

20480 Programming in HTML5 with JavaScript and CSS3

Overview

This course provides an introduction to HTML5, CSS3, and JavaScript and helps students gain basic HTML5/CSS3/JavaScript programming skills. This course is an entry point into both the Web application and Windows Store apps training paths.

Prerequisite Comments

Before attending this course, students must have at least three months of 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:
 - How to use HTML tags to display text content.
 - How to use HTML tags to display graphics.
 - How to use HTML APIs.
- Understand how to style common HTML elements using CSS, including:
 - How to separate presentation from content
 - How to manage content flow.
 - How to control the position of individual elements.
 - How to implement basic CSS styling.
- Understand how to write JavaScript code to add functionality to a web page:
 - How to create and use variables
 - How to use:
 - arithmetic operators to perform arithmetic calculations involving one or more variables
 - relational operators to test the relationship between two variables or expressions
 - logical operators to combine expressions that contain relational operators
 - How to control the program flow by using if ... else statements.
- How to implement iterations by using loops.
- How to write simple functions.

1-3 months experience creating Web applications, including writing simple JavaScript code
1 month experience creating Windows client applications
1 month of experience using Visual Studio 2017

Target Audience

This course is intended for professional developers who have 6-12 months of programming experience and who are interested in developing applications using HTML5 with JavaScript and CSS3 (either Windows Store apps for Windows 8 or web applications).

Course Objectives

After completing this course, students will be able to:

Explain how to use Visual Studio 2017 to create and run a Web application.

Describe the new features of HTML5, and create and style HTML5 pages.

Add interactivity to an HTML5 page by using JavaScript.

Create HTML5 forms by using different input types, and validate user input by using HTML5 attributes and JavaScript code.

Send and receive data to and from a remote data source by using XMLHttpRequest objects and Fetch API.

Style HTML5 pages by using CSS3.

Create well-structured and easily-maintainable JavaScript code.

Write modern JavaScript code and use babel to make it compatible to all browsers.

Use common HTML5 APIs in interactive Web applications.

Create Web applications that support offline operations.

Create HTML5 Web pages that can adapt to different devices and form factors.

Add advanced graphics to an HTML5 page by using Canvas elements, and by using and Scalable Vector Graphics.

Enhance the user experience by adding animations to an HTML5 page.

Use Web Sockets to send and receive data between a Web application and a server.

Improve the responsiveness of a Web application that performs long-running operations by using Web Worker processes.

Use WebPack to package web applications for production.

Course Outline

1 - Overview of HTML and CSS

Overview of HTML

Overview of CSS

Creating a Web Application by Using Visual Studio 2017

Lab : Exploring the Contoso Conference Application

2 - Creating and Styling HTML5 Pages

Creating an HTML5 Page

Styling an HTML5 Page

Lab : Creating and Styling HTML5 Pages

3 - Introduction to JavaScript

Overview of JavaScript

Introduction to the Document Object Model

Lab : Displaying Data and Handling Events by Using JavaScript

4 - Creating Forms to Collect and Validate User Input

Creating HTML5 Forms

Validating User Input by Using HTML5 Attributes

Validating User Input by Using JavaScript

Lab : Creating a Form and Validating User Input

5 - Communicating with a Remote Server

Async programming in JavaScript
Sending and Receiving Data by Using the XMLHttpRequest Object
Sending and Receiving Data by Using the Fetch API
Lab : Communicating with a Remote Data Source

6 - Styling HTML5 by Using CSS3

Styling Text by Using CSS3
Styling Block Elements
Pseudo-Classes and Pseudo-Elements
Enhancing Graphical Effects by Using CSS3
Lab : Styling Text and Block Elements using CSS3

7 - Creating Objects and Methods by Using JavaScript

Writing Well-Structured JavaScript Code
Creating Custom Objects
Extending Objects
Lab : Refining Code for Maintainability and Extensibility

8 - Creating Interactive Pages by 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

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

10 - Implementing an Adaptive User Interface

Supporting Multiple Form Factors
Creating an Adaptive User Interface
Lab : Implementing an Adaptive User Interface

11 - Creating Advanced Graphics

Creating Interactive Graphics by Using SVG
Drawing Graphics by Using the Canvas API
Lab : Creating Advanced Graphics

12 - Animating the User Interface

Applying CSS Transitions
Transforming Elements
Applying CSS Keyframe Animations
Lab : Animating the User Interface

13 - Implementing Real-time Communication by Using Web Sockets

Introduction to Web Sockets
Using the WebSocket API
Lab : Performing Real-time Communication by Using Web Sockets

14 - Performing Background Processing by Using Web Workers

Understanding Web Workers
Performing Asynchronous Processing by Using Web Workers
Lab : Creating a Web Worker Process

15 - Packaging JavaScript for Production Deployment

Understanding Transpilers And Module bundling
Creating Separate Packages for Cross Browser Support
Lab : Setting Up Webpack Bundle for Production
