React Hooks Visual Cheat Sheet

useState

```
Manage local component state.

const [value, setValue] = useState(initialValue)

Use case: Counter, form input, toggle, etc.

Visual:
[State Value]  useState(0)  [Function to Update It]
```

useEffect

```
Handle side effects like data fetching, timers, or subscriptions.

useEffect(() => {
  // side-effect code
  return () => {
    // optional cleanup
  }
}, [dependencies])

Use case: API calls, DOM events, intervals

Visual:
Component Mounts or Updates

useEffect runs

Runs again only if dependencies change
```

useContext

```
Access values from React Context.

const value = useContext(MyContext)

Use case: Theme, user authentication, language

Visual:
<Context.Provider value="dark">
  <Component />  useContext() reads "dark"
</Context.Provider>
```

useRef

```
Refer to DOM elements or hold a persistent value without triggering re-renders.

const ref = useRef(initialValue)
```

React Hooks Visual Cheat Sheet

```
Use case: Accessing DOM, storing timers, previous state

Visual:
+------------+
| useRef()   |
+------------+

  .current  persistent value
```

useMemo

```
Optimize performance by memoizing expensive computations.

const result = useMemo(() => computeExpensiveValue(), [deps])

Use case: Expensive calculations, derived data

Visual:
useMemo(fn, [deps])  caches result unless deps change
```

useCallback

```
Memoize a function to avoid re-creation on every render.

const memoFn = useCallback(() => doSomething(), [deps])

Use case: Prevent unnecessary child re-renders

Visual:
useCallback(fn, [deps])  same function unless deps change
```

Custom Hooks (useYourHook)

```
Encapsulate and reuse logic using other hooks.

function useCustomHook() {
  const [state, setState] = useState(0)
  useEffect(() => { ... }, [])
  return state
}

Use case: Window size, form validation, fetching data

Visual:
[Custom Hook]

Reusable logic inside functional components
```

React Hooks Visual Cheat Sheet

Rules of Hooks

1. Only call hooks at the top level (not inside loops or conditions).
2. Only call hooks in React functions (components or custom hooks).