# Phần 1 React

## Install

### Phần mềm cần sử dụng

Extends trình duyệt chorm

* react-detecter tool : xác định xem page nào dung reactjs
* cài đặt môi trường phát triển Nodejs (phía server) nodejs.org download và cài đặt.
* su dung yarn cung giống nodejs
* IDE có dụng babel, jsx, emmet

## MAIN CONCEPTS

### Khởi tạo PR

<https://reactjs.org/>

* Cai đặt npm install –g create-react-app
* -g là global gọi dc bất kì đâu
* Tạo mới project

npx create-react-app name\_project

cd name\_project

npm start (chạy project)

có thể đổi port mở package.json

"scripts": {

"start": "react-scripts start",

},

Them

"scripts": {

"start": "set port=4200 && react-scripts start",

},

Sau đó chạy lại start

Cấu trúc thư mục

* Public chứa file index.php,
* Src làm việc với thư mục này nhiều
* File package.json khai báo các thông tin và thư viện sử dụng cho pj

Component là phần xây dựng page ví dụ header, menu, footer, …

* Tạo hello word tạo file /src/Demo.js

import React, { Component } from 'react';  
class Demo extends Component {  
 render(){  
 return(  
 <div className="demo">  
 hello  
 </div>  
 );  
 }  
}  
export default Demo;

* Import Demo.js vào src/index.js

import React from 'react';  
import ReactDOM from 'react-dom';  
import './index.css';  
// import App from './App';  
import Demo from './Demo';  
import \* as serviceWorker from './serviceWorker';  
  
ReactDOM.render(<Demo />, document.getElementById('root'));  
  
// If you want your app to work offline and load faster, you can change  
// unregister() to register() below. Note this comes with some pitfalls.  
// Learn more about service workers: https://bit.ly/CRA-PWA  
serviceWorker.unregister();

render ra public/index.html với phần tử có id = root

### Component

Chèn boostrap vào project

* Download và cop vào public
* Gọi vào index.html sử dụng (jquery,scrip, css, …) js gọi ở dưới cùng trong body

%PUBLIC\_URL%/ : khai báo chỉ để thư mục public

* Component là chia nhỏ template để quản lý dễ dàng
* Tạo thư mục /src/component chứa các component con(header, footer, menu, …)
* Tạo dc 1 header component khi gọi trong appcomponent thì header component nằm trong app component và có thể gọi <Header /> hoặc <Header></Header> giống như html
* Tạo teamplate thì App.js là component cha gọi các component con vào (Home. Menu)

import React,{Component} from 'react';  
import './App.css';  
import Home from "./component/Home";  
import Menu from "./component/Menu";  
  
class App extends Component {  
 render()  
 {  
 return (  
 <div className="App">  
 <Menu></Menu>  
 <Home />  
 </div>  
 );  
 }  
 }  
  
 export default App;

* Chú ý phải viết đúng cú pháp thẻ đóng mở của component
* Nên chia nhỏ các component để dễ quản lý

### JSX

# Phần 2 Redux

# Phần 3 Router

# Phần 4 Connect API