



# 3\_Props

## React Props

- React에서 컴포넌트에 데이터를 전달하는 방법 중 하나
- 부모 컴포넌트 → 자식 컴포넌트로 전달
- props로 전달 받은 값은 컴포넌트에서 변경할 수 없음(읽기 전용)
- 컴포넌트간의 데이터 전달이 간단해지고 유지보수성, 재사용성이 증가됨.

## Props 예제 1

### 1. src/components/R03\_Props1.js 생성 후 작성

```
import React from 'react';

// props : 부모 컴포넌트가 자식 컴포넌트에 데이터 전달 시 사용하는 객체
// * props는 자식 컴포넌트에서 수정 불가능 *
const PropsEx1 = (props) => {

  // 부모 컴포넌트(App.jsdml Function App())로 부터 전달 받은 값(props) 확인
  console.log('PropsEx1');
  console.log(props);

  return(
    /* 부모 컴포넌트로 부터 전달 받은 값(props)를 이용해서 자식 컴포넌트 만들기*/
    <h3>번호 : {props.num} / 이름 : {props.name}</h3>
  )
}

export default PropsEx1;
```

## 2. src/App.js 작성

```
import './App.css';

import Props1 from './components/R03_Props1';

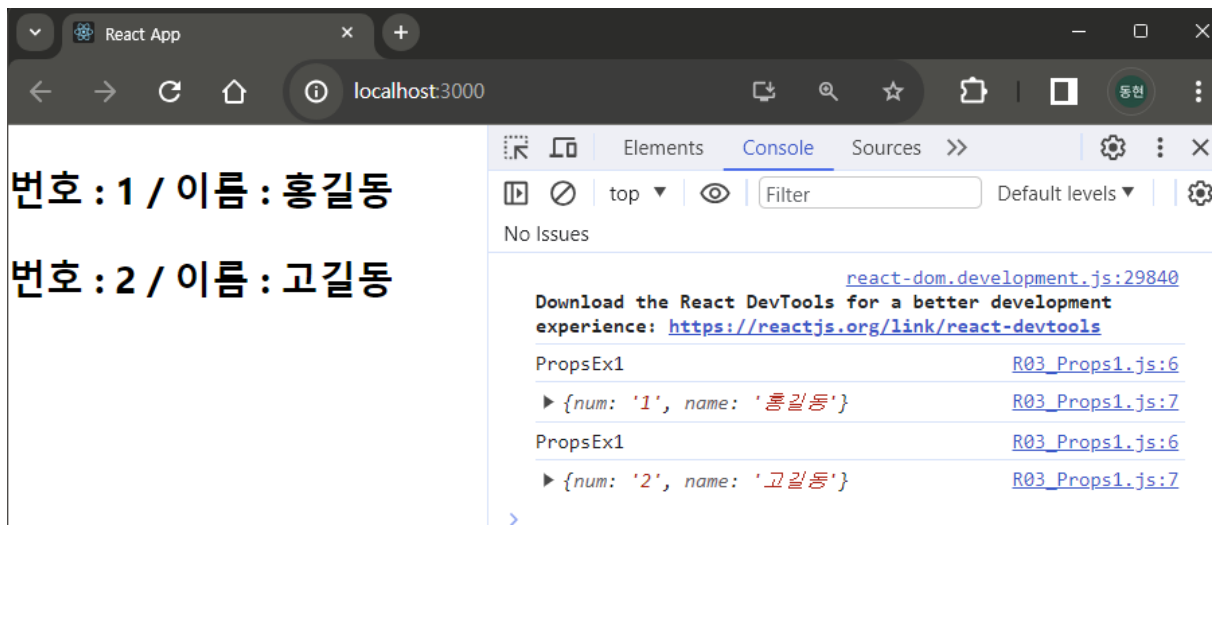
function App() {
  return (

    <>
      {/* 부모 컴포넌트인 App() 에서 자식 컴포넌트인 Props에게 num, name을 전달 */}
      <Props1 num='1' name='홍길동' />
      <Props1 num='2' name='고길동' />
    </>

  );
}

export default App;
```

## 3. 결과 화면



## Props 예제 1

## 1. src/components/R04\_Props2.js 생성 후 작성

```
import React from 'react';

const PropsEx2 = (props) => {

  // 부모 컴포넌트(App.js)로 부터 전달 받은 값(props) 확인
  console.log('PropsEx2');
  console.log(props); // 객체 형태 {K:V, K:V}

  // props에서 key와 변수명이 일치하는 경우 value를 해당 변수에 대입
  let {name, age, address, gender} = props;

  return(
    <ul className='info'>
      <li>이름 : {name}</li>
      <li>나이 : {age}</li>
      <li>주소 : {address}</li>
      <li>성별 : {gender}</li>
    </ul>
  )
}

export default PropsEx2;
```

