

## **1. What is the purpose of useEffect in React?**

useEffect is used to handle side effects in a React component. Side effects include API calls, subscriptions, timers, event listeners, and DOM updates. It runs after the component renders and can also clean up resources when the component unmounts or before re-running the effect.

## **2. What problem does useMemo solve?**

useMemo solves performance issues caused by expensive calculations running on every render. It memoizes the computed value and recalculates it only when its dependencies change.

## **3. Why do we use useCallback in React?**

useCallback is used to memoize functions. It prevents functions from being recreated on every render, which helps avoid unnecessary re-renders of child components.

## **4. Difference between useEffect and useMemo?**

useEffect is used for performing side effects after rendering. useMemo is used for performance optimization by memoizing values during rendering.

## **5. Difference between useMemo and useCallback?**

useMemo memoizes a returned value, whereas useCallback memoizes a function reference. Internally, useCallback is a special case of useMemo.

## **6. What happens if you don't provide a dependency array to these hooks?**

If no dependency array is provided, the hook runs on every render. This may cause performance issues or infinite loops, especially in useEffect.

## **7. When does useMemo recompute its value?**

useMemo recomputes its value only when one or more values in its dependency array change.

## **8. How does useCallback prevent unnecessary child re-renders?**

By returning the same function reference between renders (when dependencies don't change), React.memo-wrapped child components do not re-render unnecessarily.

## **9. Can we perform API calls inside useMemo or useCallback? Why?**

No. API calls are side effects and should only be performed inside useEffect. useMemo and useCallback must be pure and synchronous.

## **10. Real-time scenario using useEffect, useMemo, and useCallback together**

Example: An e-commerce product list. useEffect fetches products from an API. useMemo filters products based on search input. useCallback memoizes the add-to-cart function passed to child components to avoid unnecessary re-renders.