

Slot33 – Memo, callback, transition, lazy (Tối ưu hiệu năng ReactJS)

1. Memo: - ghi nhớ components

=> ngăn component render lại không cần thiết khi props không thay đổi

=> memo: ngăn không cho component reloading

ví dụ:

Khi 1 component cha re-render => mọi component con đều re-render mặc định => sử dụng memo để chỉ render component có props thay đổi

=>

Ngoài ra:

memo có thể lưu giá trị tính toán phức tạp để lần sau dùng mà không cần tính toán lại

2. callback: ghi nhớ hàm

Tránh tạo hàm mới mỗi lần rendering

Tránh truyền props khác nhau

Tránh re-render con

ví dụ:

nếu không sử dụng callback => hàm mới sẽ liên tục được gọi mỗi khi rendering

=> nếu sử dụng callback => hàm được ghi nhớ => sẽ được gọi chỉ khi có sự thay đổi dependencies

3. transition:

giúp giao diện mượt (React 18+)

tách xử lý nền và hiển thị trên giao diện

4. Lazy: Tải component theo nhu cầu (lazy loading)
Chia nhỏ các bundle => chỉ tải component khi cần
(Code Splitting)

===

-Memo: Ghi nhớ component (dùng khi component nhận props không đổi)

Lưu trữ tính toán phức tạp

-Callback: ghi nhớ hàm (khi truyền hàm xuống component con)

-Transititon: ưu tiên cập nhật giao diện (sử dụng khi thao tác nặng khiến UI giật lag)

-Lazy: Tải component theo nhu cầu (Sử dụng khi bundle lớn, cần tối ưu tốc độ tải)

===