

# Appendix A:

# CIW JavaScript Specialist

# Objectives and Locations

The CIW *JavaScript Specialist* courseware and this appendix are designed to help students prepare for the CIW JavaScript Specialist exam (1D0-735). Students can use this appendix as a study guide to locate content within the *JavaScript Specialist* course that corresponds to the specific skills objectives of the CIW JavaScript Specialist program. You can register for the CIW JavaScript Specialist exam at any VUE ([www.vue.com](http://www.vue.com)) testing center. If your learning institution is a CIW Certification Testing Center (CTC) participant, then your instructor may be able to deliver CIW certification exams directly in your classroom. For more information about the CIW JavaScript Specialist certification or any other CIW certification, visit [www.CIWcertified.com](http://www.CIWcertified.com).

JavaScript Specialist Objective	JavaScript Specialist Courseware Lesson(s) and Section(s)
<b>Domain 1: Essential JavaScript Principles and Practices</b>	
<b>1.1: Identify characteristics of JavaScript and common programming practices.</b>	
1.1.1 List key JavaScript characteristics, including object-based nature, events, platform-independence, and differences between scripting languages and programming languages.	<b>Lesson 1: Introduction to JavaScript</b> <ul style="list-style-type: none"> <li>- JavaScript Characteristics</li> <li>- JavaScript Flavors and Versions</li> </ul>
1.1.2 Identify common programming concepts, including objects, properties and methods.	<b>Lesson 1: Introduction to JavaScript</b> <ul style="list-style-type: none"> <li>- JavaScript and Common Programming Concepts</li> </ul>
1.1.3 Describe various JavaScript versions and flavors, including ECMA standards, JScript and similarities with proprietary scripting languages.	<b>Lesson 1: Introduction to JavaScript</b> <ul style="list-style-type: none"> <li>- JavaScript Flavors and Versions</li> <li>- Technologies based on JavaScript</li> </ul>
1.1.4 Distinguish between server-side and client-side JavaScript applications, including JavaScript interpreters and rendering engines.	<b>Lesson 1: Introduction to JavaScript</b> <ul style="list-style-type: none"> <li>- Server-Side vs. Client-Side Applications</li> </ul>
1.1.5 Describe acceptable coding practices, including appropriate use of comment tags and the <code>&lt;noscript&gt;</code> tag.	<b>Lesson 1: Introduction to JavaScript</b> <ul style="list-style-type: none"> <li>- Server-Side vs. Client-Side Applications</li> <li>- Annotating Your Code with Comments</li> </ul>
1.1.6 Examine the evolution of the role of JavaScript in simple Web page design, such as gaming, virtual reality (VR), native development, mobile device scripting and backend development.	<b>Lesson 1: Introduction to JavaScript</b> <ul style="list-style-type: none"> <li>- Expanding the Role of JavaScript</li> <li>- Lab 1-3: Research gaming and VR industries job requirements</li> </ul>

JavaScript Specialist Objective	JavaScript Specialist Courseware Lesson(s) and Section(s)
<b>1.2: Work with variables and data in JavaScript.</b>	
1.2.1 Use attributes and methods to communicate with users, including the <i>type</i> attribute.	<b>Lesson 2: Working with Variables and Data in JavaScript</b> <ul style="list-style-type: none"> <li>- Using JavaScript to Communicate with the User</li> <li>- Lab 2-1: Using the JavaScript <i>document.write()</i> method</li> </ul>
1.2.2 Define variables, data types and scope.	<b>Lesson 2: Working with Variables and Data in JavaScript</b> <ul style="list-style-type: none"> <li>- Using Data More Than Once: Variables</li> </ul>
1.2.3 List keywords and reserved words.	<b>Lesson 2: Working with Variables and Data in JavaScript</b> <ul style="list-style-type: none"> <li>- JavaScript Keywords and Reserved Words</li> </ul>
1.2.4 Store user input in variables and evaluate for appropriate responses using the console and built-in methods such as <i>alert()</i> and <i>prompt()</i> .	<b>Lesson 2: Working with Variables and Data in JavaScript</b> <ul style="list-style-type: none"> <li>- Using Data More Than Once: Variables</li> <li>- Lab 2-2: Storing user data in a JavaScript variable</li> <li>- Operators</li> <li>- Lab 2-3: Demonstrate and compare operators</li> <li>- JavaScript Alerts and Prompts</li> <li>- Lab 2-4: Using the JavaScript <i>alert()</i> method</li> <li>- Lab 2-5: Using the JavaScript <i>prompt()</i> method</li> <li>- Lab 2-6: Using the JavaScript <i>confirm()</i> method</li> <li>- Lab 2-7: Assigning and adding variables in JavaScript</li> </ul>
1.2.5 Distinguish between concatenation and addition.	<b>Lesson 2: Working with Variables and Data in JavaScript</b> <ul style="list-style-type: none"> <li>- Using JavaScript to Communicate with the User</li> <li>- JavaScript Operators</li> </ul> <b>Lesson 4: Debugging and Troubleshooting JavaScript</b> <ul style="list-style-type: none"> <li>- Lab 4-3: Troubleshooting a logic error in JavaScript</li> </ul>

