Module 10: **Using JavaScript and jQuery for Responsive MVC 5 Web Applications**

# ****Contents:****

|  |  |
| --- | --- |
|  | [Module Overview](https://www.skillpipe.com/api/2.1/content/urn:uuid:4ed7876c-8864-4cf5-91ee-491726dfe322@2020-12-11T17:05:09Z/OPS/html/20486C10.html#P1) |
| **Lesson 1:** | [Rendering and Executing JavaScript Code](https://www.skillpipe.com/api/2.1/content/urn:uuid:4ed7876c-8864-4cf5-91ee-491726dfe322@2020-12-11T17:05:09Z/OPS/html/20486C10.html#P2) |
| **Lesson 2:** | [Using jQuery and jQueryUI](https://www.skillpipe.com/api/2.1/content/urn:uuid:4ed7876c-8864-4cf5-91ee-491726dfe322@2020-12-11T17:05:09Z/OPS/html/20486C10.html#P3) |
| **Lab:** | [Using JavaScript and jQuery for Responsive MVC 5 Web Applications](https://www.skillpipe.com/api/2.1/content/urn:uuid:4ed7876c-8864-4cf5-91ee-491726dfe322@2020-12-11T17:05:09Z/OPS/html/20486C10.html#P4) |
|  | [Module Review and Takeaways](https://www.skillpipe.com/api/2.1/content/urn:uuid:4ed7876c-8864-4cf5-91ee-491726dfe322@2020-12-11T17:05:09Z/OPS/html/20486C10.html#P5) |

# ****Module Overview****

Responsive web design is a web designing approach that helps create visually rich and interactive web applications. JavaScript plays an important role in the development of responsive web applications. You need to know how to use JavaScript to implement application logic and resize interface elements, without triggering a full-page refresh. To simplify adding JavaScript to your web application, you need to know how to use libraries such as jQuery, jQuery UI, and jQuery Mobile.

### ****Objectives****

After completing this module, you will be able to:

|  |  |
| --- | --- |
| • | Add JavaScript code to your web application. |
| • | Use the jQuery and jQuery UI libraries, in your web application. |

# Lesson 1: ****Rendering and Executing JavaScript Code****

You can create interactive HTML elements in your web application by using JavaScript. ASP.NET renders these interactive elements on your webpages. You can add packaged JavaScript libraries to your project by using the NuGet tool. You should know how to use AJAX to update the contents of webpages. By using AJAX, you can optimize the performance of your web application. In addition, you should know how the content delivery network (CDN) helps take content geographically closer to users.

## ****Lesson Objectives****

After completing this lesson, you will be able to:

|  |  |
| --- | --- |
| • | Describe how to add JavaScript files to an MVC application. |
| • | Describe how to call JavaScript functions in JavaScript libraries. |
| • | Describe how to use JavaScript Libraries in MVC 5 web applications. |
| • | List the benefits of using CDN to improve the performance of JavaScript libraries. |
| • | Describe how to use the NuGet tool to add packages. |
| • | Use the NuGet tool to add a JavaScript Library. |

## ****Adding JavaScript Files****

You can add JavaScript code to web applications by:

|  |  |
| --- | --- |
| • | Adding the JavaScript code to HTML. |
| • | Defining the JavaScript code in dedicated JavaScript files. |

**Inserting a JavaScript Function**

<body>

<script type="text/javascript">

function HelloWorld() {

alert('Hello World');

}

</script>

<div>

...

</div>

</body>

If you have multiple HTML pages in a web application, you need to add JavaScript code for each HTML page. You cannot simultaneously add JavaScript code for multiple HTML pages. Therefore, you can define the JavaScript code in a JavaScript file (.js file). Then, you can reference the JavaScript file in multiple HTML pages. This enables you to maintain a single JavaScript file, to edit the JavaScript code for multiple HTML pages. You can also have multiple JavaScript code files for a single HTML page.

The following image displays the Add New Item dialog box that helps to add a JavaScript file.

**FIGURE 10.1: ADDING A JAVASCRIPT FILE**

**Referencing JavaScript Files**

<script src="~/Scripts/JavaScript1.js" type="text/javascript"></script>

You can create the Scripts folder in the root folder of your MVC project, and then save all JavaScript files in the Scripts folder.

