Course Outline

20480-Programming in HTML5 with JavaScript and CSS3

Duration: 5 days (30 hours)

Target Audience:

The course is intended for developers who have at least six months of professional experience and who are interested in developing applications using HTML5 with JavaScript and CSS3 (either Windows Store apps or IE10 apps for the Web). While the students may have little or no HTML5 coding experience, they should have some experience with HTML4.

Students choosing to attend this course without a developer background should pay special attention to the training prerequisites. Developers who have more than 5 years programming experience may find that portions of this training are fundamental in nature when presenting the syntax associated with certain programming tasks.

Individuals who are interested in taking exam 70-480: Programming in HTML5 with JavaScript and CSS3, can also attend this course.

Prerequisites:

Before attending this course, students must have at least three months professional development experience.
In addition to their professional experience, students who attend this training should have a combination of practical and conceptual knowledge related to HTML5 programming. This includes the following prerequisites:

- Understand the basic HTML document structure:
    - Use HTML tags to display text content.
    - Use HTML tags to display graphics.
    - Use HTML APIs.
- Understand how to style common HTML elements using CSS, including:
    - Separating presentation from content.
    - Managing content flow.
    - Managing positioning of individual elements.
    - Managing content overflow.
    - Basic CSS styling.

Topics Covered:

- Module 1: Overview of HTML and CSS
    - Overview of HTML
    - Overview of CSS
    - Creating a Web Application by Using Visual Studio 2012
        - Lab : Exploring the Contoso Conference Application

- Walkthrough of the Contoso Conference Application
- Examining and Modifying the Contoso Conference Application

After completing this module, students will be able to:

- Describe basic HTML elements and attributes.
- Explain the structure of CSS.
- Describe the tools available in Visual Studio 2012 for building Web applications.

➢ Module 2: Creating and Styling HTML5 Pages

- Creating an HTML5 Page
- Styling an HTML5 Page
  - Lab : Creating and Styling HTML5 Pages
    - Creating HTML5 Pages
    - Styling HTML5 Pages

After completing this module, students will be able to:

- Create static pages using the new features available in HTML5.
- Use CSS3 to apply basic styling to the elements in an HTML5 page.

➢ Module 3: Introduction to JavaScript

- Overview of JavaScript Syntax
- Programming the HTML DOM with JavaScript
- Introduction to jQuery
  - Lab : Displaying Data and Handling Events by Using JavaScript
    - Displaying Data Programmatically
    - Handling Events

After completing this module, students will be able to:

- Explain the syntax of JavaScript and describe how to use JavaScript with HTML5.
- Write JavaScript code that manipulates the HTML DOM and handles events.
- Describe how to use jQuery to simplify code that uses many common JavaScript APIs.

➢ Module 4: Creating Forms to Collect Data and Validate User Input

- Overview of Forms and Input Types
- Validating User Input by Using HTML5 Attributes
- Validating User Input by Using JavaScript
  - Lab : Creating a Form and Validating User Input
    - Creating a Form and Validating User Input by Using HTML5 Attributes
    - Validating User Input by Using JavaScript

After completing this module, students will be able to:

- Create forms that use the new HTML5 input types.
- Validate user input and provide feedback by using the new HTML5 attributes.
- Write JavaScript code to validate user input and provide feedback in cases where it is not suitable to use HTML5 attributes

➢ Module 5: Communicating with a Remote Data Source

- Sending and Receiving Data by Using XMLHTTPRequest

- Sending and Receiving Data by Using jQuery AJAX operations
    - Lab : Communicating with a Remote Data Source
        - Retrieving Data
        - Serializing and Transmitting Data
        - Refactoring the Code by Using jQuery ajax method

After completing this module, students will be able to:

- Serialize, deserialize, send, and receive data by using XMLHTTPRequest objects.
- Simplify code that serializes, deserializes, sends, and receives data by using the jQuery ajax method.

➢ Module 6: Styling HTML5 by Using CSS3

- Styling Text
- Styling Block Elements
- CSS3 Selectors
- Enhancing Graphical Effects by Using CSS3
    - Lab : Styling Text and Block Elements using CSS3
        - Styling the Navigation Bar
        - Styling the Page Header
        - Styling the About Page

After completing this module, students will be able to:

- Style text elements on an HTML5 page by using CSS3.
- Apply styling to block elements by using CSS3.
- Use CSS3 selectors to specify the elements to be styled in a Web application.
- Implement graphical effects and transformations by using the new CSS3 properties.

