**1.**

const JSX = <h1>Xin chào JSX!</h1>;

**2.**

const JSX = (

<div>

<h1>Tiêu đề</h1>

<p>Đoạn văn bản</p>

<ul>

<li>Mục 1</li>

<li>Mục 2</li>

<li>Mục 3</li>

</ul>

</div>

);

**3.**

const JSX = (

<div>

{*/\* Đây là comment trong JSX \*/*}

<h1>This is a block of JSX</h1>

<p>Here's a subtitle</p>

</div>

);

**4.**

ReactDOM.render(JSX, document.getElementById('jsx-node'));

**5.**

const JSX = (

<div className="myDIV">

<h1>This is a block of JSX</h1>

<p>Here's a subtitle</p>

</div>

);

**6.**

const JSX = (

<div>

<h2>Welcome to React!</h2>

<br />

<p>Be sure to close all tags!</p>

<hr />

</div>

);

**7.**

*// Dạng hàm thông thường*

const MyComponent = function() {

return <div>Nội dung bất kỳ</div>;

}

*// Dạng arrow function*

const MyComponent = () => {

return <div>Nội dung bất kỳ</div>;

}

**8.**

class MyComponent extends React.Component {

render() {

return <div>Nội dung bất kỳ</div>;

}

}

**9.**

const Fruits = () => {

return (

<div>

<TypesOfFruit />

</div>

);

};

class TypesOfFood extends React.Component {

constructor(props) {

super(props);

}

render() {

return (

<div>

<h1>Types of Food:</h1>

<Fruits />

</div>

);

}

}

**10.**

const CurrentDate = (props) => {

return (

<div>

<p>The current date is: {props.date}</p>

</div>

);

};

class Calendar extends React.Component {

constructor(props) {

super(props);

}

render() {

return (

<div>

<h3>What date is it?</h3>

<CurrentDate date={Date()} />

</div>

);

}

}

**11.**

const List = (props) => {

return <p>{props.tasks.join(", ")}</p>

};

class ToDo extends React.Component {

constructor(props) {

super(props);

}

render() {

return (

<div>

<h1>To Do Lists</h1>

<h2>Today</h2>

<List tasks={["taskA", "taskB"]} />

<h2>Tomorrow</h2>

<List tasks={["taskC", "taskD", "taskE"]} />

</div>

);

}

}

**12.**

class StatefulComponent extends React.Component {

constructor(props) {

super(props);

this.state = {

firstName: 'Your Name'

};

}

render() {

return (

<div>

<h1>{this.state.firstName}</h1>

</div>

);

}

}

**13.**

class StatefulComponent extends React.Component {

constructor(props) {

super(props);

this.state = {

firstName: 'Your Name'

};

}

render() {

const firstName = this.state.firstName;

return (

<div>

<h1>{firstName}</h1>

</div>

);

}

}

**14.**

class MyComponent extends React.Component {

constructor(props) {

super(props);

this.state = {

name: 'Initial State'

};

this.handleClick = this.handleClick.bind(this);

}

handleClick() {

this.setState({

name: 'React Rocks'

});

}

render() {

return (

<div>

<button onClick={this.handleClick}>Click Me</button>

<h1>{this.state.name}</h1>

</div>

);

}

}

**15.**

*// Cách 1: Arrow function*

class MyComponent extends React.Component {

constructor(props) {

super(props);

this.state = {

text: "Hello"

};

}

handleClick = () => {

this.setState({

text: "You clicked!"

});

}

render() {

return (

<div>

<button onClick={this.handleClick}>Click Me</button>

<h1>{this.state.text}</h1>

</div>

);

}

}

*// Cách 2: Bind*

class MyComponent extends React.Component {

constructor(props) {

super(props);

this.state = {

text: "Hello"

};

this.handleClick = this.handleClick.bind(this);

}

handleClick() {

this.setState({

text: "You clicked!"

});

}

render() {

return (

<div>

<button onClick={this.handleClick}>Click Me</button>

<h1>{this.state.text}</h1>

</div>

);

}

}

**16. Counter component**

*// Cách 1: Bind*

class Counter extends React.Component {

constructor(props) {

super(props);

this.state = {

count: 0

};

this.increment = this.increment.bind(this);

this.decrement = this.decrement.bind(this);

this.reset = this.reset.bind(this);

}

increment() {

this.setState(state => ({

count: state.count + 1

}));

}

decrement() {

this.setState(state => ({

count: state.count - 1

}));

}

reset() {

this.setState({

count: 0

});

}

render() {

return (

<div>

<button className='inc' onClick={this.increment}>Increment!</button>

<button className='dec' onClick={this.decrement}>Decrement!</button>

<button className='reset' onClick={this.reset}>Reset</button>

<h1>Current Count: {this.state.count}</h1>

</div>

);

}

}

*// Cách 2: Arrow function*

class Counter extends React.Component {

constructor(props) {

super(props);

this.state = {

count: 0

};

}

increment = () => {

this.setState(state => ({

count: state.count + 1

}));

}

decrement = () => {

this.setState(state => ({

count: state.count - 1

}));

}

reset = () => {

this.setState({

count: 0

});

}

render() {

return (

<div>

<button className='inc' onClick={this.increment}>Increment!</button>

<button className='dec' onClick={this.decrement}>Decrement!</button>

<button className='reset' onClick={this.reset}>Reset</button>

<h1>Current Count: {this.state.count}</h1>

</div>

);

}

}

**17. Controlled Input**

class ControlledInput extends React.Component {

constructor(props) {

super(props);

this.state = {

input: ''

};

this.handleChange = this.handleChange.bind(this);

}

handleChange(event) {

this.setState({

input: event.target.value

});

}

render() {

return (

<div>

<input

value={this.state.input}

onChange={this.handleChange}

/>

<h4>Controlled Input:</h4>

<p>{this.state.input}</p>

</div>

);

}

}

**18.**

class MyForm extends React.Component {

constructor(props) {

super(props);

this.state = {

input: '',

submit: ''

};

this.handleChange = this.handleChange.bind(this);

this.handleSubmit = this.handleSubmit.bind(this);

}

handleChange(event) {

this.setState({

input: event.target.value

});

}

handleSubmit(event) {

event.preventDefault();

this.setState({

submit: this.state.input

});

}

render() {

return (

<div>

<form onSubmit={this.handleSubmit}>

<input

value={this.state.input}

onChange={this.handleChange}

/>

<button type='submit'>Submit!</button>

</form>

<h1>{this.state.submit}</h1>

</div>

);

}

}

**19.**

class MyApp extends React.Component {

constructor(props) {

super(props);

this.state = {

name: 'CamperBot'

}

}

render() {

return (

<div>

<Navbar name={this.state.name} />

</div>

);

}

}

class Navbar extends React.Component {

constructor(props) {

super(props);

}

render() {

return (

<div>

<h1>Hello, my name is: {this.props.name}</h1>

</div>

);

}

}

**20.**

class Colorful extends React.Component {

render() {

return (

<div style={{

color: 'red',

fontSize: '72px',

border: '2px solid blue'

}}>

Big Red

</div>

);

}

}

**21.**

class MyComponent extends React.Component {

constructor(props) {

super(props);

this.state = {

display: true

}

this.toggleDisplay = this.toggleDisplay.bind(this);

}

toggleDisplay() {

this.setState((state) => ({

display: !state.display

}));

}

render() {

return (

<div>

<button onClick={this.toggleDisplay}>Toggle Display</button>

{this.state.display && <h1>Displayed!</h1>}

</div>

);

}

}