by this field when displaying data. Can be used with fields of type CHECKS, COMBOBOX, RADIOS and TYPE_AHEAD.

Returns

Number

visible

The visible property of the component, default true.

Returns

Boolean - visible property