

## 7. React\_LifeCycle\_Functional\_Component\_useEf (March 9th, 21)

```
useEffect(function(){
  console.log('%cfunc => useEffect'+(++funcId), funcStyle);
  document.title = number;
  return function(){
    console.log('%cfunc => useEffect componentWillUnmount', funcStyle);
  }
});
```

number가 바뀌지 않았을 때, 위의 useEffect는 실행이 될까?

⇒ 실행이 된다.

그렇다면, number가 바뀌지 않았을 때는 위의 코드가 실행되지 않도록 하는 방법을 알아보자.

```
useEffect(function(){
  console.log('%cfunc => useEffect'+(++funcId), funcStyle);
  document.title = number;
  return function(){
    console.log('%cfunc => useEffect componentWillUnmount', funcStyle);
  }
}, [number]); #useEffect 마지막에 원하는 배열을 추가해줌으로써,
              해당 배열 안의 원소가 변할 때만 useEffect가 실행되도록 만들 수 있다.
```

```
function FuncComp(props){
  useEffect(function(){
    console.log('%cfunc => useEffect number'+(++funcId), funcStyle);
    document.title = number;
    return function(){
      console.log('%cfunc => useEffect componentWillUnmount number', funcStyle);
    }
  }, [number]);

  useEffect(function(){
    console.log('%cfunc => useEffect _date'+(++funcId), funcStyle);
    document.title = _date;
    return function(){
      console.log('%cfunc => useEffect componentWillUnmount _date', funcStyle);
    }
  }, [_date]);
}
```

```
func => render5
func => useEffect componentWillUnmount number
func => useEffect number7
func => render8
func => useEffect componentWillUnmount _date
func => useEffect _date10
```

number가 바뀔 때는 render를 추적하는 useEffect만, \_date가 바뀔 때는 \_date를 추적하는 useEffect만 실행된다.

이렇게, **변한 값에 대한 처리**만 할 수 있도록 할 수 있다.

- **UseEffect**는 class형 컴포넌트의 **componentDidMount**와 **componentDidUpdate**에 해당하는 기능을 하고, 두 경우에 해당하는 단계 마다 모두 실행된다.
- 그런데, UseEffect를 **componentDidMount**나 **componentDidUpdate** 중 한 가지 경우에서만 실행하고 싶다면 어떻게 해야 할까?

```
useEffect(function(){
  console.log('%cfunc => useEffect (componentDidMount)' + (++funcId), funcStyle);
  document.title = number;
  return function(){
    console.log('%cfunc => useEffect componentWillUnmount (componentDidMount) number', funcStyle);
  }
}, []);
```

**useEffect**의 마지막에 빈 배열을 추가해주면, **useEffect**가 **componentDidMount**의 역할만 실행한다.

- 하위 컴포넌트를 부모 컴포넌트를 이용해서 없애버리는 **Unmount**시에는

```
useEffect(function(){
  console.log('%cfunc => useEffect (componentDidMount)' + (++funcId), funcStyle);
  document.title = number;
  return function(){
    console.log('%cfunc => useEffect componentWillUnmount number', funcStyle);
  }
}, []);
```

**useEffect**의 **return**함수가 실행된다.