

Introduction to Web Development

Introduction to Web Development
Quick reference guide for beginners

1. What Is Web Development?

Aspect	Description
Definition	The process of building, testing, and maintaining websites or web applications that run on the Internet or an intranet.
Goal	Deliver functional, performant, and user-friendly experiences across devices (desktop, tablet, mobile).
Main Areas	Frontend (client-side) – what users see. Backend (server-side) – data processing, storage, and business logic. Fullstack – combination of both.

2. Core Technologies

Layer	Primary Languages / Tools	Typical Tasks
HTML	Markup language (HTML5)	Structure content: headings, paragraphs, links, tables, forms, semantic tags (<code><article></code> , <code><nav></code> , etc.).
CSS	Styling language (CSS3)	Visual design: layout (Flexbox, Grid), colors, typography, animations, responsive design (media queries).
JavaScript	Scripting language (ES6+)	Interactivity: DOM manipulation, event handling, fetch API, client-side validation, SPA frameworks.
Server-Side	Node.js, Python (Django/Flask), Ruby (Rails), PHP, Java (Spring), .NET, Go	Process requests, business logic, authentication, API endpoints.
Databases	Relational (MySQL, PostgreSQL) • NoSQL (MongoDB, Redis)	Store, query, and manage persistent data.
Version Control	Git (GitHub, GitLab, Bitbucket)	Track changes, collaborate, manage releases.

Introduction to Web Development

Layer	Primary Languages / Tools	Typical Tasks
Build Tools	npm/Yarn, Webpack, Vite, Parcel, Gulp	Bundle assets, transpile (Babel, TypeScript), optimize for production.
Deployment	Cloud platforms (Vercel, Netlify, AWS, Azure, GCP) • Containers (Docker) • CI/CD pipelines	Host sites, automate testing and release.

3. Front?End Foundations

1. HTML Basics

- Doctype (`<!DOCTYPE html>`)
- `<head>` vs. `<body>`
- Semantic elements (`<header>`, `<section>`, `<footer>`)
- Forms & input types (`<input type="email">`, `<select>`)

2. CSS Essentials

- Selectors (class, id, attribute, pseudo?classes)
- Box model (margin, border, padding, content)
- Layout modules: **Flexbox** (one?dimensional) & **CSS Grid** (two?dimensional)
- Responsive design: fluid units (% , vh/vw), media queries, mobile?first approach

3. JavaScript Core

- Variables (`let`, `const`), data types, arrow functions
- DOM API (`document.querySelector`, `addEventListener`)
- Asynchronous patterns: **Promises**, **async/await**, `fetch`
- Module system (`import / export`)

4. Popular Front?End Frameworks

- **React** – component?based, JSX, virtual DOM.
 - **Vue** – template?driven, reactivity system.
 - **Angular** – full?featured, TypeScript?centric, MVC.
-

Introduction to Web Development

4. Backend Basics

Concept Typical Implementation