

HTML and CSS Tutorial: Build Responsive Websites

A hands-on guide for beginners to learn web development with HTML, CSS, Flexbox, animations, and SVGs from scratch.

Web & Mobile Design

Instructor: Morgan Blake • 5 sections • 56 lessons • 17.00h total

[View this course online](#)

[Browse all courses](#)

About this course

Many aspiring web developers feel overwhelmed by the technologies needed to create modern websites. HTML and CSS are the core building blocks, but understanding how they work together can be tricky without clear guidance. This course demystifies the process, starting from the ground up.

You'll learn how to write semantic HTML, style pages with CSS, master layouts using Flexbox and Grid, add smooth transitions and animations, integrate scalable SVG graphics, and build a fully responsive portfolio site. Each module includes video lessons with real-world examples and hands-on projects to apply your skills immediately.

Course curriculum

Web Fundamentals & Core HTML

- **How Websites Work: Clients, Servers, and Browsers (15m)**
We'll cover the basic flow of how a webpage gets from a server to your screen, so you understand the technology behind every site you build.
- **Setting Up Your Development Environment (12m)**
Get your tools ready with a step-by-step guide to installing VS Code and configuring browsers for efficient web development.
- **Introduction to HTML and Semantic Markup (18m)**
Learn to write clean, meaningful HTML that improves accessibility and SEO, using tags that describe your content's purpose.
- **Working with Headings and Paragraphs (15m)**
Structure text content properly with headings and paragraphs to create a clear document outline for better readability.
- **Creating Lists and Links (15m)**
Build navigable content with unordered, ordered lists, and hyperlinks, including tips for effective link text.
- **Adding Images and Media (15m)**
Insert images and other media with proper attributes for alt text and responsive sizing, enhancing user experience.
- **Building a Simple Profile Page (25m)**
Apply your HTML skills in a mini-project to create a well-structured personal profile page from scratch.

Styling with CSS & Modern Layouts

- **CSS Basics: Selectors and Properties (18m)**
Start styling with CSS by learning how selectors target HTML elements and how properties define their appearance.
- **Applying Colors and Fonts (15m)**
Choose and apply colors and web-safe fonts to set the visual tone of your pages, including using modern color formats.
- **Understanding the CSS Box Model (20m)**
Break down how every element is a box with content, padding, border, and margin, and why this concept is key to layout control.
- **Margin, Border, Padding, and Content (18m)**
Dive into each part of the box model with practical examples to space and style elements precisely.
- **Introduction to Flexbox: Flex Containers and Items (22m)**
Create flexible layouts by understanding flex containers and items, with a focus on the main axis and cross axis.

- **Flexbox Properties: Alignment and Direction (18m)**
Control how items are aligned and distributed using justify-content and align-items for responsive designs.
- **Creating Responsive Layouts with Flexbox (20m)**
Make your layouts adapt to different screen sizes using Flexbox properties like wrap and grow.
- **Styling Your Profile Page (25m)**
Mini-project: Apply CSS to your profile page from Section 1, adding colors, fonts, and a flexible layout structure.
- **Linking CSS Files and Inline Styles (12m)**
Organize your styles by linking external CSS files and use inline styles sparingly for specific overrides.
- **CSS Units and Values (15m)**
Choose the right units for fonts, spacing, and sizing with an overview of pixels, percentages, and more.
- **Flexbox Wrap and Order (15m)**
Handle content overflow with flex-wrap and reorder items visually without changing HTML structure.
- **Mini-Project: Implement Flexbox Layout (30m)**
Build a responsive card-based layout using Flexbox, applying alignment and wrapping techniques in a real scenario.

Mastering CSS Specificity & The Cascade

- **The CSS Cascade: How Styles are Applied (18m)**
Understand the order in which CSS rules are resolved, from browser defaults to author styles, to avoid unexpected results.
- **Inheritance and the Universal Selector (15m)**
Learn which properties are inherited by child elements and how the universal selector can simplify your code.
- **Specificity: ID, Class, Element Selectors (20m)**
Calculate specificity scores to predict which styles will win when conflicts arise, using ID, class, and element selectors.
- **Calculating Specificity Scores (15m)**
Practice with real examples to compute specificity values, making you confident in debugging style issues.
- **Debugging CSS Conflicts (18m)**
Use strategies to identify and fix style conflicts, such as checking specificity and cascade order.
- **Using Browser Developer Tools (20m)**
Inspect and edit live CSS with browser tools to test changes and troubleshoot layout problems efficiently.
- **Modern CSS Units: rem, em, vw, vh (15m)**
Explore relative units like rem and viewport units for more scalable and responsive typography and spacing.
- **Relative vs Absolute Units (12m)**
Compare when to use relative units for flexibility and absolute units for precision in your designs.

