

# useCallback() vs useMemo() 😎

Presented by: Hau Nguyen | Easy Frontend

1. useCallback() là gì?
2. useMemo() là gì?
3. So sánh useCallback() vs useMemo()
4. Có nên sử dụng useCallback() vs useMemo() hay không?
5. Link tham khảo

## 1. useCallback() là gì?

Là một react hooks giúp mình tạo ra một **memoized callback** và chỉ tạo ra callback mới khi **dependencies thay đổi**.

- Nhận vào 2 tham số: 1 là **function**, 2 là **dependencies**.
- Return **memoized callback**
- Chỉ tạo ra function mới khi dependencies thay đổi.
- Nếu dùng empty dependencies thì không bao giờ tạo ra function mới.

```
// Mỗi lần App re-render
// --> tạo ra một function mới
// --> Chart bị re-render
function App() {
  const handleChartTypeChange = (type) => {}
  return <Chart onChange={handleChartTypeChange} />;
}
```

```
// Mỗi lần App re-render
// --> nhờ có useCallback() nó chỉ tạo function một lần đầu
// --> Nên Chart ko bị re-render.
function App() {
  const handleChartTypeChange = useCallback((type) => {}, [])
  return <Chart onChange={handleChartTypeChange} />;
}
```

## 2. useMemo() là gì?

Là một react hooks giúp mình tạo ra một **memoized value** và chỉ tính toán ra value mới khi **dependencies thay đổi**.

- Nhận vào 2 tham số: 1 là **function**, 2 là **dependencies**.
- Return **memoized value**
- Chỉ tính toán value mới khi dependencies thay đổi.
- Nếu dùng empty dependencies thì không bao giờ tính toán lại value mới.

```
// Mỗi lần App re-render
// --> tạo ra một mảng mới
// --> Chart bị re-render do props thay đổi
function App() {
  const data = [{}, {}, {}];
  return <Chart data={data} />;
}
```

```
// Mỗi lần App re-render
// --> nhờ có useMemo() nó chỉ tạo ra cái mảng 1 lần đầu
// --> Nên Chart ko bị re-render.
function App() {
  const data = useMemo(() => [{}, {}, {}], [])
  return <Chart data={data} />;
}
```

## 3. So sánh useCallback() vs useMemo()

### GIỐNG NHAU

- Đều áp dụng kĩ thuật memoization.
- Đều nhận vào 2 tham số: function và dependencies.
- Đều là react hooks, dùng cho functional component.
- Dùng để hạn chế những lần re-render dư thừa (micro improvements).

### KHÁC NHAU

#	useCallback()	useMemo()
return	memoized <b>callback</b>	memoized <b>value</b>
code	useCallback( <b>(type)</b> => {}, [])	useMemo(() => [{}, {}, {}], [])

## 4. Có nên sử dụng useCallback() vs useMemo() hay không?

- Không nên dùng cho tất cả components.
- Nên dùng cho: đồ thị, biểu đồ, animations, những component nặng phần render.
- Chỉ là micro improvements.

## 5. Link tham khảo

- useCallback() hooks: <https://reactjs.org/docs/hooks-reference.html#usecallback>
- useMemo() hooks: <https://reactjs.org/docs/hooks-reference.html#usememo>
- Một bài blog rất hay về useMemo và useCallback: <https://kentcdodds.com/blog/usememo-and-usecallback>
- <https://medium.com/@jan.hesters/usecallback-vs-usememo-c23ad1dc60>
- <https://dev.to/dinhhuysam/introduction-to-react-memo-usememo-and-usecallback-5ei3>

Cảm ơn các bạn đã theo dõi video của mình! ❤️

Nhớ like, share và subscribe để mọi người cùng xem nhé 😊

Còn bạn nào muốn donate cho mình ly cafe thì tìm link trong phần mô tả video nha 😊