

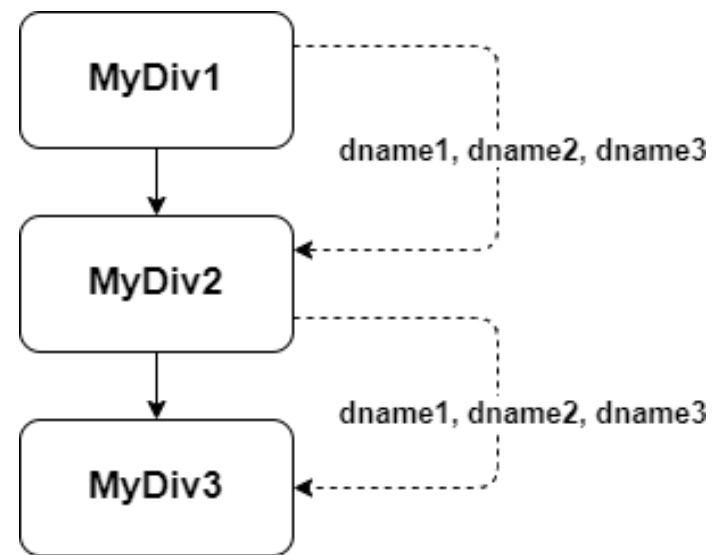
소프트웨어융합기초1

React Probs와 State

김경민



prob 예제



컴포넌트 probs

• 컴포넌트 간의 값을 전달하는 수단

```
import MyDiv2 from "./MyDiv2";

export default function MyDiv1() {
  const div1Name = "div1" ;
  const div2Name = "div2" ;
  const div3Name = "div3" ;

  return (
    <div className="flex flex-col justify-center items-center"
      w-2/3 h-2/3
      □text-white ■bg-lime-900
      rounded-sm m-10 p-10">
      <p className="flex w-full font-bold justify-start">{div1Name}</p>
      {MyDiv2 dname1 = {div1Name} dname2 = {div2Name} dname3 = {div3Name} />}
    </div>
  )
}
```

MyDiv1

```
import MyDiv3 from "./MyDiv3" ;
export default function MyDiv2(probs) {
  return (
    <div className="flex flex-col justify-center items-center"
      w-4/5 h-4/5
      text-white bg-lime-700
      rounded-sm m-10 p-10">
      <p className="flex w-full justify-start">{probs.dname1} > {probs.dname2}</p>
      {MyDiv3 dname1 = {probs.dname1} dname2 = {probs.dname2} dname3 = {probs.dname3} />}
    </div>
  )
}
```

MyDiv2

```
export default function MyDiv3(probs) {
  return (
    <div className="flex flex-col justify-center items-center"
      w-4/5 h-4/5
      □text-white ■bg-lime-500
      rounded-sm m-10 p-10">
      <p className="flex w-full justify-start">
        {probs.dname1} > {probs.dname2} > {probs.dname3}
      </p>
    </div>
  )
}
```

MyDiv3

컴포넌트 probs

• 구조 분해 할당

```
import MyDiv2 from "./MyDiv2";

export default function MyDiv1() {
  const div1Name = "div1";
  const div2Name = "div2";
  const div3Name = "div3";

  return (
    <div className="flex flex-col justify-center items-center"
      w-2/3 h-2/3
      □text-white ■bg-lime-900
      rounded-sm m-10 p-10">
      <p className="flex w-full font-bold justify-start">{div1Name}</p>
      {MyDiv2 dname1 = {div1Name} dname2 = {div2Name} dname3 = {div3Name} />}
    </div>
  )
}
```

MyDiv1

```
import MyDiv3 from "./MyDiv3";
export default function MyDiv2({dname1, dname2, dname3}) {
  return (
    <div className="flex flex-col justify-center items-center"
      w-4/5 h-4/5
      text-white bg-lime-700
      rounded-sm m-10 p-10">
      <p className="flex w-full justify-start">` ${dname1} > ${dname2}`</p>
      {MyDiv3 dname1 = {dname1} dname2 = {dname2} dname3 = {dname3} />}
    </div>
  )
}
```