JavaScript Specialist Objective	JavaScript Specialist Courseware Lesson(s) and Section(s)
<p>1.2.6 Apply operators, including string concatenation ( += ), strict comparison ( === , !==), mathematical precedence and bitwise operators.</p>	<p><b>Lesson 2: Working with Variables and Data in JavaScript</b></p> <ul style="list-style-type: none"> <li>- Using JavaScript to Communicate with the User</li> <li>- Using Data More Than Once: Variables</li> <li>- Lab 2-2: Storing user data in a JavaScript variable</li> <li>-</li> <li>- JavaScript Operators</li> <li>- JavaScript Expressions</li> <li>- Lab 2-5: Storing user data in a JavaScript variable</li> <li>- Lab 2-6: Assigning and adding variables in JavaScript</li> </ul>
<p>1.2.7 Demonstrate how to use expressions.</p>	<p><b>Lesson 2: Working with Variables and Data in JavaScript</b></p> <ul style="list-style-type: none"> <li>- JavaScript Expressions</li> <li>- Lab 2-6: Assigning and adding variables in JavaScript</li> </ul>
<p>1.2.8 Implement simple event handlers, including keyboard, mouse and mobile (gestures or touch) events.</p>	<p><b>Lesson 2: Working with Variables and Data in JavaScript</b></p> <ul style="list-style-type: none"> <li>- Using JavaScript to Communicate with the User</li> <li>- Inline Scripting, Simple User Events, and Basic Event Handlers</li> <li>- Lab 2-8: Using the JavaScript <i>onLoad</i> event handler and inline scripting</li> </ul>
<p><b>1.3: Use JavaScript functions, methods, and events.</b></p>	
<p>1.3.1 Define and use methods as functions.</p>	<p><b>Lesson 3: Functions, Methods and Events in JavaScript</b></p> <ul style="list-style-type: none"> <li>- Introduction to Functions</li> <li>- Defining a Function</li> <li>- Lab 3-1: Creating a user-defined function in JavaScript</li> <li>- Calling a Function</li> <li>- Lab 3-2: Using functions, arguments and return values in JavaScript</li> <li>- Lab 3-3: Calling a function from within another function in JavaScript</li> <li>- Methods as Functions</li> </ul>
<p>1.3.2 Demonstrate the use of various types of functions and function elements including prototype functions, anonymous functions, closure functions, arguments and the use of return values.</p>	<p><b>Lesson 3: Functions, Methods and Events in JavaScript</b></p> <ul style="list-style-type: none"> <li>- Calling a Function</li> <li>- Lab 3-2: Using functions, arguments and return values in JavaScript</li> <li>- Lab 3-3: Calling a function from within another function in JavaScript</li> <li>- Types of Functions</li> <li>- Lab 3-5: Compare function types</li> </ul>

