# Creating React Project Using NPM Packages

Make sure you have nodejs installed. If not installed, go through this tutorial for nodejs (<https://www.guru99.com/node-js-tutorial.html>) installation.

Once you have nodejs installed, create a folder **reactproj/**.

To start with project setup, run command **npm init**.

This is how the folder structure will look like:

reactproj\

package.json

Now we will install the packages that we need.

Here are the list of packages for reactjs:

npm install react --save-dev

npm install react-dom --save-dev

npm install react-scripts --save-dev

or

npm install react react-dom react-scripts -–save-dev

Note: For dependencies vs dev-dependencies

<https://www.matthewburfield.com/do-npm-dependencies-and-dev-dependecies-effect-your-production-build/>

Open the command prompt and run above commands inside the folder reactproj/.-

Create a folder **src/** where all the js code will come in that folder. All the code for reactjs project will be available in the src/ folder. Create a file index.js and add import react and react-dom, as shown below.

**index.js**

import React from 'react';

import ReactDOM from 'react-dom';

ReactDOM.render(

<h1>Hello, user!</h1>,

document.getElementById('root')

);

We have returned the basic code for reactjs. We will explain the details of it in the next chapter. We want to display **Hello, user**and the same is given to the dom element with id “root”.It is taken from the index.html file, which is the start file, as shown below.

Create a folder public/ and add index.html in that as shown below

**index.html**

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8" />

<title>ReactJS Demo</title>

</head>

<body>

<div id = "root"></div>

</body>

</html>

The package react-scripts will take care of compiling the code and starting the server to display the html file i.e index.html. You need to add the command in package.json that will take care of using react-scripts to compile the code and start server as shown below:

"scripts": {

"start": "react-scripts start"

}

After installing all the packages and adding the above command, the final package.json is as follows:

**Package.json**

{

"name": "reactproj",

"version": "1.0.0",

"description": "",

"main": "index.js",

"scripts": {

"start": "react-scripts start"

},

"author": "",

"license": "ISC",

"devDependencies": {

"react": "^16.9.0",

"react-dom": "^16.9.0",

"react-scripts": "^3.1.1"

}

}

To start testing reactjs run the command

npm run start

C:\reactproj>npm run start

> reactproj@1.0.0 start C:\reactproj

> react-scripts start

It will open browser with url http://localhost:3000/ as shown below:

**public/index.html**

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8" />

<title>ReactJS Demo</title>

</head>

<body>

<div id = "root"></div>

</body>

</html>

We are going to use the same process to execute the javascript files in the next chapters too. Add all your js and .jsx file in src/ folder .The file structure will be as follows:

reatproj/

src/

index.js

node\_modules/

public/

index.html

package.json

**How to Create Your First React Project Setup**

Here is a step by step guide in this ReactJS Tutorial to start with the first react application.

**Step 1)** Import the react packages.  
1. To start with ReactJS, we need to first import the react packages as follows.

import React from 'react';

import ReactDOM from 'react-dom';

2. Save the file as index.js in src/ folder

**Step 2)** Write Simple Code.  
We will write a simple code in this tutorial React JS, wherein we will display the message **“Hello, user”**

ReactDOM.render(

<h1>Hello, user!</h1>,

document.getElementById('root')

);

**ReactDOM.render**will add the <h1> tag to the element with id root. Here is the html file we are having:

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8" />

<title>ReactJS Demo</title>

</head>

<body>

<div id = "root"></div>

</body>

</html>

**Step 3)** Compile the Code.

Next in this React.js Tutorial, we need to compile the code to get the output in the browser.

Here is the folder structure:

reactproj/

node\_modules/

src/

index.js

package.json

public/

index.html

We have added the commands to compile the final file in package.json as follows:

"scripts": {

"start": "react-scripts start"

},

To compile the final file run following command:

npm run start

When you run above command, it will compile the files and notify you if any error, if all looks good, it will open the browser and the run the index.html file at http://localhost:3000/index.html

Command: npm run start:

C:\reactproj>npm run start

> reactproj@1.0.0 start C:\reactproj

> react-scripts start

**Step 4)** Check Output.

The URL http://localhost:3000 will open in the browser once the code is compiled.

**Note :**

**Its mandatory to have index.js and index.html file otherwise it will give this error**

**Could not find a required file.**

**Name: index.html**

**Searched in: C:\Users\user\Desktop\React JS\React\4 React using NPM example\ReactProject\public**

**Could not find a required file.**

**Name: index.js**

**Searched in: C:\Users\user\Desktop\React JS\React\4 React using NPM example\ReactProject\src**

**When the React.js application is loaded in a web browser, the browser first loads the index.html file. Within the index.html file, there is usually a <div> element with an id attribute, such as <div id="root"></div>. This <div> acts as a placeholder or mounting point for the React application.**

**Next, the index.js file is responsible for rendering the root component of the React application into that <div> element. It typically uses the ReactDOM.render() function to accomplish this. The index.js file is imported or included in the index.html file using a <script> tag, and the browser executes the JavaScript code within the index.js file to initialize and run the React application.**

**In summary, while both the index.html and index.js files are important for a React.js application, the index.html file is the initial file loaded by the browser, whereas the index.js file contains the code that sets up and runs the React application within the designated element of the index.html file.**

**Refer : https://www.c-sharpcorner.com/article/how-react-application-gets-loaded-or-started/**

****

**Note :**

**Here webpack(development server) is installed automatically when we install react and react-dom and react-scripts. But if you face any problem regarding webpack server, or we gets webpack compiled with 1 error or warning so in that case try to install webpack through npm. Like :**

**npm install webpack webpack-dev-server webpack-cli –save-dev**

**Output :**

**C:\Users\user\Desktop\React JS\React\4 React using NPM example\ReactProject>npm run start**

**> reactproject@1.0.0 start**

**> react-scripts start**

**Compiled successfully!**

**You can now view reactproject in the browser.**

**Local: http://localhost:3000**

**On Your Network: http://192.168.90.171:3000**

**Note that the development build is not optimized.**

**To create a production build, use npm run build.**

**webpack compiled successfully**