

Web Development

Build the structure. Design the experience. Bring it to life.

Web development is the process of creating websites and interactive experiences for the internet. HTML structures content, CSS controls presentation and layout, and JavaScript adds behaviour and interactivity.

12-week learning programme

HTML • CSS • JavaScript

Practical coding sessions, guided projects, and a final web application



SkillBridge Academy's Web Development programme introduces learners to the core front-end technologies through guided instruction, coding demonstrations, practical exercises, projects, debugging, and feedback. Over twelve weeks, learners will progress from creating their first HTML page to building responsive, interactive web applications using HTML, CSS, and JavaScript.

Is this course for you?

This programme is designed for beginners, secondary school and university students, graduates, and adults who want to build practical digital skills. No previous web development experience is required, although basic computer skills and regular practice are important.

What you'll learn

By the end of the programme, you'll have practical experience in how to:

- Understand how websites work: Explain the roles of browsers, servers, HTML, CSS, and JavaScript in a front-end website.
- Structure web content: Create clear, semantic HTML pages using headings, links, images, lists, tables, and forms.
- Style and lay out pages: Apply CSS selectors, the box model, Flexbox, Grid, and responsive design techniques.
- Write JavaScript logic: Use variables, data types, operators, conditionals, loops, functions, arrays, and objects.
- Build interactive interfaces: Use the DOM and event listeners to respond to user actions and update page content.
- Handle and store user data: Validate form input and use browser local storage for simple persistent data.
- Test and debug: Use browser developer tools to identify and fix common HTML, CSS, and JavaScript errors.
- Build and present a project: Plan, develop, test, and explain a responsive front-end website or application.

Your 12-week learning journey

Duration: 12 weeks

Schedule: 2 sessions per week

Session length: 2 hours

Total guided learning: 48 hours across 24 sessions

MODULE 1: Web Development Foundations

Week 1: Introduction to Web Development and HTML

Session 1: Understanding the Web and Setting Up

- Understanding what websites and web applications are and how they are used.
- Exploring front-end and back-end development at a beginner level.
- Understanding browsers, servers, URLs, domains, and website hosting.
- Setting up a code editor, project folders, and browser developer tools.



- Creating a first HTML file and opening it in a browser.

Session 2: HTML Document Structure

- Understanding HTML elements, tags, attributes, and nesting.
- Creating a standard HTML document with doctype, html, head, and body elements.
- Using headings, paragraphs, line breaks, and horizontal rules appropriately.
- Adding comments and organising files and folders.
- Building a simple personal introduction page.

Week 2: HTML Content and Semantic Structure

Session 3: Links, Images, Lists, and Tables

- Creating links and navigating between pages.
- Adding images with appropriate alternative text and file paths.
- Using ordered lists, unordered lists, and nested lists.
- Creating basic tables when tabular data is appropriate.
- Organising page content so it is easy to read and navigate.

Session 4: Semantic HTML and Multi-Section Pages

- Using semantic elements such as header, nav, main, section, article, and footer.
- Understanding why meaningful HTML structure matters.
- Organising content into clear sections.
- Reviewing common HTML mistakes and validating page structure.
- Building a structured profile or information page.

Week 3: HTML Forms and Accessibility

Session 5: Building Forms

- Understanding form purpose, action, and method at an introductory level.
- Using labels, text fields, email fields, password fields, radio buttons, checkboxes, and select menus.
- Adding text areas, buttons, and helpful placeholder text.
- Using required fields and built-in browser validation attributes.
- Building a student registration or contact form.

Session 6: Accessibility and HTML Review

- Connecting labels to form controls and writing useful alternative text.
- Considering keyboard navigation, clear instructions, and readable content structure.
- Reviewing semantic HTML and accessible form practices.
- Testing pages in the browser and correcting common errors.
- Completing an HTML practical exercise.

