

# Performance Best Practices - memo(componentName)

- Memo Higher Order Component can be used to stop re-rendering child components after the parent has been updated.
- By default, all child components will render again if the parent has been updated

```
import { memo } from "react";

function ChildComponent() {
  return (
    <div className="App">
      Child Component
    </div>
  );
}

export default memo(ChildComponent);
```

## `useCallback( () => {}, [dependencies] )`

The `useCallback` is a react hook that returns a **memoized callback** when passed a function and a list of dependencies as parameters.

It's very useful to avoid Re-Rendering of child components **because by default, React will render all child components after any state change inside the parent**

It only changes the callback when one of its dependencies gets changed.

Is useful for to improve performance of an app, you will only re-render components that need re-rendering