Using jQuery to retrieve information from other websites

Objectives

- JQuery to simplify interactions
- Add interactivity
- Add content from another website

Introduction to JQuery

In the last module we spent time going over JavaScript to add interactivity to your web page. We looked at validating input from a user as well as adding HTML content using code.

In this module we want to introduce a library for JavaScript called JQuery. JQuery is a set of JavaScript functions code that allow you to interact with your content in a very simple way. First let's add JQuery to our web page, and then we will look at what it can do.

Open up your **index.html** file and add the "jquery-1.10.2.js" line to the <head> portion of your web page.

```
<html>

<head>
    <title>My First Web Page</title>
    <link href="app.css" rel="stylesheet" />
    ...
    <script src="https://code.jquery.com/jquery-1.10.2.js" type="application/javasc
    <script src="app.js" type="application/javascript"></script>
</head>
```

Now that you have a reference to jQuery, you can begin using it to simplify your JavaScript.

Simplify your code

Remember in the last module we added content to our page when the number was invalid. As a reminder, here is the code.

```
function checkNumber() {
    var theNumber, theMessage;

    // Get the value of the input field with id="numb"
    theNumber = document.getElementById("smallnumber").value;
```

```
    // If x is Not a Number or less than one or greater than 10
    if (isNaN(theNumber) || theNumber < 1 || theNumber > 10) {
        theMessage = "Number was expected to be between 1 and 10";
    } else {
        theMessage = "Number is Good";
    }
    document.getElementById("numberMessage").innerHTML = theMessage;

}
```

With JQuery, we can simplify the lines that use `document.getElementById('id')` to use the jQuery function `$('id')`. The function does things a little differently, but here's how the new code could look with using jQuery.

```
function checkNumber() {
    var theNumber, theMessage;

    // Get the value of the input field with id="numb"
    theNumber = $('#smallnumber').val();

    // If x is Not a Number or less than one or greater than 10
    if (isNaN(theNumber) || theNumber < 1 || theNumber > 10) {
        theMessage = "Number was expected to be between 1 and 10";
    } else {
        theMessage = "Number is Good";
    }
    $('#numberMessage').text(theMessage);
}
```

Notice we have something new in our code...the '$'. This actually represents all of the code that was loaded by the jQuery library. Next, inside the parenthesis, we use the same approach that we use in CSS for selecting an HTML Element. '#' for an ID, '.' for a class or just that tag name for a tag. In other words, `$(#smallnumber)` means "select the element with an ID of 'smallnumber'".

Now, don't be confused but it is **NOT** the same thing as `document.getElementById("smallnumber")`. Notice, that you have to use the jQuery `val()` function and not the javascript `value` function. If you are going to use jQuery to modify the document, then you should *always* use jQuery. Don't switch back and forth...it will get confusing for you (and potentially other developers you are working with).

Getting content from another website

Great, now that we have a handle on how jQuery is going to help us, we are going to use a very useful function for calling other websites. There is lots of great information out on the internet that we can take advantage of. For example, weather is one of those things that there are sensors all over the world that we can ask for the current temperature.

Programmers expose this information through an API or an "Application Programming Interface". Sometimes, you will get this information for free, sometimes you will have to sign up and pay a fee. Everything from weather, to mapping, sending text messages to a phone, to stock market prices...you name it, there is probably an API for you to take advantage of.

In this section, we have a simple API to call to get an image from the web.

First, we will need to create a placeholder for the image. Let's go back to our final Bootstrap row and add the content to our page. We'll take the center six grid section `<div class="col-sm-6">` and change it to be five columns and add a one column div. Next, we'll add a paragraph "<p>" tag to that and give it an id of "badge".

```
    <div class="container-fluid">
        <div class="row">
            <div class="col-sm-3"></div>
            <div class="col-sm-5">
                <a name="contact">Contact Us</a>
                <p>Anytown Chamber of Commerce</p>
                <p>123 Main St.</p>
                <p>Anytown, OH 43210</p>
            </div>
            <div class="col-sm-1">
                <p id="badge"></p>
            </div>
        </div>
    </div>
```

Now we need to add some JavaScript to our app.js file. In this script, we are defining a function "getAPIBadge" that points to an API running in Microsoft Azure. We are going to use jQuery to "post" data to the API. Copy this code and modify the SchoolName, ZipCode and Level ("Beginner" or "Experienced").

When this function is complete, the ".done" function will run and jQuery will replace the contents on the paragraph tags with the HTML that comes back from the API.

```
function getAPIBadge() {
    var ctcAPI = "http://ChooseToCodeAPI.azurewebsites.net/api/values/";
```

```
    $.post( ctcAPI, {
        SchoolName:"YourSchoolNameHere",
        ZipCode: "YourZipCodeHere",
        Level:"Beginner"
    }).done(function( data ) {
        $("#badge").html(data);
    });
}
```

Finally, modify the onLoad method to call the "getAPIBadge" method once the document has completed loading.

```
function onLoad() {
    document.getElementById("timestamp").innerHTML = Date();
    getAPIBadge();
}
```

Congratulations

You have completed this module and called an API on another web page to pull data into your website. There are many other APIs that you can investigate and take advantage of. Almost all common APIs have sample projects to help you get started with integration into your own website. Here are a few favorites to look into.

API Data Type	URL
Bing Maps	http://www.microsoft.com/maps/choose-your-bing-maps-API.aspx
Twitter	https://dev.twitter.com/rest/public
Weather.Com	http://openweathermap.org/API