JavaScript Roadmap for Web Development

Unit 1: JavaScript Basics

1. Introduction to JavaScript
2. Setting up the Environment (Browser, Node.js)
3. Variables and Data Types (var, let, const, primitive & reference types)
4. Operators and Expressions
5. Conditional Statements (if-else, switch)
6. Loops (for, while, do-while)
7. Functions
   - Function Declaration vs Expression
   - Arrow Functions
   - Function Parameters & Return Values

Unit 2: DOM Manipulation & Events

1. Understanding the DOM (Document Object Model)
2. Selecting Elements (getElementById, querySelector, etc.)
3. Modifying Elements (text, styles, attributes)
4. Event Handling (onclick, addEventListener)
5. Event Bubbling & Delegation
6. Form Validation with JavaScript

Unit 3: Advanced JavaScript Concepts

1. Scope and Hoisting
2. Closures and Lexical Scope
3. The 'this' keyword

4. Prototype and Prototypal Inheritance

5. Higher-Order Functions & Callbacks

6. ES6+ Features:

- Template Literals
- Destructuring
- Spread and Rest Operators
- Default Parameters
- Modules (import/export)

Unit 4: Asynchronous JavaScript

1. Callbacks and Callback Hell

2. Promises (resolve, reject, then, catch)

3. Async/Await

4. The Event Loop & Concurrency Model

5. Fetch API for HTTP requests

6. Error Handling (try-catch)

Unit 5: Object-Oriented Programming (OOP) in JavaScript

1. Objects and Object Literals

2. Constructor Functions

3. Classes and Inheritance

4. Encapsulation and Abstraction

5. Getters and Setters

6. The 'new' keyword and 'instanceof'

Unit 6: JavaScript in the Browser

1. Local Storage, Session Storage, and Cookies

2. Working with Forms and Validations

3. BOM (Browser Object Model)

4. Geolocation and Browser Events

5. Fetching and Displaying Data from APIs

Unit 7: Working with JSON & AJAX

1. What is JSON?

2. Parsing JSON data

3. AJAX with XMLHttpRequest

4. Modern AJAX with fetch()

5. Handling API responses and errors

Unit 8: Error Handling and Debugging

1. Common JavaScript Errors and Fixes

2. Debugging with Browser DevTools

3. console.log vs debugger

4. Writing Clean and Maintainable Code

5. Performance Optimization Tips

Unit 9: Essential Tools & Best Practices

1. ESLint for Code Quality

2. Prettier for Code Formatting

3. Version Control with Git & GitHub

4. Deployment Basics (Netlify, Vercel)

5. Writing Reusable Functions and Code

Unit 10: Introduction to Frameworks & Libraries

1. Why Use Frameworks?

2. Introduction to React.js Basics

3. Understanding Components and Props

4. State Management Basics

5. Setting up a Simple React Project