- **Refactoring CSS for Maintainability (18m)**
Reorganize your CSS code to be cleaner and easier to maintain, with tips on naming conventions and structure.
- **Mini-Project: CSS Refactoring Challenge (30m)**
Take a messy CSS file from a multi-element page and refactor it using specificity knowledge and modern units.

Dynamic Interfaces with Transitions & Animations

- **Introduction to CSS Transitions (15m)**
Add smooth changes between CSS states, like hover effects, to make your interfaces feel more interactive.
- **Transition Properties: duration, timing, delay (18m)**
Control the speed and easing of transitions with properties for duration, timing functions, and delay.
- **Creating Hover Effects (15m)**
Design engaging hover effects for buttons and links that provide visual feedback to users.
- **State Changes with Transitions (15m)**
Apply transitions to different states, such as focus and active, for a cohesive user experience.
- **Introduction to CSS Animations (18m)**
Move beyond transitions to create complex, multi-step animations using keyframes.
- **Defining Keyframes (20m)**
Write keyframe rules to define the start, end, and intermediate points of your animations.
- **Animation Properties: iteration, direction, fill-mode (18m)**
Fine-tune animations with properties for looping, direction, and how they behave before and after playing.
- **Combining Transitions and Animations (15m)**
Layer transitions and animations to create dynamic effects that respond to user interactions.
- **Working with SVGs: Inline SVG Basics (18m)**
Embed scalable vector graphics directly in your HTML for crisp icons and illustrations that load fast.
- **Styling SVGs with CSS (15m)**
Apply CSS properties to change SVG colors, sizes, and styles, making them integrate with your design.
- **Creating Custom SVG Icons (20m)**
Design your own simple SVG icons from scratch or edit existing ones to match your site's branding.
- **SVG Animations (18m)**
Animate SVG elements with CSS to create loading spinners, interactive graphics, and more.
- **Mini-Project: Animated Portfolio Elements (30m)**
Build interactive elements like animated buttons and custom SVG icons for a portfolio page.

Integrated Project & Advanced Concepts

- **Planning a Multi-Page Website (15m)**
Sketch out the structure and content for a multi-page site, focusing on user flow and navigation.
- **Structuring HTML for Multiple Pages (18m)**
Use semantic HTML consistently across pages with headers, footers, and main content areas.

- **Linking Pages and Navigation (15m)**
Create clear navigation menus and internal links to connect your pages smoothly.
- **Responsive Design Principles (18m)**
Learn the core ideas behind responsive design, like fluid grids and flexible images, for all devices.
- **Media Queries for Breakpoints (20m)**
Use media queries to apply different styles at specific screen widths, adapting layouts for mobile and desktop.
- **Introduction to CSS Grid (22m)**
Handle complex two-dimensional layouts with CSS Grid, defining rows and columns for precise control.
- **Grid Templates and Areas (20m)**
Define grid templates and named areas to create magazine-style layouts with ease.
- **Advanced Layouts with Grid (18m)**
Combine Grid with Flexbox for hybrid layouts that handle both rows and columns efficiently.
- **Organizing CSS with Modules (15m)**
Break your CSS into modular files for better organization, making large projects easier to manage.
- **Accessibility Best Practices (18m)**
Ensure your website is usable by everyone with proper contrast, keyboard navigation, and ARIA labels.
- **Performance Optimization (15m)**
Optimize images, minify CSS, and use efficient selectors to keep your site fast and lightweight.
- **Final Project: Build Portfolio Website (32m)**
Apply all your skills to create a complete, animated portfolio site with multiple pages and custom SVG artwork.
- **Testing and Debugging (18m)**
Test your site across browsers and devices, and debug common issues to ensure a consistent experience.
- **Deployment and Next Steps (15m)**
Learn how to deploy your site to a live server and explore paths for further learning in web development.

Discover our blog

Learn more with free articles written by our professors and guest experts.

Latest article: **Flight Attendant Preparation: What to Study Before You Apply**

[Visit the blog](#) [All courses](#) [This course](#)