CheckBox	Options	Description	Default Value	Acceptable Values
	accessKey	specifies a short cur key that focus on editor	undefined	"A", "1","b", etc.
	activeStateEnabled	true -> editor will change its appearance (e.g. color, border, etc.) when user is interacting like clicking or tapping	TRUE	true, false
	disabled	making it unresponsive to user interaction	FALSE	true, false
	elementAttr	add custom html attributes such as id, class	{}	{ "id" : "CustomId", "class" : "my-custom-class" }
	focusStateEnabled	editor can be focused using keyboard (used for accessibility & imroved UI/UX)	TRUE	true, false
	height	set height of editor	undefined	50, 50px, 50%
	hint	when user hovers a editor hint is visible (contains text simplifies usecase of editor)	undefined	a string
	hoverStateEnabled	editor will change its appereance on hover	TRUE	true, false
	isValid	defines editor is valid or not accordig to validation rules	TRUE	true, false
	name	sets the name attribute to editor (useful when submitting the form)	undefined	"customName"
	onContentReady	a callback function that executes when editor is fully rendered	null	function()
	onDisposing	a callback function that executes just before the editor is removed from the DOM.	null	function()
	onInitialized	a callback function that executes when editor is initialized	null	function()
	onOptionChanged	a callback function that executes when editor's option is changed	null	function()
	onValueChanged	a callback funcion that executes when editor's value is changed	null	function()
	readOnly	makes field read only. No modification permission is given to user.	FALSE	true, false
	rtlEnabled	enables right to left text direction. (useful for languages like urdu)	FALSE	true, false
	tabIndex	Specifies the order in which elements gain focus when navigating through the page using the Tab key.	0	A number
	text	Sets the text label displayed next to the Editor.	""	a string
	validationError	Contains information about a validation error when a single error is present.	null	{ message : "myMsg"}
	validationErrors	Contains information about multiple validation errors if more than one is present.	null	array of msg objects
	validationMessageMode	Determines how validation messages are displayed .. Auto -> for some time, always -> continuesly visible, none -> not	auto	always, auto, none
	validationStatus	Indicates the current validation status of the Editor	valid	valid, invalid, pending
	value	Represents the current value of the Editor	FALSE	true, false
	visible	Defines visibility of the editor	TRUE	true, false
	width	sets the width os the editor.	undefined	50, 50px, 50%

DateBox	Options	Description	Default Value	Acceptable Values
	acceptCustomValue	allows user ro enter custom values	TRUE	true, false
	accessKey	specifies a short cur key that focus on editor	undefined	"A", "1","b", etc.
	activeStateEnabled	true -> editor will change its appearance (e.g. color, border, etc.) when user is interacting like clicking or tapping	TRUE	true, false
	adaptivityEnabled	adapts widget for small screens	FALSE	true, false
	applyButtonText	text for the apply button	"Apply"	a string
	applyValueMode	when to apply the new value	instantly	instantly, useButtons

buttons	custom buttons for editors	undefined	an array of predefined button configurations
calendarOptions	options for the calander configurations	{}	disabledDates, dateSerializationFormat, etc.
cancelButtonText	text for the cancel; button	Cancel	a string
dateOutOfRangeMessage	message for out-of-range dates	Value is out of range.	a string
dateSerializationFormat	Format for the data serialization	undefined	any string representing date format
deferRendering	defers rendering until the wifget is visible	FALSE	true, false
disabled	making it unresponsive to user interaction	FALSE	true, false
disabledDates	specifies date cannot be selected	null	an array of dates
displayFormat	Format of displaying date	undefined	a string consists display format
dropDownButtonComponent	custom comonent for the dropdown	undefined	any custom component
dropDownButtonRender	custom render function for the dropdown button	undefined	function()
dropDownButtonTemplate	template for the dropdown button	undefined	function()
dropDownOptions	Options for the drop down	{}	object contains options

elementAttr	add custom html attributes such as id, class	{}	{
			"id" : "CustomId",
			"class" :
			"my-custom-class"
			}

focusStateEnabled	editor can be focused using keyboard (used for accessibility & imroved UI/UX)	TRUE	true, false
height	set height of editor	undefined	50, 50px, 50%
hint	when user hovers a editor hint is visible (contains text simplifies usecase of editor)	undefined	a string
hoverStateEnabled	editor will change its appereance on hover	TRUE	true, false
inputAttr	to add html attributes to input	{}	object contains html attributes
invalidDateMessage	Message for invalid dates	"Invalid Date"	a string
isValid	defines editor is valid or not accordig to validation rules	TRUE	true, false
max	maximum acceptable value	undefined	value
maxLength	maximum number of characters	null	a number
min	minimum acceptable value	undefined	value
name	minimum number of characters	null	a number
onChange	a callbackfunction executes on change	null	function()
onClosed	a callbackfunction executes when dropdown is closed	null	function()
onContentReady	a callback function that executes when editor is fully rendered	null	function()
onCopy	a callback function executes on copy event	null	function()
onCut	a callback function executes on cut event	null	function()
onDisposing	a callback function that executes just before the editor is removed from the DOM.	null	function()
onFocusIn	a callback function executes on focus-in event	null	function()
onEnterKey	a callback function executes when user presses enter key	null	function()
onFocusOut	a callback function executes on focus-out event	null	function()
onInitialized	a callback function that executes when editor is initialized	null	function()
onInput	event handler for input event	null	function()
onKeyDown	a callback function executes on key-down event	null	function()
onKeyUp	a callback function executes on key-up event	null	function()
onOpened	event handler for open event	null	function()

