**1. Basics of JavaScript**

* Introduction to JavaScript
* Variables (var, let, const)
* Data types (String, Number, Boolean, Undefined, Null, Object, Symbol)
* Operators (Arithmetic, Comparison, Logical, Assignment)
* Control structures (if-else, switch-case)
* Loops (for, while, do-while)
* Functions (Declaration, Expression, Arrow Functions)
* Scope (Global, Local, Block)
* Hoisting and Closures

**2. Intermediate JavaScript**

* Arrays & Array Methods (map, filter, reduce, etc.)
* Objects & Object Methods (Object.keys(), Object.values(), Object.entries())
* The this keyword
* Prototypes & Inheritance
* ES6+ Features (Destructuring, Spread & Rest, Template Literals, Modules)
* Asynchronous JavaScript (setTimeout, setInterval, Promises, Async/Await)
* Error Handling (try...catch)

**3. DOM Manipulation**

* Selecting elements (getElementById, querySelector)
* Modifying HTML & CSS using JavaScript
* Event Listeners & Handling (click, mouseover, keydown, etc.)
* Forms & Validations
* Local Storage & Session Storage

**4. Advanced JavaScript**

* Fetch API & HTTP Requests (AJAX)
* JSON (JavaScript Object Notation)
* Regular Expressions (Regex)
* Web APIs (Geolocation, Notifications, Clipboard, etc.)
* Debouncing & Throttling
* Callbacks, Promises, Async/Await deep dive

**5. Object-Oriented Programming (OOP) in JavaScript**

* Classes & Objects
* Constructor Functions
* Prototypal Inheritance
* Encapsulation, Abstraction, Polymorphism

**6. JavaScript in the Browser**

* Event Loop & Call Stack
* Web Storage (Local Storage, Session Storage, IndexedDB)
* Cookies & Authentication
* Performance Optimization

**7. JavaScript Frameworks & Libraries (After mastering JS)**

* Learn React.js (Since you’ve already started)
* Learn Next.js (Good for full-stack development)
* Explore other frameworks (Vue.js, Angular, Svelte)

**8. Backend Development with JavaScript**

* Learn Node.js (Since you want to be a backend developer)
* Learn Express.js
* Work with Databases (MongoDB, PostgreSQL, MySQL)
* Build REST APIs & GraphQL APIs

**9. Advanced Topics**

* WebSockets & Real-time Applications
* Serverless Functions
* Testing in JavaScript (Jest, Mocha)
* TypeScript (for better type safety)

**10. Projects**

* Build small projects (Calculator, To-Do List, Weather App)
* Work on a full-fledged project (E-commerce site, Task Manager, Blog Platform)
* Contribute to open-source

Would you like me to suggest some beginner-friendly projects to practice?