# **Overview**

DevExtreme is a set of enterprise-ready UI component suites for Angular, React, Vue, and jQuery. It is everything you need to create responsive web apps for touch devices and traditional desktops: data grid, interactive charts, data editors, navigation and multi-purpose widgets. These controls are designed to look great and to provide powerful functionality in any browser.

DevExtreme provides ready to use templates and demos for **Data grid, Pivot Grid, Tree List, Scheduler, HTML Editor, Diagram, Charts, Gantt, Gauges, Navigation, Editors, Forms and Multi-Purpose, File Management, Actions and List, Maps, Dialogs and Notification.**

DevEtreme Provide collection of UI elements like **Accordion, ActionSheet, Autocomplete, BarGauge, Box, Bullet, Button, ButtonGroup, Calendar, Chart, CheckBox, CircularGauge, ColorBox, ContextMenu, DataGrid, DateBox, Diagram, Drawer, DropDownBox, DropDownButton, FileManager, FileUploader, FilterBuilder, Floating Action Button, Form, Funnel, Gallery, Gantt, HtmlEditor, LinearGauge, List, LoadIndicator, LoadPanel, Lookup, Map, Menu, MultiView, NumberBox, PieChart, PivotGrid, PolarChart, Popover, Popup, ProgressBar, RadioGroup, RangeSelector, RangeSlider, Resizable, ResponsiveBox, Sankey, Scheduler, ScrollView, SelectBox, Slider, Sparkline, Switch ,TabPanel, Tabs ,TagBox ,TextArea , TextBox, TileView, Toast, Toolbar ,Tooltip ,TreeList ,TreeView, VectorMap.**

# **DevExtrem with Jquery**

# **DevExtreme jQuery Components**

The DevExtreme jQuery Component Suite is a feature-complete set of 65+ responsive and touch-enabled UI components implemented as jQuery plugins. The components are a data grid, interactive charts, data editors, navigation and multi-purpose UI components.

In addition to components, the suite includes the following features:

* [Data layer](https://js.devexpress.com/Documentation/20_2/Guide/Data_Binding/Data_Layer/)
* [Client-side data validation](https://js.devexpress.com/Documentation/20_2/Guide/UI_Components/Common/UI_Widgets/Data_Validation/)
* [Themes and styles](https://js.devexpress.com/Documentation/20_2/Guide/Themes_and_Styles/Predefined_Themes/)
* [Localization](https://js.devexpress.com/Documentation/20_2/Guide/Common/Localization/)
* [Modularity](https://js.devexpress.com/Documentation/20_2/Guide/Common/Modularity/Link_Modules/)
* [Customization using templates](https://js.devexpress.com/Documentation/20_2/Guide/UI_Components/Common/Templates/)

These CDN links needed to use DevExtreme Library in Our Jquery Application.

* <script type="text/javascript" src="https://code.jquery.com/jquery-3.5.1.min.js"></script>
* <link rel="stylesheet" href="https://cdn3.devexpress.com/jslib/20.2.10/css/dx.common.css">
* <link rel="stylesheet" href="https://cdn3.devexpress.com/jslib/20.2.10/css/dx.light.css">
* <script type="text/javascript" src="https://cdn3.devexpress.com/jslib/20.2.10/js/dx.all.js"></script>

# **Component Configuration Syntax**

Every UI Component must be within container call <div> and must have a unique ID.

Then later this container will be used for plugin DevExtreme UI components via JQuery.

Later we can configure that component by passing an argument to that plugin.

## **Get and Set Properties**

For getting Property we can create an instance of UI Component bypassing “instance” and access get all the properties of that instance via its option method or get single property via passing the name of the property to that option method.

We can also directly get all the properties or single properties without crating instances bypassing the ‘option’ parameter to that components.

We can set property value via the same get method with an extra parameter that contains a value for that property.

We can set multiple properties via passing a single object literal which contains a key as the property name and value as the value for that property.

We can do that with or without creating an instance of that component.

## **Call Methods**

We can call the components method via the same way we get and set property.

We just need to pass the name of the method and its argument to the plugin or by creating its instance and we can just call that method.

By creating an instance we can just call the method we don’t need to pass the name of that method as an argument to that instance.

## **Handle Events**

We can subscribe to the event in the same way to set property.

All we need to just pass an object literal to that plugin that contains the key as event name and value as a function that handles that event.

We can pass multiple events to one object literal.

We can also call the .on method of instance to set an event also.

To unsubscribe that event we can call the .off method which contains the event name and method which we want to reattach.

## **Dispose of a UI Component**

For Disposing UI component, we can pass ‘dispose’ as a parameter and then remove its container via calling the remove() method.

# **Editor (Overview)**

## **CheckBox**

This element is used when user want select or deselect something on display.

These are importans properties of CheckBox.

accessKey :- this property is used to access this element when user press key on keyboard.

Disabled:- this property is used to enable disable checkbox element.

elementAttr:- this property is used to set value to to current element’s attribute. We can pass object which contains all the value for this element.

Height:- this property is used to set height of checkbox.

Hint:- this property is used to display hint for checkbox.

isValid:- this property is used to check that checkbox value passed all the validation rule or not.

Name: this property is used to set name of checkbox container.

readOnly:- This property is used to set whether this checkbox is readonly or not.

Text:- This property is used to set lable for current checkbox.

validationError:- this property is used to check all the validation errors.

validationMessageMode:- this property is used to set how error message will be displayed. Value could be auto or always.

value:- either true, false or undefined,

visible:- this property indicate that whether checkbox should be visible or not.

Width:- this property used to set width of checkbox.

