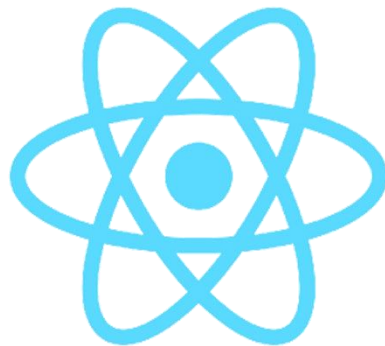


4장 React Hooks



React

목차

- ✓ 1. React Hooks 개요
- ✓ 2. 함수형 컴포넌트의 생명주기
- ✓ 3. 리액트 불변성

1. React Hooks 개요

- React Hooks는 React v16.8에서 함수형 컴포넌트의 기능 확장을 위해 사용되는 API 함수 ← Use ~.
- 최신 리액트에서는 클래스형 컴포넌트 대신 함수형 컴포넌트와 Hooks 사용을 권장

18

주요 Hook	설명
useState	컴포넌트의 상태값을 선언하고 관리하는 Hook
useRef	컴포넌트의 DOM 요소에 대한 참조값을 설정하고 참조하기 위한 Hook
useReducer	<ul style="list-style-type: none">• 컴포넌트의 다양한 상태 업데이트를 위한 Hook• 현재 상태와 액션을 인자로 받아 새로운 상태를 반환
useMemo	<ul style="list-style-type: none">• 연산 비용이 큰 함수의 결과를 메모이제이션하여 성능을 최적화하기 위한 Hook• 이전에 계산된 값으로 재사용하기 때문에 불필요한 처리 방지
useCallback	성능 최적화를 위해 렌더링 시에 함수가 매번 새로 생성되는 것을 방지하는 Hook
useEffect	컴포넌트의 생명주기에 따라 특정 작업을 처리하기 위한 Hook

19 버전 제미 예제.

2. 함수형 컴포넌트의 생명주기 useEffect()

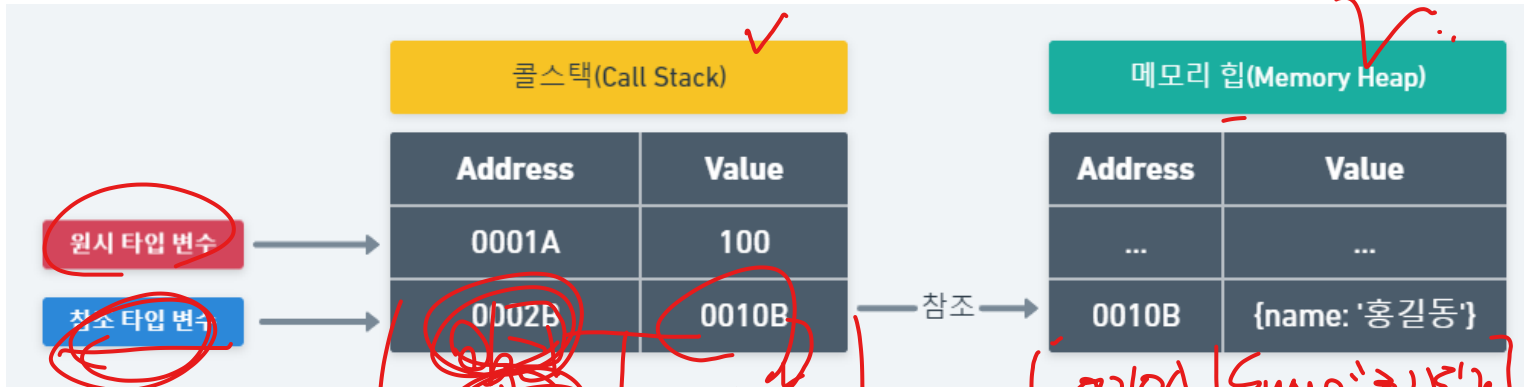
- 마운트 mount 는 컴포넌트가 브라우저에 출력되는 단계
- 업데이트 update 는 컴포넌트의 state가 변경되어 렌더링되는 단계
- 언마운트 unmount 는 컴포넌트가 브라우저에서 제거되는 단계



useEffect (렌더, [의존성 배열])

3. 리액트 불변성

- 불변성(immutability)은 기존의 상태 객체나 배열을 직접 수정하지 않고, 새로운 객체나 배열을 생성해 상태를 업데이트 하는 특성
- 리액트는 불변성을 지킴으로써 상태변화를 감지하고, 이 변경사항에 따라 필요한 컴포넌트만 업데이트(re-render)
- 리액트에서 데이터 불변성을 유지하기 위해 객체나 배열을 업데이트 할때 전개 연산자(Spread operator) 사용



const obj1 = { "name": "홍길동" }
const obj2 = { ...obj1 };

출처: <https://issue.tistory.com/143>

원시 타입 → 값 변경 유지
참조 타입 → 불변성 유지

원시 타입 → 불변성 유지 X