**React**

*React has web pack that does the bundling for us*

**Install react :**

1. **Npm install react (or)**
2. **A tool for react : create-react-app**

In the terminal : **npm install -g create-react-app**

Now after installation :in terminal **create-react-app robofriends //** we simply call create-react-app followed by **name of the app (**here robofriends)

It will install set of components required for our app

**cd robofriends**

**Npm start //** starts react and opens local host with react page

In src -> index.js

**ReactDOM.render(<App />, document.getElementById('root'));**

Make changes and save

ReactDOM.render(**<h1> welcome to react**</h1>**,** document.getElementById('root'));

**To upgrade to newer version of react :** go to json file and change the version and type npm install in the terminal

"react": "^16.7.0",

  "react-dom": "^16.7.0",

  "react-scripts": "2.1.3"

<https://www.npmjs.com/package/create-react-app>

<https://github.com/facebook/create-react-app>

After pull request  from hit hub , use **install npm** to install the changed environment

**React introduction :**

**Import react** works here because react has web packs that takes care of

**React and ReactDom**

React is a view library - core that does dom manipulations

React can be used for mobile devices , virtual reality

plug and play different libraries like this (react-dom) based on for what screen you are rendering into .. so ReactDom for dom websites ….ReactNative that renders to mobile devices

**Import “./index.css” :**

seperate css file for separate components ./ indicates that it is present here

**Registerserviceworker -** faster and work offline

**ReactDOM.render(<h1> Hello World </h1>, document.getElementById('root'));**

I want the reactdom package to use the render function .

**ReactDOM.render(<App />, document.getElementById('root'));**

**Import App from “./App”  //**./App.js file , App.js file from the source ,

It is imported

**Go to App.js**

**import React, { Component } from 'react';**

**//** importing , destructuring and getting the component

**import logo from './logo.svg';**

**import './App.css';**

**class App extends Component { //**class App extends component that comes from react and this is a standard syntax

**render() {  //** component always has a rule that it should render something

**return ( //** we render by returing the html piece of the website

**<div className="App">**

**<header className="App-header">**

**<img src={logo} className="App-logo" alt="logo" />**

**<p>**

**Edit <code>src/App.js</code> and save to reload.**

**</p>**

**</header>**

**</div>**

**);**

**}**

**}**

**export default App;**

// we are creating a custom component here and using it as we need

**Creating own component:**

**ReactDOM.render(<Hello />, document.getElementById('root'));**

**import Hello from “./Hello”    //**or Hello.js

**Right click src and create new file  Hello.js//** first letter should be capital and create Hello.js

**import React,{Component} from "react";**

// import React and component from react instead of React.component

**import "./Hello.css"; //**create new css file and import it

**import "tachyons"; //** package similar to boot strap , we can use class names

Add tachyons by using **npm install tachyons**

**class Hello extends Component{ //** Hello component creation

**render(){ //** always render function should be there

**return ( //** tell render what we should return . brackets to add multiple lines

**<div className="f1 tc"> //**f1 and tc are tachyon classes

// not using class and using classname , because it is actually not a html here , it is a js file . **JSx file .react allows to combine html and js.**react allows components to be components with all it needs . it allows us to change only that particular component . react creates its own virtual or fake dom using JSX , and real dom takes the content from this virtual dom and thats why it is really really fast . so class is reserved keyword in js and thats why we use classname

**<h1> React introduction</h1>**

**<p> welcome to react . It is an useful concept to learn</p>**

**<p>{this.props.greetings}</p> //** see below

**</div>**

**);**

**}**

**}**

**export default Hello;  //**if we want another file to use this we should export . default - it exports only one thing i.e., app

**ReactDOM.render(<Hello greetings={'wow react is good'+ 'but heavy'}/>, document.getElementById('root'));**

**//**in main js file we are using greeting property . and this can be used to get the property inside the html file.

**<p>{this.props.greetings}</p>**

**This is like a function //**react is like a function that takes parameter and rendered

Const Hello = (props) =>{

return (

**<div className="f1 tc">**

**<h1> React introduction</h1>**

**<p> welcome to react . It is an useful concept to learn</p>**

**<p>{props.greetings}</p>  //this is deleted as it is not an object anymore**

**</div>**

)

}

[**https://developers.google.com/web/fundamentals/primers/service-workers/**](https://developers.google.com/web/fundamentals/primers/service-workers/)

**Robo Friends**