

Day (Classes)	TOPICS	PARTICULARS	LEARNING OBJECTIVES	LEVEL
1 - 2	Intro to Web Dev	HTML overview, understanding the Web Browser, Basic HTML document structure (<!DOCTYPE>, <html>, <head>, <body>), and creating your first HTML page.	Student will understand the core architecture of the web and basic document structure.	Basic
2 - 3	HTML Tags & Text	Basic HTML tags: paragraphs (<p>), headings (<h1> through <h6>), horizontal lines (<hr>), text formatting (, <i>, <u>), and using comments for code organization.	Student will be able to structure text-heavy content correctly and use formatting tags.	Basic
3 - 4	PROJECT 1: Personal Bio	Build: A single-page professional profile. Particulars: Semantic structure, multiple headings, text formatting, and code comments [user request].	Student will apply basic tags to create a meaningful digital identity.	Basic (Build)
3 - 4	Multimedia & Data	HTML lists (ordered and unordered), embedding images () with attributes (src, alt, width/height), creating hyperlinks (<a>), and building data tables (<table>, <tr>, <td>).	Student will be able to work with HTML lists, links, tables and embed images into documents.	Basic
3 - 4	PROJECT 2: Digital Menu	Build: A restaurant menu. Particulars: Using nested tables for layout, structured lists for menu items, and optimized images [user request, 139].	Student will master data organization and advanced structured content display.	Basic (Build)
3 - 4	CSS Foundations	Style rules, simple selectors (element), properties (color, background-color), and the power of id and class selectors to target specific elements.	Student will understand how to separate content from presentation using basic CSS rules.	Intermediate
4 - 5	Layout & Box Model	Box Model mastery: Padding, Border, and Margin. Controlling element dimensions (width/height), float, position (absolute/relative), and z-index.	Student will use the box model to design layouts that align precisely on a screen.	Intermediate
4 - 5	PROJECT 3: Landing Page	Build: A professional product landing page template. Particulars: Multi-column layout using floats, stylized headers, and precise box-model spacing [user request, 118, 141].	Student will apply complex layout logic to modern UI design.	Intermediate (Build)
4 - 5	Advanced Styling	Fonts (family, weight), inheritance, CSS multiple classes, descendant selectors, pseudo-classes (e.g., :hover), and CSS specificity.	Student will have more leverage working with advanced selectors to design interactive UI elements.	Intermediate
3 - 4	PROJECT 4: Interactive Gallery	Build: A responsive image gallery. Particulars: Hover effects, absolute positioning for captions, and grouped styling for a uniform look [user request, 118, 141].	Student will master advanced selectors and interactive CSS visual effects.	Intermediate (Build)
Row	SEPARATOR	ADVANCED & PROFESSIONAL LEVEL	JAVASCRIPT & DYNAMIC WEB APPS	---

5 - 6	JavaScript Basics	JS Syntax, variables (let, const), data types (strings, numbers, booleans), basic math operators, and using the Developer Console for debugging [user request].	Student will move from static design to logic-based programming.	Advanced
4 - 5	PROJECT 5: Smart Calculator	Build: A web-based calculator. Particulars: Performing math logic using JS variables and functions, and displaying results in an HTML element [user request].	Student will apply programming logic to create a functional, interactive web tool.	Advanced (Build)
6 - 8	DOM Manipulation	Connecting JS to HTML: Selecting elements (getElementById), changing content/styles via code, and Event Listeners (click, input) to trigger actions [user request].	Student will learn to make web pages that update dynamically without refreshing.	Professional
6 - 8	PROJECT 6: Dynamic To-Do App	Build: A task manager. Particulars: Creating elements dynamically, deleting tasks, and marking items as complete using DOM manipulation [user request].	Student will build a functional CRUD (Create, Read, Update, Delete) application.	Professional (Build)
6 - 8	Logic & API Integration	Complex conditionals, loops in JS, fetching data from Web APIs (using the fetch API), and JSON parsing for real-world data handling [user request].	Student will build "smart" apps that communicate with external servers to fetch live data.	Professional
6 - 8	PROJECT 7: Real-Time Weather	Build: A weather dashboard. Particulars: Fetching live data for any city, parsing JSON, and updating the UI dynamically with images and temperature [user request].	Student will master asynchronous programming and live data handling in a professional app.	Professional (Build)
8 - 12	PROJECT 8: Full Capstone	Build: An independent, professional-grade site (e.g., a community portal or game) optimized for all screens with a custom JavaScript engine.	Student will follow an Agile process to design, build, and Showcase a professional software artifact.	Final Showcase