

Khái niệm, tác dụng của Hook

Hooks mới được thêm vào trong React 16.8. Chúng cho phép sử dụng state và những tính năng khác của React mà không cần phải dùng tới class.

Hooks không hoạt động bên trong các class. Nhưng chúng ta có thể sử dụng hooks thay vì dùng class.

Hook là gì?

- Hooks là một bổ sung mới trong React 16.8.
- Hook là một hàm đặc biệt cho phép bạn sử dụng các tính năng của React (mà không cần phải tạo class). Ví dụ, `useState` là một hook cho phép bạn thêm React state vào function components

Vì sao cần dùng React Hook

Các vấn đề khi gặp phải khi làm việc với React:

- “Wrapper hell” các component được lồng (nested) vào nhau nhiều tạo ra một DOM tree phức tạp.
- Component quá lớn.
- Sự rắc rối của Lifecycles trong class

Lợi ích của Hook

- Khiến các component trở nên gọn nhẹ hơn
- Giảm đáng kể số lượng code, dễ tiếp cận
- Cho phép chúng ta sử dụng state ngay trong function component

Sử dụng hook `useState`

Khai báo một biến `state`

Bên trong một function component, chúng ta không có `this`, cho nên chúng ta không thể cài đặt hoặc đọc `this.state`. Thay vào đó, chúng ta gọi `useState` Hook trực tiếp bên trong component:

```
import React, { useState } from 'react';

function Example() {
  // Khai báo một biến state mới, gọi nó là "count"
  const [count, setCount] = useState(0);}
```

Hàm `useState` làm gì khi được gọi?

Nó khai báo một “state variable” (biến state). Biến này gọi là `count` nhưng ta có thể gọi nó với bất kì tên nào, ví dụ gọi là `banana`. Đây là cách để “lưu giữ” các giá trị giữa các lần gọi hàm — `useState` là một cách mới để sử dụng như là cách `this.state` được dùng trong class. Thông thường, các biến này “biến mất” khi hàm kết thúc nhưng các biến state này được React giữ lại.

Hàm `useState` nhận tham số gì?

Tham số duy nhất được truyền vào hook `useState()` là state ban đầu. Không giống như khai báo với Class, state không cần thiết phải là object mà có thể là số hoặc chuỗi. Với ví dụ trên, ta chỉ cần biết người dùng click bao nhiêu lần, vì vậy ta truyền vào giá trị khởi tạo là `0`. (Nếu ta muốn lưu hai giá trị khác nhau, ta sẽ gọi `useState()` hai lần.)

Hàm `useState` trả về gì?

Nó trả về một cặp giá trị dưới dạng mảng: `state` hiện tại và một hàm để update nó. Đây là lý do chúng ta viết `const [count, setCount] = useState()`. Cặp này tương tự như `this.state.count` và `this.setState` trong class, khác là ta dùng chúng theo cặp.

Đọc state

Khi muốn hiển thị giá trị hiện tại của state `count` trong hàm (sử dụng với hooks), chúng ta dùng trực tiếp biến `count`:

```
<p>Bạn đã click{count} lần</p>
```

Updating state

Sử dụng biến `setCount`

```
<button onClick={() => setCount(count + 1)}>
  Click vào
</button>
```