

1. 라이프사이클이란?

☰ 다중 선택

라이프사이클

Lifecycle

- 컴포넌트가 생성, 업데이트, 제거되는 과정에서 실행되는 일련의 단계와 이를 제어하기 위한 메서드 또는 Hook을 의미함

1. 라이프 사이클 단계

- React의 컴포넌트 라이프 사이클은 크게 3단계로 나뉩니다:

(1) Mount (탄생)

- 컴포넌트가 탄생하는 순간
- 화면에 처음 렌더링되는 순간
- A 컴포넌트가 Mount 되었다 == A 컴포넌트가 화면에 처음으로 렌더링되었다
- 주요 메서드/Hook:
 - `constructor` (클래스 컴포넌트): 초기 상태 및 프로퍼티 설정
 - `render`: JSX를 반환하여 DOM을 구성
 - `componentDidMount` (클래스 컴포넌트): DOM에 렌더링된 후 실행 (API 호출 등 초기화 작업)
 - `useEffect` (함수형 컴포넌트): 빈 의존성 배열을 사용해 초기화

```
useEffect(() => {  
  console.log("Component mounted");  
}, []);
```

(2) Update (변화)

- 컴포넌트가 다시 렌더링 되는 순간
- 리렌더링 될 때를 의미
- A 컴포넌트가 업데이트 되었다 == A 컴포넌트가 리렌더링 되었다
- 주요 메서드/Hook:
 - `render` : 업데이트된 상태를 기반으로 새로운 JSX 반환
 - `componentDidUpdate` (클래스 컴포넌트): 업데이트 후 작업 수행
 - `useEffect` (함수형 컴포넌트): 의존성 배열을 설정해 업데이트 작업 수행

```
useEffect(() => {
  console.log("Component updated");
}, [dependency]); // dependency가 변경될 때 실행
```
 - `shouldComponentUpdate` (클래스 컴포넌트): 렌더링 여부를 결정 (성능 최적화)
 - `useMemo` / `useCallback` : 계산된 값 또는 함수를 캐싱하여 불필요한 렌더링 방지

(3) UnMount (죽음)

- 컴포넌트가 화면에서 사라지는 순간
- 렌더링에서 제외되는 순간을 의미
- A 컴포넌트가 언마운트 되었다 == A 컴포넌트가 화면에서 사라졌다
- 주요 메서드/Hook:
 - `componentWillUnmount` (클래스 컴포넌트): 제거 전에 실행 (정리 작업 수행)
 - `useEffect` (함수형 컴포넌트): 클린업 함수를 반환

```
useEffect(() => {
  return () => {
    console.log("Component unmounted");
  };
});
```

```
};  
}, []);
```