***JSX***

1. ***JSX là gì? Vì sao nên sử dụng JSX?***

**JSX là: JSX = Javascript + XML**

* JSX là viết tắt của JavaScript XML.
* JSX là một ngôn ngữ cho phép bạn kết hợp JavaScript và các thẻ giống HTML để xác định các phần tử giao diện người dùng (UI) và chức năng của chúng.
* JSX là một phần mở rộng cho JavaScript cho phép viết những gì trông giống như HTML và XML trong JavaScript.
* JSX là cách viết tắt để gọi React.createElement () để tạo các phần tử và thành phần (Babel biên dịch JSX thành những câu gọi hàm React.createElement()).
* JSX trông giống như HTML và cung cấp một cú pháp thân thiện để tạo các phần tử DOM.
* JSX chấp nhận bất kỳ biểu thức JavaScript hợp lệ nào bên trong dấu ngoặc nhọn.

**Vì sao nên sử dụng JSX**

* Thứ nhất, JSX với cú pháp gần giống XML, cấu trúc cây khi biểu thị các attributes, điều đó giúp ta dễ dàng định nghĩa, quản lý được các component phức tạp, thay vì việc phải định nghĩa và gọi ra nhiều hàm hoặc object trong javascript. Khi nhìn vào cấu trúc đó cũng dễ dàng đọc hiểu được ý nghĩa của các component. Code JSX ngắn hơn, dễ hiểu hơn code JS.
* Thứ 2, JSX không làm thay đổi ngữ nghĩa của Javascript
* Thứ 3, với cách viết gần với các thẻ HTML, nó giúp những developers thông thường (ví dụ như các designer) có thể hiểu được một cách dễ dàng, từ đó có thể viết hoặc sửa code mà không gặp nhiều khó khăn.

***2. Trình duyệt có thể đọc được JSX không? Làm sao để trình duyệt có thể đọc được jsx***

+)Trình duyệt **không thể trực tiếp đọc JSX** vì nó không phải là JavaScript thuần. JSX là một cú pháp mở rộng của JavaScript, được dùng trong React để viết mã dễ dàng hơn bằng cách cho phép nhúng mã HTML trực tiếp vào JavaScript. Để trình duyệt có thể đọc được JSX, bạn cần chuyển đổi nó thành JavaScript thông qua **trình biên dịch**.

+)Công cụ phổ biến nhất để chuyển đổi JSX thành JavaScript là **Babel**. Babel biên dịch mã JSX thành mã JavaScript mà trình duyệt có thể hiểu được.

***3. Làm sao có thể viết mã js và trong JSX.***

Nhúng mã JavaScript vào JSX bằng cách sử dụng cặp ngoặc nhọn {}.

Vd:

const isLoggedIn = true;

const element = <h1>{isLoggedIn ? "Welcome back!" : "Please log in"}</h1>;

***4. Các lưu ý khi viết mã JSX***

 **JSX phải trả về một phần tử đơn:**  
Tất cả các phần tử JSX phải được bọc trong một phần tử bao ngoài (như <div>, <React.Fragment>, hoặc <> - fragment ngắn gọn).

 **Các thuộc tính của HTML trong JSX theo camelCase:**  
Trong JSX, các thuộc tính HTML phải sử dụng định dạng camelCase.

 **Tránh viết logic phức tạp trong JSX:**  
JSX được thiết kế để biểu diễn UI, nên việc viết logic phức tạp trong JSX có thể làm cho mã khó đọc. Nên tách logic ra các hàm hoặc sử dụng biến để giữ JSX đơn giản.

 **Không thể sử dụng từ khóa JavaScript như "class", "for":**  
Những từ khóa như class và for là từ khóa của JavaScript, vì vậy chúng được thay thế trong JSX:

* class trở thành className.
* for trở thành htmlFor.