onOptionChanged	a callback function that executes when editor's option is changed	null	function()
onPaste	a callback function executes on paste event	null	function()
onValueChanged	event hanlder for value change event	null	function()
opened	event hanlder for open event	null	function()
openOnFieldClick	Opens the dropdown when the field is clicked.	TRUE	true, false
pickerType	Specifies the type of date picker.	calender	calender, list, rollers
placeholder	placeholder to editor's input field	""	a string
readOnly	makes field read only. No modification permission is given to user.	FALSE	true, false
rtlEnabled	enables right to left text direction. (useful for languages like urdu)	FALSE	true, false
showAnalogClock	Displays an analog clock in the time picker.	TRUE	true, false
showClearButton	Shows a button to clear the selected value.	TRUE	true, false
showDropDownButton	Displays a dropdown button.	TRUE	true, false
spellcheck	Enables or disables the spellchecker.	FALSE	true, false
stylingMode	Specifies the styling mode for the widget.	outlined	outlined, contained, underlined
tabIndex	Specifies the order in which elements gain focus when navigating through the page using the Tab key.	0	A number
text	Sets the text label displayed next to the Editor.	""	a string
type	specifies the type of the element	date	date, time, datetime
useMaskBehavior	determines whether a maskbehaviour is applied to the input	FALSE	true, false
validationError	Contains information about a validation error when a single error is present.	null	{ message : "myMsg"}
validationErrors	Contains information about multiple validation errors if more than one is present.	null	array of msg objects
validationMessageMode	Determines how validation messages are displayed .. Auto -> for some time, always -> continuesly visible, none -> not	auto	always, auto, none
validationStatus	Indicates the current validation status of the Editor	valid	valid, invalid, pending
value	Represents the current value of the Editor	FALSE	true, false
valueChangeEvent	event handler for value change event	null	function()
visible	Defines visibility of the editor	TRUE	true, false
width	sets the width os the editor.	undefined	50, 50px, 50%

DropDownBox	**Options**	**Description**	**Default Value**	**Acceptable Values**
	acceptCustomValue	allows user ro enter custom values	TRUE	true, false
	accessKey	specifies a short cur key that focus on editor	undefined	"A", "1","b", etc.
	activeStateEnabled	true -> editor will change its appearance (e.g. color, border, etc.) when user is interacting like clicking or tapping	TRUE	true, false
	buttons	custom buttons for editors	undefined	an array of predefined button configurations
	contentTemplate	allows you to specify a custom template for rendering the content within the Editor.	undefined	function()
	dataSource	define data source for editor.	null	array or data scource object
	deferRendering	defers rendering until the wifget is visible	FALSE	true, false
	disabled	making it unresponsive to user interaction	FALSE	true, false
	displayExpr	determines how the selected value is displayed in editor input field	undefined	object
	dropDownButtonTemplate	template for the dropdown button	undefined	function()
	dropDownOptions	Options for the drop down	{}	object contains options
	elementAttr	add custom html attributes such as id, class	{}	{ "id" : "CustomId", "class" : "my-custom-class" }

