



JSX

1. Viết code JSX
2. Biểu thức trong JSX
3. Chèn một khối HTML
4. Thuộc tính
5. Câu lệnh if

## JSX là gì?

- × JSX là viết tắt của JavaScript XML
- × JSX cho phép người lập trình viết HTML trong React.
- × JSX giúp việc viết và thêm HTML trong React dễ dàng hơn.

- × JSX cho phép người lập trình viết các phần tử HTML bằng JavaScript và đặt chúng vào DOM mà không cần bất kỳ phương thức `createElement()` và/hoặc `appendChild()` nào.

## Ví dụ: sử dụng JSX

```
const myElement = <h1>I Love JSX!</h1>;  
  
const root = ReactDOM.createRoot(document.getElementById('root'));  
root.render(myElement);
```

## Ví dụ: không sử dụng JSX

```
const myElement = React.createElement('h1', {}, 'I do not use JSX!');  
  
const root = ReactDOM.createRoot(document.getElementById('root'));  
root.render(myElement);
```

Có thể thấy trong ví dụ đầu tiên, JSX cho phép viết HTML trực tiếp trong mã JavaScript. JSX là phần mở rộng của ngôn ngữ JavaScript dựa trên ES6 và được dịch sang JavaScript thông thường khi chạy.

- × Với JSX chúng ta có thể viết biểu thức bên trong dấu ngoặc nhọn { }.
- × Biểu thức có thể là biến React, thuộc tính hoặc bất kỳ biểu thức JavaScript hợp lệ nào khác. JSX sẽ thực thi biểu thức và trả về kết quả

Ví dụ:

```
const myElement = <h1>React is {5 + 5} times better with JSX</h1>;
```

- × Để viết HTML trên nhiều dòng, hãy đặt HTML bên trong dấu ngoặc đơn

```
const myElement = (  
  <ul>  
    <li>Apples</li>  
    <li>Bananas</li>  
    <li>Cherries</li>  
  </ul>  
);
```

Một thẻ Html cấp 1.

- × Code HTML phải nằm trong một thẻ html cấp 1 (Có nghĩa là chỉ có một thẻ html cấp 1 duy nhất).

```
const myElement = (  
  <div>  
    <p>I am a paragraph.</p>  
    <p>I am a paragraph too.</p>  
  </div>  
);
```

Một thẻ Html cấp 1.

- × JSX sẽ đưa ra lỗi nếu HTML không chính xác hoặc nếu HTML thiếu phần tử gốc.

```
const myElement = (  
  <>  
    <p>I am a paragraph.</p>  
    <p>I am a paragraph too.</p>  
  </>  
);
```

Một thẻ Html cấp 1.

- × JSX sẽ đưa ra lỗi nếu HTML không chính xác hoặc nếu HTML thiếu phần tử gốc.

```
const myElement = (  
  <>  
    <p>I am a paragraph.</p>  
    <p>I am a paragraph too.</p>  
  </>  
);
```

Thuộc tính: `class = className`

- × Thuộc tính `class` là thuộc tính được sử dụng nhiều trong HTML, nhưng vì JSX được hiển thị dưới dạng JavaScript và từ khóa lớp là một từ dành riêng trong JavaScript nên bạn không được phép sử dụng nó trong JSX.

- × Thay vào đó, JSX đã giải quyết vấn đề này bằng cách sử dụng `className`. Khi JSX được hiển thị, nó sẽ dịch các thuộc tính `className` thành thuộc tính `class`.

Ví dụ:

```
const myElement = <h1 className="myclass">Hello World</h1>;
```

- × React hỗ trợ các câu lệnh if, nhưng không hỗ trợ bên trong JSX.
- × Để có thể sử dụng các câu lệnh có điều kiện trong JSX, nên đặt các câu lệnh if bên ngoài JSX hoặc thay vào đó có thể sử dụng biểu thức ba ngôi

## Ví dụ 1:

```
const x = 5;
let text = "Goodbye";
if (x < 10) {
  text = "Hello";
}

const myElement = <h1>{text}</h1>;
```

## Ví dụ 2:

```
const x = 5;  
  
const myElement = <h1>{(x) < 10 ? "Hello" : "Goodbye"}</h1>;
```