# Getting Started

For more information on this topic, see: <http://docs.kendoui.com/getting-started/what-is-kendo>

Getting and adding KendoUI to your project.

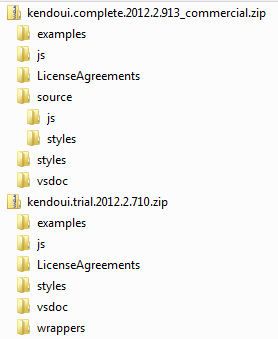
## Getting KendoUI

For more information on this topic, see: <http://docs.kendoui.com/getting-started/downloading-kendo>

KendoUI is divided into three parts: Web, Mobile, and DataViz.

The preferred method of getting KendoUI is to browse to <http://www.kendoui.com/download> and download KendoUI as a .zip file.

Depending on whether you are using the Trial version, or have purchased the full Commercial version, you will have a .zip file that follows one of these directory structures:



The \js and \styles folders in the root of the .zip file contain the “minified” versions of the KendoUI .js and .css files. If you have the commercial version of KendoUI, you may also notice the \source folder. This folder also contains \js and \styles folders, but these are the full, non-minified versions of the .js and .css. These would typically be used for local development and debugging, but the minified versions are the ones that should be deployed in a real application. For that reason, we will use the minified files in the remainder of this example project.

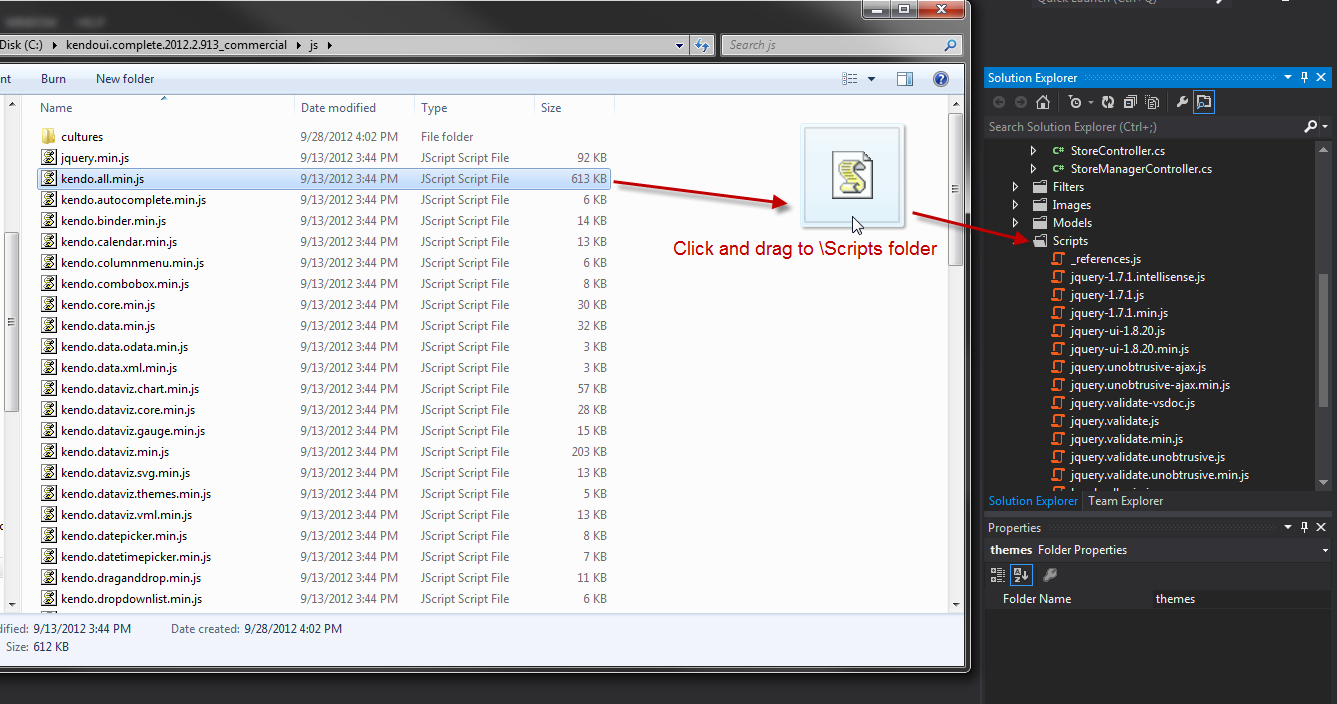
## Adding KendoUI to your ASP.NET MVC Project