fieldTemplate	customize the template for editors	undefined	function()
focusStateEnabled	editor can be focused using keyboard (used for accessibility & imroved UI/UX)	TRUE	true, false
height	set height of editor	undefined	50, 50px, 50%
hint	when user hovers a editor hint is visible (contains text simplifies usecase of editor)	undefined	a string
hoverStateEnabled	editor will change its appereance on hover	TRUE	true, false
inputAttr	to add html attributes to input	{}	object contains html attributes
isValid	defines editor is valid or not accordig to validation rules	TRUE	true, false
items	An array of predefined items that can be displayed in the editor dropdown.	null	array
maxLength	maximum number of characters	null	a number
name	sets the name attribute to editor (useful when submitting the form)	undefined	"customName"
onChange	a callbackfunction executes on change	null	function()
onClosed	a callbackfunction executes when dropdown is closed	null	function()
onCopy	a callback function executes on copy event	null	function()
onCut	a callback function executes on cut event	null	function()
onDisposing	a callback function that executes just before the editor is removed from the DOM.	null	function()
onEnterKey	a callback function executes when user presses enter key	null	function()
onFocusIn	a callback function executes on focus-in event	null	function()
onFocusOut	a callback function executes on focus-out event	null	function()
onInitialized	a callback function that executes when editor is initialized	null	function()
onInput	event handler for input event	null	function()
onKeyDown	a callback function executes on key-down event	null	function()
onKeyUp	a callback function executes on key-up event	null	function()
onOpened	event handler for open event	null	function()
onOptionChanged	a callback function that executes when editor's option is changed	null	function()
onPaste	a callback function executes on paste event	null	function()
onValueChanged	event hanlder for value change event	null	function()
opened	event hanlder for open event	null	function()
openOnFieldClick	Opens the dropdown when the field is clicked.	TRUE	true, false
placeholder	placeholder to editor's input field	""	a string
readOnly	makes field read only. No modification permission is given to user.	FALSE	true, false
rtlEnabled	enables right to left text direction. (useful for languages like urdu)	FALSE	true, false
showClearButton	Shows a button to clear the selected value.	TRUE	true, false
showDropDownButton	Displays a dropdown button.	TRUE	true, false
stylingMode	Specifies the styling mode for the widget.	outlined	outlined, contained, underlined
tabIndex	Specifies the order in which elements gain focus when navigating through the page using the Tab key.	0	A number
text	Sets the text label displayed next to the Editor.	""	a string
validationError	Contains information about a validation error when a single error is present.	null	{ message : "myMsg"}
validationErrors	Contains information about multiple validation errors if more than one is present.	null	array of msg objects
validationMessageMode	Determines how validation messages are displayed .. Auto -> for some time, always -> continuesly visible, none -> not	auto	always, auto, none
validationStatus	Indicates the current validation status of the Editor	valid	valid, invalid, pending
value	Represents the current value of the Editor	FALSE	true, false
valueChangeEvent	event handler for value change event	null	function()

	valueExpr	specifies which data field provides the widget value.	undefined	function()
	visible	Defines visibility of the editor	TRUE	true, false

NumberBox	Options	Description	Default Value	Acceptable Values
	accessKey	Sets a keyboard shortcut for focusing the widget.	undefined	Single character string (e.g., "A")
	activeStateEnabled	Enables the widget's active state.	TRUE	true, false
	buttons	Custom buttons for the widget.	undefined	Array of predefined or custom button configurations
	disabled	Disables the widget.	FALSE	true, false
	elementAttr	HTML attributes for the root element.	{}	Object
	focusStateEnabled	Enables focus state on interaction.	TRUE	true, false
	format	Specifies the format for the number.	undefined	String or function
	height	Height of the widget.	undefined	Number (pixels) or string (e.g., "100px")
	hint	Tooltip text.	undefined	String
	hoverStateEnabled	Enables hover state on interaction.	TRUE	true, false
	inputAttr	HTML attributes for the input element.	{}	Object
	invalidValueMessage	Message for invalid values.	"Value is invalid"	String
	isValid	Validation state.	TRUE	true, false
	max	Maximum value.	undefined	Number
	min	Minimum value.	undefined	Number
	mode	Specifies the input mode.	"text"	"text", "number"
	name	Name attribute of the editor.	undefined	String
	onChange	Event handler for change event.	null	function()
	onContentReady	Event handler for when the content is ready.	null	function()
	onCopy	Event handler for copy event.	null	function()
	onCut	Event handler for cut event.	null	function()
	onDisposing	Event handler for disposing event.	null	function()
	onEnterKey	Event handler for enter key event.	null	function()
	onFocusIn	Event handler for focus-in event.	null	function()
	onFocusOut	Event handler for focus-out event.	null	function()
	onInitialized	Event handler for initialization event.	null	function()
	onInput	Event handler for input event.	null	function()
	onKeyDown	Event handler for key-down event.	null	function()
	onKeyUp	Event handler for key-up event.	null	function()
	onOptionChanged	Event handler for option changed event.	null	function()
	onPaste	Event handler for paste event.	null	function()
	onValueChanged	Event handler for value changed event.	null	function()
	placeholder	Placeholder text for the input field.	""	String
	readOnly	Specifies whether the editor is read-only.	FALSE	true, false
	rtlEnabled	Switches the widget to a right-to-left representation.	FALSE	true, false
	showClearButton	Shows a button to clear the selected value.	TRUE	true, false
	showSpinButtons	Displays spin buttons.	FALSE	true, false
	step	Specifies the step by which a value changes.	1	Number
	stylingMode	Specifies the styling mode for the widget.	"outlined"	"outlined", "underlined", "filled"
	tabIndex	Specifies the widget's order in the tab sequence.	0	Number
	text	Text of the widget.	""	String
	useLargeSpinButtons	Uses large spin buttons.	FALSE	true, false
	validationError	Specifies information about the first validation error.	null	Object
	validationErrors	Specifies information on all validation errors.	null	Array
	validationMessageMode	Specifies how the message about the validation rules should be displayed.	"auto"	"auto", "always"

validationStatus	Indicates whether the editor's value is valid.	"valid"	"valid", "invalid", "pending"
value	Specifies the current value of the widget.	0	Number
valueChangeEvent	Specifies the DOM event that triggers a value change.	"change"	String
visible	Specifies whether the widget is visible.	TRUE	true, false
width	Specifies the widget's width.	undefined	Number (pixels) or string (e.g., "100px")

