

State trong ReactJS

- State được tạo ra và quản lý bởi component hiện tại.
- State được dùng cho những dữ liệu có khả năng sẽ thay đổi.

Ví dụ component ColorBox có state `color` sẽ thay đổi khi click các nút để đổi màu tương ứng.

```
// Functional Component
function ColorBox(props) {
  const [color, setColor] = useState('white');

  return (
    <div>
      {color}

      <button onClick={() => setColor('black')}>Change to black</button>
      <button onClick={() => setColor('white')}>Change to white</button>
    </div>
  );
}
```

```
// Class component
class ColorBox extends PureComponent {
  constructor(props) {
    super(props);

    this.state = {
      color: 'white',
    }
  }

  render() {
    return (
      <div>
        {this.state.color}

        <button onClick={() => this.setState({ color: 'black' })}>
          Change to black
        </button>

        <button onClick={() => this.setState({ color: 'white' })}>
          Change to white
        </button>
      </div>
    );
  }
}
```

Ví dụ component Counter có state `count` sẽ thay đổi mỗi khi click nút `Increase`.

```
// Functional Component
function Counter(props) {
  const [count, setCount] = useState(0);

  return (
    <div>
      {count}

      <button onClick={() => setCount(x => x + 1)}>Increase</button>
    </div>
  );
}
```

```
class Counter extends PureComponent {
  constructor(props) {
    super(props);

    // State declaration
    this.state = {
      count: 0,
    }
  }

  render() {
    return (
      <div>
        {this.state.count}

        {/* use this.setState() function to update state */}
        <button
          onClick={() => this.setState(prevState => ({
            count: prevState.count + 1
          })))}
        >
          Increase
        </button>
      </div>
    );
  }
}
```