**Question**: What are the advantages of using a JavaScript file?

## ****Calling JavaScript Procedures****

You can call functions defined in JavaScript files by using script blocks or event handlers.

**Using a Script Block**

<body>

<script type="text/javascript">

HelloWorld();

</script>

<div>

Hello

</div>

</body>

Before calling a JavaScript function, you need to define the function by using a script block. Then, you must reference the JavaScript file from the HTML pages.

If you want to avoid calling the JavaScript function directly, you can use the **onclick** JavaScript event to trigger JavaScript functions. The **onclick** event initiates the JavaScript function assigned to an HTML file, when you click the corresponding HTML element. JavaScript functions that are attached to document object model (DOM) events are called event handlers.

**Using an Event Handler**

<body>

<div>

Hello

<input type="button" value="Hello" onclick="HelloWorld();" />

</div>

</body>

**Question**: What is the advantage of initiating JavaScript functions by using JavaScript events?

## ****JavaScript Libraries****

You can reduce the time taken to develop applications by using JavaScript libraries. JavaScript libraries help to:

|  |  |
| --- | --- |
| • | Reduce the amount of code you need to write to add functions. |
| • | Reduce the time the system takes to debug the application. |

Some commonly used JavaScript libraries include the following:

|  |  |
| --- | --- |
| • | jQuery (<http://go.microsoft.com/fwlink/?LinkID=288974&clcid=0x409>) |
| • | jQuery UI (<http://go.microsoft.com/fwlink/?LinkID=288975&clcid=0x410>) |
| • | jQuery Mobile (<http://go.microsoft.com/fwlink/?LinkID=288976&clcid=0x411>) |
| • | jQuery Validation (<http://go.microsoft.com/fwlink/?LinkID=288977&clcid=0x412>) |
| • | jQuery Cycle (<http://go.microsoft.com/fwlink/?LinkID=288978&clcid=0x413>) |
| • | jQuery DataTables (<http://go.microsoft.com/fwlink/?LinkID=288979&clcid=0x409>) |
| • | Prototype (<http://go.microsoft.com/fwlink/?LinkID=299651&clcid=0x409>) |
| • | MooTools (<http://go.microsoft.com/fwlink/?LinkID=299652&clcid=0x409>) |

You can use JavaScript libraries to make your application more interactive. The functioning of JavaScript codes depends on the version of the library you use. Not all code may work with all versions of a library.

The jQuery library (and its related libraries) has the additional advantage of dealing with the differences in the DOM across different browsers and different browser versions.

**Question**: What is the advantage of using JavaScript libraries?

## ****Using Content Delivery Networks for JavaScript Libraries****

A content delivery network (CDN) is a group of geographically distributed servers used for hosting contents for web applications. In many cases, you can bring web content geographically closer to your applications users by using a CDN to host libraries. This will also improve the scalability and robustness of the delivery of that content.

The amount of content stored in a CDN varies among different web applications. Some applications store all their content on a CDN, while other applications store only some of their content.

Microsoft has a dedicated CDN called Microsoft Ajax CDN that hosts some popular JavaScript libraries, such as:

|  |  |
| --- | --- |
| • | jQuery |
| • | jQuery UI |
| • | jQuery Mobile |
| • | jQuery Validation |
| • | jQuery Cycle |
| • | jQuery DataTables |
| • | Ajax Control Toolkit |
| • | ASP.NET Ajax |
| • | ASP.NET MVC JavaScript Files |

Note that Microsoft does not own the license of the JavaScript libraries mentioned in the preceding list. Microsoft only hosts the libraries for developers.

You can often reduce the loading time of your web application, by using the JavaScript libraries hosted on Microsoft Ajax CDN. Web browsers can cache these JavaScript libraries on a local system.

