**useRef의 Hook으로 컴포넌트 안에서 조회 및 수정할 수 있는 변수를 관리하기**

-useRef로 관리하는 변수는 값이 바뀐다고 컴포넌트가 리렌더링 되지 않음

-리액트 컴포넌트에서의 상태는 상태를 바꾸는 함수를 호출하고 나서 렌더링 이후 업데이트 된 상태를

조회할 수 있는 반면, useRef로 관리하고 있는 변수는 설정 후 바로 조회 할 수 있다.

-다음과 같은 값을 관리할 수 있다.

-setTimeout, setInterval을 통해서 만들어진 id

-외부 라이브러리를 사용하여 생성된 인스턴스

-scroll 위치

**배열에 새 항목을 추가할 때, 새 항목에서 사용할 고유 id를 관리하는 용도의 변수를 관리**

//App.js

function App() {

// App.js에서 관리하기 위해 여기에 배열을 선언함

const users = [

{

id: 1,

username: 'velopert',

email: 'public.velopert@gmail.com'

},

{

id: 2,

username: 'tester',

email: 'tester@example.com'

},

{

id: 3,

username: 'liz',

email: 'liz@example.com'

}

];

const nextId = useRef(4); // useRef()를 사용할 때 파라미터를 넣어주면,

//이 값이 .current값의 기본값이 됨

const onCreate = () => {

//나중에 구현 할 “배열에 항목을 추가하는 로직”

nextId.current += 1;

};

return <UserList users={users} />;

}

-useRef()를 사용할 때, 파라미터를 넣어주면, 이값이 .current 값의 기본값이 됨.

-그리고 이 값을 수정할 때에는 .current 값을 수정하면 되고 조회할 때에는 .current 를 조회하면 됨.