➢ Module 7: Creating Objects and Methods by Using JavaScript

- Writing Well-Structured JavaScript
- Creating Custom Objects
- Extending Objects
    - Lab : Refining Code for Maintainability and Extensibility
        - Inheriting From Objects
        - Refactoring Code to Use Objects

After completing this module, students will be able to:

- Describe the benefits of structuring JavaScript code carefully to aid maintainability and extensibility.
- Explain best practices for creating custom objects in JavaScript.
- Describe how to extend custom and native objects to add functionality.

➢ Module 8: Creating Interactive Pages using HTML5 APIs

- Interacting with Files
- Incorporating Multimedia
- Reacting to Browser Location and Context
- Debugging and Profiling a Web Application

- Lab : Creating Interactive Pages by Using HTML5 APIs
  - Incorporating Video
  - Incorporating Images
  - Using the Geolocation API

After completing this module, students will be able to:

- Use the Drag and Drop, and the File APIs to interact with files in a Web application.
- Incorporate audio and video into a Web application.
- Detect the location of the user running a Web application by using the Geolocation API.
- Explain how to debug and profile a Web application by using the Web Timing API and the Internet Explorer Developer Tools.

➢ Module 9: Adding Offline Support to Web Applications

- Reading and Writing Data Locally
- Adding Offline Support by Using the Application Cache
  - Lab : Adding Offline Support to a Web Application
    - Implementing the Application Cache
    - Implementing Local Storage

After completing this module, students will be able to:

- Save and retrieve data locally on the user's computer by using the Local Storage API.
- Provide offline support for a Web application by using the Application Cache API.

➢ Module 10: Implementing an Adaptive User Interface

- Supporting Multiple Form Factors
- Creating an Adaptive User Interface
  - Lab : Implementing an Adaptive User Interface
    - Creating a Print-Friendly Stylesheet
    - Adapting Page Layout To Fit a Different Form Factor

After completing this module, students will be able to:

- Describe the need to detect device capabilities and react to different form factors in a Web application.
- Create a Web page that can dynamically adapt its layout to match different form factors.

➢ Module 11: Creating Advanced Graphics

- Creating Interactive Graphics by Using Scalable Vector Graphics
- Programmatically Drawing Graphics by Using a Canvas
  - Lab : Creating Advanced Graphics
    - Creating an Interactive Venue Map by Using Scalable Vector Graphics
    - Creating a Speaker Badge by Using a Canvas Element

After completing this module, students will be able to:

- Use Scalable Vector Graphics to add interactive graphics to an application.
- Draw complex graphics on an HTML5 Canvas element by using JavaScript code.

➢ Module 12: Animating the User Interface

- Applying CSS Transitions

- Transforming Elements
- Applying CSS Key-frame Animations
  - Lab : Animating User Interface Elements
    - Applying Transitions to User Interface Elements
    - Applying Key-Frame Animations

After completing this module, students will be able to:

- Apply CSS transitions to elements on an HTML5 page, and write JavaScript code to detect when a transition has occurred.
- Describe the different types of 2D and 3D transitions available with CSS3
- Implement complex animations by using CSS key-frames and JavaScript code.

➢ Module 13: Implementing Real-Time Communications by Using Web Sockets

- Introduction to Web Sockets
- Sending and Receiving Data by Using Web Sockets
  - Lab : Implementing Real-Time Communications by Using Web Sockets
    - Receiving Data from Web Socket
    - Sending Data to a Web Socket
    - Sending Multiple Types of Messages To or From a Web Socket

After completing this module, students will be able to:

- Explain how Web Sockets work and describe how to send and receive data through a Web Socket.
- Use the Web Socket API with JavaScript to connect to a Web Socket server, send and receive data, and handle the different events that can occur when a message is sent or received.

➢ Module 14: Creating a Web Worker Process

- Introduction to Web Workers
- Performing Asynchronous Processing by Using a Web Worker
  - Lab : Creating a Web Worker Process
    - Improving Responsiveness by Using a Web Worker
    - Providing User Feedback During a Long-Running Process

After completing this module, students will be able to:

- Describe the purpose of a Web Worker process, and how it can be used to perform asynchronous processing as well as provide isolation for sensitive operations.
- Use the Web Worker APIs from JavaScript code to create, run, and monitor a Web Worker process.