**Linking to JavaScript Libraries**

<script src="http://ajax.aspnetcdn.com/ajax/jquery/jquery-3.2.1.js" type="text/javascript"></script>

**Additional Reading:**For more information about Microsoft AJAX CDN, go to <http://go.microsoft.com/fwlink/?LinkID=293689&clcid=0x409>

**Question**: How can CDN help improve the performance of a web application?

## ****Using the NuGet Tool to Add Packages****

You can use the NuGet package manager to manage JavaScript libraries. You can avoid adding JavaScript libraries manually to your web application by installing the NuGet package manager in your application. This practice helps reduce the need for configuration tasks, while adding JavaScript libraries to an application.

Microsoft Visual Studio 2017 supports installing and using NuGet packages. You can search for NuGet packages in the NuGet store of Microsoft Visual Studio 2017. Then, you can directly install them into your MVC application.

The following image shows the application page that you can use to manage NuGet packages.

**FIGURE 10.2: MANAGING NUGET PACKAGES**

After you select the NuGet package that you wish to install, click **Install**, to download and install the package into your project.

**Additional Reading:**To search for NuGet packages and analyze the details of each package, go to <http://go.microsoft.com/fwlink/?LinkID=288981&clcid=0x410>

**Question**: Why should you use NuGet packages to add JavaScript libraries to your web application?

## ****Demonstration: How to Use NuGet to Add a JavaScript Library****

In this demonstration, you will see how to:

|  |  |
| --- | --- |
| • | Add the jQueryUI library to an application by using NuGet Package Manager. |
| • | Access the location where jQueryUI components are added in the MVC application. |
| • | Link to a script file in the site template view. |

### ****Demonstration Steps****

|  |  |
| --- | --- |
| • | You will find the steps in the “Lesson 1: Rendering and Executing JavaScript Code“ section on the following page: <https://github.com/MicrosoftLearning/20486-DevelopingASPNETMVCWebApplications/blob/master/Instructions/20486C/20486C_MOD10_DEMO.md>. |

# Lesson 2: ****Using jQuery and jQueryUI****

jQuery is a JavaScript library that simplifies the adding of JavaScript to HTML pages. jQuery is an open-source software that you can use for free. It helps reduce the amount of code that you need to write, to perform tasks such as accessing and modifying HTML elements on a webpage. You can use the **ajax** function in jQuery to call a web service in your application. You can also use jQuery UI to add interactions, animations, effects, and widgets to your web applications.

## ****Lesson Objectives****

After completing this lesson, you will be able to:

|  |  |
| --- | --- |
| • | Describe jQuery. |
| • | Describe how to link web applications to jQuery Libraries for client-side scripting. |
| • | Describe how jQuery helps access the HTML elements of a webpage. |
| • | Describe how to modify elements by using jQuery. |
| • | Describe how to call a web service by using jQuery. |
| • | Describe jQuery UI. |
| • | Add a jQuery UI widget to an MVC 5 web application. |

## ****Introduction to jQuery****

jQuery is a JavaScript library that you can use with different browsers. jQuery was first released in 2006. jQuery helps query the HTML Document Object Model (DOM) and obtain a set of HTML DOM elements. This feature of jQuery helps:

|  |  |
| --- | --- |
| • | Reduce the amount of code that you need to write, to perform a task. |
| • | Reduce the development time of HTML applications. |

jQuery includes the following features:

|  |  |
| --- | --- |
| • | DOM element selections |
| • | DOM traversal and modification |
| • | DOM manipulation, based on CSS selectors |
| • | Events |
| • | Effects and animations |
| • | AJAX |
| • | Extensibility through plug-ins |
| • | Utilities |
| • | Compatibility methods |
| • | Multi-browser support |

**Additional Reading:**For more information about jQuery, go to: <http://go.microsoft.com/fwlink/?LinkID=288982&clcid=0x411>

The jQuery family includes the following two companion modules:

|  |  |
| --- | --- |
| • | jQuery UI. This library adds functions and other supporting elements that help implement a rich interface to HTML-based applications. |
| • | jQuery mobile. This library adds functions that optimize the application interface for mobile devices. |

**Question**: Why should you use jQuery while developing web applications?

## ****Linking to jQuery Libraries****

jQuery Original Version and jQuery Minified Version provide similar functionalities; however, they optimize web applications for different purposes:

|  |  |
| --- | --- |
| • | jQuery Original Version (jQuery-<version>.js). This is the uncompressed version of the jQuery library. |
| • | jQuery Minified Version (jQuery-<version>.min.js). This includes the compressed and gZip versions of jQuery. |

When you develop the production environment, you can use jQuery Minified Version to reduce the loading time of the web application. If you use the minified version while working on the development environment, you cannot access the source code of the JavaScript libraries during the debug operation. Therefore, you can use the original version of jQuery, while creating the development environment.

