Course Outline

Modern Web Development with HTML5, CSS3, & Modern JavaScript (ES6+)

Course Summary

Description

This course helps both new and experienced developers boost their productivity and gain awareness of the newest features and industry best practices in core web technologies. This course is a solid foundation for those moving on to work with Angular or React. A balanced mixture of theory and practical labs are designed to arm students with the skills needed to be more effective web developers. Students develop multiple applications using AJAX & REST APIs to create, read, update and delete data (CRUD) from a server.

We begin with an overview of Modern Web Development and discussion of current popular frameworks including Angular, React, and Ember. This helps students get a sense of the current development environment for browser, desktop and mobile applications.

Students will learn and practice fundamental concepts with hands-on development using the 3 pillars of Web Dev: HTML5, CSS3 and Modern JavaScript (ES6+). Students leverage many development tools during the course, including working with Node.js, npm, build tools, linters (code quality), and automated testing frameworks. Numerous demo projects and resources are shared for continued, deeper learning.

Objectives

At the completion of this course, students will learn;

- Basic and best practices with HTML, CSS and JavaScript
- Leverage features and extensions of VSCode to write, debug and test code faster
- Understand HTML5 semantic tags, forms, features, and simplified page markup
- Practice with HTML5 APIS: Geolocation, Media, Canvas, and Web Storage
- Separate design from content and address accessibility with CSS3
- Responsive web design concepts and Bootstrap
- Use AJAX to work with JSON data from a REST Based server API
- Define Modern JavaScript (ES6) and practice with new syntax and modules
- Get a feel for the most popular options between libraries and frameworks
- Use Karma and Jasmine for automated testing of code

Audience

This course is designed for new or experienced web developers who have been self-taught or can benefit from focused training & understanding of the best practices followed in web development. Those with no web dev experience are positioned to learn, as this course focuses on best practices with fundamentals from the ground up.

Course Outline

Modern Web Development with HTML5, CSS3, & Modern JavaScript (ES6+)

Course Summary (cont.)

Topics

- Introduction
- Overview of Modern Web Development & Node
- Working with Browsers
- HTML5 Elements
- CSS3 Features
- Responsive Web Design with Bootstrap
- Modern JavaScript Essentials
- Client-Side APIs
- Working with HTML5Forms
- JavaScript ES6 Modules
- Client-Side Web APIs
- Unit Testing JavaScript

Prerequisites

Students should be able to navigate the Windows file system, and understand variables, functions and loops.

Duration

Five Days

Course Outline

Modern Web Development with HTML5, CSS3, & Modern JavaScript (ES6+)

Course Outline

I. *Introduction*
- A. Course Objectives and Overview
- B. Course Logistics and Introductions
- C. Development Environment Setup: Git, VSCode, Node, npm
- D. Getting Course Demos & Solutions

II. *Overview of Modern Web Development & Node*
- A. Front vs Back vs Full-stack Development
- B. Traditional Web Development
- C. Overview of Modern Web Development
- D. HTML5 W3C and WHATWG specs
- E. EcmaScript / JavaScript standard
- F. Accessibility Guidelines Section 508
- G. Single Page Applications: Angular, Ember, Vue, React
- H. Using Node to setup development servers and tools
- I. Basics of calling and receiving data from a REST API
- J. Modifying the package.json for dependencies and scripts
- K. Using npm commands

III. *Working with Browsers*
- A. Using browser developer tools for Dom inspection and manipulation
- B. Testing network traffic
- C. JS debugging
- D. Shims, shivs, polyfills and Modernizr

IV. *HTML5 Elements*
- A. HTML doctype, elements & attributes
- B. Adding a custom favicon to your site
- C. Using validators to ensure adherence to standards
- D. Block-level and inline elements
- E. HTML5 Semantic Tags
- F. Image optimizations and alt tags for accessibility
- G. Working with multi-level lists
- H. Tables - for Data only!

V. *CSS3 Features*
- A. CSS HTML selectors, classes, and ids
- B. Understanding why your CSS rules are being overridden
- C. Specificity & Order when multiple CSS rules are applied
- D. Using media queries for targeting different devices
- E. Ways to include CSS for screen and print
- F. Text Properties, Font Properties, Backgrounds
- G. CSS Box Model, MBP
- H. Using floats, margins and positioning techniques
- I. Custom Fonts and Color in CSS3
- J. Ensuring text is readable with backgrounds being used
- K. Combinators (>, +, ~)
- L. Pseudo-Class Selectors
- M. Shadows and Other Effects
- N. Performance Considerations

VI. *Responsive Web Design with Bootstrap*
- A. How grids and media queries are used for Responsive web design
- B. How time can be saved using LESS or Sass
- C. What is Flexbox
- D. Getting Started with Bootstrap 4

VII. *Modern JavaScript Essentials*
- A. JavaScript variables, types, arrays and functions
- B. New syntax in EcmaScript 6
- C. Garbage collection, avoiding memory leaks
- D. Global functions: setInterval and setTimeout
- E. JavaScript arrays and functions
- F. Objects and properties
- G. Arrow functions and this

Course Outline

Modern Web Development with HTML5, CSS3, & Modern JavaScript (ES6+)

Course Outline (cont.)

VIII. *Client-Side APIs*

A. Strategies for adding JS to web pages
B. Drawing with the canvas
C. Storing data locally with HTML5 Web Storage
D. Using Geolocation to get position
E. Using AJAX to communicate with the server
F. What is Fetch?
G. Introducing promises

IX. *Working with HTML5Forms*

A. Using new HTML5 Form tags and attributes
B. Enforcing constraints using regular expression and patterns

X. *JavaScript ES6 Modules*

A. Common Pitfalls in classic JavaScript
B. Operating in “strict mode” and work-arounds such as IIFEs
C. ES6 Modules & Module Loading
D. Browserify & Webpack

XI. *Client-Side Web APIs*

A. Overview of drawing with the canvas
B. Storing data locally with HTML5 Web Storage
C. Making calls to the server with AJAX and fetch
D. Introducing promises and observables

XII. *Unit Testing JavaScript*

A. Learn about unit testing JavaScript code
B. Demonstrate assertion frameworks and testrunners
C. Mocha and Chai examples
D. Using Karma & Jasmine for front-end testin