JavaScript

for Developers

55244AC / MAX003AC

-- Instructor Notes --

AUTHOR

Michael T. Smith  
MCT, MVP SharePoint  
Senior Instructor, Max Technical Training

COPYRIGHT

© 2017 MAX Technical Training, Inc. All Rights Reserved. This manual and any training materials supplied with it are copyrighted with all rights reserved. No part of this publication may be reproduced, transmitted, transcribed, stored in a retrieval system or translated into any language in any form by any means without the written permission of MAX Technical Training, Inc.

TRADEMARKS

Brand names, company names and product names used herein are trademarks or registered trademarks of their respective companies.

MAX Technical Training, Inc.   
4900 Parkway Drive, Suite 160   
Mason, OH 45040  
513.322.8888

CWM: 55244AC / MAX003AC

08/29/17

Table of Contents

[Additional Support 1](#_Toc491803965)

[Class Length and Delivery Options 2](#_Toc491803966)

[Class Length 2](#_Toc491803967)

[Instructor Skills Needed 3](#_Toc491803968)

[Target Audience 3](#_Toc491803969)

[Lab Setup 4](#_Toc491803970)

[Local or Hosted Virtual Machines 4](#_Toc491803971)

[Lab Web Service 5](#_Toc491803972)

[Files Supplied with the Class 7](#_Toc491803973)

[Appendix 1: Editors and Tools 8](#_Toc491803974)

[Visual Studio Code 8](#_Toc491803975)

# Additional Support

For additional support for this class visit the MAX Technical Training courseware support site:

http://courseware.maxtrain.com

There you will find:

* Updates to the class.
* The lab files.

Or, email [courseware@maxtrain.com](mailto:courseware@maxtrain.com) with any questions or suggestions about the course or the setup.

# Class Length and Delivery Options

### Class Length

This class can be delivered in two to five days.

**Target: 2 days**

A basic introduction to JavaScript and jQuery by covering only modules 1-3.

**Target: 3 days**

A basic introduction to JavaScript, jQuery and AJAX by covering only modules 1-4.

**Target: 5 days**

An in depth class on JavaScript, jQuery and AJAX, including more advanced topics such as anonymous functions, closures, self-invoking functions and recursion along with a detailed study of data types and objects.

**Opportunity for Expansion**

You may want to include additional content to suit the needs of your students.

* ECMAScript 2016: This class is designed to cover JavaScript as it is commonly supported mid-2017. Many of the new features defined ECMAScript 2016 are not covered here due to incomplete browser support. You may want to add a “what’s coming soon” demo and lecture. Check our website for updates…
* ECMAScript Classes.
* Mobile Development.
* Additional libraries and tools such as Prototype, React, Bootstrap, Angular or Underscore.js.

We will be adding a “B” version and/or “optional modules” to cover many of these in the future.

**Time Management**

Depending on your knowledge and real world use of JavaScript, a class like this one may be hard to stretch to four days, and can extend to fill two weeks! You will need to manage your time carefully to deliver a well-balanced class in the scheduled number of days. Many JavaScript features are listed or mentioned for completeness, and can be skimmed over in less than five day classes.

### Instructor Skills Needed

I can’t stress enough that the instructor needs very strong JavaScript and HTML skills. This type of class can often feel like a “stump the instructor” session due to the range of JavaScript language features along with HTML details. Time management and “student management” skills are very important.

We encourage you to contact MAX Technical Training and attend one of our live classes remotely.

Skills needed or useful:

* JavaScript, of course!
* HTML 5 and CSS 3, and some knowledge of their history and browser support.
* Knowledge of UI and accessibility best practices.
* Experience with the editor(s) and browsers used in the class.
* Back end development.
* Web services including both SOAP and REST.

### Target Audience

This class was created for:

* The new developer learning JavaScript and HTML.
* The new to JavaScript developer with some experience in other languages.

# Lab Setup

### Local or Hosted Virtual Machines

This class can be delivered on various platforms including virtual machines, Microsoft Azure, Amazon AWS and third party hosting services such as Learn on Demand. This is just about the easiest class environment you ever need to build!

For each student:

* A PC or virtual machine running Windows 8, 8.1 or 10. Use Windows 10 if you want to use the Edge browser in class.
* A collection of web browsers:
  + Internet Explorer 11
  + Microsoft Edge (if running on Windows 10)
    - If you would like to demo Edge, but are using Windows 7, 8 or 8.1 lab machines, you can download prebuilt Windows 10 90-day evaluation virtual machines here: <https://developer.microsoft.com/en-us/microsoft-edge/tools/vms/>
  + Latest version of Firefox
  + Latest version of Chrome
* One or more HTML / JavaScript editors. The choices will depend on your audience and other training you may be combining with this course.
  + Microsoft Visual Code – Free and light weight (see Appendix 1)
  + Microsoft Visual Studio Professional
  + Microsoft Visual Studio Community - Free
  + Aptana Studio
  + Adobe Dreamweaver
* A place to host a Lab Web Service project. (see below)

For the instructor:

* The same setup as for the student, but ideally with network access so students can open your demos.
* Or, a cloud based virtual machine configured to allow student access using classroom machines, phones or tablets.

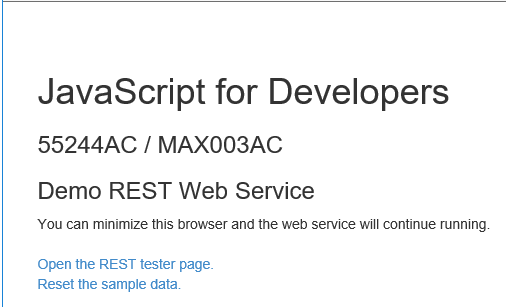
### Lab Web Service

The AJAX module labs use a sample Visual Studio project to provide a demo REST web service.