FileUploader	Options	Description	Default Value	Acceptable Values
	abortUpload	Cancels the file upload.	-	function()
	accept	Specifies the file types that can be accepted.	-	String
	accessKey	Shortcut key to focus on the widget.	-	String
	activeStateEnabled	Allows the widget to be focused using keyboard navigation.	FALSE	Boolean
	allowCanceling	Allows the file upload to be canceled.	TRUE	Boolean
	allowedFileExtensions	List of file extensions accepted for upload.	[]	Array
	chunkSize	Size (in bytes) of the file chunks uploaded.	0	Number
	dialogTrigger	Custom element to trigger the file upload dialog.	-	Element
	disabled	Disables the widget if true.	FALSE	Boolean
	dropZone	Custom drop zone for file uploads.	-	Element
	elementAttr	Custom attributes for the widget's root element.	-	Object
	focusStateEnabled	Allows the widget to be focused using keyboard navigation.	TRUE	Boolean
	height	Height of the widget.	auto	Number, String
	hint	Text hint displayed when the widget is hovered over.	-	String
	hoverStateEnabled	Changes widget state on hover.	TRUE	Boolean
	inputAttr	Custom attributes for the file input element.	-	Object
	invalidFileExtensionMessage	Message when a file with an invalid extension is selected.	"The file type is not allowed."	String
	invalidMaxFileSizeMessage	Message when a file exceeds the maximum size.	"The file size exceeds the maximum allowed size."	String
	invalidMinFileSizeMessage	Message when a file is smaller than the minimum size.	"The file size is smaller than the minimum allowed size."	String
	isValid	Indicates if the selected files are valid.	TRUE	Boolean
	labelText	Text displayed on the file upload label.	"Select file"	String
	maxFileSize	Maximum file size (in bytes) for upload.	0 (no limit)	Number
	minFileSize	Minimum file size (in bytes) for upload.	0	Number
	multiple	Allows multiple file selection.	FALSE	Boolean
	name	Name attribute of the file input element.	-	String
	onBeforeSend	function() executed before a file is sent.	-	function()
	onContentReady	function() executed when the widget's content is ready.	-	function()
	onDisposing	function() executed before the widget is disposed.	-	function()
	onDropZoneEnter	function() executed when a file is dragged into the drop zone.	-	function()
	onDropZoneLeave	function() executed when a file is dragged out of the drop zone.	-	function()
	onFilesUploaded	function() executed when all files are uploaded.	-	function()
	onInitialized	function() executed when the widget is initialized.	-	function()
	onOptionChanged	function() executed when a widget option is changed.	-	function()
	onProgress	function() executed when the file upload progress changes.	-	function()
	onUploadAborted	function() executed when the file upload is aborted.	-	function()

	onUploaded	function() executed when a file is successfully uploaded.	-	function()
	onUploadError	function() executed when a file upload error occurs.	-	function()
	onUploadStarted	function() executed when a file upload starts.	-	function()
	onValueChanged	function() executed when the widget's value is changed.	-	function()
	progress	Upload progress percentage.	0	Number
	readOnly	Makes the widget read-only.	FALSE	Boolean
	readyToUploadMessage	Message displayed when files are ready to be uploaded.	"Ready to upload"	String
	rtlEnabled	Supports right-to-left (RTL) languages.	FALSE	Boolean
	selectButtonText	Text displayed on the file selection button.	"Select File"	String
	showFileList	Displays the list of selected files.	TRUE	Boolean
	tabIndex	Tab index of the widget.	0	Number
	uploadAbortedMessage	Message displayed when the file upload is aborted.	"Upload aborted"	String
	uploadButtonText	Text displayed on the upload button.	"Upload"	String
	uploadChunk	function() executed when a file chunk is uploaded.	-	function()
	uploadCustomData	Additional data sent with the file upload request.	-	Object
	uploadedMessage	Message displayed when a file is successfully uploaded.	"Uploaded"	String
	uploadFailedMessage	Message displayed when a file upload fails.	"Upload failed"	String
	uploadFile	function() executed when a file is uploaded.	-	function()
	uploadHeaders	HTTP headers sent with the file upload request.	-	Object
	uploadMethod	HTTP method used for file uploads.	"POST"	String
	uploadMode	Upload mode.	"instantly"	"instantly", "useButtons", "useForm"
	uploadUrl	URL to which the files are uploaded.	-	String
	validationError	Validation error message.	-	String
	validationErrors	Multiple validation error messages.	[]	Array
	validationStatus	Validation status of the selected files.	"valid"	"valid", "invalid", "pending"
	value	Files to be uploaded.	[]	Array
	visible	Specifies widget visibility.	TRUE	Boolean
	width	Width of the widget.	auto	Number, String
Validator	**Options**	**Description**	**Default Value**	**Acceptable Values**
	adapter	Custom validation implementation.	-	Object
	elementAttr	Custom attributes for the widget.	-	Object
	height	Height of the widget.	undefined	Number, String
	name	Name of the validation rule.	-	String
	onDisposing	function() executed before the widget is disposed.	-	function()
	onInitialized	function() executed after the widget is initialized.	-	function()
	onOptionChanged	function() executed when a widget option is changed.	-	function()
	onValidated	function() executed when validation is complete.	-	function()
	validationGroup	Group for validation.	undefined	String
	validationRules	Validation rules to be checked.	-	Array
	width	Width of the widget.	undefined	Number, String
Radio Group	**Options**	**Description**	**Default Value**	**Acceptable Values**
	accessKey	Shortcut key to focus the widget.	-	String
	activeStateEnabled	Enables focus using keyboard navigation.	FALSE	Boolean
	dataSource	Data source for the widget.	-	Array, DataSource
	disabled	Disables the widget if true.	FALSE	Boolean
	displayExpr	Specifies the data field whose values are displayed.	this	String, function()
	elementAttr	Custom attributes for the widget's root element.	-	Object
	focusStateEnabled	Enables focus using keyboard navigation.	TRUE	Boolean

