Week 1: JavaScript + DSA Basics

Day 1:

Full Stack (JavaScript Basics)

- Variables (`var`, `let`, `const`), Data Types

- Operators, Conditionals, Loops, Functions

- Hands-on: Write basic programs (Palindrome, Factorial, Fibonacci)

DSA (Arrays & Strings)

- Basics of Arrays, String Manipulation

- Problems: Reverse a String, Two Sum

Day 2:

Full Stack (ES6 & DOM)

- ES6 Features: Destructuring, Spread, Template Literals

- DOM Manipulation (`querySelector`, Events, Forms)

- Hands-on: Build a simple to-do list

DSA (Sorting Algorithms)

- Bubble Sort, Selection Sort, Insertion Sort

- Hands-on: Implement sorting in JavaScript

Day 3:

Full Stack (Asynchronous JS)

- Callbacks, Promises, Async/Await, Event Loop

- Hands-on: Fetch data from a public API

DSA (Recursion & Backtracking)

- Basics of Recursion, Factorial, Fibonacci

- Problems: Generate Parentheses, N-Queens

Day 4:

Full Stack (OOP in JavaScript)

- Classes, Objects, Inheritance, Prototypes

- Hands-on: Create a `User` class with methods

DSA (Linked List)

- Singly & Doubly Linked List, Reversal

- Problems: Detect Loop, Merge Two Lists

Day 5:

Full Stack (Advanced JavaScript Concepts)

- Debouncing, Throttling, Memory Management

- Hands-on: Implement a debounced search bar

DSA (Stack & Queue)

- Stack & Queue Basics, Implement using Arrays

- Problems: Next Greater Element, Min Stack

Day 6:

Full Stack (JavaScript Mini Project)

- Build a Weather App using JavaScript

- Revise JavaScript Concepts

DSA (Hashing & HashMap)

- HashMap Basics, Frequency Count

- Problems: Two Sum, Subarray Sum Equals K

Day 7:

Full Stack (React Basics)

- JSX, Components, Props, State

- Hands-on: Create a counter app

DSA (Binary Search & Searching Algorithms)

- Binary Search, Lower & Upper Bound

- Problems: Find Peak Element, Rotated Array Search

Week 2: React + Backend Basics + DSA Intermediate

Day 8:

Full Stack (React Hooks & Forms)

- `useState`, `useEffect`, Forms & Validation
- Hands-on: Build a simple login form

DSA (Bit Manipulation & Number Theory)

- XOR Operations, GCD, Prime Numbers
- Problems: Single Number, Power of Two

Day 9:

Full Stack (React Router & Navigation)

- `react-router-dom`, Private Routes
- Hands-on: Implement Navigation in React

DSA (Sliding Window Technique)

- Fixed & Variable Window Problems
- Problems: Maximum Sum Subarray, Longest Substring Without Repeating Characters

Day 10:

Full Stack (API Integration in React)

- Fetching Data with Axios, Error Handling

- Hands-on: Fetch data from a public API

DSA (Dynamic Programming - Basics)

- Introduction to DP, Memoization

- Problems: Climbing Stairs, Fibonacci DP

Day 11:

Full Stack (Context API & Global State Management)

- `useContext`, Reducers vs Redux

- Hands-on: Implement a global theme switcher

DSA (DP - Knapsack & Subsets)

- 0/1 Knapsack, Subset Sum Problem

- Problems: Partition Equal Subset Sum, Coin Change

Day 12:

Full Stack (Frontend UI & Forms)

- Material-UI, Tailwind CSS, `react-hook-form`

- Hands-on: Build a form with validations

DSA (Graphs Basics - BFS & DFS)

- Adjacency List, BFS, DFS

- Problems: Number of Islands, Shortest Path

Day 13:

Full Stack (Frontend Mini Project)

- Build a Notes App using React

- Revise React Basics

DSA (Heaps & Priority Queue)

- Max Heap, Min Heap, Heap Sort

- Problems: Kth Largest Element, Merge K Sorted Lists

Day 14:

Full Stack (Backend Basics - Node.js & Express)

- Node.js & Express.js Introduction

- Hands-on: Create a simple API

DSA (Graph Advanced - MST & Dijkstra's Algorithm)

- Dijkstra's, Kruskal's, Prim's Algorithm

- Problems: Network Delay Time, MST Cost

Week 3: Full Stack Integration + Advanced DSA

Day 15-21:

- Build a **Full Authentication System (MERN)**

- Implement JWT Authentication

- Integrate APIs in React

- Implement Role-Based Access Control

DSA Focus:

- Trees (BST, AVL Trees)

- Trie Data Structure

- Segment Tree & Fenwick Tree

Week 4: Full-Stack Capstone Project + Interview Prep

Day 22-28:

- Build a **Job Board App (MERN Full Stack)**
- Deploy on Netlify & Render
- Optimize Performance (Lazy Loading, Code Splitting)
- Mock Interviews & Debugging

DSA Final Topics:

- Hard DP Problems
- Graph Hard Problems
- Advanced System Design Topics

Day 29-30:

- Revise All Topics (Full Stack + DSA)
- Fix Bugs, Improve Documentation
- Solve Common Interview Questions