This project can be hosted:

* On the student’s lab machine / VM.
  + Ask the students to open Visual Studio, open the project and then click the Run button. They can then minimize Visual Studio and still use any editor for their web projects.
* On a cloud based service such as Azure or Amazon, publishing the Visual Studio project to a free Azure Web App. (All students accessing the same web service.)
* A server on the classroom network or the instructor’s PC. (All students accessing the same web service.)

**If deploying to the student machines or a shared classroom server:**

1. Confirm that Visual Studio Pro or Community 2015 or later is installed.
2. Confirm that the machine has internet access. (Needed to update the NuGet packages.)
3. Create a folder named JavaScript (your choice of names) in the root of drive C.
4. Copy the 55244AC MAX003AC REST Demo.zip to the new folder.
5. Double-click the file and extract the files.
6. Open the RestDemo folder and double-click RestDemo.sln. (This will launch Visual Studio and open the project.)
7. Click the green run button, or press F5.
   1. The NuGet packages should download.
   2. The project should build.
   3. The home page should open:  
      

**If deploying to the Azure:**

You will need an Azure subscription to continue.

The following steps can be run from any PC with Visual Studio installed.

1. Confirm that Visual Studio Pro or Community 2015 or later is installed.
2. Confirm that the machine has internet access. (Needed to update the NuGet packages and to publish to Azure.)
3. Create a folder named JavaScript (your choice of names) in the root of drive C.
4. Copy the 55244AC MAX003AC REST Demo.zip to the new folder.
5. Double-click the file and extract the files.
6. Open the RestDemo folder and double-click RestDemo.sln. (This will launch Visual Studio and open the project.)
7. Click the green run button, or press F5.
   1. The NuGet packages should download.
   2. The project should build.

**Create the Web App in Azure**

1. Logon to Azure. (https://portal.azure.com)
2. Click the “+ New” link.
3. Click “Web + Mobile”.
4. Click “Web App”.
5. Click the Create button.
6. Enter an app name. Must be unique in the azurewebsites.net domain. (Example: “yourcompanyRestDemo”)
7. Click “App Service plan/Location” and click Create New.
8. Give the plan a name. Must be unique within your subscription. Example: “RestDemo”
9. Select a location close to you.
10. Click “Pricing Tier” and set the pricing tier to “F1 Free”.
11. Click Select, OK and Create.
12. Wait for the Web App to be created…
13. Navigate to your new Web App. (Easiest way to get there is to click the “Deployments succeeded” item in the notifications dropdown menu (bell).)
14. Click the Overview tab if not selected.
15. Click “Get publish profile” at the top of the page.
16. Download the file to your local PC to the directory where you unzipped the project (C:\JavaScript\RestDemo) or other location of your choice.

**Deploy the Project**

1. Return to Visual Studio.
2. Right-click the project, “RestDemo” (not the solution), and click Publish.
3. Click the Profile tab.
4. Click the Import button and navigate to your Publish Profile file and open it.
5. Click Publish.
6. When the publish is complete, a web browser will open with the site’s home page.
7. You can now close the browser and Visual Studio.

**Test the Site**

1. Open a browser on any PC/Device and navigate to your test site. (Something like http://yourcompanyRestDemo.azurewebsites.net)
2. The home page should open.
3. Enter the URL to the rest services and confirm that you either see JSON data or get the option to download a file. (Bikes.json)  
    http://yourcompanyRestDemo.azurewebsites.net/odata/Bikes  
   (Note: “odata/Bikes” is case sensitive!)

Tip: If you get this error: “Could not find a part of the path 'D:\home\site\wwwroot\App\_Data”, return to Visual Studio, right-click the App\_Data folder and click “Publish App\_Data”.

### Files Supplied with the Class

* The sample REST web service for module 4
  + 55244AC MAX003AC REST Demo.zip
* The class lab files
  + 55244AC MAX003AC JavaScript labs.zip
* This document
  + 55244AC MAX003AC JavaScript for Developers Instructor Notes.docx

# Appendix 1: Editors and Tools

### Visual Studio Code

Visual Studio Code is a free editor from Microsoft for Windows, macOS and Linux. While it has built-in support for JavaScript, TypeScript and Node.js, its real power comes from a large library of extensions.

Features:

* Multiple editors, both built-in and via extensions, including:
  + JavaScript, JSON, CSS, Sass, Less, TypeScript, Markdown, PHP, PowerShell, Python, Go, Dockerfile, T-SQL, C# and C++
* Coloring, auto indenting
* IntelliSense and autocomplete
* Snippets
* GIT support
* Debugging
* Integrated terminal
  + $SHELL on Linux and OS X, PowerShell on Window 10 and cmd.exe on earlier versions of Windows
* Many extensions



Links:

* Visual Studio Code: https://code.visualstudio.com/
* Download: https://code.visualstudio.com/Download
* Available Extensions: https://marketplace.visualstudio.com/VSCode

Getting Started with Visual Studio Code:

* JavaScript: https://code.visualstudio.com/Docs/languages/javascript
* HTML: https://code.visualstudio.com/docs/languages/html
* CSS: https://code.visualstudio.com/docs/languages/css
* GIT: https://code.visualstudio.com/docs/editor/versioncontrol

To add extensions:

1. Open the editor, click the View menu and then Extensions.
   1. A list of currently listed extensions will be displayed.
2. Search for an extension.
3. Click Install. (A few extensions may require a restart of VSCode.)