JavaScript Specialist Objective	JavaScript Specialist Courseware Lesson(s) and Section(s)
1.3.3 Distinguish between global and local variables.	<b>Lesson 3: Functions, Methods and Events in JavaScript</b> - Calling a Function
1.3.4 Use the conditional operator.	<b>Lesson 3: Functions, Methods and Events in JavaScript</b> - Calling a Function - Lab 3-3: Calling a function from within another function in JavaScript
1.3.5 Identify user events and event handlers.	<b>Lesson 3: Functions, Methods and Events in JavaScript</b> - User Events and JavaScript Event Handlers
1.3.6 Demonstrate the use of function specific methods including <i>calling</i> , <i>binding</i> and <i>applying</i> .	<b>Lesson 3: Functions, Methods and Events in JavaScript</b> - User Events and JavaScript Event Handlers - Lab 3-4: Using event handlers
1.3.7 Use built-in functions and cast variables.	<b>Lesson 3: Functions, Methods and Events in JavaScript</b> - Methods as Functions
<b>Domain 2: Intermediate JavaScript Programming Techniques</b>	
<b>2.1: Debug and troubleshoot JavaScript code.</b>	
2.1.1 Demonstrate common steps for debugging JavaScript code, including reviewing code and testing code in different browsers and various devices.	<b>Lesson 4: Debugging and Troubleshooting JavaScript</b> - What Is Debugging? - Lab 4-1: Debugging with the Chrome - Lab 4-2: Debugging with Apple Safari - Testing Code in Various Browsers - Debugging Logical Errors - Lab 4-3: Troubleshooting a logical error in JavaScript - JavaScript and Mobile Devices
2.1.2 Demonstrate how to use various native and supplemental debugging tools, including enabling/disabling display.	<b>Lesson 4: Debugging and Troubleshooting JavaScript</b> - Tools for Debugging Code - Lab 4-1: Debugging with the Chrome - Lab 4-2: Debugging with Apple Safari
<b>2.2: Use JavaScript statements to control program flow.</b>	
2.2.1 Use the <i>if...</i> statement.	<b>Lesson 5: Controlling Program Flow in JavaScript</b> - The <i>if...else</i> Statement - Lab 5-1: Using <i>if</i> statements

JavaScript Specialist Objective	JavaScript Specialist Courseware Lesson(s) and Section(s)
2.2.2 Use the <i>while...</i> statement.	<b>Lesson 5: Controlling Program Flow in JavaScript</b> - The <i>while</i> Statement - Lab 5-2: Using a <i>while</i> statement
2.2.3 Use the <i>do...while</i> statement.	<b>Lesson 5: Controlling Program Flow in JavaScript</b> - The <i>do...while</i> Statement
2.2.4 Use the <i>for...</i> statement.	<b>Lesson 5: Controlling Program Flow in JavaScript</b> - The <i>for</i> Statement - Lab 5-3: Using a <i>for</i> statement
2.2.5 Use the <i>forEach</i> statement.	<b>Lesson 5: Controlling Program Flow in JavaScript</b> - The <i>forEach</i> Statement - Lab 5-4: Using a <i>forEach</i> statement
2.2.6 Use the <i>break</i> and <i>continue</i> statements.	<b>Lesson 5: Controlling Program Flow in JavaScript</b> - The <i>break</i> Statement - Lab 5-5: Nesting <i>if</i> and <i>break</i> statements inside a <i>while</i> loop - The <i>continue</i> Statement - Lab 5-6: Using a <i>continue</i> statement
2.2.7 Use the <i>switch...</i> statement.	<b>Lesson 5: Controlling Program Flow in JavaScript</b> - The <i>switch</i> Statement - Lab 5-7: Using a <i>switch</i> statement
<b>2.3: Use the JavaScript Document Object Model (DOM).</b>	
2.3.1 Use JavaScript to manipulate the Document Object Model (DOM).	<b>Lesson 6: The JavaScript Document Object Model (DOM)</b> - The JavaScript Document Object Model (DOM)
2.3.2 Use the <i>window</i> object of the DOM.	<b>Lesson 6: The JavaScript Document Object Model (DOM)</b> - The <i>window</i> Object - Lab 6-1: Launching a new window with the <i>open()</i> method - Lab 6-2: Writing content to new windows - Lab 6-3: Changing status bar text
2.3.3 Manipulate properties and methods of the <i>document</i> object within the DOM.	<b>Lesson 6: The JavaScript Document Object Model (DOM)</b> - The <i>document</i> Object - Lab 6-4: Assigning properties to a remote document dynamically
2.3.4 Use the <i>image</i> object of the DOM, including image rollover creation.	<b>Lesson 6: The JavaScript Document Object Model (DOM)</b> - The <i>image</i> Object - Lab 6-5: Preloading and swapping images to create an active link