## 2. src/App.js 작성

```
import './App.css';

import Props2 from './components/R04_Props2';

function App() {
  return (

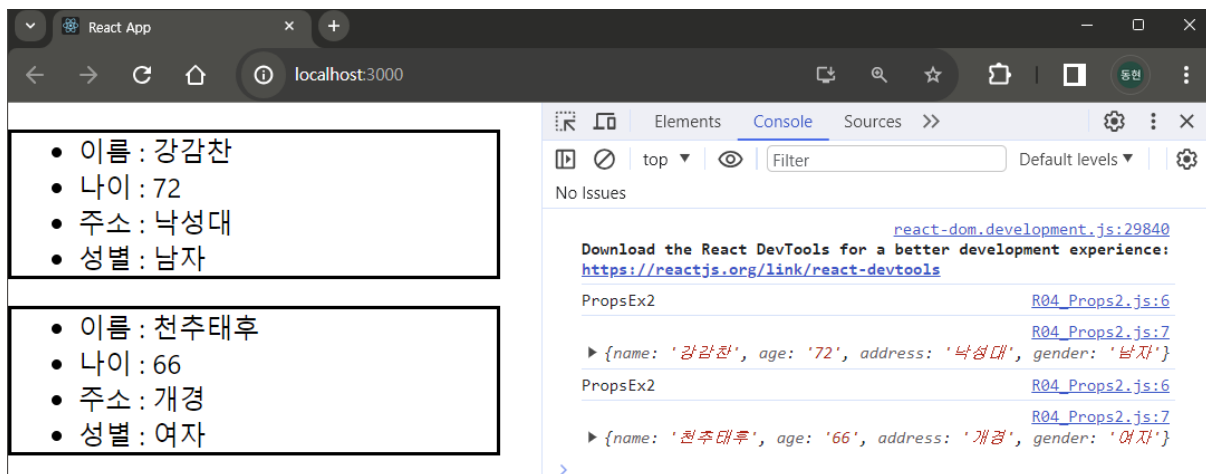
    <>
      <Props2 name="강감찬" age="72" address="낙성대" gender="남자"/>
      <Props2 name="천추태후" age="66" address="개경" gender="여자"/>
    </>
  );
}

export default App;
```

## 3. App.css 작성

```
.info{
  width: 250px;
  border: 2px solid black;
}
```

## 4. 실행 화면



## Props Drilling (상태 내리꽂기)

- props를 통해 하위 컴포넌트에 계속 props를 계속해서 전달하는 것
- 장점 : 명시적이며, 값의 추적에 요이함
- 단점 : 중간 컴포넌트가 필요없는 props를 계속 전달 받아야함

### 1. src/components/R05\_Props3.js 생성 후 작성

```
import React from 'react';

const ChildComponent = (props) => {
  /* 여기서 props : PropsEx3로 부터 전달 받은 데이터 */
  return(
    <>
```

```

        <h3>App에서 부터 전달된 name : {props.name}</h3>
        <h3>PropsEx3에서 전달된 age : {props.age}</h3>
      </>
    );
  }

  const PropsEx3 = (props) => {
    /* 여기서 props : App으로 부터 전달 받은 데이터 */
    return(
      <>
        { /* Props Drilling : props를 하위 컴포넌트로 계속 전달 */ }
        <ChildComponent name={props.name} age={props.name === '김영희' ? 20 : 25}/>
      </>
    );
  }

  export default PropsEx3;

```

## 2. src/App.js 작성

```

import './App.css';

import Props3 from './components/R05_Props3';

function App() {

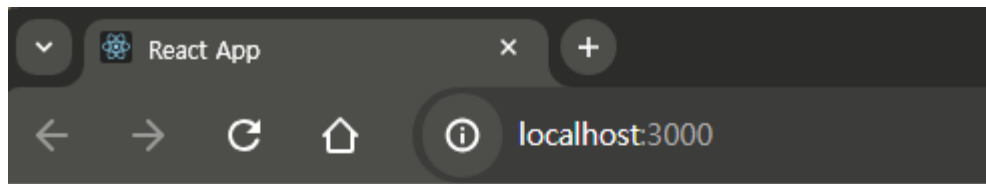
  return (

    <>
      <Props3 name="김영희"/>
      <Props3 name="김철수"/>
    </>
  );
}

export default App;

```

## 3. 실행 화면



**App에서 부터 전달된 name : 김영희**

**PropsEx3에서 전달된 age : 20**

**App에서 부터 전달된 name : 김철수**

**PropsEx3에서 전달된 age : 25**