	height	Specifies the widget's height.	undefined	Number, String
	hint	Text hint displayed on hover.	-	String
	hoverStateEnabled	Enables hover state change.	TRUE	Boolean
	isValid	Specifies if the widget's value is valid.	TRUE	Boolean
	items	Items displayed by the widget.	[]	Array
	itemTemplate	Template for items in the widget.	-	function(), String
	layout	Specifies item layout.	vertical	vertical', 'horizontal'
	name	The value of the name attribute of the underlying HTML element.	-	String
	onContentReady	function() executed when the widget's content is ready.	-	function()
	onDisposing	function() executed before the widget is disposed.	-	function()
	onInitialized	function() executed after the widget is initialized.	-	function()
	onOptionChanged	function() executed when a widget option is changed.	-	function()
	onValueChanged	function() executed when the widget's value changes.	-	function()
	readOnly	Specifies if the widget is read-only.	FALSE	Boolean
	rtlEnabled	Enables right-to-left (RTL) languages.	FALSE	Boolean
	tabIndex	Specifies the widget's tab index.	0	Number
	validationError	Validation error message.	-	String
	validationErrors	Validation error messages.	-	Array
	validationMessageMode	Specifies how validation messages are displayed.	auto	auto', 'always'
	validationStatus	Validation status.	valid	valid', 'invalid', 'pending'
	value	Specifies the selected value.	undefined	Any
	valueExpr	Specifies the data field providing widget values.	this	String, function()
	visible	Specifies widget visibility.	TRUE	Boolean
	width	Specifies the widget's width.	undefined	Number, String

SelectBox	Options	Description	Default Value	Acceptable Values
	acceptCustomValue	Specifies whether the SelectBox allows users to enter a custom value.	FALSE	true, false
	accessKey	Specifies a shortcut key to activate/focus the SelectBox.	None	Any single character
	activeStateEnabled	Specifies whether the SelectBox changes its state when interacted with.	TRUE	true, false
	buttons	Configures the buttons displayed in the SelectBox.	undefined	Array
	dataSource	Specifies the data source for the SelectBox.	null	Array, DataSource
	deferRendering	Specifies whether to defer widget rendering until the widget is shown.	TRUE	true, false
	disabled	Specifies whether the SelectBox is disabled.	FALSE	true, false
	displayExpr	Specifies the data field whose values should be displayed.	undefined	String, function()
	displayValue	Specifies the display value of the SelectBox.	undefined	String
	dropDownButtonTemplate	Specifies a custom template for the drop-down button.	null	String, function()
	dropDownOptions	Configures the drop-down options.	{}	Object
	elementAttr	Specifies custom attributes for the SelectBox's root element.	{}	Object
	fieldTemplate	Specifies a custom template for the input field.	null	String, function()
	focusStateEnabled	Specifies whether the SelectBox can be focused.	TRUE	true, false
	grouped	Specifies whether data items should be grouped.	FALSE	true, false
	groupTemplate	Specifies a custom template for group headers.	null	String, function()
	height	Specifies the height of the SelectBox.	undefined	Number, String
	hint	Specifies a hint text inside the SelectBox.	""	String
	hoverStateEnabled	Specifies whether the SelectBox changes its state when hovered over.	TRUE	true, false
	inputAttr	Specifies attributes for the input element of the SelectBox.	{}	Object
	isValid	Specifies whether the current value is valid.	TRUE	true, false