JavaScript Specialist Objective	JavaScript Specialist Courseware Lesson(s) and Section(s)
2.3.5 Use the <i>history</i> object of the DOM.	<b>Lesson 6: The JavaScript Document Object Model (DOM)</b> - The <i>history</i> Object - Lab 6-6: Traversing entries in the <i>history</i> Object
2.3.6 Evaluate and change URL information with the <i>location</i> object of the DOM.	<b>Lesson 6: The JavaScript Document Object Model (DOM)</b> - The <i>location</i> Object - The <i>navigator</i> Object - Lab 6-7: Detecting browsers and devices using JavaScript
2.3.7 Use the <i>navigator</i> object of the DOM.	<b>Lesson 6: The JavaScript Document Object Model (DOM)</b> - The <i>navigator</i> Object - Lab 6-8: Identifying browser properties with the <i>navigator</i> object
2.3.8 Describe virtual DOM.	<b>Lesson 6: The JavaScript Document Object Model (DOM)</b> - The Virtual DOM
<b>2.4: Use JavaScript language objects and create expressions.</b>	
2.4.1 Use the <i>String</i> object to test user input.	<b>Lesson 7: JavaScript Language Objects</b> - The <i>String</i> Object - Lab 7-1: Using <i>String</i> object formatting methods - Lab 7-2: Applying <i>String</i> methods to text
2.4.2 Evaluate strings, including use of the <i>length</i> property, and use of the <i>indexOf()</i> , <i>lastIndexOf()</i> , <i>substring()</i> and <i>charAt()</i> methods.	<b>Lesson 7: JavaScript Language Objects</b> - Evaluating Strings - Lab 7-2: Applying <i>String</i> methods to text
2.4.3 Implement basic regular expressions and the <i>RegExp</i> object.	<b>Lesson 7: JavaScript Language Objects</b> - JavaScript Regular Expressions - Lab 7-3: Implement basic regular expressions
2.4.4 Use the <i>Array</i> object to create more efficient code.	<b>Lesson 7: JavaScript Language Objects</b> - The <i>Array</i> Object - Lab 7.4: Creating an <i>Array</i> object
2.4.5 Use the <i>map</i> method.	<b>Lesson 7: JavaScript Language Objects</b> - The <i>Map</i> Method - Lab 7-5: Use the <i>Map</i> method

JavaScript Specialist Objective	JavaScript Specialist Courseware Lesson(s) and Section(s)
2.4.6 Apply the <i>Date</i> and <i>Math</i> objects.	<b>Lesson 7: JavaScript Language Objects</b> <ul style="list-style-type: none"> <li>- The <i>Date</i> Object</li> <li>- Lab 7-6: Accessing and using dates with the <i>Date</i> object</li> <li>- Setting and Extracting Time Information</li> <li>- Lab 7-7: Creating an onscreen clock</li> <li>- The <i>Math</i> Object</li> <li>- Lab 7-8: Using the <i>Math</i> object to generate a random quotation</li> </ul>
<b>2.5: Create and use custom JavaScript objects.</b>	
2.5.1 Create a custom JavaScript object.	<b>Lesson 8: Custom JavaScript Objects</b> <ul style="list-style-type: none"> <li>- Creating Custom Objects</li> <li>- Creating a JavaScript Object: The Constructor</li> <li>- Lab 8-1: Creating a custom object</li> </ul>
2.5.2 Define properties and methods of custom objects.	<b>Lesson 8: Custom JavaScript Objects</b> <ul style="list-style-type: none"> <li>- Creating a JavaScript Object: The Constructor</li> <li>- Creating Object Methods</li> <li>- Creating Functions for Your Objects</li> <li>- Lab 8-1: Creating a custom object</li> </ul>
2.5.3 Create new object instances.	<b>Lesson 8: Custom JavaScript Objects</b> <ul style="list-style-type: none"> <li>- Creating an Instance of a Custom Object</li> <li>- Creating Object Methods</li> <li>- Creating Functions for Your Objects</li> <li>- Lab 8-1: Creating a custom object</li> </ul>
2.5.4 Create client-side arrays using custom objects.	<b>Lesson 8: Custom JavaScript Objects</b> <ul style="list-style-type: none"> <li>- Creating Object Methods</li> <li>- Creating Functions for Your Objects</li> <li>- Lab 8-1: Creating a custom object</li> </ul>
2.5.5 Create functions and methods for manipulating client-side arrays.	<b>Lesson 8: Custom JavaScript Objects</b> <ul style="list-style-type: none"> <li>- Creating Object Methods</li> <li>- Creating Functions for Your Objects</li> <li>- Lab 8-1: Creating a custom object</li> </ul>
2.5.6 Use the prototype property, concept of classes, constructors, iterators and generators.	<b>Lesson 8: Custom JavaScript Objects</b> <ul style="list-style-type: none"> <li>- Creating an Instance of a Custom Object</li> <li>- Concept of Classes</li> <li>- Lab 8-2: Implementing classes and constructors</li> <li>- Iterators and Generators</li> <li>- Lab 8-3: Using iterators and generators</li> </ul>

