

리액트 심화 (4) useEffect

1. 사이드 이펙트

- 리액트의 역할

UI 렌더링 ==> 사용자 INPUT에 반응 ==> 리렌더링

EX) 검색 창 자동완성 기능

입력 ==> onChange 이벤트 리스너 & 핸들러 ==> 리렌더링

onChange 이벤트 리스너 & 핸들러 == HTTP Request ==> 서버

서버 == HTTP Request ==> onChange 이벤트 리스너 & 핸들러

(1) 키워드 값이 바뀔 때 ==> 리렌더링

(2) 서버에서 키워드로 자동완성 된 데이터를 받아왔을때 ==> 리렌더링

(3) 사용방법

```
useEffect(() => {...}, [ dependencies ] );
```

- `() => {...}`: 매순간 dependencies 가 바뀔 때마다, 이 함수가 실행이 된다.
- `[dependencies]`: 첫 번째 인자 함수를 매번 실행을 시켜줘야 하는 의존하는 값을 배열로 넣는다.

(4) practice 4-4

```
import React, { useState } from 'react';

import Login from './components/Login/Login';
import Home from './components/Home/Home';
import MainHeader from './components/MainHeader/MainHeader';

function App() {
  const [isLoggedIn, setIsLoggedIn] = useState(false);

  const loginHandler = (email, password) => {
    setIsLoggedIn(true);
  };

  const logoutHandler = () => {
    setIsLoggedIn(false);
  };

  return (
    <React.Fragment>
      <MainHeader isAuthenticated={isLoggedIn} onLogout={logoutHandler} />
      <main>
        {!isLoggedIn && <Login onLogin={loginHandler} />}
        {isLoggedIn && <Home onLogin={logoutHandler} />}
      </main>
    </React.Fragment>
  );
}
```

```

    </React.Fragment>
  )
}

```

```

function App() {
  const [isLoggedIn, setIsLoggedIn] = useState(false);

  useEffect(() => {
    console.log('hello useEffect');
  });

  const loginHandler = (email, password) => {
    setIsLoggedIn(true);
  };

  const logoutHandler = () => {
    setIsLoggedIn(false);
  };

  return (
    <React.Fragment>
      <MainHeader isAuthenticated={isLoggedIn} onLogout={logoutHandler} />
      <main>
        {isLoggedIn && <Login onLogin={loginHandler} />}
        {isLoggedIn && <Home onLogin={logoutHandler} />}
      </main>
    </React.Fragment>
  )
}

```

렌더링이 될때마다 실행

```

useEffect(() => {
  console.log('hello useEffect');
});

```

처음에 렌더링이 될때만 실행

```

useEffect(() => {
  const storedUserLoggedInInfo = localStorage.getItem("isLoggedIn");
  //만약 로그인이 되어 있다면 isLoggedIn = true
  if(storedUserLoggedInInfo === '1') setIsLoggedIn(true);
  //만약 로그인이 되어 있지 않다면 isLoggedIn = false
  else setIsLoggedIn(false);
}, []);

```

localStorage 개념

```
const loginHandler = (email, password) => {  
  localStorage.setItem('isLoggedIn', '1');  
  setIsLoggedIn(true);  
};  
  
const logoutHandler = () => {  
  localStorage.removeItem("isLoggedIn");  
  setIsLoggedIn(false);  
};
```

-
- 사이드 이펙트는 일반적인 UI 렌더링 사이클과 다르게 동작한다.
 - `useEffect` 혹은 `useLayoutEffect`를 통해 사이드 이펙트를 컴포넌트 안에서 핸들링 할 수 있다.