

 Cập nhật tháng 8 năm 2024

[Bài Đọc] Higher order function (HOF) trong javascript

Higher Order Functions (HOF) là một trong những khái niệm quan trọng trong JavaScript, đặc biệt khi làm việc với các cấu trúc dữ liệu và lập trình hàm. Dưới đây là một bài tổng quan về Higher Order Functions:

1. Higher order function trong javascript

- **Định nghĩa:**
 - Nhận một hàm khác làm tham số
 - Trả về một hàm khác như kết quả
- Lợi ích của Higher order function
 - **Tái sử dụng mã:** Giúp viết các đoạn mã có thể tái sử dụng và dễ đọc hơn
 - **Trừu tượng hóa:** Tách biệt logic chung và các chi tiết thực hiện
 - **Dễ quản lý và bảo trì:** Giảm sự phức tạp trong mã khi xử lý dữ liệu hoặc logic

2. Ví dụ phổ biến

- HOF nhận một hàm làm tham số
 - Các hàm như **map**, **filter**, và **reduce** là những ví dụ phổ biến của HOF:
 - **map()**
 - Dùng để tạo một mảng mới bằng cách áp dụng một hàm lên từng phần tử của mảng cũ.

```
const numbers = [1, 2, 3];  
const doubled = numbers.map(callbackfn: num => num * 2);  
console.log(...data: doubled); // [2, 4, 6]
```

- **filter()**

- Trả về một mảng mới chỉ chứa các phần tử thỏa mãn điều kiện.

```
const numbers = [1, 2, 3, 4];
const evens = numbers.filter(predicate: num => num % 2 === 0);
console.log(...data: evens); // [2, 4]
```

- **reduce()**

- Giảm mảng thành một giá trị duy nhất.
- Có thể gọi là hàm tích lũy

```
const numbers = [1, 2, 3, 4];
const sum = numbers.reduce(callbackfn: (total, num) => total + num, initialValue: 0);
console.log(...data: sum); // 10
```

- **HOF trả về một hàm khác**

- Một HOF có thể tạo ra các hàm tùy chỉnh:

```
function multiplier(factor) {
  return function (num) {
    return num * factor;
  };
}

const double = multiplier(factor: 2);
console.log(...data: double(num: 5)); // 10
```

3. Ứng dụng của HOF

- **Xử lý dữ liệu mảng:** Giảm số lượng mã cần viết khi làm việc với mảng và dữ liệu phức tạp.
- **Currying và Partial Application:** Tạo ra các hàm tùy chỉnh.
- **Callbacks và Promises:** Sử dụng trong lập trình bất đồng bộ (asynchronous programming).
- **Thành phần chức năng (Functional Components):** Thường dùng trong các thư viện như React để tạo các thành phần UI.

4. Tài liệu tham khảo

<https://www.freecodecamp.org/news/higher-order-functions-in-javascript-explained/>

Danh sách các bài học