items	Specifies an array of items for the SelectBox.	[]	Array
itemTemplate	Specifies a custom template for items.	null	String, function()
maxLength	Specifies the maximum number of characters allowed.	null	Number
minSearchLength	Specifies the minimum number of characters needed to start a search.	0	Number
name	Specifies the name of the input element.	""	String
noDataText	Specifies the text displayed when no data is available.	"No data"	String
onChange	A function that is executed when the value changes.	null	function()
onClosed	A function that is executed when the drop-down list is closed.	null	function()
onContentReady	A function that is executed when the content is ready.	null	function()
onCopy	A function that is executed when text is copied.	null	function()
onCustomItemCreating	A function that is executed when a custom item is created.	null	function()
onCut	A function that is executed when text is cut.	null	function()
onDisposing	A function that is executed when the widget is disposed of.	null	function()
onEnterKey	A function that is executed when the Enter key is pressed.	null	function()
onFocusIn	A function that is executed when the SelectBox is focused.	null	function()
onFocusOut	A function that is executed when the SelectBox loses focus.	null	function()
onInitialized	A function that is executed when the widget is initialized.	null	function()
onInput	A function that is executed when text is input.	null	function()
onItemClick	A function that is executed when an item is clicked.	null	function()
onKeyDown	A function that is executed when a key is pressed down.	null	function()
onKeyUp	A function that is executed when a key is released.	null	function()
onOpened	A function that is executed when the drop-down list is opened.	null	function()
onOptionChanged	A function that is executed when a widget option is changed.	null	function()
onPaste	A function that is executed when text is pasted.	null	function()
onSelectionChanged	A function that is executed when the selection is changed.	null	function()
onValueChanged	A function that is executed when the value changes.	null	function()
opened	Specifies whether the drop-down list is opened.	FALSE	true, false
openOnFieldClick	Specifies whether the drop-down list is opened when the input field is clicked.	TRUE	true, false
placeholder	Specifies a short hint displayed in the input area.	""	String
readOnly	Specifies whether the SelectBox is read-only.	FALSE	true, false
rtlEnabled	Specifies whether the SelectBox supports right-to-left text direction.	FALSE	true, false
searchEnabled	Specifies whether the search is enabled.	FALSE	true, false
searchExpr	Specifies the data field to search.	undefined	String, Array, function()
searchMode	Specifies the search mode.	"contains"	"contains", "startswith", "equals"
searchTimeout	Specifies the time delay before the search is executed.	500	Number
selectedItem	Specifies the selected item.	null	Object
showClearButton	Specifies whether the clear button is displayed.	FALSE	true, false
showDataBeforeSearch	Specifies whether data is shown before a search is performed.	FALSE	true, false
showDropDownButton	Specifies whether the drop-down button is displayed.	TRUE	true, false

showSelectionControls	Specifies whether selection controls are displayed.	FALSE	true, false
spellcheck	Specifies whether the SelectBox should be spellchecked.	FALSE	true, false
stylingMode	Specifies the styling mode of the SelectBox.	"outlined"	"outlined", "underlined", "filled"
tabIndex	Specifies the Tab index of the SelectBox.	0	Number
text	Specifies the text displayed in the SelectBox.	""	String
useItemTextAsTitle	Specifies whether to use item text as the title attribute value.	FALSE	true, false
validationError	Specifies the validation error.	null	Object
validationErrors	Specifies an array of validation errors.	[]	Array
validationMessageMode	Specifies how validation messages are displayed.	"auto"	"auto", "always"
validationStatus	Specifies the validation status of the SelectBox.	"valid"	"valid", "invalid", "pending"
value	Specifies the current value of the SelectBox.	null	Any
valueChangeEvent	Specifies the DOM event after which the SelectBox's value should be updated.	"change"	String
valueExpr	Specifies the data field whose values should be used as the SelectBox value.	undefined	String, function()
visible	Specifies whether the SelectBox is visible.	TRUE	true, false
width	Specifies the width of the SelectBox.	undefined	Number,

TextArea	Options	Description	Default Value	Acceptable Values
	accessKey	Specifies a shortcut key to activate/focus the TextArea.	None	Any single character
	activeStateEnabled	Specifies whether the TextArea changes its state when interacted with.	TRUE	true, false
	autoResizeEnabled	Specifies whether the TextArea automatically adjusts its height to fit content.	FALSE	true, false
	disabled	Specifies whether the TextArea is disabled.	FALSE	true, false
	elementAttr	Specifies custom attributes for the TextArea's root element.	{}	Object
	focusStateEnabled	Specifies whether the TextArea can be focused.	TRUE	true, false
	height	Specifies the height of the TextArea.	undefined	Number, String
	hint	Specifies a hint text inside the TextArea.	""	String
	hoverStateEnabled	Specifies whether the TextArea changes its state when hovered over.	TRUE	true, false
	inputAttr	Specifies attributes for the input element of the TextArea.	{}	Object
	isValid	Specifies whether the current value is valid.	TRUE	true, false
	maxHeight	Specifies the maximum height of the TextArea.	null	Number, String
	maxLength	Specifies the maximum number of characters allowed.	null	Number
	minHeight	Specifies the minimum height of the TextArea.	null	Number, String
	name	Specifies the name of the input element.	""	String
	onChange	A function that is executed when the value changes.	null	function()
	onContentReady	A function that is executed when the content is ready.	null	function()
	onCopy	A function that is executed when text is copied.	null	function()
	onCut	A function that is executed when text is cut.	null	function()
	onDisposing	A function that is executed when the widget is disposed of.	null	function()
	onEnterKey	A function that is executed when the Enter key is pressed.	null	function()
	onFocusIn	A function that is executed when the TextArea is focused.	null	function()
	onFocusOut	A function that is executed when the TextArea loses focus.	null	function()
	onInitialized	A function that is executed when the widget is initialized.	null	function()
	onInput	A function that is executed when text is input.	null	function()

