

Introduction to Modern Web Development

Applied Technology Academy

Three days of practical front-end web development with HTML, CSS and JavaScript. Participants create semantic, accessible documents and forms, work through the cascade, Flexbox, Grid, custom properties and responsive design, then use JavaScript for the DOM, events, forms, page state, JSON and Fetch. Generative AI assistants are used as development partners throughout - and every generated solution is inspected, tested in the browser and refined rather than accepted on trust.

LEVEL	DURATION	COURSE CODE	DELIVERY
Introductory	3 Days	SEN-301	Instructor-led

Course Overview

- Three days, project-based: labs build on one another into a complete responsive single-page website.
- No previous web-development experience is required.
- Covers the modern platform - semantic HTML, modern CSS layout and browser JavaScript APIs.
- Generative AI (ChatGPT, Gemini) is used to explain code, suggest changes and help troubleshoot, with every suggestion reviewed and tested rather than accepted automatically.

Course Outline

- **Module 1. How the Modern Web Works**
 - Clients, servers, HTTP, the browser and the DOM; what HTML, CSS and JavaScript each own.
 - Organizing the files for a front-end project and inspecting a page with Developer Tools.
 - Generative AI as a coding assistant: describing a requirement and requesting a targeted suggestion.
- **Module 2. HTML Semantics**
 - Valid modern documents, semantic structure and logical heading order.
 - Accessible links and images, and why semantics matter for assistive technology and maintainability.
- **Module 3. HTML Forms and User Input**
 - Structured forms, labels correctly associated with controls, and appropriate input types.
 - Browser-based validation, and why it does not replace server-side validation.
- **Module 4. CSS Fundamentals**
 - Rules and selectors; the cascade, inheritance, specificity and source order.
 - CSS units, colors, typography and external stylesheets.
- **Module 5. The CSS Box Model**

- Dimensions, padding, borders, margins, box-sizing, overflow and display behavior.
- Diagnosing common overflow and sizing problems with Developer Tools.
- **Module 6. Advanced CSS Selectors**
 - Combinators, attribute selectors, pseudo-classes, functional selectors and pseudo-elements.
 - Styling UI states without unnecessary JavaScript, and the specificity consequences.
- **Module 7. CSS Custom Properties and Modern CSS Values**
 - Defining and consuming custom properties; a small reusable design-token system.
 - Scoping, overriding and fallbacks; modern CSS functions for fluid sizing and spacing.
- **Module 8. Advanced Formatting, Images and Backgrounds**
 - Background images, gradients, transparency, borders, shadows, filters and visual layering.
 - Presenting embedded images responsively while keeping text readable.
- **Module 9. Modern Layout with Flexbox**
 - Aligning, distributing, wrapping, resizing and reordering within flexible containers.
 - Practical patterns: navigation, toolbars, forms and card components.
- **Module 10. CSS Grid**
 - Rows, columns, tracks, lines, fractional units and grid areas.
 - Responsive grids with repeat(), minmax(), auto-fit and auto-fill; choosing Grid or Flexbox.
- **Module 11. Responsive Web Design**
 - Mobile-first design, fluid layouts, relative sizing, responsive typography and images.
 - Choosing breakpoints from content rather than device names, and testing with browser tools.
- **Module 12. Modern CSS Effects and Interaction**
 - Transitions, transforms, state-based styling and basic keyframe animation.
 - Respecting reduced-motion preferences, and deciding when CSS alone is enough.
- **Module 13. JavaScript for the Browser**
 - Variables, values, arrays, objects, conditions, loops and functions - for page behavior.
 - Using the browser console to test and troubleshoot, and asking AI to explain unfamiliar code.
- **Module 14. Working with the DOM**
 - Locating elements, reading and modifying content and attributes, managing CSS classes.
 - Creating elements programmatically and generating interface content from JavaScript data.
- **Module 15. Events and User Interaction**
 - Event listeners for clicks, keyboards, form controls and submission.
 - Updating interface state, and keeping interactions understandable and accessible.
- **Module 16. Building a Single-Page Website**
 - Content structure, anchors, fragment identifiers, smooth scrolling and active navigation state.
 - Introductory History API concepts, meaningful URLs, keyboard navigation and focus management.

- **Module 17. Forms in an Interactive Website**
 - Reading and processing form values, and supplementing built-in validation with JavaScript.
 - Clear success and error feedback, and dynamically changing form and interface state.
- **Module 18. Dynamic Data with JSON and Fetch**
 - JSON, fetch(), and an introduction to promises and async/await.
 - Rendering retrieved data with appropriate loading, empty and error states.
- **Module 19. Practical Single-Page UI Patterns and Browser APIs (time permitting)**
 - Mobile navigation, expandable sections, tabs, filtering, galleries and scrolling behavior.
 - Data attributes to associate HTML with behavior, and browser APIs for section awareness.
- **Module 20. Debugging JavaScript Web Applications (time permitting)**
 - Console errors, breakpoints, inspecting variables, tracing events and DOM changes.
 - Investigating failed or unexpected network requests.

Prerequisites

- No previous web-development experience is required.
- Participants should be comfortable with a modern desktop operating system, files and folders.
- Comfort with a web browser and a text or code editor is expected.
- Previous programming experience can help but is not required.
- Previous experience with ChatGPT, Gemini or another generative AI tool is not required.

Who Should Attend

- Aspiring web developers.
- Designers who want to expand their coding skills.
- Application developers moving into front-end work.
- Technical professionals who work with web content.
- Anyone who wants to understand how modern interactive websites are built.

What You'll Learn

By the end of this course, participants will be able to:

- explain the roles of HTML, CSS and JavaScript in modern web applications
- create well-structured HTML documents using semantic elements
- create accessible links, images, navigation and forms
- apply CSS using the cascade, inheritance, specificity and modern selectors
- use custom properties to build a maintainable styling system
- create modern page layouts with Flexbox and CSS Grid
- build responsive interfaces with fluid sizing, media queries and modern CSS functions

- manipulate the page through the DOM and respond to user interaction with events
- process and validate form input in the browser
- work with JSON and retrieve data using the Fetch API
- use browser Developer Tools to inspect, test and debug
- write clear prompts, then review and test AI-generated code rather than assuming it is correct

Hands-On Work

- Labs build on one another so that participants progressively develop a complete responsive and
- interactive single-page website. At selected points they use generative AI to explain code,
- suggest small changes and assist with troubleshooting - then inspect, test in the browser and
- refine what it produces before incorporating it into the project.

Ready to enroll? Call 800.674.3550 or email Info@AppliedTechAc.com — private team cohorts, GSA MAS purchasing, military credentialing funding, VA GI Bill and VR&E and student financing available.