



# Kendo UI Part 1

Telerik Framework

Telerik Software Academy  
Learning & Development Team  
<http://academy.telerik.com>

A screenshot of the Kendo UI website, which has an orange header with the Kendo UI logo and navigation links for UI Framework, Mobile, Demos, Roadmap, What's new, and Download. The main content area features a large image of various devices (laptop, tablet, smartphone) displaying charts and graphs, with text about setting the new standard for HTML5 UI.

# Table of Contents

1. Installation & Introduction
2. Basic Widgets
3. DataSource
4. Advanced Widgets
5. Templates
6. Validator



# Installation & Introduction

Getting ready for Kendo UI



- ◆ First include the style sheets
  - ◆ kendo.common.min.css (common css first)
  - ◆ kendo.{theme}.min.css (theme css)
- ◆ Then include the JavaScript files
  - ◆ jquery.min.js (jQuery is needed first)
  - ◆ kendo.web.min.js (Kendo UI web)
- ◆ More information here
  - ◆ <http://docs.telerik.com/kendo-ui/getting-started/introduction>



# Installing Kendo UI

Live Demo

# Basic Widgets

## The cool stuff!



## ◆ Instantiating Widgets

- Appended to jQuery objects (DOM elements)
- May have additional options depending on the widget and the usage of it

```
$('#menu').kendoMenu();  
  
$('#grid').kendoGrid({  
    dataSource: carsDataSource,  
    editable: true,  
    toolbar: ['create']  
});
```

## ◆ Getting Widgets Reference

- Use **jQuery data function**
- Pass the **Pascal case name of the widget prefixed with "kendo"**

```
$('#menu').data('kendoMenu');
```

```
$('#grid').data('kendoGrid');
```

## ◆ Using Widgets API

- Widgets expose functions and properties

```
$('#ac').kendoAutoComplete(['1', '2', '3']);  
  
var autocomplete = $('#ac').data('kendoAutoComplete');  
  
autocomplete.value('4');  
var value = autocomplete.value();  
  
alert(value); // Displays '4'
```

## ◆ Subscribing to events

- Specify at initialization

```
$('#ac').kendoAutoComplete({  
    change: function() { this... }  
});
```

- Use bind and unbind

```
$('#ac').kendoAutoComplete();  
var autocomplete = $('#ac').data('kendoAutoComplete');  
  
autocomplete.bind('change', function() { this... });
```

- ◆ The DOM elements
  - ◆ The element property

```
var element = $('#myWindow').data('kendoWindow').element;
```

- ◆ The wrapper property

```
var element = $('#myWindow').data('kendoWindow').wrapper;
```

- ◆ Destroying widgets
  - Using the `destroy` function
  - Deleted the widget instance
  - Removes auto generated HTML content
  - Destroys all child widgets

```
$('#myGrid').data('kendoGrid').destroy();
```

# Common Functions

Live Demo

## ◆ Kendo Color Picker

- **palette**: "basic", "websafe", array of colors
- **columns**: number
- **value**: string
- **methods**: close, open, color, enable
- **events**: change, select, open, close

```
$('#element').kendoColorPicker({  
    palette: 'basic'  
});
```

## ◆ Kendo Date Time Picker

- **options:** value, min, max, format, start, interval
- **methods:** close, open, enable, min, max
- **events:** change, open, close

```
$('#dateTimePicker').kendoDateTimePicker({  
    value: new Date(2000, 10, 10, 10, 0, 0),  
    min: new Date(1950, 0, 1, 8, 0, 0),  
    max: new Date(2049, 11, 31, 18, 0, 0)  
});
```

## ◆ Kendo Masked Text Box

- **mask**: o, 9, #, L, ?, &, C, A, a
- **promptChar**: string
- **methods**: enable, readonly, value
- **events**: change

```
$( '#maskedtextbox' ).kendoMaskedTextBox({  
    mask: '(000) 000-0000',  
    promptChar: '*'  
});
```

## ◆ Kendo Menu

```
<ul id="menu">
    <li>Normal Item
        <ul>
            <li>First</li>
            <li>Second</li>
        </ul>
    </li>
    <li>Third</li>
</ul>
```

```
$("#menu").kendoMenu();
```

## ◆ Kendo TabStrip

```
<ul>
    <li class="k-state-active">Paris</li>
    <li>New York</li>
</ul>
<div>
    <p>Rainy weather in Paris.</p>
</div>
<div>
    <p>Sunny weather in New York.</p>
</div>
```

```
$("#tabstrip").kendoTabStrip();
```

