Complete JavaScript Syllabus

(Frontend + Backend + Interview Preparation)

1. JavaScript for Frontend Development

A. Basics (Foundations)

- What is JavaScript?
- How browsers run JS (JS engine: V8, etc.)
- Linking JS to HTML (inline, internal, external)
- console.log() and debugging
- alert(), prompt(), confirm()

B. Variables and Data Types

- var, let, const
- Primitive types: Number, String, Boolean, Null, Undefined, Symbol, BigInt
- Reference types: Array, Object, Function
- Type coercion and conversion

C. Operators and Expressions

- Arithmetic, Comparison, Logical operators
- Ternary operator
- Short-circuiting (||, &&)

D. Control Structures

- if, else, else if
- switch statement
- Loops: for, while, do-while, for...of, for...in, break, continue

E. Functions

- Function declaration vs expression
- Arrow functions
- Default parameters
- Rest and spread operators
- Callback functions
- Higher-order functions
- Recursion

F. Arrays

- Creation and manipulation
- Array methods: push, pop, shift, unshift, slice, splice, indexOf, includes
- Advanced methods: map, filter, reduce, find, sort, every, some, flat

G. Objects

- Object creation (literal, constructor)
- Object methods and this keyword
- Destructuring
- Object shorthand
- Looping through objects (for...in)
- Object.keys(), Object.values(), Object.entries()

H. DOM Manipulation

- Selecting elements (getElementById, querySelector, etc.)
- Changing content and styles
- Creating, removing, and appending elements
- Handling events (click, input, submit, etc.)
- Event delegation

I. Browser APIs

- setTimeout, setInterval
- localStorage, sessionStorage
- Fetch API
- Geolocation API
- History API
- Navigator API

J. ES6+ Features

- let, const, arrow functions
- Template literals
- Destructuring
- Spread/rest
- for...of
- Modules (import/export)
- Optional chaining ?.
- Nullish coalescing ??

K. Asynchronous JavaScript

- Callbacks
- Promises
- async / await
- Error handling with try/catch
- Fetching data from APIs (REST)

L. Error Handling

- Types of errors (syntax, runtime, logical)
- try, catch, finally
- Custom error creation

2. JavaScript for Backend Development (Node.js)

A. Introduction to Node.js

- What is Node.js?
- Node.js architecture (event loop, single-threaded, async I/O)
- Installing Node.js and npm

B. Node.js Fundamentals

- Modules and require
- Built-in modules: fs, http, url, path, os, events
- Creating servers with http module
- Streams and Buffers

C. npm and Package Management

- Installing packages (global vs local)
- Semantic Versioning
- Creating package.json
- Common packages: dotenv, nodemon, express, mongoose, cors, axios

D. Express.js Framework

- Routing (GET, POST, PUT, DELETE)
- Middleware (custom, built-in, 3rd party)
- Error handling middleware
- Static file serving
- Request and response objects

E. REST APIs

- RESTful conventions
- Route parameters and query strings
- Handling JSON input
- Status codes
- CRUD operations

F. MongoDB with Mongoose

- Introduction to MongoDB
- Connecting with Mongoose
- Schema and models
- CRUD operations
- Validation
- Mongoose middleware

G. Authentication & Authorization

- Password hashing with bcrypt
- JWT (JSON Web Tokens)
- Sessions and cookies

- Role-based access

H. File Uploads

- Using multer for uploading
- Handling file types and size limits

I. Realtime with WebSockets

- Introduction to WebSockets
- Using socket.io for real-time communication

J. Deployment

- Environment variables
- Hosting with Render / Vercel / Railway
- Serving frontend with backend
- CI/CD basics

3. JavaScript Interview Preparation

A. Theoretical Questions

- Hoisting
- Closures
- Scope (Global, Local, Block, Lexical)
- Execution context and call stack
- Event loop and task queue
- this keyword in different contexts
- Shallow vs Deep copy
- Prototypes and Inheritance
- Functional programming vs OOP

B. Coding/DSA Questions

- Array & String problems: reverse, rotate, flatten, unique values, etc.
- Object manipulation
- Recursion-based problems
- Sorting algorithms (bubble, quick, merge, insertion)
- Searching (binary search, linear search)
- HashMap/Set problems
- Promise-based problems
- Event handling simulation

C. Advanced JavaScript Concepts

- Closures in-depth
- Debounce and Throttle
- Currying
- Memoization

- Call, Apply, Bind
- Prototype chain and Inheritance
- Async vs Defer
- Event bubbling vs capturing

D. System Design Basics (for backend roles)

- REST vs GraphQL
- MVC architecture
- Stateless vs Stateful
- Rate limiting
- Caching strategies
- Scalability principles