**React Developer Roadmap 2025 (Every Budding React Developer Should Know) -** [Venkatesh B </>](https://medium.com/@venkateshb-03?source=post_page-----81750df3d3be--------------------------------) Aug 11, 2022

**Introduction**

Hi readers, This is my third blog of the Roadmap Series. Here I Listed all the React topics to learn level by level to master React.

This is not the Top to End Roadmap for React. This is part of My First Complete Front End Development Roadmap (blog). If you did not read that blog mean please read at the below link. In that, At Level 2 we need to pick one JavaScript Framework. Now we chose to React as the first JS Library. So we should know what are things to learn to master React.

Link :- <https://medium.com/@venkateshb-03/front-end-roadmap-things-every-frontend-developer-should-know-for-surviving-up-to-2050-b0051f4df8c8>.

**Let’s learn….**

**Introduction to React**

React (also known as React.js or ReactJS) is a free and open-source front-end JavaScript library responsible only for the view layer of the application for building user interfaces based on UI components. And React is not the Framework, it is JavaScript Library. It is maintained by Meta (formerly Facebook) and a community of individual developers and companies. React can be used as a base in the development of single-page, mobile, or server-rendered applications with frameworks like NextJS, etc.

React is a declarative, efficient, and flexible JavaScript library for building single-page applications.

It is using Virtual DOM so it is very faster like VueJS and easy to learn when compared to other JS Frameworks and libraries. Official website -<https://reactjs.org/>

React uses a heuristic algorithm called the “diffing” algorithm for reconciliation that makes the component updates predictable and faster. The React first calculates the differences between the real DOM and the copy of DOM when it finds an update of components. Once it is finished calculating, the new update would be reflected on the real DOM.

**RoadMap**

Diagram

Description automatically generated

I found the above image from <https://roadmap.sh/frontend>. There we can see all the Roadmaps for FrontEnd, BackEnd, DevOps, Full Stack, etc.

**Basic to advanced things every React Developer should know are**

Level 1 (React Basics)

Learn HTML, CSS, and JavaScript (very Important) with all things I mentioned in the Front End Development Roadmap up to the stage of picking the JavaScript Framework or Library.(*Link-* <https://medium.com/@venkateshb-03/front-end-roadmap-things-every-frontend-developer-should-know-for-surviving-up-to-2050-b0051f4df8c8> ).

Then, Learn React Basics 1

1. Create React APP
2. React Folder Structure
3. JSX ,WebBack and Babel(JSX Compiler)
4. Node package Managers like NPM and Yarn
5. Components and Components Rendering
6. Class vs Functional Components
7. Props and State and their difference
8. Components Lifecycle
9. Conditional Rendering(using && operator)
10. Handling Events
11. Lists(arrays) and keys(key -maintain the uniqueness of each element when rendering with List)
12. Composition vs Inheritance

Learn Typescript and its concepts like(this is optional)

* Structural Typing
* Type Interface
* Union Types
* Built-in Types
* Type Guards
* Generics
* Typescript Decorators

And start developing components in the .tsx or ts file.

Level 2 (React Basics 2)

1. Forms
2. Lifting State up (maintaining state in parent component and sending state and setState function to child component via props drilling. and Child can update parent state)
3. Basic Hooks like -useState, useEffect- And also learn the syntax of useEffect for all the Lifecycles.
4. Common Hooks like-

* useCallback and useMemo and their difference
* useReducer ,useSelector
* useRef, useContext,etc

5. Learn to build your own custom hook

6. Context API

7. Refs (useRef,createRef) and the Dom

8. Rendering Props and Code Splitting

Level 3 (React Advanced Topics 1)

1. Higher Order Components (sending component as a prop to another component), Controlled(Ex: Input tag is controlled by React Component) vs Uncontrolled Components. (Forms data(input element) are handled by DOM(ref) itself .)
2. Portals and Profiler
3. Error Boundaries

* [static getDerivedStateFromError()](https://reactjs.org/docs/react-component.html#static-getderivedstatefromerror)
* [componentDidCatch()](https://reactjs.org/docs/react-component.html#componentdidcatch)

4. Fiber Architecture

5. Optimizing Performance

6. Reconciliation and Diffing Algorithm

7. Type Checking with PropTypes and Strict Mode

8. Lazy Loading, Suspense, and fallback

Level 4 (React Advanced Topics 2- Ecosystem)

1. Routers- React Router or Reach Router -(Protected Route and Public Route).

2. Forms-

* Formik (<https://formik.org/>)
* React Hook Form
* Final Form, AvForm,etc

3. State Management

* Context API/State
* Redux
* Mobx ,etc

4. Styling

* Material UI
* ReactStrap (React + Bootstrap)
* Ant Design, Chakra UI, etc
* Styled Components

5. API Calls with HTTP

* Axios
* Unfetch
* superagent, etc

6. API Calls with Graph QL

* Apollo
* Relay Model, etc

Level 5 (Testing and Other things with React)

1. Component Testing

2. Props and State Testing

3. Snapshot Testing

4. Event Testing

5. Mocking function

6. Rest API Testing, etc

Testing Libraries need to learn

1. Jest (so popular and developed by Facebook)
2. Enzyme
3. React Testing Library

Others in React

* Server Side Rendering with NextJS
* Static Site Generator with NextJS ,Gatsby
* Mobile(Android and IOS) App with React Native
* Using Web Components in React and Using React in your Web Components.
* Progressive Web APP in React with the help of service worker

Tips

Do some CRUD Operation like Todo APP with State(useState),Then do with ContextAPI and then do with state management library like Redux, etc, And do CRUD with API Integration(using any public API) with the Help of HTTP Client like Axios. And then do some Clone APP like Facebook clone, Amazon Clone, and Apply all the concepts you learned. And start to test the component with any Testing Library.

Keep updated yourself and be aware of upcoming versions and their features. And Checkout <https://reactjs.org/blog/all.html/> this official react blog page for upcoming features like Server Components, Transitions, and Auto Batching, etc.

**Conclusion**

I hope this blog is very useful for beginners and fresher who are working or training as React developers or Front-End Developer or Full Stack Developer. Thanks for reading. Have a good day ✌️.

Don’t worry this is just the RoadMap, I will write the blog for all the topics and concepts which are present in the above Image.

If you think something is missing, have questions, or would like to give feedback, go ahead and leave a comment below. I’d appreciate the feedback.

**My Roadmap Blog Series**

1. Full Front End Development Roadmap- <https://medium.com/@venkateshb-03/front-end-roadmap-things-every-frontend-developer-should-know-for-surviving-up-to-2050-b0051f4df8c8>
2. JavaScript Roadmap- Comming Soon
3. Angular Developer Roadmap- <https://medium.com/@venkateshb-03/angular-developer-roadmap-2025-things-every-budding-angular-developer-should-know-86061c84a0f9>
4. React Developer Roadmap- \*\*
5. VueJS Developer Roadmap- <https://medium.com/@venkateshb-03/vuejs-developer-roadmap-things-every-budding-vuejs-developer-should-know-fc5a0f9c7d55>

Follow my page , you will get a quick notification for upcoming blogs.

Please Share your feedbacks and topics you need to learn on Command. Based on that I will enhance my Blog.