

**Modern Web Development with HTML, CSS, and JavaScript**

**Course Number:** SCRPT-192
**Duration:** 5 days

**Overview**

Creating modern web applications demands a deep understanding of HTML, CSS, advanced JavaScript techniques, as well as at least one client-side framework like Angular, React, Vue, or Svelte.

This Web Development training course teaches attendees how to create web applications using well-formed HTML, concise and flexible CSS, and scalable and elegant JavaScript, preparing learners to subsequently master a client-side framework. This course uses modern techniques and practices that stretch even experienced developers but are still within the grasp of JavaScript novices.

**Prerequisites**

No prior experience is presumed.

**Materials**

All Web Development training attendees receive comprehensive courseware.

**Software Needed on Each Student PC**

* Google Chrome
* Other modern browsers as desired
* IDE/development environment of your choice
* Other free software and lab files that Accelebrate would specify

**Objectives**

* Create web pages and websites using HTML5 and CSS3
* Debug HTML, CSS, and JavaScript
* Expertly apply styles with the advanced CSS3 selectors, including pseudo-elements and pseudo-classes
* Compare and contrast flexbox and grid and know when to use each
* Use HTML5 semantic tags like <main>, <section>, <header>, <article> and more
* Lay out pages using CSS flexbox and grids
* Write well-organized and properly structured JavaScript modules
* Handle multi-threading in JavaScript with promises and async/await
* Consume a RESTful API with Ajax using the fetch API

**Outline**

* Course Intro
* The Secrets of Web Development
	+ Architecture of the web
	+ How the W3C works
	+ How to exploit modern browser capabilities
* Perfect Page Setup
	+ The proper structure of HTML
	+ The most critical elements
	+ SMS, phone calls, and emails from pages
* JavaScript Quickstart
	+ Just enough JavaScript to write a program (if, while, for, comments)
* How to Control the DOM
	+ What is the DOM?
	+ Querying the DOM
	+ Wiring up raw event handlers
	+ Altering the DOM for dynamic views
	+ So who needs Angular, React, Vue?
* Debugging Tools
	+ Node inspector
	+ On-the-fly HTML/CSS changes
	+ Inspecting HTTP packets
	+ Emulating phones and tablets
* Operators
	+ Arithmetic
	+ Auto-operators
	+ Logical operators
	+ Truthy and falsey
	+ Short-circuiting
* Variables
	+ let, var, and const: when to use each
	+ How hoisting works
	+ Destructuring
	+ Easy string templates with `${}`
* Arrays in JavaScript
	+ Iterating arrays
	+ for-in vs. for-of
	+ How to spread arrays
	+ Array.prototype.\*
	+ map()
	+ filter()
* Semantic Grouping
	+ Why use them?
	+ Section, article, nav, aside
	+ Header, footer, main
* Effective CSS Styling
	+ Fundamental separation of concerns
	+ Best practices on placing styles
	+ Basic selectors
	+ Selector specificity
	+ !important, and why not to use it
* Positioning with CSS
	+ Position: absolute vs. relative vs. fixed vs. static
	+ The box model
	+ The art of centering
* How to Layout Pages with CSS
	+ Why tables are a fail
	+ Floating divs
	+ Display: inline-block
	+ Flexbox layouts
	+ Grid layouts
* Deep dive into Flexbox
	+ How flex works
	+ The two ways to think about flex
	+ Wrapping flexbox
	+ No-wrap flexbox
* Deep dive into Grids
	+ How grid works
	+ Lines, tracks, cells, and areas
	+ Defining the grid
	+ Placing elements in the grid
* Advanced CSS Selectors
	+ Basic selectors reminder
	+ Compound selectors
	+ Relationship selectors
	+ Attribute selectors
	+ Pseudo-classes
	+ :nth-child()
	+ :not()
	+ Pseudo-elements
* Functional JavaScript
	+ Variadicity
	+ Function statements
	+ Function expressions
	+ Arrow functions
	+ Default parameters
* Object-oriented JavaScript
	+ Functional JavaScript vs. object-oriented JavaScript - Which should I choose?
	+ How prototypal inheritance is different from traditional inheritance
	+ Classes
	+ Constructors
	+ get() and set()
	+ Classless objects
* Modules with JavaScript
	+ requireJS
	+ import
	+ export vs. export default
	+ IIFEs
* Asynchronous JavaScript
	+ Multithreading
	+ How the event loop works
	+ Promises
	+ Async and await
* Ajax
	+ Making RESTful calls
	+ JSON.stringify() and parse()
	+ The fetch API
* Deep Dive into Tables (time-permitting)
	+ The legal structure of tables
	+ Spanning
	+ Styling techniques
* Best Practices with Forms (time-permitting)
	+ A complete form tag
	+ The most useful inputs
	+ Selects
	+ Configuring the soft keyboard
	+ How to write declarative data validations
* Conclusion