

REACT NATIVE

Lecture 12: React Hooks

Shubhang Sharma
Taraksh Goyal
Deepak Soni

Cognition 4.0
2025

UPES MTC

Hooks in React ~

- State Hooks: `useState` , `useReducer`
- Context Hooks: `useContext`
- Ref Hooks: `useRef` , `useImperativeHandle`
- Effect Hooks: `useEffect`, `useLayoutEffect`, `useInsertionEffect`
- Performance Hooks: `useMemo` , `useCallback` , `useTransition`,
`useDeferredValue`
- Other Hooks: `useDebugValue`, `useId` , `useActionState`

useState

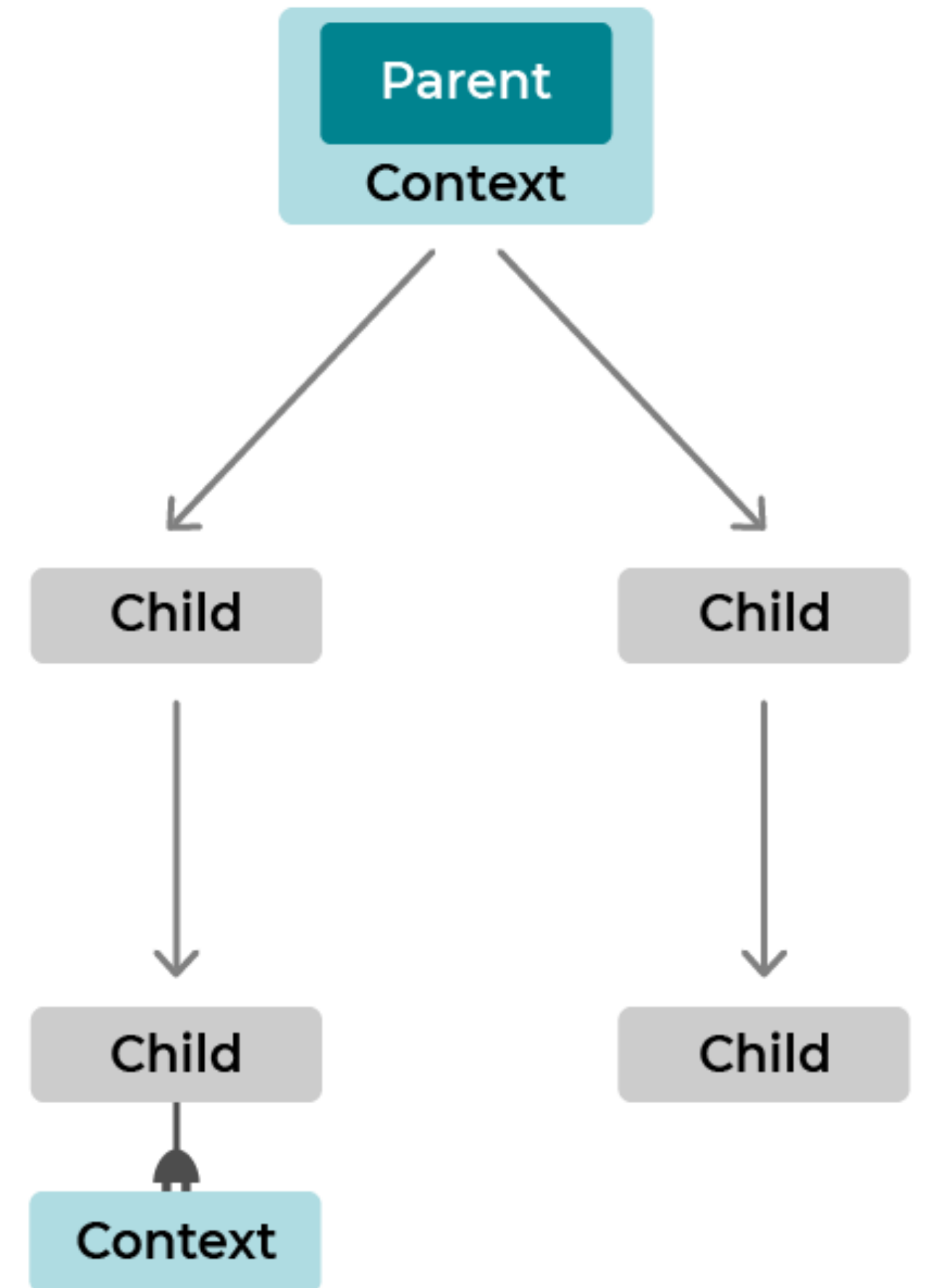
- Used for adding a state variable.
- `const [age, setAge] = useState(17);`
- The convention is to name state variables like `[something, setSomething]` using array destructuring.

useEffect

- Use to execute function on initial render or according to our use case.
- Common use case is for fetching data from a api.

useContext

- Allows you to share values between multiple levels of components of components without passing props through each level.



useReducer

- useReducer is a React Hook that lets you add a reducer to your component.
- Used for updating complex states.
- Use State Reducer Pattern.

useRef

- useRef is a React Hook that lets you reference a value that's not needed for rendering.
- useRef returns a ref object with a single current property initially set to the initial value you provided.

useMemo

- `useMemo` is a React Hook that lets you cache the result of a calculation between re-renders.

useTransition

- `useTransition` is a React Hook that lets you render a part of the UI in the background.
- `const [isPending, startTransition] = useTransition()`
- Generally used for Loading screens.

useLayoutEffect

[API REFERENCE](#) > [HOOKS](#) >

useLayoutEffect

Pitfall

`useLayoutEffect` can hurt performance. Prefer `useEffect` when possible.

`useLayoutEffect` is a version of `useEffect` that fires before the browser repaints the screen.

```
useLayoutEffect(setup, dependencies?)
```

useLayoutEffect

- `useLayoutEffect` is a version of `useEffect` that fires before the browser repaints the screen.
- Helps in avoiding flickering issues and some other specific use cases.

useInsertionEffect

useInsertionEffect

Pitfall

`useInsertionEffect` is for CSS-in-JS library authors. Unless you are working on a CSS-in-JS library and need a place to inject the styles, you probably want `useEffect` or `useLayoutEffect` instead.

`useInsertionEffect` allows inserting elements into the DOM before any layout Effects fire.

```
useInsertionEffect(setup, dependencies?)
```

useId

- Gives unique ID.
- Can be used when getting duplicated keys errors.
- Relatively New ~

Thank You !