

# Creating React Folder using Vite

## Steps to Follow

- ❖ Create one folder in desktop
- ❖ Open command prompt inside this folder
- ❖ `npm create vite@latest`
- ❖ Project name : anything
- ❖ Select Framework: react
- ❖ `cd project_name_you_have_give`
- ❖ `npm install`
- ❖ `code .` (Open in Vs code)
- ❖ Open Terminal `ctrl+j`
- ❖ Command to run app : `npm run dev`
- ❖ Delete existing src folder create new src folder
- ❖ Inside src folder create main.jsx file

## 1. What is Vite?

It is a more efficient way to build web projects and it is a modern built tool.

## 2. Why we have to use Vitejs instead of other ways?

→ Faster development

→ It uses native ES modules to create a development server that is quicker and more efficient

### 3. What is Rollup?

→ It is a module bundler. A module bundler is a tool used in web development to combine multiple separate modules or files into a single file, typically for the purpose of improving performance, optimizing loading times, and organizing code.

#### Folder structure of ReactJS

After successful installation of reactjs app we will be getting default folder and files.

##### 1) **node\_modules :**

- It is a folder where all the predefined codes of react will be present.
- Warning (do not touch this folder)

##### 2) **public :**

- Static files can be placed here . Ex: Images, fonts

##### 3) **src :**

- It is a source folder where we are writing the code
- The 2 important files we have to maintain
  - ✓ Main.jsx (it is considered as a root file execution starts from here)
  - ✓ App.jsx (it is a Component)

##### 4) **package-lock.json & package.json:**

- These two files are considered as directories of react folder which contain some information

##### 5) **vite.config.js:**

- The Vite configuration file. This is where you can customize the Vite build settings, plugins, and other project-specific configurations.

## Libraries needed for ReactJS :

**1)react :** The 'react' library provides fundamental tool and components needed to build react app.

## 2) react-dom :

- ✓ The 'react-dom' library responsible for rendering React components in the UI.
- ✓ **createRoot( )** inside this method we need to target the `<div id="root">` `</div>` present inside index.html
- ✓ **render()** method used to render the content on UI. Inside this method we can pass either normal string, JSX, component.

### main.jsx

```
import React from "react"
import ReactDOM from "react-dom/client"

console.log(React);
console.log(ReactDOM);

ReactDOM.createRoot(document.getElementById("root")).render("hii")
```

