name.substring (0, name.length-1) Xóa ký tự cuối cùng trong chuỗi

str.search('window.location.href') Tìm một chuỗi trong 1 chuỗi 🡺 trả về index

# Jquery

## Disable Div:

$('div.amsify-suggestags-area').css({'pointer-events':'none','opacity':'0.5'});

$(this).closest("#row-data-setting").find("div#tag-input").css({'pointer-events':'auto','opacity':'1'});

## Mouseout and mouseleave

$( "div.overout" )

.mouseout(**function**() {

$( "p:first", **this** ).text( "mouse out" );

$( "p:last", **this** ).text( ++i );

})

.mouseover(**function**() {

$( "p:first", **this** ).text( "mouse over" );

});

Lưu ý:

mouseout 🡪 đi ra

mouseover 🡪 đi vào

# REACT

## Thông tin cần thiết:

## Cài đặt môi trường

Sublime, nodejs

## Install react

Vào htdocs

* Mở git bash here

Tạo app mới: npx create-react-app my-app

cd my-app

Chạy local: npm start

## Component

React được xây dựng xung quanh component

class App extends Component { 🡺 “App” chính là một component

render() {

return (

<div className="App">

<header className="App-header">

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

<a

className="App-link"

href="https://reactjs.org"

target="\_blank"

rel="noopener noreferrer"

>

Learn React

</a>

</header>

</div>

);

}

}

## Thực hành

### Tạo app khác:

* Tạo trang src/Hello.js
* Muốn sử dụng được thì phải export ra export default Hello;
* Truy cập vào trang index.js để import app Hello vào

### Tích hợp bootstrap

Download bootstrap

Cóp tất cả thư mục (css, js) vào thư mục public

Nạp css:

* <link rel="stylesheet" href="%PUBLIC\_URL%/css/bootstrap.css" />

Nạp js:

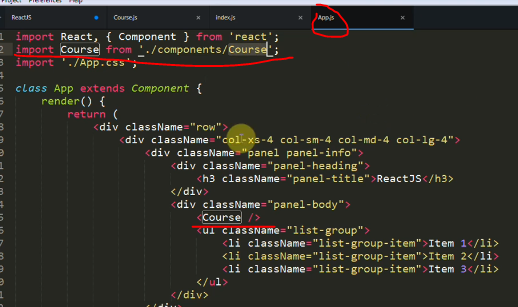
* <script src="%PUBLIC\_URL%/js/jquery-3.3.1.slim.min.js" type="text/javascript" charset="utf-8" async defer></script>
* <script src="/js/bootstrap.min.js" type="text/javascript" charset="utf-8" async defer></script>

### Tạo component

Tạo thư mục src/components và di chuyển Hello.js vào thư mục component.

Vào index.js để sửa đường dẫn:

* Nạp một component vào một component khác



Đây là ngôn ngữ jsx

C2: Không return về html

class Lesson extends Component {

render() {

return React.createElement('li',{className: 'list-group-item'}, '123');

}

}

* Có thể sử dụng các biểu thức expression:

{this.formatName(user)} <br/>

{2 + 2}

### Sử dụng Props, Event, Ref và State

#### Props:

<Course name="nguyen" />

<Course name="thi"/>

<Course name="ngoc"/>

<div className="panel-title">{this.props.name}</div> 🡪 Element này nằm

* Trường hợp để lấy giá trị trong thẻ Course

<Course name="nguyen" type="Họ">Nguyễn Thị Ngọc Trân</Course>

🡪 {this.props.children}

* Tạo hàm trong jsx

formatName(user){

return user.firstName + ' ' + user.lastName;

}

🡪 {this.formatName(user)}

#### Foreach

const elemCourses = arrays.map((item, index)=>

<Course key={index} name={item.name} type={item.time} free={item.free}/>

)

🡺 Lưu ý: phải có index để phân biệt được các phần tử

#### Event

* Không truyền tham số

handleClick(){

alert("hello");

};

<button onClick={this.handleClick} className="save-button btn btn-warning" type="button">View 1</button>

* Có truyền tham số

handleClick2(content){

alert(content);

};

<button onClick={() => this.handleClick2("ngoctran")} className="save-button btn btn-danger" type="button">View 2</button>

* Hàm dùng props

constructor(props){

super(props);

🡪 đẩy (this) vào handleClick3 để nó có thể hiểu là Course để nó có thể đọc được các thuộc tính props

this.handleClick3 = this.handleClick3.bind(this)

}

handleClick3(){

alert(this.props.name)

}

<button onClick={this.handleClick3} className="save-button btn btn-success" type="button">View 3</button>

#### Ref

constructor(props){

super(props);

this.handleClick3 = this.handleClick3.bind(this)

this.registerCourse = this.registerCourse.bind(this)

}

registerCourse(){

console.log(this.refs.username.value);

}

<div className="card-footer">

<div className="input-group mb-3">

<input type="text" className="form-control" placeholder="Search" ref="username"/>

<div className="input-group-append">

<button onClick={this.registerCourse} className="btn btn-success" type="submit">Go</button>

