Faysal Aziz

# JavaScript and CSS

## Build a responsive, interactive webpage with HTML foundations

FA-0635 · 2 days

Software Development

Build and refine one small webpage using HTML, CSS, JavaScript, form validation and a prepared Fetch API example.

---

## Introduction

This two-day course introduces HTML structure, CSS styling and JavaScript interactivity through one small webpage developed across the sessions. It is suitable for learners beginning front-end development who are comfortable with basic computer tasks.

Participants practise selected layout techniques, DOM events, forms and data retrieval. More advanced CSS features are demonstrations or short examples rather than mastery targets; the project is a learning prototype, not a complete production web application.

## Learning outcomes

The course teaches participants to:

- Create and style a simple HTML page.
- Use variables, functions, conditions and loops in small JavaScript examples.
- Apply the box model and compare basic responsive layout techniques.
- Select and modify DOM elements and respond to user events.
- Use HTML validation and JavaScript feedback for a simple form.
- Fetch sample data and handle response status, loading and failure states.
- Integrate and troubleshoot the page’s styling and interactivity.

# Prerequisites

- Basic computer, file and browser skills; no prior JavaScript is required.
- A prepared editor, current browser and local web-serving environment.
- Access to the supplied demonstration API or equivalent prepared sample data.

# Training outline

## 1. Day 1 — HTML, CSS and JavaScript Foundations

### Introduction — 30 minutes

- Course objectives
- Importance of HTML, CSS, and JavaScript in web development

### HTML Basics — 1 hour

- What is HTML?
- Basic structure of an HTML document
- Common HTML elements (headings, paragraphs, links, images, lists)
- Hands-on: Create a simple HTML page

### CSS Introduction — 1.5 hours

- What is CSS?
- Inline, internal, and external CSS
- CSS selectors (element, class, and ID selectors)
- Basic styling properties (color, background-color, font-size, font-family)
- Hands-on: Style the HTML page created earlier

### Box Model and Layout — 1.5 hours

- Understanding the box model
- Margin, padding, border, and content
- Display property (block, inline, inline-block)
- Floats and clears for text wrapping and recognition of legacy layouts; use Flexbox/Grid examples for contemporary page layout.

- Hands-on: Create a simple page layout using CSS

## JavaScript Introduction — 2 hours

- What is JavaScript?
- Basic syntax and data types
- Variables, constants, and operators
- Conditional statements and loops
- Functions
- Hands-on: Add interactivity to the HTML page with JavaScript

# 2. Day 2 — Layout, DOM, Forms and Data

## Further CSS Techniques — 1.5 hours

- CSS positioning (static, relative, absolute, fixed, sticky)
- Pseudo-classes and pseudo-elements
- Selected transitions and animations, with restrained use.
- Introductory comparison of Flexbox and Grid through prepared examples.
- Responsive design with media queries
- Enhance the same mini-project's layout and responsiveness.

## DOM and Events — 1.5 hours

- What is the DOM?
- Selecting and manipulating HTML elements with JavaScript
- Adding and removing elements
- Event handling, including appropriate click and keyboard interactions.
- Hands-on: Add DOM manipulation to the HTML page

## Forms and Validation — 1 hour

- HTML forms
- Accessing form data with JavaScript
- Start with built-in HTML constraints and enhance feedback with JavaScript; client-side checks do not replace server-side validation.
- Create a sample contact form using fictional inputs; do not collect real personal information.

## AJAX and Fetch — 1 hour

- What is AJAX?
- Recognise XMLHttpRequest in existing code; focus new exercises on Fetch.
- Fetch API
- Check HTTP response status and handle network/parse failures; recognise cross-origin restrictions.
- Fetch prepared sample data and display it with loading and error feedback.

## Project Review — 1 hour

- Recap of key concepts covered
- Identify suitable next learning steps rather than reproducing an unnecessary reference list.
- Q&A and troubleshooting session
- Review the mini-project and collect feedback.

---

© 2026 Faysal Aziz · Kuala Lumpur