## ◆ Kendo Tooltip

- **options:** position, width, height, content, showOn, autoHide
- **methods:** show, hide, refresh
- **events:** show, hide

```
$("#target").kendoTooltip({ content: "Tooltip content" });
```

# DataSource

## Data, data, data



- ◆ Abstraction over local or remote data
- ◆ Play central role in Kendo UI applications
- ◆ Retrieve data from local or remote end point
- ◆ Provides CRUD operations and serialization
- ◆ Provides filtering, grouping, page sizing
- ◆ Synchronizing updates (batch and normal)
- ◆ And many more



- ◆ Initialization with new kendo.data.DataSource
- ◆ Takes an JSON object as parameter
- ◆ The JSON object contains variable options
- ◆ data option – array of same objects or string



```
var cars = [  
  { make: 'Opel', model: 'Insignia' , year: '2009'},  
  { make: 'Audi', model: 'A5', year: '2008'},  
  { make: 'BMW', model: 'M3', year: '2010'},  
  { make: 'Mercedes', model: 'CL', year: '2011'}  
];  
var carsDataSource = new kendo.data.DataSource({  
    data: cars  
});
```

- ◆ **columns option – array of objects**
  - ◆ **field, width, title**
- ◆ **aggregate option – array of objects**
  - ◆ **field, aggregate**
  - ◆ **"average", "count", "max", "min", "sum"**



```
...
columns: [
    { field: 'make', width: 50, title: 'Make' },
    { field: 'model', width: 50, title: 'Model' }
],
aggregate: [
    { field: 'power', aggregate: 'sum' },]
...
...
```

- ◆ filter option – array of objects

- ◆ logic option: 'and', 'or'
- ◆ filters option: array of objects
  - ◆ field, operator, value
  - ◆ operators: "eq", "neq", "lt", "lte", "gt", "gte"



```
...
filter: {
    logic: 'and',
    filters: [
        { field: 'make', operator: 'eq', value: 'Audi' },
        { field: 'year', operator: 'gt', value: 2006 }
    ]
}
...
...
```

- ◆ group option – array of objects
  - ◆ field, dir, aggregates
  - ◆ dir: 'asc' and 'desc'



```
...
group: {
    field: 'make',
    dir: 'desc',
    aggregates: [
        { field: 'power', aggregate: 'max' },
        { field: 'make', aggregate: 'count' }
    ]
}
...
```

- ◆ sort option – array of objects
  - ◆ field, dir
  - ◆ dir: 'asc' and 'desc'



```
...
sort: [
  { field: 'year', dir: 'desc' },
  { field: 'make', dir: 'asc' }
]
...
...
```

- ◆ **transport option – array of objects**

- ◆ **create, read, update, destroy**
  - ◆ **url, dataType**
  - ◆ **parameterMap – function for parsing data**



```
...
transport: {
    read: {
        url: 'http://someurl.com/api/read',
        dataType: 'json'
    },
...
}
```

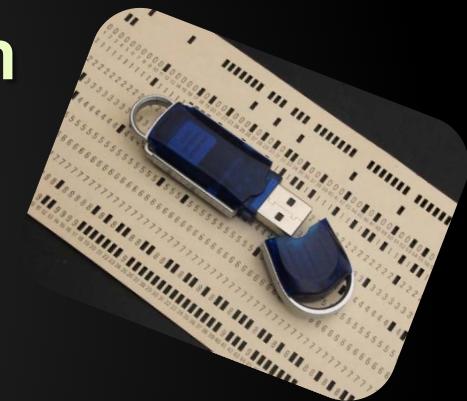
- ◆ batch option – boolean
- ◆ page option – number
- ◆ pageSize option – number
- ◆ serverPaging option – boolean
- ◆ serverSorting option - boolean
- ◆ events – change, error, sync



```
...  
change: function (e) {  
    ...  
}  
...  
}
```

## ◆ Methods

- ◆ add – object or Kendo.data.model
- ◆ aggregate – get or set configuration
- ◆ aggregates – returns result
- ◆ at – indexator
- ◆ data – gets or sets the data array
- ◆ fetch – reads the data, needs ready function
- ◆ filter – gets or sets the configuration
- ◆ group – gets or sets the configuration
- ◆ indexOf – return the index of an object in data



## ◆ Methods

- **insert** – inserts data at specified index
- **page** – gets or sets the current page
- **pageSize** – gets or sets the page size
- **read** – reads the data
- **remove** – removes item
- **sort** – gets or sets the configuration
- **sync** – syncs data with remote service
- **total** – number of items in data
- **view** – return corresponding data (with fetch)

