



---

# JSX

# Mục tiêu

---



- Giới thiệu cú pháp JSX
- Cách sử dụng, khai báo , truyền props trong JSX
- Phân biệt giữa cú pháp JSX và `React.createElement`

# React Element ,cơ chế Rendering Elements

---



## Giới thiệu

- Trong React, Element là những mảnh ghép nhỏ nhất của các ứng dụng React.
- Không giống như những element DOM của trình duyệt, React element là những “đối tượng đơn giản” (plain object) và rất dễ để tạo ra.
- React DOM giữ vai trò cập nhật DOM để phù hợp với các React element.

# React Element ,cơ chế Rendering Elements

---

Nội dung bài đọc sẽ có các phần chính sau:

- React Element
- Rendering Element

# React Element là gì?

---



- React element chỉ là một đối tượng JavaScript cũ đơn thuần mà không có các phương thức riêng, là những gì được trả về từ các components.
- Nó mô tả những gì chúng ta muốn thấy trên màn hình, nó là thành phần đại diện của DOM
- Với function component, element là đối tượng mà hàm trả về
- Với class component, là đối tượng mà hàm render của thành phần đó trả về

# React Element là gì?

---



## React Element là gì?

- React element không phải là những gì chúng ta thấy trong trình duyệt. Chúng chỉ là những đối tượng trong bộ nhớ và chúng ta không thể thay đổi.
- React element có thể có các thuộc tính kiểu khác với các phần tử HTML gốc

# React Element là gì?

---



React.createElement có cú pháp sau:

**React.createElement(type, [props], [...children])**

- type có thể là một thẻ HTML như h1, div hoặc nó có thể là React component
- prop là các thuộc tính bạn muốn phần tử có
- children chứa các thẻ HTML khác hoặc có thể là component

# React Element là gì?

---



## Ví dụ:

```
const hello = React.createElement(  
  "H1",  
  {id: "msg", className: "title"},  
  "Hello React Element"  
);  
ReactDOM.render(  
  hello,  
  document.getElementById('root')  
);
```

## Kết quả

```
{  
  type: 'H1',  
  props: {  
    id: 'msg',  
    class: 'title'  
    children:  
      'HelloReactElement '  
  }  
}
```



# Rendering Elements trong DOM

---



Để render một phần tử React vào root DOM, thì ta dùng phương thức **ReactDOM.render()**

**Cú pháp :**

**ReactDOM.render(element, container[, callback])**

# Rendering Elements trong DOM

---



- Phương thức sẽ render phần tử React vào DOM trong vùng chứa được cung cấp và trả về một tham chiếu đến thành phần (hoặc trả về null cho các thành phần không trạng thái).
- Nếu phần tử React trước đó đã được hiển thị vào vùng chứa, thì phần tử này sẽ thực hiện cập nhật nó và chỉ thay đổi DOM khi cần thiết để phản ánh phần tử React mới nhất.
- Nếu lệnh gọi lại tùy chọn được cung cấp, nó sẽ được thực thi sau khi thành phần được hiển thị hoặc cập nhật.

# Rendering Elements trong DOM

---



**Ví dụ:**

```
const items = ["C++", "PHP", "JAVA"];
const programming = React.createElement("section", {id:
"coding"},
  React.createElement("h1", null, "Programming Languages"),
  React.createElement("ul", {className: "coding"},
    items.map((coding, i) =>
      React.createElement("li", { key: i }, coding),
    );
  );

ReactDOM.render(programming, document.getElementById('root')
);
```

# Tổng kết

---



- Khái niệm React Element, và phương thức tạo element trong React
- Phương thức để render trong DOM