MyDiv2

```
export default function MyDiv3({dname1, dname2, dname3}) {
  return (
    <div className="flex flex-col justify-center items-center"
      w-4/5 h-4/5
      □text-white ■bg-lime-500
      rounded-sm m-10 p-10">
      <p className="flex w-full justify-start">
        { ${dname1} > ${dname2} > ${dname3} }
      </p>
    </div>
  )
}
```

MyDiv3

React Developer Tools

The image displays two overlapping browser windows. The background window shows the Chrome Web Store page for React Developer Tools, featuring the React logo, a 4.0 star rating, and a user count of 4,000,000. The foreground window shows the React App interface running on localhost:3000. The app displays a visual tree of nested divs: 'div1' contains 'div1 > div2', which contains 'div1 > div2 > div3'. The React DevTools component inspector is open on the right, showing the component hierarchy and props for the selected 'MyDiv2' component. The props include 'dname1', 'dname2', 'dname3', and 'new entry'. The rendered by section shows the component tree path: 'MyDiv11' -> 'App' -> 'createRoot()' -> 'react-dom@18.2.0'. The source is 'bundle.js:108'.

React Developer Tools

chrome 웹 스토어

확장 프로그램 및 테마 검색

React Developer Tools

추천 4.0 ★ (평점 1.5천개)

확장 프로그램 개발자 도구 4,000,000 사용자

React App

localhost:3000

리액트 실습

div1

div1 > div2

div1 > div2 > div3

MyDiv2

props

- dname1: "div1"
- dname2: "div2"
- dname3: "div3"
- new entry: ""

rendered by

- MyDiv11
- App
- createRoot()
- react-dom@18.2.0

source

- bundle.js:108

© 2024 kimkyungmin. All rights reserved.

컴포넌트 이벤트추가

MyDiv1

■ MyDiv1

■ MyDiv11



onClick이벤트 추가하고 함수직접 작성

```
const MyDivAUser1 = ({ user }) => {
  let cnt = 0;

  return (
    <>
      <ul className={style.div1ul}>
        <li>{user}</li>
        <li>{user + '1'}</li>
      </ul>
      <footer>
        <h2><span onClick={() => {
          cnt++;
          console.log("cnt =", cnt);
        }}>♥</span> {cnt}</h2>
      </footer>
    </>
  );
};
```

onClick이벤트 추가하고 함수호출

```
const MyDivAUser1 = ({user}) => {
  let cnt = 0 ;

  const likeBt = () => {
    cnt++ ;
    console.log("cnt =", cnt);
  }

  return (
    <>
      <ul className={style.div1ul}>
        <li>{user}</li>
        <li>{user+'1'}</li>
      </ul>
      <footer>
        <h2><span onClick={likeBt}>♥</span> {cnt}</h2>
      </footer>
    </>
  );
};
```

onClick이벤트 추가하고 인수를 전달할 경우는 반드시 함수형식으로 호출

```
const MyDivAUser1 = ({user}) => {
  let cnt = 0 ;

  const likeBt = (n) => {
    cnt = cnt + n ;
    console.log("cnt =", cnt);
  }

  return (
    <>
      <ul className={style.div1ul}>
        <li>{user}</li>
        <li>{user+'1'}</li>
      </ul>
      <footer>
        <h2><span onClick={()=>likeBt(1)}>♥</span> {cnt}</h2>
      </footer>
    </>
  );
};
```

