

AI 활용 빅데이터분석 풀스택웹서비스 SW 개발자 양성과정

React



부산대학교 소프트웨어교육센터
PUSAN NATIONAL UNIVERSITY SOFTWARE EDUCATION CENTER



Hook : useRef()

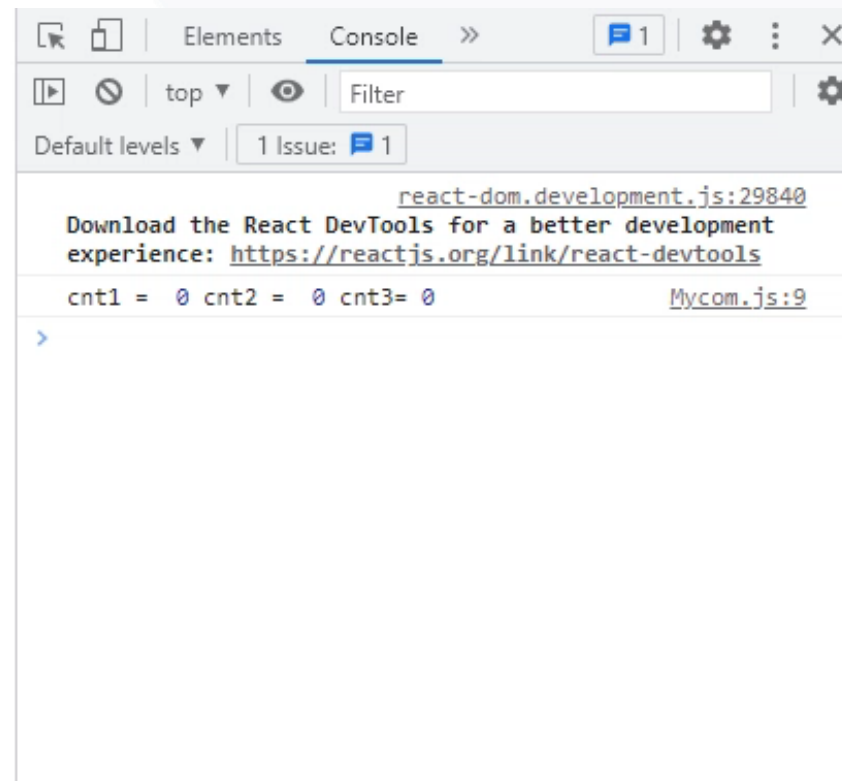
- **.current** 프로퍼티로 전달된 인자(initialValue)로 초기화된 변경 가능한 ref 객체를 반환하고 반환된 객체는 컴포넌트의 전 생애주기를 통해 유지
- **변수 관리**
 - useRef로 관리하는 값은 값이 변해도 화면이 렌더링되지 않음
- **DOM 요소 선택**
 - useRef() 를 사용하여 Ref객체를 만들고, 이 객체를 선택하고 싶은 DOM 에 ref값으로 설정
 - Ref객체의 .current값은 DOM 을 가리킴

Hook : useRef()

• 변수 관리

- 컴포넌트 변수 cnt1 : 0
- state 변수 cnt2 : 0
- ref 변수 cnt3 : 0

varAdd stateAdd refAdd



```
let cnt1 = 0 ;  
const [cnt2, setCnt2] = useState(0);  
const cnt3 = useRef(0);
```

```
const showCnt = () => {  
  console.log("cnt1 = ", cnt1, "cnt2 = ", cnt2, "cnt3=", cnt3.current);  
}
```

//변수 비교

```
const varAdd= (e) => {  
  e.preventDefault();  
  cnt1 = cnt1 + 1 ;  
  showCnt();  
}
```

컴포넌트에서 선언된 변수는
렌더링에 영향을 주지 않음으로 화면에 반영되지 않고
렌더링이 되는 경우 초기화

```
const stateAdd= (e) => {  
  e.preventDefault();  
  setCnt2(cnt2 + 1);  
}  
useEffect(()=>{showCnt();}, [cnt2] );
```

State변수값이 변경되면 화면이 재 렌더링이
되고 State변수값 유지

```
const refAdd= (e) => {  
  e.preventDefault();  
  cnt3.current = cnt3.current + 1 ;  
  showCnt();  
}
```

current속성에 값저장

ref변수 값은 변경이되어도 재렌더링이 되지 않고
재렌더링이 된 경우에도 이전 값을 그대로 유지

Hook : useRef DOM 핸들링

```
function MyCom() {  
  const txtRef = useRef();  
  const [txtState, setTxtState] = useState([]);  
  const [txtList, setTxtList] = useState([]);  
  
  useEffect(() => {  
    txtRef.current.focus();  
  }, []);  
  
  useEffect(() => {  
    setTxtList(txtState.map((item) => {return <li key={item}>{item}</li>}));  
  }, [txtState]);  
}
```