MODULE 2: CSS Styling and Responsive Layout

Week 4: CSS Fundamentals

Session 7: Introducing CSS

- Understanding the role of CSS in the presentation of web pages.
- Writing inline, internal, and external CSS and understanding when to use external stylesheets.
- Using element, class, and ID selectors.



- Applying colours, backgrounds, borders, and basic typography.
- Understanding the cascade, inheritance, and specificity at an introductory level.

Session 8: Typography and the Box Model

- Styling text with font family, size, weight, line height, and text alignment.
- Understanding content, padding, border, and margin.
- Using width, height, max-width, and box-sizing.
- Creating consistent spacing and visual hierarchy.
- Styling the profile page created earlier in the course.

Week 5: CSS Layout and Positioning

Session 9: Display, Positioning, and Flexbox

- Understanding normal document flow and common display values.
- Using relative, absolute, fixed, and sticky positioning appropriately.
- Arranging elements with Flexbox.
- Controlling alignment, spacing, direction, and wrapping.
- Creating a navigation bar and aligned content sections.

Session 10: CSS Grid and Page Layouts

- Understanding grid columns, rows, gaps, and alignment.
- Comparing practical uses of Flexbox and CSS Grid.
- Building card layouts and multi-column sections.
- Combining layout methods to create coherent page structures.
- Building a landing page with a hero section, service cards, and footer.

Week 6: Responsive Design and HTML/CSS Project

Session 11: Responsive and Mobile-First Design

- Understanding responsive design and the mobile-first approach.
- Using viewport settings, flexible sizing, and media queries.
- Making images and layouts adapt to different screen sizes.
- Testing pages at mobile, tablet, and desktop widths.
- Identifying layout issues and improving readability across devices.

Session 12: HTML/CSS Project and Review

- Planning a small business or community website.
- Combining semantic HTML, CSS styling, Flexbox, Grid, and media queries.
- Reviewing navigation, spacing, typography, and visual consistency.
- Testing basic accessibility and responsive behaviour.
- Presenting a working website and receiving constructive feedback.

MODULE 3: JavaScript Programming Foundations

Week 7: JavaScript Basics and Decision-Making

Session 13: Introducing JavaScript

- Understanding the role of JavaScript in interactive websites.

- Adding JavaScript to a page and using the browser console.



- Declaring variables with let and const.
- Working with strings, numbers, booleans, and basic operators.
- Displaying output and inspecting errors in developer tools.

Session 14: Conditions and Logical Thinking

- Using comparison and logical operators.
- Writing if, else if, else, and switch statements.
- Understanding truthy and falsy values at a beginner level.
- Converting input values and avoiding common type-related mistakes.
- Building a grade calculator or simple decision-based program.

Week 8: Functions, Loops, Arrays, and Objects

Session 15: Functions and Reusable Code

- Understanding function declarations, parameters, arguments, and return values.
- Writing arrow functions and calling functions with different inputs.
- Understanding local and global scope at an introductory level.
- Breaking a problem into smaller, reusable steps.
- Practising with calculation and conversion functions.

Session 16: Loops and Data Structures

- Using for and while loops to repeat tasks.
- Creating and accessing arrays and objects.
- Working with common array methods such as push, includes, and forEach.
- Reading and updating object properties.
- Building a simple program that processes a list of student scores.

MODULE 4: Interactive Web Pages

Week 9: DOM Manipulation and Events

Session 17: Selecting and Updating Page Elements

- Understanding the Document Object Model (DOM).
- Selecting elements with querySelector and querySelectorAll.
- Changing text, attributes, and CSS classes with JavaScript.
- Creating, appending, and removing HTML elements.
- Building a page that updates content based on user input.

Session 18: Events and Interactive Features

- Understanding event listeners and common browser events.
- Responding to clicks, keyboard input, and form submissions.
- Reading event information and preventing default form behaviour when needed.
- Building interactive buttons, counters, menus, or simple filters.
- Creating the first version of an interactive to-do list.

