# 0. Basic information

React có 2 dạng render:

\_ render phía client

\_ render phía server

<https://toidicodedao.com/2018/09/11/su-khac-biet-giua-server-side-rendering-va-client-side-rendering/>

## File index.js

File index.js nằm trong src là file script ở phía server. Nó dùng để render html, và có thể nhúng react vào để code.

## Public

Chứa tất cả các file tải về cho client. Bây giờ mọi request từ phía client gửi lên thì vào public để tìm.

## Div root

Div root dùng để hứng kết quả mà react render

## Render

Trong phần render của react của 2 phần:

\_ render cái gì

\_ render ở đâu

Render nằm trong Component, nó là 1 trạng thái

Component có rất nhiều trạng thái nhưng render là trạng thái quan trọng nhất.

## Autobind

<https://viblo.asia/p/react-es6-khong-con-autobind-EoDkQOZbGbV>

# 1. Environment and Setup

ReactJs cần NodeJs để chạy

NodeJs là một mã nguồn mở, đa nền tảng cho phát triển các ứng dụng phía Server và các ứng dụng liên quan đến mạng. Ứng dụng Node.js được viết bằng Javascript và có thể chạy trong môi trường Node.js trên hệ điều hành Window, Linux...

Node.js chỉ là một môi trường – điều này có nghĩa bạn tự phải làm mọi thứ. Sẽ chẳng có bất kỳ máy chủ mặc định nào cả !!! Một đoạn script xử lý tất cả các kết nối với Client. Điều này làm giảm đáng kể số lượng tài nguyên được sử dụng trong ứng dụng.

## setup nodejs.

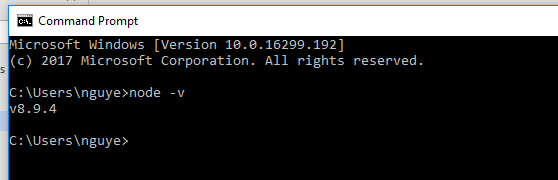
Download từ trang chủ nodejs.org rồi setup

node-v8.9.4-x64.msi

Default location: C:\Program Files\nodejs\

Check cmd:

Node –v



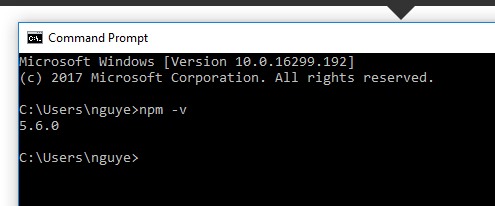
## 1.2 Setup npm

Cài đặt nodeJs là có sẵn npm rồi.

Tác dụng của npm:

Check version npm:

npm –v



## 1.3 List khoa phạm

<https://www.youtube.com/watch?v=tprWMUZc_ig&list=PLzrVYRai0riSPcINVFvaCaM7Ul55DzpLd&index=2>

# 2. Jsx là gì?

 JSX chỉ đơn giản là một cú pháp thay thế JavaScript, mặc dù cú pháp đó là đặc trưng của React (ít nhất ở thời điểm hiện tại). Ví dụ bên dưới được viết lại như sau:

React.render(

React.createElement('div', {}, 'Hello, world!'),

document.body

);

# 3. Hello world

React.render(

<div>Hello, world!</div>,

document.body

);

## 3.1 Hello world theo Khoa Pham thành công

Setup exress và ejs:

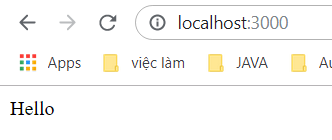
npm install express ejs

**var express** = ***require***(**"express"**);  
**var** app = **express**();  
app.use(**express**.**static**(**"public"**));  
app.set(**"view engine"**, **"ejs"**);  
app.set(**"views"**, **"./views"**);  
app.listen(3000); *// open port 3000 for client send request to server*app.get(**"/"**, **function** (req, res) {  
 res.render(**"home"**);  
});

mở command prompt và nhập lệnh sau:

node index.js

Result:



# 4. State là gì?

State cũng là 1 thuộc tính của React giống props, nhưng nó có thể thay đổi, và kéo theo cả lay-out thay đổi.

# 5. Khác nhau giữa Component và Purecomponent

Pureconponent có 1 phần shallow comparison có thể re-render bất cứ khi nào shouldComponentUpdate gọi. Do đó nó sẽ có hiệu suất tốt hơn component.

Tuy nhiên theo lời Duy nói thì chỉ nên sử dụng nó cho các data đơn giản.

# 6. Props

Props là thuộc tính của component. Props không bao giờ thay đổi.

<**ThuatNguyen name="ReactJS" teacher="Mr.Khoa"**>React Subject</**ThuatNguyen**>

Có 1 props đặc biệt là children

Ở đây React Subject là children

# 7. Fetch API

<https://viblo.asia/p/tim-hieu-fetch-api-p1-bWrZngbrlxw>

Fetch API là một API đơn giản cho việc gửi và nhận requesst bằng js. Với fetch thì việc thực hiện các yêu cầu web và xử lý phản hồi dễ dàng hơn so với XMLHttpRequest cũ.

Nếu trả về OK:

response.**ok**: true  
response.json(): là 1 Promise  
response.**status: 200**