JavaScript Specialist Objective	JavaScript Specialist Courseware Lesson(s) and Section(s)
<b>Domain 3: Applied JavaScript</b>	
<b>3.1: Modify HTML with JavaScript.</b>	
<p>3.1.1 Identify steps and methods for changing HTML "on the fly," including the <i>getElementById</i>, <i>getElementsByName</i>, <i>getElementsByTagName</i> and <i>getElementsByClassName</i> methods of the DOM.</p>	<p><b>Lesson 9: Changing HTML on the Fly</b></p> <ul style="list-style-type: none"> <li>- Changing HTML on the Fly</li> <li>- Using the <i>getElementById</i> Method</li> <li>- Lab 9-1: Redirecting a page based on user input with <i>getElementById</i></li> <li>- Using the <i>getElementsByName</i> Method</li> <li>- Using the <i>getElementsByTagName</i> Method</li> <li>- Lab 9-2: Debugging the <i>getElementsByTagName</i> method</li> <li>- Using the <i>getElementsByClassName</i> Method</li> <li>- Lab 9-3: Change the DOM using <i>getElementsByClassName</i></li> </ul>
<p>3.1.2 Modify attributes in HTML using DOM elements.</p>	<p><b>Lesson 9: Changing HTML on the Fly</b></p> <ul style="list-style-type: none"> <li>- Lab 9-1: Redirecting a page based on user input with <i>getElementById</i></li> <li>- Modifying Attributes within the DOM</li> <li>- Lab 9-4: Getting, setting and removing HTML attributes</li> </ul>
<p>3.1.3 Modify form object values.</p>	<p><b>Lesson 9: Changing HTML on the Fly</b></p> <ul style="list-style-type: none"> <li>- Modifying Attributes within the DOM</li> </ul>
<b>3.2: Use JavaScript to develop interactive forms.</b>	
<p>3.2.1 Identify and use form controls, including HTML5 form elements.</p>	<p><b>Lesson 10: Developing Interactive Forms with JavaScript</b></p> <ul style="list-style-type: none"> <li>- Overview of Form Elements</li> </ul>
<p>3.2.2 Define the <i>form</i> object.</p>	<p><b>Lesson 10: Developing Interactive Forms with JavaScript</b></p> <ul style="list-style-type: none"> <li>- The <i>form</i> Object</li> </ul>
<p>3.2.3 Refer to form objects, including <i>input</i>, <i>text</i>, <i>textarea</i>, <i>radio</i>, <i>checkbox</i>, <i>select</i>, <i>button</i>, <i>password</i>, <i>hidden</i>, <i>file</i>, and <i>submit</i>.</p>	<p><b>Lesson 10: Developing Interactive Forms with JavaScript</b></p> <ul style="list-style-type: none"> <li>- Referring to Form Elements</li> <li>- The <i>input</i> Elements and <i>text</i> Objects</li> <li>- Lab 10-1: Working with <i>input</i> elements</li> <li>- The <i>textarea</i> Object</li> <li>- Lab 10-2: Working with <i>textarea</i> elements</li> <li>- The <i>select</i> Object</li> <li>- Lab 10-3: Working with selection lists</li> <li>- Lab 10-4: Working with a multiple-selection list box</li> </ul>