# DataSource

Live Demo

# Templates

## Mustache, Beard, Eyebrows



- ◆ Kendo UI templates – in script tags
- ◆ Type of tag – “text/x-kendo-template”
- ◆ Should have id attribute
- ◆ Initialized with kendo.template(\$('#id').html())
- ◆ Takes object parameter with data
- ◆ Appended to other DOM (jQuery) elements

```
<script type="text/x-kendo-template" id="some-id">
// template </script>

var tmpl = kendo.template($('#some-id').html());

$('#some-tag').append(template({ /* data obj */ }));
```



## ◆ Syntax

- Normal HTML syntax
- # Between number sign you can write JS code
- #: Encodes the values
- #= Takes the value from a parameter

```
<script type="text/x-kendo-template" id="car">
    <div>
        <span>#: make #, </span>
        <a href="/cars/#= id #">#: model #</a>
    </div>
</script>
var carTemplate = kendo.template($('#car').html());
$('#some-tag').append(template({
    id: i, make: car.make, model: car.model }));
```

# Templates

Live Demo

# Validator

Client-side data validation



## ◆ Validator

- Data is validated through attributes on tags
- required – makes field required
- pattern – defines RegEx for validation
- min – defines minimum value
- max – defines maximum value
- validationMessage – message if not valid

```
<input type="tel" id="tel" name="tel" placeholder="Phone"
       pattern="\d{10}"
       required
       validationmessage="Phone should have 10 digits"
/>
```

## ◆ Validator

- Initialized on jQuery object with  
`.kendoValidator().data('kendoValidator');`
- Validity is checked with `.validate()`

```
var validator =  
    $('#id').kendoValidator().data('kendoValidator');  
  
if (validator.validate()) {  
    // do something  
}  
else {  
    // do something else ☺  
}
```

# Validator

Live Demo

# Advanced Widgets

## The cooler stuff!



## ◆ Kendo Editor

- **tools:** variety of tools
- **methods:** refresh, update, value
- **events:** change, keyup, keydown, select

```
$("#editor").kendoEditor({  
    tools: [  
        "bold",  
        "italic",  
        "underline",  
        "foreColor"  
    ]  
});
```

## ◆ Kendo TreeView

```
<ul id="treeView">
    <li>Item 1
        <ul>
            <li>Item 1.1</li>
            <li>Item 1.2</li>
        </ul>
    </li>
    <li>Item 2</li>
</ul>
```

```
$("#treeView").kendoTreeView();
```

## ◆ Kendo TreeView

- **methods:** append, expand, findByText, parent
- **events:** expand, change, select

```
dataSource: [
    {
        text: "Item 1",
        items: [
            { text: "Item 1.1" }
        ]
    },
    { text: "Item 2" }
]
```

## ◆ Kendo Window

- **options:** position, actions, resizable, visible, draggable, modal, title
- **methods:** open, close, center, maximize, minimize
- **events:** close, open, resize

```
$("#window").kendoWindow();
```

## ◆ Kendo ListView

- **options:** template, dataSource, selectable, editable, editTemplate
- **methods:** add, remove, edit, save, select
- **events:** change, edit, save

```
$("#repo-list").kendoListView({  
    dataSource: dataSource,  
    template: kendo.template($("#tpl").html())  
});
```

## ◆ Kendo Grid

- **options:** dataSource, editable, toolbar, columns, pageable, scrollable, sortable, filterable, groupable
- **methods:** addRow, editRow, editCell, refresh, saveChanges, select, cancelChanges
- **events:** cancel, change, edit, save

## ◆ Kendo Grid

```
$('#grid').kendoGrid({  
    dataSource: dataSource,  
    editable: true,  
    toolbar: ['create'],  
    pageable: {  
        input: true,  
        numeric: false  
    },  
    scrollable: false,  
    sortable: true,  
    filterable: false,  
    groupable: true,  
    columns: []  
});
```

# Additional Info

More is better!



- ◆ Documentation

- ◆ <http://docs.telerik.com/kendo-ui/>

- ◆ Demos

- ◆ <http://demos.telerik.com/kendo-ui/>

- ◆ Training room

- ◆ <http://trykendoui.telerik.com/>

- ◆ Downloads

- ◆ <http://www.telerik.com/download/kendo-ui>

# Mobile, Charts and MVC

A whole other universe!

Questions?