**Referencing jQuery**

<script src="http://ajax.aspnetcdn.com/ajax/jquery/jquery-3.2.1.min.js" type="text/javascript"></script>

**Bundling and Minification**

Bundling is a feature in ASP.NET that you can use in MVC 5 web applications. Bundling helps combine multiple JavaScript libraries into a single HTTP request.

Minification compresses code files before incorporating them in the client application. Bundling and minification help reduce the loading time of web applications by reducing both the number and size of HTTP requests.

To use bundling and minification in your web application, you should perform the following steps:

|  |  |
| --- | --- |
| 1. | Reference the **Microsoft.AspNet.Web.Optimization** library in your application by using NuGet packages. |
| 2. | In the App\_Start folder of your project, add the BundleConfig.cs file. |
| 3. | In the BundleConfig.cs file, add the following code.  public static void RegisterBundles(System.Web.Optimization.BundleCollection bundles)  {  bundles.Add(new System.Web.Optimization.ScriptBundle("~/bundles/jquery").Include(  "~/Scripts/jquery-{version}.js","~/Scripts/JavaScript1.js"));  } |
| 4. | In the **Global.asax Application\_Start** event, add the following code.  BundleConfig.RegisterBundles(System.Web.Optimization.BundleTable.Bundles); |

The **ScriptBundle** class enables you to define the bundle that helps combine multiple JavaScript libraries into a single file. You can use a special placeholder, such as **{version}**in the JavaScript path, to help update the version of jQuery libraries. ASP.NET replaces **{version}** with the latest version number of the JavaScript libraries present in the Scripts folder. If the minified version of jQuery is available in the Scripts folder, the MVC 5 engine selects the minified version for bundling. Then, in your View file, you can include the following lines of code to render the bundled JavaScript file.

<head>

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

</head>

You should add the following line of code in a page or namespace of the Web.config file in the View folder. This code helps trigger the functioning of **@Scripts.Render**.

<add namespace="System.Web.Optimization"/>

Based on the compilation setting in the Web.config file, the jQuery library renders minified versions of Javascript. For example, the following code sets the **debug** attribute of the **compilation** element to **false**, which allows jQuery to use the non-minified version of the libraries, making debugging of the libraries easier.

<compilation debug="false" />

**Question**: What are the benefits of using the minified version of jQuery in the production environment?

## ****Accessing HTML Elements by Using jQuery****

jQuery helps access HTML elements, to create interactive web applications. You use the following selector to select elements by element name, id, or CSS class, while adding jQuery code to access HTML elements.

$(element name|#id|.class)

The following jQuery selector accesses the HTML element with the **HelloButton** ID.

$(“#HelloButton”)

You can use jQuery to access or modify all instances of a specific HTML element, in an HTML page. The following jQuery selector identifies all instances of the **A** element in an HTML page.

$(“a”)

After accessing the HTML elements, you can perform actions on the elements, such as:

|  |  |
| --- | --- |
| • | Modifying the attributes on the HTML elements. |
| • | Defining event handlers to respond to events associated with the selected HTML elements. |

**Using a jQuery Event Handler**

$("#HelloButton").click(function (event) {

alert("Hello World");

});

If the jQuery scripts load before the webpage loads, you may encounter errors such as **object not defined**. You can place the jQuery code in the **document.ready** event, to prevent the code from loading until all HTML elements in the page load.

**Using the Document Ready Function**

$(document).ready(function () {

//Code placed here will not execute before the page is fully loaded.