onKeyDown	A function that is executed when a key is pressed down.	null	function()
onKeyUp	A function that is executed when a key is released.	null	function()
onOptionChanged	A function that is executed when a widget option is changed.	null	function()
onPaste	A function that is executed when text is pasted.	null	function()
onValueChanged	A function that is executed when the value changes.	null	function()
placeholder	Specifies a short hint displayed in the input area.	""	String
readOnly	Specifies whether the TextArea is read-only.	FALSE	true, false
rtlEnabled	Specifies whether the TextArea supports right-to-left text direction.	FALSE	true, false
spellcheck	Specifies whether the TextArea should be spellchecked.	TRUE	true, false
stylingMode	Specifies the styling mode of the TextArea.	"outlined"	"outlined", "underlined", "filled"
tabIndex	Specifies the Tab index of the TextArea.	0	Number
text	Specifies the text displayed in the TextArea.	""	String
validationError	Specifies the validation error.	null	Object
validationErrors	Specifies an array of validation errors.	[]	Array
validationMessageMode	Specifies how validation messages are displayed.	"auto"	"auto", "always"
validationStatus	Specifies the validation status of the TextArea.	"valid"	"valid", "invalid", "pending"
value	Specifies the current value of the TextArea.	""	String
valueChangeEvent	Specifies the DOM event after which the TextArea's value should be updated.	"change"	String
visible	Specifies whether the TextArea is visible.	TRUE	true, false
width	Specifies the width of the TextArea.	undefined	Number, String

TextBox	Options	Description	Default Value	Acceptable Values
	accessKey	Specifies a shortcut key to activate/focus the TextBox.	None	Any single character
	activeStateEnabled	Specifies whether the TextBox changes its state when interacted with.	TRUE	true, false
	buttons	Configures buttons to be displayed within the TextBox.	None	Array of button configurations
	disabled	Specifies whether the TextBox is disabled.	FALSE	true, false
	elementAttr	Specifies custom attributes for the TextBox's root element.	{}	Object
	focusStateEnabled	Specifies whether the TextBox can be focused.	TRUE	true, false
	height	Specifies the height of the TextBox.	undefined	Number, String
	hint	Specifies a placeholder text inside the TextBox.	""	String
	hoverStateEnabled	Specifies whether the TextBox changes its state when hovered over.	TRUE	true, false
	inputAttr	Specifies attributes for the input element of the TextBox.	{}	Object
	isValid	Specifies whether the current value is valid.	TRUE	true, false
	mask	Specifies a mask to use when entering values.	""	String
	maskChar	Specifies the character used to represent masked characters.	"_"	String
	maskInvalidMessage	Specifies a message displayed when the mask rule is violated.	"Value is invalid"	String
	maskRules	Specifies custom mask rules.	{}	Object
	maxLength	Specifies the maximum number of characters allowed.	null	Number
	mode	Specifies the input type (e.g., "text", "password").	"text"	"text", "password", "email", etc.
	name	Specifies the name of the input element.	""	String
	onChange	A function that is executed when the value changes.	null	function()
	onContentReady	A function that is executed when the content is ready.	null	function()