JavaScript Specialist Objective	JavaScript Specialist Courseware Lesson(s) and Section(s)
3.2.4 Use <i>form</i> objects, including <i>radio</i> , <i>select</i> , <i>button</i> , <i>text</i> , <i>input</i> , <i>textarea</i> , <i>checkbox</i> , <i>password</i> , <i>hidden</i> , <i>file</i> and <i>submit</i> .	<b>Lesson 10: Developing Interactive Forms with JavaScript</b> <ul style="list-style-type: none"> <li>- Referring to Form Elements</li> <li>- The <i>input</i> Elements and <i>text</i> Objects</li> <li>- Lab 10-1: Working with <i>input</i> elements</li> <li>- The <i>textarea</i> Object</li> <li>- Lab 10-2: Working with <i>textarea</i> elements</li> <li>- The <i>select</i> Object</li> <li>- Lab 10-3: Working with selection lists</li> <li>- Lab 10-4: Working with a multiple-selection list box</li> </ul>
3.2.5 Conduct form validation.	<b>Lesson 10: Developing Interactive Forms with JavaScript</b> <ul style="list-style-type: none"> <li>- Form Validation</li> <li>- Lab 10-5: Conducting form validation</li> </ul>
3.2.6 Identify common form security issues.	<b>Lesson 10: Developing Interactive Forms with JavaScript</b> <ul style="list-style-type: none"> <li>- Common Form Security Issues</li> <li>- Lab 10-6: Identifying form security errors</li> </ul>
<b>3.3: Address JavaScript security issues involving browsers and cookies.</b>	
3.3.1 Distinguish between the browser and the operating system in relation to the elements responsible for security.	<b>Lesson 11: JavaScript Security</b> <ul style="list-style-type: none"> <li>- Browser vs. Operating System</li> </ul>
3.3.2 Discuss browser security issues relevant to JavaScript, including script blocking, frame-to-frame URL changing, and <i>document.write</i> behavior differences among browsers.	<b>Lesson 11: JavaScript Security</b> <ul style="list-style-type: none"> <li>- Browser-Specific Security Issues</li> <li>- Script Blocking</li> <li>- Differences in <i>document.write</i> Among Browsers</li> <li>- Malicious and Accidental Coding</li> <li>- Lab 11-2: Locking the browser with malicious code</li> <li>- Frame-to-Frame URL Changing</li> </ul>
3.3.3 Define signed scripts.	<b>Lesson 11: JavaScript Security</b> <ul style="list-style-type: none"> <li>- Signed Scripts</li> </ul>
3.3.4 Perform client-side browser detection and determine browser compatibility.	<b>Lesson 11: JavaScript Security</b> <ul style="list-style-type: none"> <li>- Browser Compatibility and Client-Side Detection</li> <li>- Lab 11-1: Performing client-side browser detection</li> </ul>
3.3.5 Identify common issues and procedures for creating secure JavaScript code.	<b>Lesson 11: JavaScript Security</b> <ul style="list-style-type: none"> <li>- Malicious and Accidental Coding</li> <li>- Lab 11-2: Locking the browser with malicious code</li> <li>- Creating Secure JavaScript Code</li> </ul>
3.3.6 Define cross-site scripting and the associated security risks.	<b>Lesson 11: JavaScript Security</b> <ul style="list-style-type: none"> <li>- Cross-Site Scripting (XSS)</li> </ul>

