

HTML	Basic Structure of an HTML Page HTML defines the structure and meaning of a web page. “ What contents exist “ “ How to arrange it”
	CSS :- It controls Presentation and layout of web content “ How content looks across devices and screen size“ - Responsive web design
	Javascript JS adds behavior and interactivity to web pages. “ How page should respond to user action”
	Separate duties for content, Design and its behavior Enable scalability, maintainability and UI development
	React/Angular/Vue _ Front End framework
	Various Tags and Attributes
	Inline and Block Elements
	Forms
	Semantic elements
	HTML 5 Features
	Paper based Layout
	Hero unit
	Audio Video Players
	Semantic Markup
	Geo Location, A11Y (Accessibility)

css	Intro to CSS
	CSS Properties and their syntaxes
	Various Selectors in CSS and their preference orders
	Inline , Internal and External Styling
	Box Model
	CSS 3
	Advance Units like rem, em and vh vw
	New elements added like Round Corners
	Working with flex box
	Working with Bootstrap

Bootstrap

	Develop a static webpage,Forms and Form Validations
	Navs and Navbar,Pagination and Progress
	Navigation and Pagination

Basics of JavaScript	Basics of JavaScript - Introduction to JavaScript, Data type, variables, Functions and scope Type Conversions, Basic Operators Branching, Conditional statements and Looping Statements, Functions, Arrays and Strings
Advanced Java Script	Modern JavaScript (ES6): Javascript - Fetch API , Arrow functions, Template literals, Destructuring, Spread and rest operators Date and Time Objects, Closures, Promises and Async Programming, Closures and Scoping, DOM Manipulation: DOM Objects, Accessing elements in the DOM, Modifying HTML content, Adding and removing elements Functional Programming, Error Handling client side scripting JSON, JSON - Objects, arrays etc, JSON-Ajax Advanced DOM Manipulation, jQuery Plugins, Event Delegation, Ajax and Deferred Objects, Optimization and Performance

DEmo 1: Create a portfolio website : Digital resume

DEmo 2: Blog website/Ecommerce Website (Image gallery , RWD, Navigation etc)

Demo 3: Calculator , To Do list

Creating a registration form

Creating SPA: Single page application

Working with NPM(Node package manager)

Typescript

Feature	Explanation & Industry Usage
Structured Markup (HTML)	Defines headings, forms, media, and layout used in all web apps
Styling & Layout (CSS)	Controls branding, responsiveness, and UI consistency
Client-Side Interactivity (JS)	Handles clicks, form validation, dynamic updates

Semantic Elements	Improve SEO, accessibility, and maintainability
Responsive Design	Adapts UI across mobile, tablet, desktop
Media & Rich Content	Audio, video, and animations
Standards-Based	Browser-supported, framework-independent

Typical Execution Flow of web page :

Step 1: Browser loads HTML file

Step 2: HTML defines structure and semantic meaning

Step 3: CSS applies layout, colors, spacing, responsiveness

Step 4: JavaScript executes logic and event handling

Step 5: User interactions trigger JS functions

Step 6: DOM updates reflect changes dynamically

Layer	Responsibility
HTML	Structure & semantics
CSS	Layout, responsiveness, design
JavaScript	Logic, validation, interaction

Case Study: Frontend Web Development Using HTML, CSS & JavaScript

Demo–1: Portfolio Website (Digital Resume)

User Stories – Demo 1

Story ID	User Role	User Story	Acceptance Criteria	Technology
US-1.1	Visitor	As a visitor, I want to view a personal introduction so that I understand the candidate's profile	Name, role, short bio visible	HTML
US-1.2	Visitor	As a visitor, I want to see education and skills details so that I can assess qualifications	Sections clearly structured	HTML
US-1.3	Visitor	As a visitor, I want the website to look professional and readable	Consistent fonts, colors, layout	CSS
US-1.4	Visitor	As a visitor, I want the site to be responsive so I can view it on mobile devices	Layout adapts on screen resize	CSS (RWD)
US-1.5	User	As a user, I want navigation links so I can jump to sections easily	Smooth navigation working	HTML + CSS
US-1.6	User	As a user, I want a contact section so I can reach out	Contact form visible	HTML
US-1.7	User	As a user, I want basic form validation so incorrect inputs are restricted	Mandatory fields enforced	JavaScript

Demo–2: Blog Website / E-Commerce Website

User Stories – Demo 2

Story ID	User Role	User Story	Acceptance Criteria	Technology
US-2.1	Visitor	As a visitor, I want to view blog posts / products so I can explore content	Multiple items displayed	HTML
US-2.2	Visitor	As a visitor, I want to see images in a gallery format	Images aligned and responsive	HTML + CSS
US-2.3	Visitor	As a visitor, I want a navigation menu to move across pages	Navigation links functional	HTML + CSS
US-2.4	Visitor	As a visitor, I want the site to be mobile responsive	Layout adjusts automatically	CSS (RWD)

US-2.5	User	As a user, I want hover effects for better interaction	Hover states visible	CSS
US-2.6	User	As a user, I want to filter or view categories	Category-wise layout	HTML + CSS
US-2.7	Admin (UI)	As an admin, I want a structured layout to add future content	Reusable components	HTML

Demo–3: JavaScript Applications (Calculator & To-Do List)

User Stories – Demo 3 (Calculator)

Story ID	User Role	User Story	Acceptance Criteria	Technology
US-3.1	User	As a user, I want to enter numbers using buttons	Buttons capture input	HTML
US-3.2	User	As a user, I want to perform basic arithmetic operations	Correct calculation shown	JavaScript
US-3.3	User	As a user, I want a clear/reset option	Display resets correctly	JavaScript
US-3.4	User	As a user, I want a clean UI for calculator	Proper alignment and spacing	CSS

User Stories – Demo 3 (To-Do List)

Story ID	User Role	User Story	Acceptance Criteria	Technology
US-3.5	User	As a user, I want to add tasks to a list	Task appears in list	JavaScript
US-3.6	User	As a user, I want to delete completed tasks	Task removed from list	JavaScript
US-3.7	User	As a user, I want tasks to persist during session	Tasks remain until refresh	JavaScript
US-3.8	User	As a user, I want visual feedback for completed tasks	Completed task styled	CSS + JS

GitHub Upload Instructions (For Students)

Repository Structure (Mandatory)

Folder/File	Purpose
/portfolio	Demo-1 project
/blog-ecommerce	Demo-2 project
/js-apps	Calculator & To-Do
index.html	Main HTML file
/css/style.css	Styling
/js/script.js	JavaScript logic
README.md	Project description

Steps to Upload on GitHub

Step	Instruction
1	Create an account on GitHub
2	Click New Repository
3	Name repository (example: frontend-html-css-js-demos)
4	Set visibility to Public
5	Upload folders using Upload files option
6	Add README.md describing demos
7	Commit changes
8	Share repository link for evaluation

README.md – Mandatory Content

Section	Description
Project Title	Frontend Web Development Demos
Technologies	HTML, CSS, JavaScript
Demo List	Portfolio, Blog/E-commerce, Calculator, To-Do
Learning Outcome	UI design, responsiveness, DOM manipulation
Author	Student name