**ReactJS developer roadmap 2022 –**

[Melih Yumak](https://melih193.medium.com/?source=post_page-----76ca119188bd--------------------------------)Jun 21, 2022

**Roadmap for ReactJS developer in 2022**

ReactJS is a very popular UI framework for creating user interfaces. There are also so many benefits of using react. In this tutorial, we will look at the tools and languages needed to be known to be a successful ReactJS developer

We can create

* Progressive web applications
* Scalable web applications
* Reusable components
* Small to big-scale projects

Here in this tutorial, we will discuss which languages and tools you need to learn to become a successful **ReactJS developer**.

**HTML**

Html has a crucial impact on creating web components in ReactJS applications.

Html is used in the .jsx or .tsxfiles to define our component elements

We can use HTML tags directly inside these components.

const componentExample = () => (  
 <div>  
 <h1> Heading </h1>  
 </div>  
)

As the example above we have defined our example component with HTML tags. To be able to use these tags you need to learn how to use HTML elements inside ReactJS components.

**CSS**

Styling is one of the most important things in web development. Especially if you are front end developer you will spend so much time making perfect stylings

These changes require so much styling knowledge, there is another language you should learn along with HTML and that language is CSS.

There are other alternatives to CSS like SCSS or SASS. But before learning these 2 I suggest you learn CSS completely and go further steps.

**JavaScript**

JavaScript is one of the key languages while learning ReactJS, you should learn JavaScript as well. JavaScript evolves every year and increases its power more than ever.

Except than the creating HTML and CSS we will mostly use JavaScript for writing ReactJS code.

**NPM**

Npm is a package manager for the JavaScript libraries and frameworks.

While you are learning how to start ReactJS projects and also creating one you will need to use package manager to install required packages to your local device.

You can search for the packages in npm through their website.

Learning npm also will be done along with the projects you make with ReactJS as well.

**Git**

In programming, git is one of the most important tools you need to learn to submit your code and work with another developer's code.

If you are started learning git, you can have a **look**[**most used 8 GitHub commands**](https://medium.com/codex/most-used-8-github-commands-81b35f95bdf3)

**[Most used 8 GitHub commands](https://medium.com/codex/most-used-8-github-commands-81b35f95bdf3" \t "_blank)**

[If you are part of development Git is one of the most important tools you can use for development.](https://medium.com/codex/most-used-8-github-commands-81b35f95bdf3" \t "_blank)

[medium.com](https://medium.com/codex/most-used-8-github-commands-81b35f95bdf3" \t "_blank)

In this article, you will learn how to use the commands below

* clone repository
* check status
* add to git
* how to commit
* how to push
* how to pull
* opening new branches and using these branches

**React**

Learning ReactJS will enable you to make faster user interfaces with cleaner code. In this step, you will start using your knowledge from the parts described above. Now, these languages and tools will be your helper along the journey of learning ReactJS.

While learning ReactJS there is very good [documentation](https://reactjs.org/) on the official ReactJS website. You can follow the tutorial to learn ReactJS.

I know the title of the article states ReactJS developer map, but after you learn ReactJS here are some other libraries we need to use along with ReactJS for state management :)

**Redux**

After you have gained a basic understanding of using ReactJS and feeling comfortable enough to create your own projects there is another topic will appear in the ReactJS development life cycle and that is **state management.**

**State management** is a really important topic in ReactJS development. There are many libraries you can use for state management, but I prefer using redux.

The tutorial below **about redux** and **how to use redux with react**

**[How to use Redux with React](https://itnext.io/how-to-use-redux-with-react-143de57d0bab" \t "_blank)**

[Redux Basic explanation for ReactJS applications](https://itnext.io/how-to-use-redux-with-react-143de57d0bab" \t "_blank)

[itnext.io](https://itnext.io/how-to-use-redux-with-react-143de57d0bab" \t "_blank)

After you learn regarding to redux as well there is another tool which is broadly used in the JavaScript ecosystem.

**Typescript**

Typescript is used for to use types inside the JavaScript. For the large-scale applications typescript helps a lot along the way.

I strongly use typescript in all my projects. It helps me to identify possible issues when I write the code itself. That provides me much more time than searching the whole code flow for some bug in the code itself.

**+Bonus GraphQL**

GraphQL is a very powerful tool, and it makes queries and requests much easier. When you make a couple of projects with ReactJS, I suggest you test out making some of the same applications with GraphQL as well.

That is all regarding the ReactJS developer roadmap.