React Developer Roadmap (2024)

A thorough React developer roadmap for 2024 that addresses all aspects of React and beyond.

0. Before you start React

You should know and be comfortable with all of the following:

- Basic HTML
  - HTML Elements, Attributes, Headings, Paragraphs, Colors & Styles
  - HTML Links, Images, Tables, Lists, Block & Inline, Div, Classes, Id
  - HTML Forms
  - HTML Layout, Responsiveness & Semantic
- Basic CSS
  - CSS Basics - Syntax, Selectors, Colors, Backgrounds, Borders, Margin, Padding, Height/Width, Box Model, Outline, Text, Fonts, Links etc.
  - CSS More - Lists, Tables, Display, Position, z-index, Overflow, Float, Inline Block, Align, Combinators, Pseudo-classes & elements, Opacity etc.
  - CSS Forms & Layouts
  - CSS Flexbox
  - CSS Grid
  - Advanced CSS - CSS Units, Shadows, Gradients, Transitions, Animations, Specificity, etc.
- Basic Tailwind CSS
  - Tailwind Utilities
  - Responsive Variants
  - Hover, focus and other states
  - Dark Mode variant
  - Tailwind Directives
  - Tailwind Configurations
  - Theme Configurations
  - Tailwind cn() utility
- Document Object Model (DOM)
  - DOM Basics - Basics, Method, Document, Elements, Forms, CSS, Events, Navigation, Nodes and Collections
  - DOM Advanced

- Basic JavaScript
    - JS Basics - Statements, Expressions, Syntax, Variables, Operators, Data Types, Functions, Objects, Arrays, Events, Array and String Methods, Object Methods, Date, Conditionals, Error Handling, JavaScript OOP - classes and inheritance and Debugging
    - JS Web APIs - Forms, History, Geolocation, Storage, Worker and Fetch API
    - JS JSON
- JavaScript Advanced
    - Solid JS Concepts - Scope, Hosting, Execution Context, Closures, Prototype, Recursion, Primitive vs Reference Data Types, Currying, Intersection Observer, Memoization, Event Propagation, Debounce etc.
    - Asynchronous JavaScript - Callbacks, Promises and async-await
- Modern JavaScript
    - Different ES6+ JS Syntaxes and concepts eg. Arrow function, Truthy/Falsy values, Ternary Operator, Different Array methods like find, filter, map, reduce, slice, splice, push, pop, concat, different looping strategies, Spread & Rest Operator, Array and Object Destructuring, Imports/Exports syntax, Template Literals, Sorting etc.
- Git/GitHub
    - Basics of Git
    - Important Git Commands

1. React Fundamentals

You should know and be comfortable with all of the following:

- Getting Started with React
    - Introduction to React - Why React - Comparison with Vanilla JS
    - React Installation & Editor Setup with Vite
    - How React works - Virtual DOM
    - Basics of React Components
    - Basics of JSX: React's Markup
    - JavaScript in JSX
    - Passing Props to Components
    - Conditional Rendering
    - Rendering Lists
    - Pure Components

    - How to split larger components into smaller ones
- Adding Interactivity
    - Responding to Events - Event Handlers
    - Understanding States - React Component's Memory - useState
    - How State Works in React
    - How Rendering works in React
    - Updating complex states immutably in React
- React State Management Deep Dive
    - Declarative vs Imperative UI
    - Thinking UI Declaratively
    - Finding & Structuring React States
    - Connecting Event Handlers to React
    - Sharing State between components
    - Lifting State up
    - Extracting State Logic into Reducers
    - useReducer Hook
    - How to use Immer with React for concised immutable State Update
    - Passing Data Deeply inside React Components
    - Avoiding Prop Drilling - Context API & useContext Hook
    - Combine context and reducer to write scalable code

2. Advanced React
    - Referencing values with Refs - useRef hook
    - Manipulating the DOM with Refs
    - Synchronizing with Effects - useEffect hook
    - Separating events from Effects
    - Removing Effect Dependencies
    - Performance optimization with useCallback and useMemo hook
    - Reusing logic with Custom Hooks
    - Calling APIs from Back-end with React
3. Advanced State Management
    - Using Redux / Toolkit
    - Using Zustand
    - Using Jotai
    - Using Recoil

- Using MobX

4. Styling Solutions
   - Tailwind
   - CSS Modules
   - Styled Components
   - React UI Component Library - Shadcn
   - React UI Component Library - Keep React
   - Material UI
   - Chakra UI
   - Ant Design
5. React Ecosystem & Use Cases
   - React Router DOM
   - API Request with Axios in React
   - React Suspense & Error Boundaries
   - React Lazy Load
   - React Infinite Scroll
   - Uncommon React Hooks - useDebugValue, useDeferredValue, useId, useImperativeHandle, useInsertionEffect, useLayoutEffect and useTransition
   - React Authentication
     - How to handle user sign in (email, password, JWT)
     - How to handle access tokens and token refreshes
     - Social sign in (Google, Facebook, GitHub, etc.)
     - Using Supabase
     - Using Firebase
     - Using Clerk
   - Form Handling in React
     - How to validate user input in forms (emails, passwords, etc.)
     - How to send form data to server
     - How to handle file uploads
     - Using React Hook Form
     - Using Formik
   - Accessibility
     - Understanding why accessibility is important
     - Using semantic HTML

    - How to implement keyboard navigation
    - How to add aria labels
    - Using React Aria
- Testing
    - How to implement unit tests
        - Using React Testing Library
        - Using JestHow to implement e2e integration tests
        - Using Cypress
        - Using Playwright

6. React Frameworks

You should have worked with one of the following:

- Vite
    - How to run a simple React application
- Next.js
    - Understanding file-based routing
    - Understanding Next Auth
    - Understanding server components
    - Understanding server actions
- Remix

7. Beyond React

- Team player
    - How to work within a team
    - How to perform code reviews
    - How to give and receive feedback
- Efficiency
    - How to prioritise tasks
    - How to handle tech debt
    - How to meet deadlines and goals
- Continuous Learning
    - How to continuously learn and grow
    - How to stay up to date with your skills
- Networking & Communication - Going to meetups or events - Contributing to open source projects - Networking within the company you work in

Resources from Learn with Sumit

Some free and paid resources from Learn with Sumit that might help you achieve your goal to become a React Developer

- Think in a React way: Free React Starter Course
- JavaScript Beginners Course
- DOM Course
- Think in a JavaScript way - Advanced Conceptual JavaScript Course
- Modern JavaScript Course
- CSS Grid
- CSS Flexbox
- Tailwind CSS Course
- Reactive Accelerator: React-Next.js Course
- Think in a Redux way - Redux paid course