onCopy	A function that is executed when text is copied.	null	function()
onCut	A function that is executed when text is cut.	null	function()
onDisposing	A function that is executed when the widget is disposed of.	null	function()
onEnterKey	A function that is executed when the Enter key is pressed.	null	function()
onFocusIn	A function that is executed when the TextBox is focused.	null	function()
onFocusOut	A function that is executed when the TextBox loses focus.	null	function()
onInitialized	A function that is executed when the widget is initialized.	null	function()
onInput	A function that is executed when text is input.	null	function()
onKeyDown	A function that is executed when a key is pressed down.	null	function()
onKeyUp	A function that is executed when a key is released.	null	function()
onOptionChanged	A function that is executed when a widget option is changed.	null	function()
onPaste	A function that is executed when text is pasted.	null	function()
onValueChanged	A function that is executed when the value changes.	null	function()
placeholder	Specifies a short hint displayed in the input area.	""	String
readOnly	Specifies whether the TextBox is read-only.	FALSE	true, false
rtlEnabled	Specifies whether the TextBox supports right-to-left text direction.	FALSE	true, false
showClearButton	Specifies whether the clear button is displayed.	FALSE	true, false
showMaskMode	Specifies when to display the mask in the TextBox.	"always"	"always", "onFocus"
spellcheck	Specifies whether the TextBox should be spellchecked.	TRUE	true, false
stylingMode	Specifies the styling mode of the TextBox.	"outlined"	"outlined", "underlined", "filled"
tabIndex	Specifies the Tab index of the TextBox.	0	Number
text	Specifies the text displayed in the TextBox.	""	String
useMaskedValue	Specifies whether to use the masked value in the TextBox.	FALSE	true, false
validationError	Specifies the validation error.	null	Object
validationErrors	Specifies an array of validation errors.	[]	Array
validationMessageMode	Specifies how validation messages are displayed.	"auto"	"auto", "always"
validationStatus	Specifies the validation status of the TextBox.	"valid"	"valid", "invalid", "pending"
value	Specifies the current value of the TextBox.	""	String, Number
valueChangeEvent	Specifies the DOM event after which the TextBox's value should be updated.	"change"	String
visible	Specifies whether the TextBox is visible.	TRUE	true, false
width	Specifies the width of the TextBox.	undefined	Number, String

Button	Options	Description	Default Value	Acceptable Values
	accessKey	Specifies a shortcut key to activate/focus the Button.	None	Any single character
	activeStateEnabled	Specifies whether the Button changes its state when interacted with.	TRUE	true, false
	disabled	Specifies whether the Button is disabled.	FALSE	true, false
	elementAttr	Specifies custom attributes for the Button's root element.	{}	Object
	focusStateEnabled	Specifies whether the Button can be focused.	TRUE	true, false
	height	Specifies the height of the Button.	undefined	Number, String
	hint	Specifies a hint text for the Button.	""	String
	hoverStateEnabled	Specifies whether the Button changes its state when hovered over.	TRUE	true, false
	icon	Specifies an icon to be displayed in the Button.	""	String
	onClick	A function that is executed when the Button is clicked.	null	function()

onContentReady	A function that is executed when the content is ready.	null	function()
onDisposing	A function that is executed when the widget is disposed of.	null	function()
onInitialized	A function that is executed when the widget is initialized.	null	function()
onOptionChanged	A function that is executed when a widget option is changed.	null	function()
rtlEnabled	Specifies whether the Button supports right-to-left text direction.	FALSE	true, false
stylingMode	Specifies the styling mode of the Button.	"contained"	"contained", "text", "outlined"
tabIndex	Specifies the Tab index of the Button.	0	Number
template	Specifies a custom template for the Button.	null	String, function()
text	Specifies the text displayed in the Button.	""	String
type	Specifies the type of the Button.	"normal"	"normal", "submit", "reset"
useSubmitBehavior	Specifies whether the Button behaves as a submit button.	FALSE	true, false
validationGroup	Specifies the validation group the Button belongs to.	undefined	String
visible	Specifies whether the Button is visible.	TRUE	true, false
width	Specifies the width of the Button.	undefined	Number, String

Methods	Method name	Description
	blur()	Removes focus from the component.
	clear()	Clears the value of the component.
	dispose()	Disposes of the component instance.
	focus()	Sets focus on the component.
	reset()	Resets the component to its default value.
	validate()	Validates the current value. Returns a validation result object.
	getDataSource()	Gets the DataSource instance.
	getElement()	Gets the root HTML element of the component.
	option(optionName)	Gets the value of the specified option.
	option(optionName, value)	Sets the value of the specified option.
	on(eventName, handler)	Subscribes to the specified event using the provided handler.
	off(eventName, handler)	Unsubscribes from the specified event using the provided handler.
	registerKeyHandler(key, handler)	Registers a handler for a specified key.
	repaint()	Redraws the component.

Events	Event name	Description
	onDisposing	Triggered when the component is disposed of.
	onInitialized	Triggered when the component is initialized.
	onOptionChanged	Triggered when an option of the component is changed.
	onFocusIn	Triggered when the component receives focus.
	onFocusOut	Triggered when the component loses focus.
	onValueChanged	Triggered when the component's value is changed.
	onInput	Triggered when the input event occurs on the component.
	onChange	Triggered when the change event occurs on the component.
	onKeyDown	Triggered when a key is pressed down while the component is focused.
	onKeyUp	Triggered when a key is released while the component is focused.
	onEnterKey	Triggered when the Enter key is pressed while the component is focused.

onCut	Triggered when the cut event occurs on the component.
onCopy	Triggered when the copy event occurs on the component.
onPaste	Triggered when the paste event occurs on the component.
onCustomItemCreating	Triggered when a custom item is created in the component.
onContentReady	Triggered when the content of the component is ready.
onClosed	Triggered when the drop-down list is closed.
onOpened	Triggered when the drop-down list is opened.
onItemClick	Triggered when an item is clicked.
onSelectionChanged	Triggered when the selection is changed.
onCustomItemCreating	Triggered when a custom item is being created.