리액트 Hook

- 함수 컴포넌트에서 React state와 생명주기 기능(lifecycle features)을 “연동(hook into)”할 수 있게 해주는 함수
 - React 버전 16.8부터 React 요소로 새로 추가
 - 상태 관련 로직을 추상화해 독립적인 테스트와 재사용이 가능해 레이어 변화 없이 재사용
 - 기존의 라이프사이클 메서드 기반이 아닌 로직 기반으로 나눌 수 있어서 컴포넌트를 함수 단위로 잘게 쪼갤 수 있다는 이점
- Hook 사용 규칙
 - 최상위에서만 Hook을 호출
 - 반복문, 조건문, 중첩된 함수 내에서 Hook을 실행하면 안됨
 - 이 규칙을 따르면 컴포넌트가 렌더링될 때마다 항상 동일한 순서로 Hook이 호출되는 것이 보장
 - 리액트 함수 컴포넌트에서만 Hook을 호출
 - JS함수에서는 Hook을 호출해서는 안됨



Hook : useState

- 컴포넌트 state

- 컴포넌트가 가지는 상태값
- 상태값이 변경되면 **화면이 재 렌더링**

- useState Hook

- 동적으로 변경되는 값을 관리할 때 사용하며 상태 유지값과 그 값을 갱신하는 함수를 반환
- const [state, setState] = useState(initialState);

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

const MyDivAUser1 = ({user}) => {
  const [cnt, setCnt] = useState(0);

  const likeBt = () => {
    setCnt(cnt + 1);
    cnt = cnt, cnt);

    ame={style.div1ul}>
    ser}</li>
    <li>{user+'1'}</li>
  </ul>
  <footer>
    <h2><span onClick={(()=>likeBt())}>❤️</span> {cnt}</h2>
  </footer>
</>
};
```

1. useState import

2. useState()는 배열 반환
=>구조분해 할당

구조분해(Destructuring) 할당
배열이나 객체의 속성을 분해해서
그 값을 변수에 담을 수 있게 하는
표현식

4. setcount()에
의해 state count
변수 값이변경

3. onclick이벤트 처리

5. State변수 값이 변경되면
변경되면 DOM에 바로 반영

해결문제

- 카드는 컴포넌트로 생성
- ‘좋아요’ 클릭 시 숫자 증가

리액트 실습



HTML

HTML(HyperText Markup Language)은 웹을 이루는 가장 기초적인 구성 요소로, 웹 콘텐츠의 의미와 구조를 정의할 때 사용

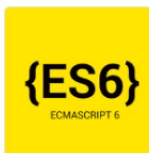
❤️ 좋아요 0



CSS

Cascading Style Sheets(CSS)는 HTML이나 XML(XML의 방언인 SVG, XHTML 포함)로 작성된 문서의 표시 방법을 기술하기 위한 스타일 시트 언어

❤️ 좋아요 0



JavaScript

웹 페이지를 위한 스크립트 언어로 가벼운, 인터프리터 혹은 just-in-time 컴파일 프로그래밍 언어로, 일급 함수를 지원

❤️ 좋아요 0



React

사용자 인터페이스를 만들기 위한 JavaScript 라이브러리

❤️ 좋아요 0

해결문제

• 영화진흥위원회 박스오피스 json활용

<https://www.kobis.or.kr/kobisopenapi/homepg/apiservice/searchServiceInfo.do?serviceId=searchDailyBoxOffice>

리액트 실습



순위	영화명	매출액	관객수	증감율
1	파묘	3,425,663,683 원	344,954명	-
2	들텐: 파트2	999,638,885 원	81,802명	-
3	원카	205,239,775 원	21,158명	↑4
4	가여운 것들	96,761,307 원	9,410명	↓1
5	브레이브하트: 셸럽 인 베이커리타운	78,874,350 원	9,343명	↑14
6	패스트 라이브즈	63,657,712 원	6,363명	-
7	랜드 오브 배드	54,673,073 원	6,109명	↓2
8	용감한 돌고래 벨루와 바닷속 친구들	32,544,720 원	4,144명	↑16
9	밥 말리: 원 러브	39,719,054 원	4,081명	↓5
10	건국전쟁	36,644,851 원	3,812명	-
[20234789] 패스트 라이브즈 개봉일 : 2024-03-06 누적관객수 : 84,256명				