</div>

</div>

</div>

#### State

<p><button className="btn btn-success" type="submit">Toggle Outline</button></p>

* Thiết lập trạng thái ban đầu của một component

constructor(props){

super(props);

this.state = {

isShowOut: true

}

}

console.log(this.state); 🡪 In ra trong render

let ElmOutline = null;

if(this.state.isShowOut){

ElmOutline = <ul className="list-group">

<Lesson />

<Lesson />

<Lesson />

<Lesson />

<Lesson />

</ul>

}

## LifeCycle

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

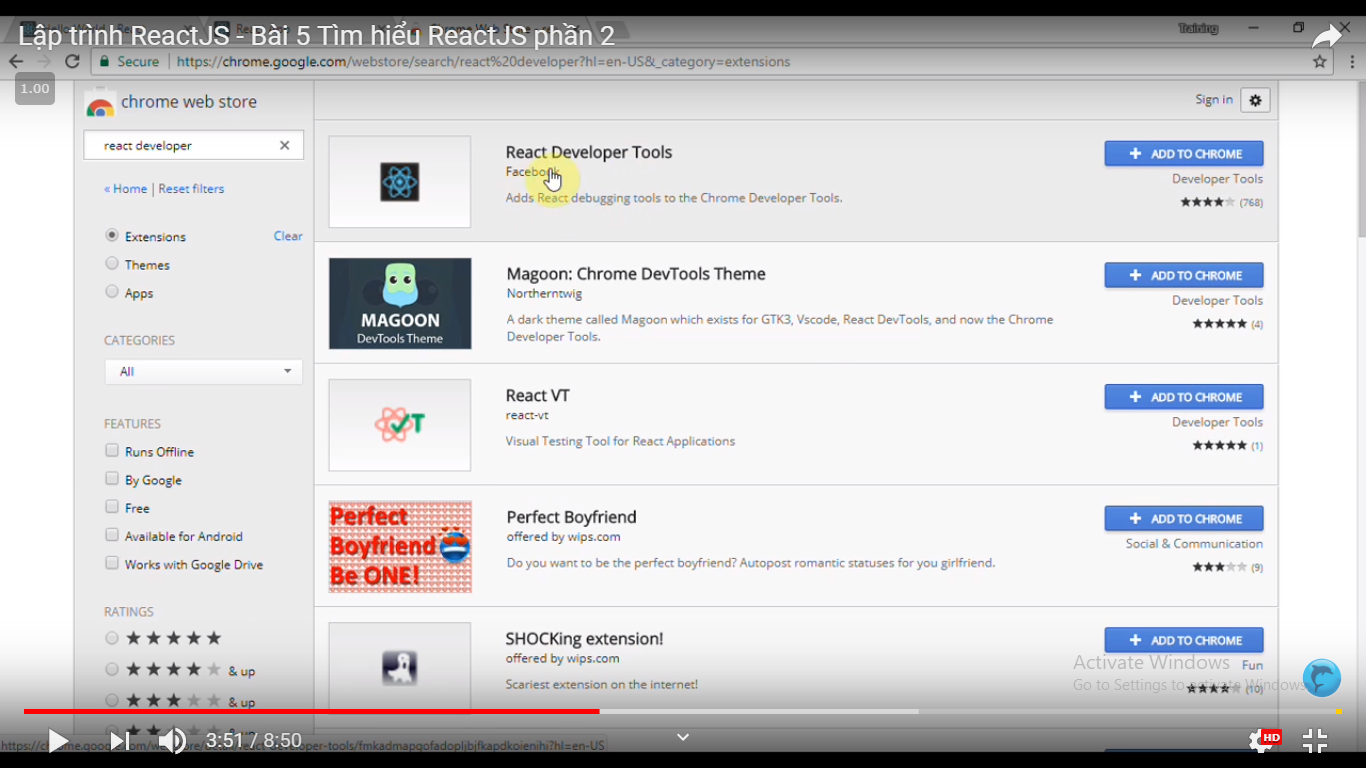
## Hàm

[React.createElement()](https://reactjs.org/docs/react-api.html#createelement) : Tạo element

## Khác

Cài đặt extend:

Vào phần cài đặt của trình duyệt -> công cụ khác 🡪 tiện ích mở rộng 🡪 cửa hang chrome trực tuyến



- Thư mục chúng ta sử dụng thường xuyên nhất đó src, mã nguồn chúng ta sẽ đặt trong src này

- Thực tế chúng ta không làm việc gì với thư mục node\_modules

Error:

Nếu project ko thể chạy được ko có module: node\_modules thì chạy:

Npm install 🡪 npm audit fix

# JSX – javascript systax extension

JSX – Javascript + XML cú pháp để thay thế javascript, cách viết gần giống XML thay thế React.createElement()

JSX không phải là HTML

<JSXElementName JSXAttributesopt>…</JSXElementName>

# Lưu ý:

Error: Syntax error: Adjacent JSX elements must be wrapped in an enclosing tag. Did you want a JSX fragment <>...</>?

🡪 Phải đặt tất cả các component trong <div></div>

* Khi một function chạy thì hàm render sẽ chạy lại

# Link:

Chuyển đổi từ html sang jsx: <https://magic.reactjs.net/htmltojsx.htm>

<https://www.youtube.com/watch?v=Zhq7cFIiRAw>

<https://www.youtube.com/watch?v=YTPu_XwyqpQ&list=PLei96ZX_m9sUs4aFBvxQDKAE1tg2kN01e&index=2>