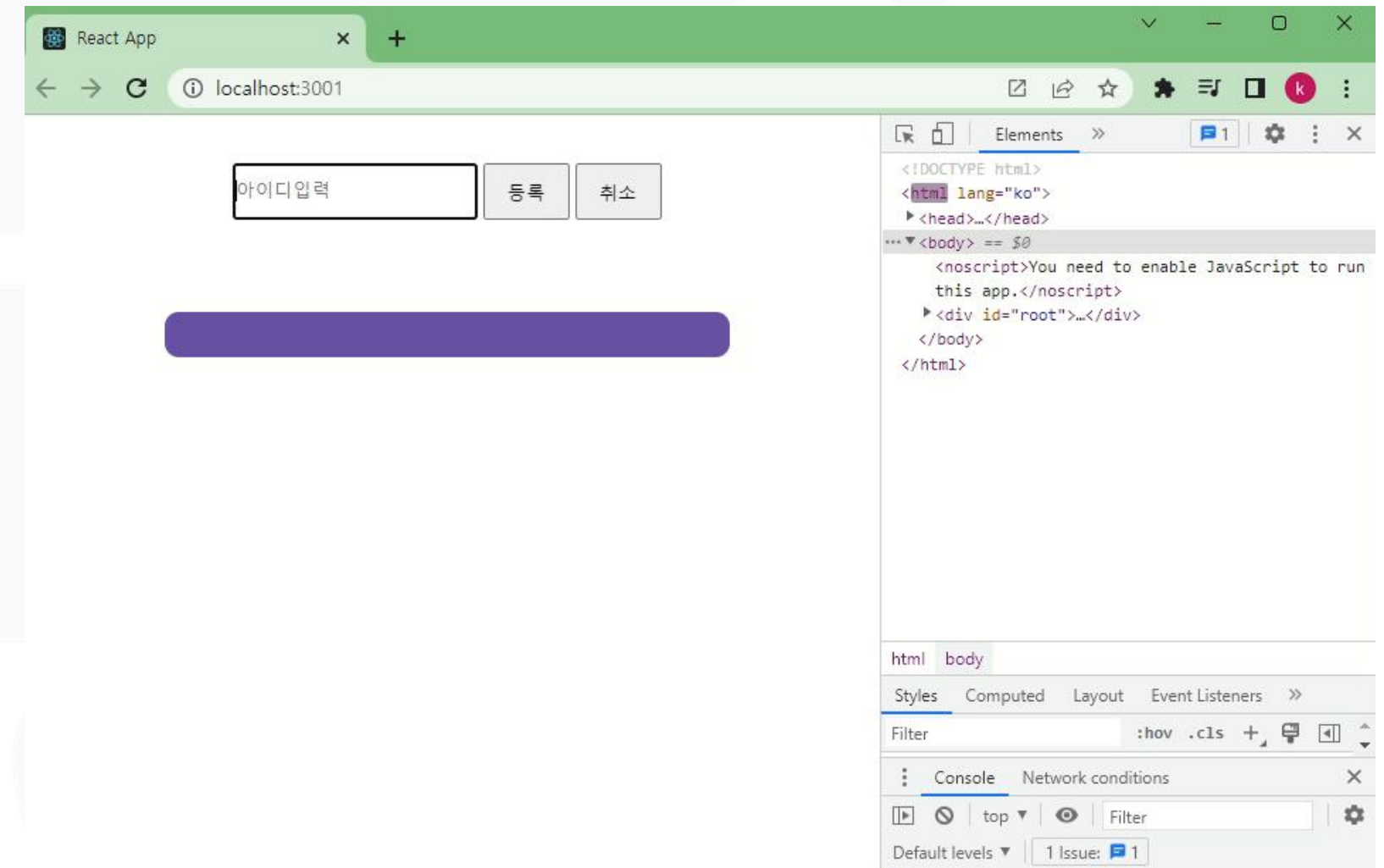
컴포넌트가 맨처음 렌더링되면
input 요소에 포커스

```
const handleClick = (e) => {  
  e.preventDefault();  
  setTxtState([txtRef.current.value, ...txtState]);  
  txtRef.current.value = '';  
  txtRef.current.focus();  
}
```

기존 배열에 추가

```
const handleReset = () => {  
  txtRef.current.focus();  
  setTxtState([]);  
  setTxtList([]);  
}
```

```
return (  
  <>input 요소에 ref속성을 지정하여 useRef 훅으로 지정한 변수 지정</>  
  <form>  
    <input ref={txtRef} type="text" placeholder='아이디입력' />  
    <button onClick={handleClick}>등록</button>  
    <button type="reset" onClick={handleReset}>취소</button>  
  </form>  
  <section className='msg'>  
    { txtList }  
  </section>  
</>  
)  
);
```



해결문제

• 간단한 할 일 목록 만들기