});

**A Complete Example**

<body>

<div>

Hello

<input type="button" value="Hello" id="HelloButton" />

</div>

<script type="text/javascript">

$(document).ready(function () {

$("#HelloButton").click(function (event) {

alert("Hello World");

});

});

</script>

</body>

**Question**: Why should you include the jQuery code in the **document.ready**event?

## ****Modifying HTML Elements by Using jQuery****

You can use jQuery to query HTML DOM and obtain HTML elements. You can use jQuery functions to modify the attributes associated with the HTML elements. The following are some commonly used jQuery functions, which enable you to modify HTML elements:

**The val Function**

$('#HelloButton').val('Hello World');

**The css Function**

$('#HelloButton').css('background-color','blue);

**The addclass Function**

$('#HelloButton').addClass('input\_css\_class');

**Additional Reading:**For more information about jQuery functions, go to <http://go.microsoft.com/fwlink/?LinkID=288983&clcid=0x412>

**Question**: If querying HTML DOM returns multiple HTML elements, how will jQuery functions handle these elements?

## ****Calling a Web Service by Using jQuery****

jQuery includes the **ajax** function that helps:

|  |  |
| --- | --- |
| • | Perform asynchronous calls to web services. |
| • | Retrieve the data returned from web services. |

**Using the ajax Function**

var req= $.ajax({

type: "POST",

dataType: "json",

url: "Customer.asmx/GetCustomerInfo",

data: "{'ID': '123'}",

contentType: "application/json; charset=utf-8",

success: function (msg) {

alert("Data Saved: " + msg);

},

failure: function (msg) {

alert(msg);

}

});

The **ajax** function uses the parameters **type**, **datatype**, **url**, **data**, **contentType**, **success**, and **failure** to control how to call the web services. These parameters are described as follows:

|  |  |
| --- | --- |
| • | type. This parameter controls the request type that you should use while querying the web services. |
| • | dataype. This parameter defines the data type to be sent for AJAX services. |
| • | url. This parameter provides the URL of the web services. |
| • | data. This parameter defines the data that you should provide as a parameter to the web services. |
| • | contentType. This parameter defines the HTTP content type that you should use, when you submit HTTP requests to web services. |
| • | success. This parameter defines the name of the function, which will be triggered when the call completes successfully. |
| • | failure. This parameter defines the name of the function, which will be triggered when the call completes with errors. |

When calls to the web services complete, jQuery triggers one of two callback functions based on the success of the call.

**Additional Reading:**For more information about the AJAX function, go to <http://go.microsoft.com/fwlink/?LinkID=288984&clcid=0x413>

**Question**: Why should you call web services by using jQuery?

## ****Introduction to jQueryUI****

jQuery simplifies interacting with JavaScript elements, by providing a simple query-based syntax. jQuery UI is a library that includes widgets, animations, and themes that help you to build a rich user interface.

**jQuery Widgets**

You can add different types of widgets to your pages by using the following jQuery functions:

|  |  |
| --- | --- |
| • | Accordion. This function adds accordion containers to your webpage. |
| • | Autocomplete. This function adds auto-complete boxes that are based on user input. |
| • | Button. This function adds buttons to your webpage. |
| • | Datepicker. This function adds date-pickers to your webpage. |
| • | Dialog. This function adds dialog boxes to your webpage. |
| • | Menu. This function adds a menu to your webpage. |
| • | Progressbar. This function adds animated and static progress bars to your webpage. |
| • | Slider. This function adds sliders to your webpage. |
| • | Spinner. This function adds a number spinner to a data entry box. |
| • | Tabs. This function adds tabs to your webpage. |
| • | Tooltip. This function displays a tooltip for interface elements. |

**jQuery Effects**

You can add various effects by using the following jQuery functions:

|  |  |
| --- | --- |
| • | Color Animation. This function animates colors. |
| • | Toggle Class, Add Class, Remove Class, and Switch Class. This function adds or removes CSS classes. |
| • | Effect. This function adds a variety of effects, such as appear, slide-down, explode, and fade-in. |
| • | Toggle. This function toggles any jQuery effect on and off. |
| • | Hide and Show. This function displays or hides any jQuery effect. |

**jQuery Utility**

You can align your webpage content by using the **Position** jQuery utility. This utility helps align an element in relation to the position of another element.

