

## REACT JS:

### WHAT IS REACT JS?

ReactJS is a declarative, efficient, and flexible JavaScript library for building reusable UI components. It is an open-source, component-based front-end library responsible only for the view layer of the application. It was created by **Jordan Walke**, who was a software engineer at **Facebook**.

### WHY WE SHOULD LEARN REACTJS?

Today, many JavaScript frameworks are available in the market (like angular, node), but still, React came into the market and gained popularity amongst them. The previous frameworks follow the traditional data flow structure, which uses the DOM (Document Object Model). DOM is an object which is created by the browser each time a web page is loaded. It dynamically adds or removes the data at the back end and when any modifications were done, then each time a new DOM is created for the same page. This repeated creation of DOM makes unnecessary memory wastage and reduces the performance of the application. Therefore, a new technology ReactJS framework was invented which removes this drawback. ReactJS allows you to divide your entire application into various components.

## Getting Started

### YOUTUBE REFERENCES:

1) [https://www.youtube.com/watch?v=-mJFZp84TIY&list=PLu0W\\_9III9agx66oZnT6IyhcmIbUMNMdt&index=1](https://www.youtube.com/watch?v=-mJFZp84TIY&list=PLu0W_9III9agx66oZnT6IyhcmIbUMNMdt&index=1)

[https://www.youtube.com/watch?v=hnVOvvbQrwA&list=PLu0W\\_9III9agx66oZnT6IyhcmIbUMNMdt&index=2](https://www.youtube.com/watch?v=hnVOvvbQrwA&list=PLu0W_9III9agx66oZnT6IyhcmIbUMNMdt&index=2)

2) <https://www.youtube.com/watch?v=zuyH4QUuTZk>

### STEPS:

#### 1)INSTALL NODE JS

FROM: <https://nodejs.org/en/download>

#### 2)SETUP NODEJS

#### 3)INSTALL VS CODE

## 4)Creating an App ON VS CODE

To create a new app, you may choose one of the following methods and type in the terminal:

### **npx**

```
npx create-react-app my-app
```

### **npm**

```
npm init react-app my-app
```

## **Selecting a template**

You can now optionally start a new app from a template by appending `--template [template-name]` to the creation command.

If you don't select a template, we'll create your project with our base template.

Templates are always named in the format `cra-template-[template-name]`, however you only need to provide the `[template-name]` to the creation command.

```
npx create-react-app my-app --template [template-name]
```

## **Creating a TypeScript app**

You can start a new TypeScript app using templates. To use our provided TypeScript template, append `--template typescript` to the creation command.

```
npx create-react-app my-app --template typescript
```

## **Selecting a package manager**

When you create a new app, the CLI will use [npm](#) to install dependencies, depending on which tool you use to run `create-react-app`. For example:

```
# Run this to use npm  
npx create-react-app my-app
```

## Output

Running any of these commands will create a directory called `my-app` inside the current folder. Inside that directory, it will generate the initial project structure and install the transitive dependencies:

```
my-app
├── README.md
├── node_modules
├── package.json
├── .gitignore
├── public
│   ├── favicon.ico
│   ├── index.html
│   ├── logo192.png
│   ├── logo512.png
│   ├── manifest.json
│   └── robots.txt
└── src
    ├── App.css
    ├── App.js
    ├── App.test.js
    ├── index.css
    ├── index.js
    ├── logo.svg
    ├── serviceWorker.js
    └── setupTests.js
```

No configuration or complicated folder structures, only the files you need to build your app. Once the installation is done, you can open your project folder:

```
cd my-app
```

## Scripts

Inside the newly created project, you can run some built-in commands:

```
npm start
```

Runs the app in development mode. Open <http://localhost:3000> to view it in the browser.

The page will automatically reload if you make changes to the code. You will see the build errors and lint warnings in the console.

```
npm test
```

Runs the test watcher in an interactive mode. By default, runs tests related to files changed since the last commit.

```
npm run build
```

Builds the app for production to the `build` folder. It correctly bundles React in production mode and optimizes the build for the best performance.

The build is minified and the filenames include the hashes.

Your app is ready to be deployed👍.