For more information on this topic, see: <http://docs.kendoui.com/getting-started/using-kendo-with/aspnet-mvc/introduction>

To add KendoUI to your project, you just need to copy the files you need into your project. For a standard ASP.NET MVC project, files from the Kendo .zip \js folder are copied to ASP.NET MVC’s \Scripts folder, and files from the Kendo .zip \styles folder are copied to ASP.NET MVC’s \Content\themes\base folder.

If you know exactly which Kendo controls and styles you are going to use in your project, you can include individual JavaScript files for just those controls. This can provide a much smaller download for your end users. Optionally, you can use the single file “kendo.all.min.js” which contains the minified files for all of Web, Mobile, and DataViz. You do not need to add kendo.all.min.js *and* each individual kendo.\*.min.js file. Include just one or the other. For more information, see: <http://docs.kendoui.com/getting-started/javascript-dependencies>

When copying the files to your ASP.NET MVC project, you can simply drag and drop the files you want from Windows Explorer to the Solution Explorer in VisualStudio. This will automatically copy the files to your project’s directory, and add them to the project.



## VisualStudio Integration and Templates for ASP.NET MVC

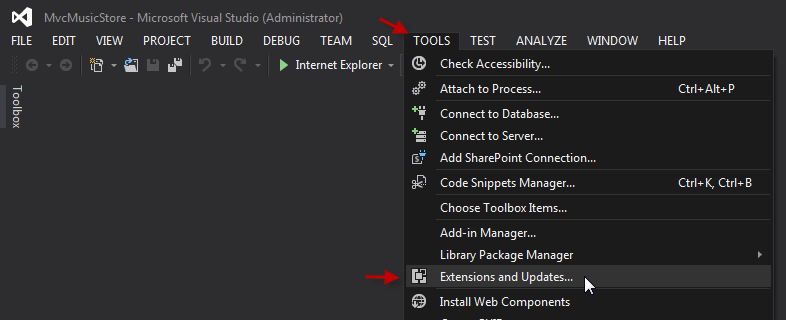
For more information on this topic, see: <http://docs.kendoui.com/getting-started/using-kendo-with/aspnet-mvc/introduction>

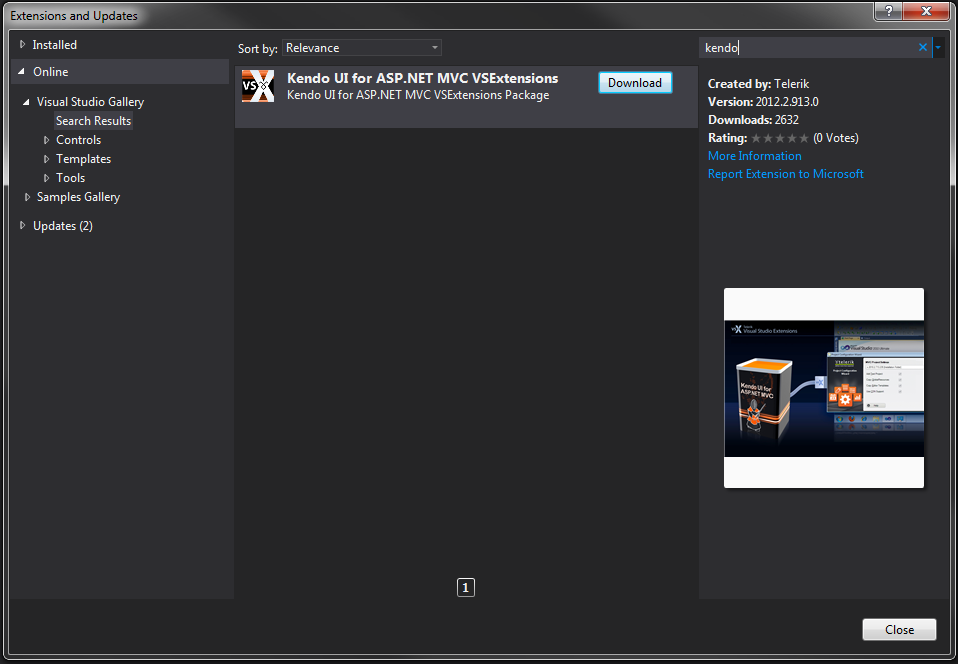
Optionally, Kendo UI provides Kendo UI Complete for ASP.NET MVC.