Week 10: Forms, Validation, and Local Storage

Session 19: Form Handling and Validation

- Retrieving values from form controls with JavaScript



- Validating required input and checking simple conditions.
- Displaying useful error and success messages.
- Handling empty values and invalid input without confusing the user.
- Building a form that processes and displays submitted information.

Session 20: Saving Data in the Browser

- Understanding the purpose and limitations of localStorage.
- Saving and retrieving simple application data.
- Converting data to and from JSON with JSON.stringify and JSON.parse.
- Updating the interface when saved data changes.
- Extending the to-do list so tasks remain after a page refresh.

MODULE 5: Integrated Project Development

Week 11: Planning, Building, and Testing an Application

Session 21: Project Planning and Interface Design

- Choosing a practical project and identifying its intended users.
- Writing a concise feature list and defining what the project must do.
- Sketching a page layout or simple wireframe.
- Planning HTML structure, CSS layout, and JavaScript behaviour.
- Setting achievable milestones for the final project.

Session 22: Implementation and Debugging

- Building the page structure and styling the main interface.
- Implementing interactive features with JavaScript and DOM events.
- Adding form validation or local storage where appropriate.
- Using browser developer tools to diagnose and fix errors.
- Reviewing code organisation, naming, and readability.

MODULE 6: Final Project and Demonstration

Week 12: Testing, Presentation, and Continued Development

Session 23: Final Testing and Refinement

- Testing core features and correcting identified bugs.
- Checking layout and usability on different screen sizes.
- Reviewing semantic structure, accessibility basics, and form feedback.
- Cleaning up code and preparing a short project explanation.
- Preparing a project demonstration and, where feasible, publishing a static site.

Session 24: Final Project Presentations and Evaluation

- Demonstrating a completed website or interactive front-end application.
- Explaining the purpose, structure, styling, and JavaScript features of the project.
- Answering questions and responding to constructive feedback.
- Reflecting on strengths and identifying skills that need more practice.
- Creating a personal plan for continued web development learning.



Throughout the programme, learners will write code, test ideas, solve small problems, and improve projects through feedback. Activities progress from short HTML exercises to styled layouts, JavaScript programming tasks, DOM-based interactions, and a complete final application. Students should practise between sessions so that each new topic builds on skills already learned.

How you'll be assessed

Progress will be evaluated through practical exercises, milestone projects, code explanations, and a final demonstration. Assessment focuses on:

- HTML structure and semantics: Clear, well-organised, meaningful page structure.
- CSS styling and responsiveness: Consistent visual design and layouts that adapt to screen sizes.
- JavaScript logic and functionality: Correct use of programming concepts and working interactive features.
- Problem-solving and debugging: Ability to identify errors, test solutions, and explain corrections.
- Usability and accessibility: Readable content, understandable controls, and basic accessibility practices.
- Code organisation: Readable naming, sensible file structure, and maintainable code.
- Project explanation: Ability to demonstrate features and explain how the project works.

Suggested assessment structure

- Weekly exercises and homework — 15%
- HTML practical assessment — 10%
- HTML/CSS website project — 20%
- JavaScript fundamentals test — 15%
- Integrated final project — 30%
- Presentation and code explanation — 10%

What to expect

- Twelve weeks of structured learning and 24 guided sessions.
- Forty-eight hours of classroom instruction, plus independent practice.
- Hands-on work with HTML, CSS, and JavaScript.
- Responsive website design, interactive page development, and debugging.
- Milestone exercises and constructive feedback.
- A final project that can be demonstrated in a portfolio.
- A personal plan for continued learning after the course.

Build the skills to bring ideas to the web.

By the end of this programme, learners should be able to create small, responsive, interactive front-end websites and explain the code behind them. Professional-level mastery requires continued practice and further projects, but this course provides a structured foundation for continued study and development.

SkillBridge Academy: Building practical digital skills for learning, work, and everyday problem-solving.

