Full-Stack JavaScript Developer Roadmap

Timeline: September 2024 - April 2025
Daily Learning Hours: 6 Hours

Phase 1: Core JavaScript (September - October 2024)

Weeks 1-2: JavaScript Basics

- **Introduction to JavaScript:**
  - Variables: `var`, `let`, `const`
  - Data Types: Primitive & Non-Primitive
  - Type Coercion and Conversion
  - Operators: Arithmetic, Comparison, Logical, Bitwise
  - Control Flow: `if`, `else`, `switch`
  - Loops: `for`, `while`, `do-while`
  - Functions: Declarations, Expressions, Arrow Functions
- **Projects**:
  - Temperature Converter
  - Simple Calculator

Weeks 3-4: Intermediate JavaScript

- **Objects and Arrays:**
  - Creating and Manipulating Objects and Arrays
  - Array Methods: `map`, `filter`, `reduce`
  - Deep vs Shallow Copy
- **Prototypes & Inheritance:**
  - Prototypal Inheritance
  - ES6 Classes
  - Object-Oriented Programming Concepts
- **Error Handling:**
  - `try...catch...finally`
- **Projects**:
  - To-Do List App
  - Custom Array Methods

Weeks 5-6: Advanced JavaScript Concepts

- **Asynchronous JavaScript:**
  - Callbacks, Promises, and Async/Await
  - Event Loop
- **DOM Manipulation:**

- o   DOM Traversal and Event Handling
- o   Form Handling
- **Modules:**
  - o   Import/Export Syntax
- **Projects**:
  - o   Dynamic Quiz App
  - o   DOM-based Calculator

Phase 2: Data Structures & Algorithms (November - January 2025)

Weeks 1-2: Arrays & Strings

- **Data Structures Introduction**:
  - o   Time Complexity, Big-O Notation
- **Array Operations**:
  - o   Traversing, Searching, Sorting (Bubble, Selection, Insertion)
  - o   Two-Pointer, Sliding Window Techniques
- **String Manipulation**:
  - o   Substrings, Palindromes, Anagrams
  - o   String Matching (KMP Algorithm)
- **Projects**:
  - o   Anagram Checker
  - o   Palindrome Checker

Weeks 3-4: Linked Lists

- **Types of Linked Lists**:
  - o   Singly, Doubly, Circular
- **Operations**:
  - o   Insertion, Deletion, Reversing, Loop Detection
- **Projects**:
  - o   Stack with Linked Lists
  - o   Merge Two Linked Lists

Weeks 5-6: Stacks & Queues

- **Stack Operations**:
  - o   Push, Pop, Peek
  - o   Applications: Balanced Parentheses
- **Queue Operations**:
  - o   Enqueue, Dequeue, Circular Queue
- **Projects**:
  - o   Browser History with Stacks
  - o   Ticketing System with Queues

Weeks 7-8: Trees

- **Binary Trees**:
  - Inorder, Preorder, Postorder Traversals
- **Binary Search Tree (BST)**:
  - Insertion, Deletion, Searching
- **Heaps**:
  - Max/Min Heaps
- **Trie**:
  - Prefix Tree Insertion and Searching
- **Projects**:
  - Contacts Search using Trie
  - Binary Search Tree Implementation

Weeks 9-12: Graphs

- **Graph Basics**:
  - Representation: Adjacency Matrix, Adjacency List
- **Graph Traversal**:
  - Breadth-First Search (BFS), Depth-First Search (DFS)
- **Advanced Algorithms**:
  - Dijkstra's Algorithm, Kruskal's Algorithm
- **Projects**:
  - Social Network Graph
  - Pathfinding Algorithm (A*)

Phase 3: Backend JavaScript (February - April 2025)

Weeks 1-2: Node.js Basics

- **Introduction to Node.js**:
  - Node.js Environment Setup
  - Core Modules (File System, Path)
  - Event-Driven Programming
- **Projects**:
  - Build a File Reader/Writer in Node.js

Weeks 3-4: REST API Development

- **HTTP & REST**:
  - RESTful API Design Principles
  - CRUD Operations with Express.js
- **Projects**:

- o Book Store API

Weeks 5-6: Database Integration

- **Databases Overview**:
  - o SQL vs NoSQL Databases
  - o MongoDB Basics (CRUD Operations)
- **Database Design**:
  - o Schema Design with Mongoose
- **Projects**:
  - o Integrate MongoDB into the Book Store API

Weeks 7-8: Authentication & Security

- **Authentication in Node.js**:
  - o JSON Web Tokens (JWT)
  - o OAuth2.0
- **Security Best Practices**:
  - o Password Hashing, Helmet.js for Security
- **Projects**:
  - o Add JWT Authentication to Book Store API

Weeks 9-12: Final Full-Stack Projects

- **Full-Stack Application Development**:
  - o Frontend-Backend Integration
  - o User Authentication System
  - o Deployment Strategies
- **Final Projects**:
  - o Task Management System
  - o Blogging Platform

Outcome:

By the end of this 7-month roadmap, you will have a deep understanding of **Vanilla JavaScript** and **DSA** for full-stack development. You will also be equipped with skills in **Node.js**, **REST API** development, and **database management**, along with practical experience in building full-stack applications.