Theses are importants methods of checkbox:

defualtOption(rule): this method accept object which contains device and options attribute to set particular options for specific type of device.

Dispose() This method is used to dispose checkbox instance.

Element() this method is used to get HTMLEelement of checkbox.

Focus() this method is used to focus on checkbox.

Getinstanse() this method is used to getinstance of HTML element.

Instance() this method is used to get instance current instance.

Off(eventname) this method is used to deattach event handler from particular event.

On(eventname) this method is used to attach event handle for specific event.

Option(name,value) this method is used to set option for checkbox.

registerHandler(key,method): this method is used to register specific keyboard key to element.

These are usefull events that used in checkbox.

contentReady: Raised when element is ready.

Disposing: Raised when element is disposing.

Initialized: Raised when element is initialized.

Optionchanged: Raised when element optionchanged.

valueChanged: Raised when element ValueChanged.

## **DateBox**

Date box is used to get Date or Time or Date and Time from user.

It will show user a Datebox where user can choose Date or time which he/she wants to enter.

These are the usefull properties of DateBox:

acceptCustomValue: Accept boolen value that indicate that whether user can add custome date or not by typing.

adaptivityEnabled: This property is used to set adaptivity for different type of devices and screens. It will indicate that whether clock is displayed or not based on screen and device.

applyButtonText: Text for Ok Button.

applyValueMode: Indicate that value will be instantly selectected or user have to use ok button.

Buttons[]: it is used to display button withing DateBox. Only accepted button name as string here is clear and dropdown. We can also provide custome buton by providing object of it.

Button objcect contains location, name, and option property.

Where option contains button’s configurationProperty like , stylingMode, width, elementAttr, onClick.

calendarOptions: These option is useful when pickertye is calender and we can configure calander property by using this property.

dateOutOfRangeMessage: this property is usefull for display message when date is out of Ranged.

Default max is 50 years from current date and defult min is 1900 year.

dateSerializationFormat: is used to serilized date formate vlkue.

disabledDates: this property usefult when pickertype is calander. This property accepts array of Dates which we want to disable on calnder so user cannot pickeup dates.

displayFormate: This property is set formate of date and time in UI element that displays to user.

inputAttr: This property is used to set attributes value of input element of this Datebox.

invalidDateMessage: This property is usefult to set Error message when user enters invalid date and time formate.

Interval: This property is used when type is time and pickertype is list. This property is used to set interval between Time.

Max: This property is used to set max date user can enter. Defualt value is 50 years from current date.

Min: This property is used to set minimum of date user can enter. Default value is 1900 year.

pickerType: This is very useful property. This property is used to set Date picker type. User can set native, calender, roller or List Property here.

Placeholder: This property is used set placeholder of dateBox.

showAnalogClock: This property is used to set whether analogclock should displayed or not. This prorty is only useful when pickertye is calender.

showClearButton: This property is used to set whether Clear Button shold displayed or not.

showDropDownButton: This property is used to set whether Dropdown Button shold displayed or not.

Spellcheck: This property is used to set whether UI displays speling mistake or not.

stylingMode: This property is used to stylingmode for Datebox. This could be outlined, underlined and filled.

Text: This property is used to display Text value of Datebox. This proprety is Readonly.

Type: This property is used to set Type of DateBox. It could be date , datetime and time.

useMaskBehavior: This property is used to Masked user valued to specidifc formate.

Value: This property is used to set or get actual value of DateBox.

These are the important methods of DateBox.

close() – This method is used to close() DropDown Widow.

content() – This method is used to Gets the popup window's content.

getButton(name) – This method is used to get speccifice button based on its name.

open() – This method is used to open dropDown Window.

## **DropDownBox**

acceptCustomValue: This property is used to allow custome value to DropDownBox.

contentTemplate: This method is used to render drop down template.

dataSource: this property is used to Bind data with dropdown.

Difference between DropDownBox and SelectBox.

|  |  |  |
| --- | --- | --- |
|  | **SelectBox** | **DropDownBox** |
| Accept custom values | [Yes](https://js.devexpress.com/Documentation/Guide/UI_Components/SelectBox/Create_a_User-Defined_Item/) | [Yes](https://js.devexpress.com/Documentation/ApiReference/UI_Components/dxDropDownBox/Configuration/#acceptCustomValue) |
| Custom buttons | [Yes](https://js.devexpress.com/Documentation/ApiReference/UI_Components/dxSelectBox/Configuration/buttons/) | [Yes](https://js.devexpress.com/Documentation/ApiReference/UI_Components/dxDropDownBox/Configuration/buttons/) |
| Keyboard navigation | [Yes](https://js.devexpress.com/Documentation/Guide/UI_Components/SelectBox/Keyboard_Support/) | Requires custom configuration |
| Search box in a drop-down field | - | Requires custom configuration |
| Display groups | [Yes](https://js.devexpress.com/Documentation/ApiReference/UI_Components/dxSelectBox/Configuration/#grouped) | Requires custom configuration |
| Search based on user input | [Yes](https://js.devexpress.com/Documentation/ApiReference/UI_Components/dxSelectBox/Configuration/#minSearchLength) | Requires custom configuration |
| Multiple selection | - | Requires custom configuration |
| Paging | [Yes](https://js.devexpress.com/Documentation/Guide/UI_Components/SelectBox/Enable_Paging/) | Requires custom configuration |

[Reference Link: https://js.devexpress.com/Documentation/Guide/UI\_Components/SelectBox/How\_to\_Choose\_a\_Drop-Down\_Editor/](https://js.devexpress.com/Documentation/Guide/UI_Components/SelectBox/How_to_Choose_a_Drop-Down_Editor/)