Kendo UI Complete for ASP.NET MVC is a set of HTML helpers which help you configure Kendo UI widgets by using server-side code in ASP.NET MVC applications. It also provides ready-to-use editor templates based on various Kendo UI widgets.

There is also a VisualStudio Extension available in the VisualStudio Gallery here: <http://visualstudiogallery.msdn.microsoft.com/65b78c2c-951e-43a8-bae7-f9039f59fb9b>

Or it can be added through VisualStudio by selecting “Tools | Extensions and Updates…” from the main menu, then searching for “kendo”.





If you would like to know more about this set of utilities, please refer to the above link.

## jQuery

Kendo is dependent on jQuery. When Kendo JavaScript files are included in an HTML page, jQuery must be included before Kendo.

# Using KendoUI Web

## Creating the Music Store Consumer Web Site

### Include Kendo files in rendered Views

Add the CSS and JS to all rendered pages by adding entries to \Views\Shared\\_Layout.cshtml. We need to make sure that Kendo’s JavaScript files come after jQuery. For styling, we include kendo.common.min.css, plus one of the Kendo UI Themes. In this example we are using kendo.default.min.css, which is the Default theme.

<head>

<meta charset="utf-8" />

<title>@ViewBag.Title - Kendo UI Music Store</title>

<link href="~/favicon.ico" rel="shortcut icon" type="image/x-icon" />

<meta name="viewport" content="width=device-width" />

@Styles.Render("~/Content/css")

<link href="~/Content/kendo.common.min.css" rel="stylesheet" /> // <-- Added Here.

<link href="~/Content/kendo.default.min.css" rel="stylesheet" /> // <-- Added Here.

@Scripts.Render("~/bundles/modernizr")

</head>

<body>

...

@Scripts.Render("~/bundles/jquery")

<script src="~/Scripts/kendo.all.min.js"></script> // <-- Added Here.

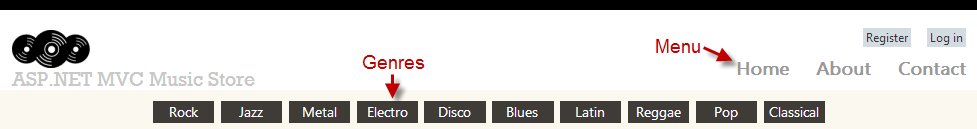
@RenderSection("scripts", required: false)

</body>

### Add the main menu

Let’s use the Kendo Menu control to add a navigation menu to the Music Store.

The standard MVC Music Store displayed three navigation items in the upper-right. Home, About, and Contact. It also listed all the Genres in the store as a row of buttons:



Instead, we will combine the Genres list into a pull-down item in the Kendo Menu control.

In the file \Views\Shared\\_Layout.cshtml the main menu is defined here:

<nav>

<ul id="menu">

<li>@Html.ActionLink("Home", "Index", "Home")</li>

<li>@Html.ActionLink("About", "About", "Home")</li>

<li>@Html.ActionLink("Contact", "Contact", "Home")</li>

</ul>

</nav>

Converting this to a Kendo menu is easy. We just add the Javascript:

$(document).ready(function () {

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

});

This uses a jQuery selector to get our <ul> tag with the id “menu,” and turns it into a Kendo UI menu.

Next we can move the Genres list into a pull-down menu. Any <li> elements that in turn contain another list <ul> element are treated as a pull-down item. Lets start with an empty pull-down menu item for our Genres:

<li>Genres

<ul>

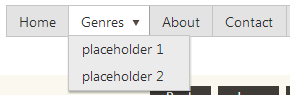
<li>placeholder 1</li>

<li>placeholder 2</li>

</ul>

</li>

This creates the pull-down item:



The MVC Music Store renders the Genre list with a partial:

@{Html.RenderAction("GenreMenu", "Store");}

## Creating the Music Store Administrative Web Site

# Using KendoUI Mobile

## Creating the Music Store Consumer Mobile Site

# Using KendoUI DataViz to add Charts

## Adding Charts to the Administrative panel.