📌 **1. Introduction to Web Development**

- Overview of web development
- Understanding client-server architecture
- Basics of HTML, CSS, and JavaScript

📌 **2. HTML Fundamentals**

- Structure of HTML documents
- HTML elements and tags
- Semantic HTML
- Forms and input elements

📌 **3. CSS Fundamentals**

- Styling HTML elements
- CSS selectors and specificity
- Box model and layout techniques
- Responsive design principles

📌 **4. JavaScript**

- Introduction to JavaScript
- Variables, data types, and operators
- Control flow and loops
- Functions and scope
- Arrays and objects

📌 **5. Version Control with Git and GitHub**

- Introduction to version control
- Setting up Git and GitHub accounts
- Basic Git commands (`init`, `add`, `commit`, `push`, `pull`, `clone`)

- Collaborative development with GitHub (`fork`, `branch`, `pull request`)

📌 **6. Introduction to Tailwind CSS**

- Overview of Tailwind CSS
- Setting up Tailwind CSS in a project
- Utility-first CSS approach
- Styling with Tailwind CSS classes

📌 **7. Introduction to React.js**

- Introduction to React.js and its benefits
- Setting up a React.js development environment
- Components and JSX syntax
- State management with `useState`

📌 **8. Advanced React.js**

- React Hooks (`useState`, `useEffect`, `useContext`, etc.)
- React Router for navigation
- Forms and form handling in React
- Context API for state management

📌 **9. Introduction to Next.js**

- Overview of Next.js and its features
- Setting up a Next.js project
- Pages and routing in Next.js
- Static Site Generation (SSG) and Server-Side Rendering (SSR)

📌 **10. Introduction to Node.js & Express.js**

- What is Node.js? Understanding the runtime
- Setting up a Node.js project
- Building a basic Express.js server
- REST API development (CRUD operations)
- Middleware and error handling

📌 **11. Building Full-Stack Applications with Next.js and Node.js**

- Integrating Next.js frontend with Node.js backend
- API routes in Next.js
- Connecting to databases (MongoDB, PostgreSQL)
- Authentication and authorization (JWT, bcrypt)

📌 **12. Deployment and Project Work**

- Deployment strategies for full-stack applications
- Deploying frontend on Vercel and backend on a cloud platform (Railway, Render)
- Building projects and portfolio development
- Collaborative project work to simulate real-world development environments