To use jQuery UI:

|  |  |
| --- | --- |
| 1. | Download the script and supporting files from <http://go.microsoft.com/fwlink/?LinkID=288985&clcid=0x414> |
| 2. | Reference the JavaScript libraries in the HTML files. |

**Linking to jQuery UI**

<script src="http://ajax.aspnetcdn.com/ajax/jquery.ui/1.12.1/jquery-ui.min.js"

type="text/javascript"></script>

**Question**: What is the key difference between jQuery and jQuery UI?

## ****Demonstration: How to Add a jQueryUI Widget****

In this demonstration, you will see how to create a set of expandable sections on a webpage by using the Accordion widget.

### ****Demonstration Steps****

|  |  |
| --- | --- |
| • | You will find the steps in the “Lesson 2: Using jQuery and jQueryUI“ section on the following page: <https://github.com/MicrosoftLearning/20486-DevelopingASPNETMVCWebApplications/blob/master/Instructions/20486C/20486C_MOD10_DEMO.md>. |

# ****Lab: Using JavaScript and jQuery for Responsive MVC 5 Web Applications****

### ****Scenario****

You have been asked to add a slideshow page to the web application that will show all the photos in the database. Unlike the **All Photos** gallery, which shows thumbnail images, the slideshow will display each photo in a large size. However, the slideshow will display only one photo at a time, and cycle through all the photos in the order of ID.

You want to use jQuery to create this slideshow because you want to cycle through the photos in the browser, without reloading the page each time. You also want to animate slide transitions and show a progress bar that illustrates the position of the current photo in the complete list. You will use jQueryUI to generate the progress bar.

Begin by importing a partially complete view that will display all photos simultaneously in the correct format. Then, change styles and add jQuery code to the application to create your slideshow.

### ****Objectives****

After completing this lab, you will be able to:

|  |  |
| --- | --- |
| • | Render and execute JavaScript code in the browser. |
| • | Use the jQuery script library to update and animate page components. |
| • | Use jQueryUI widgets in an MVC application. |

##### ****Lab Setup****

Estimated Time: 50 minutes

You will find the high-level steps on the following page: <https://github.com/MicrosoftLearning/20486-DevelopingASPNETMVCWebApplications/blob/master/Instructions/20486C/20486C_MOD10_LAB_MANUAL.md>.

You will find the detailed steps on the following page: <https://github.com/MicrosoftLearning/20486-DevelopingASPNETMVCWebApplications/blob/master/Instructions/20486C/20486C_MOD10_LAK.md>.

### ****Exercise 1: Creating and Animating the Slideshow View****

##### ****Scenario****

Your team has created a view that displays photos of the right size and format. However, the view displays all photos simultaneously, one below the other.

In this exercise, you will:

|  |  |
| --- | --- |
| • | Import the view and modify the style sheet so that the photos are displayed on top of each other. |
| • | Using jQuery, set the order for each photo so that each photo is displayed sequentially. |

### ****Exercise 2: Optional—Adding a jQueryUI ProgressBar Widget****

##### ****Scenario****

The slideshow pages you added work well. Now, you have been asked to add some indication of progress through the slideshow. You want to use a progress bar to show the position of the current photo in the list of photos in the application.

In this exercise, you will:

|  |  |
| --- | --- |
| • | Create a display by using the JQueryUI progress bar. |
| • | Test the script that you created. |

Complete this exercise if time permits.

### ****Review Question(s)****

**Check Your Knowledge**

**Discovery**

**What is the use of adding the two links to the \_MainLayout.cshtml file in Task 1 of Exercise 2?**

Show solution Reset

**Check Your Knowledge**

**Discovery**

**You added <script> tags to the \_MainTemplate.cshtml file to enable jQueryUI. Is this the optimal location for this link?**

Show solution Reset

# ****Module Review and Takeaways****

You have seen how to use JavaScript in a web application. JavaScript helps the application interact with the actions of users and provide response to users, without reloading an entire webpage. You also saw how to use the jQuery library to access the HTML DOM structure and modify HTML elements. jQueryUI is a complement to the jQuery library, which contains functions to build rich user interfaces.

### ****Review Question(s)****