JavaScript Specialist Objective	JavaScript Specialist Courseware Lesson(s) and Section(s)
3.3.7 Define the functions and common uses of cookies.	<b>Lesson 11: JavaScript Security</b> <ul style="list-style-type: none"> <li>- Cookies and Security in JavaScript</li> <li>- Lab 11-3: Setting, viewing and clearing a cookie with JavaScript</li> </ul>
3.3.8 Manipulate cookies effectively, including testing for presence of cookies, clearing cookies, enabling/disabling cookies in the browser, and deleting cookies from your hard drive.	<b>Lesson 11: JavaScript Security</b> <ul style="list-style-type: none"> <li>- Cookies and Security in JavaScript</li> <li>- Lab 11-3: Setting, viewing and clearing a cookie with JavaScript</li> </ul>
3.3.9 Discuss ethics in collecting, storing, using and protecting user data.	<b>Lesson 11: JavaScript Security</b> <ul style="list-style-type: none"> <li>- Ethics in Collecting, Storing, Using and Protecting User Data</li> <li>- Lab 11-4: What are standard ethic policies?</li> </ul>
<b>Domain 4: JavaScript Technology Extensions</b>	
<b>4.1: Implement JavaScript libraries and frameworks.</b>	
4.1.1 Identify and evaluate the benefits and drawbacks of using predefined libraries and frameworks, such as jQuery, Spry, Dojo, React.js, Angular.js and Prototype.	<b>Lesson 12: JavaScript Libraries and Frameworks</b> <ul style="list-style-type: none"> <li>- JavaScript Libraries and Frameworks</li> <li>- Choosing a Code Library</li> <li>- External and Pre-Made Scripts</li> </ul>
4.1.2 Identify steps for using libraries (such as jQuery), frameworks and available plug-ins, including optimization for faster JavaScript manipulation.	<b>Lesson 12: JavaScript Libraries and Frameworks</b> <ul style="list-style-type: none"> <li>- External and Pre-Made Scripts</li> <li>- Loading Your First Library</li> <li>- Lab 12-1: Loading a JavaScript library and running a library script</li> <li>- Lab 12-2: Using CSS and JavaScript</li> <li>- Using JavaScript Library Plug-ins</li> <li>- Lab 12-3: Loading, testing and editing a library plug-in</li> <li>- Using Frameworks</li> <li>- Best Practices with JavaScript Libraries</li> </ul>
4.1.3 Identify steps for loading and referencing external scripts and pre-made external scripts.	<b>Lesson 12: JavaScript Libraries and Frameworks</b> <ul style="list-style-type: none"> <li>- External and Pre-Made Scripts</li> <li>- Lab 12-1: Loading a JavaScript library and running a library script</li> <li>- Lab 12-2: Using CSS and JavaScript to create a basic slideshow</li> <li>- Lab 12-3: Loading, testing and editing a library plug-in</li> </ul>
4.1.4 Identify and evaluate the benefits and drawbacks of Server-side JavaScript technologies.	<b>Lesson 12: JavaScript Libraries and Frameworks</b> <ul style="list-style-type: none"> <li>- Best Practices with JavaScript Libraries and Frameworks</li> </ul>
4.1.5 Optimize page load time and user experience on various devices.	<b>Lesson 12: JavaScript Libraries and Frameworks</b> <ul style="list-style-type: none"> <li>- Best Practices with JavaScript Libraries and Frameworks</li> </ul>

JavaScript Specialist Objective	JavaScript Specialist Courseware Lesson(s) and Section(s)
<b>4.2: Use JavaScript and AJAX to create interactive Web applications.</b>	
4.2.1 Define synchronous and asynchronous, fundamental AJAX elements, and procedures.	<b>Lesson 13: JavaScript and AJAX</b> - Introduction to AJAX - Interactions Among JavaScript, Data and HTML - Usability Issues with AJAX
4.2.2 Explain the Fetch API, Promises and <i>callback</i> functions.	<b>Lesson 13: JavaScript and AJAX</b> - Interactions Among JavaScript, Data and HTML
4.2.3 Use the <i>XMLHttpRequest</i> object to retrieve data.	<b>Lesson 13: JavaScript and AJAX</b> - Using AJAX Scripts - Lab 13-1: Using AJAX to dynamically edit the DOM with button clicks
4.2.4 Describe typical AJAX-based requests.	<b>Lesson 13: JavaScript and AJAX</b> - AJAX and Servers
4.2.5 Identify key server response issues related to AJAX-based requests.	<b>Lesson 13: JavaScript and AJAX</b> - AJAX and Servers
4.2.6 Use JavaScript to communicate with databases.	<b>Lesson 13: JavaScript and AJAX</b> - AJAX and Databases
4.2.7 Identify and compare XML and JSON.	<b>Lesson 13: JavaScript and AJAX</b> - Handling Data and Content Types
<b>4.3: Implementing Web APIs.</b>	
4.3.1 Define Web API (Application Programming Interface) and benefits.	<b>Lesson 14: Introduction to Web APIs</b> - Web Application Programming Interface (API)
4.3.2 Identify and use Web APIs, including session storage, local storage, and GEO location.	<b>Lesson 14: Introduction to Web APIs</b> - Web Application Programming Interface (API) - Lab 14-1: GEO location weather apps
4.3.3 Identify and use the Canvas API.	<b>Lesson 14: Introduction to Web APIs</b> - Web Application Programming Interface (API) - Lab 14-2: Guided animation projects