

기말 시험 준비 (React)

1. 개념

- React 의 컴포넌트 기반이란 무엇인가? 사용되는 변수 및 상태의 관점에서 이해.
- React 가 선언적이란 무슨 의미를 갖는가? 어떤 변수의 값이 변경될 때 시스템이 re-rendering 을 실행하는가?

애플리케이션의 각 상태에 대한 간단한 뷰만 설계하세요. 그럼 React 는 데이터가 변경됨에 따라 적절한 컴포넌트만 효율적으로 갱신하고 렌더링합니다.

- useEffect 의 3 가지 동작 방법 : ([useEffect.pdf](#))
componentDidMount, componentDidUpdate, componentWillUnmount

2. (5 주) [hw-web-reactJSX.pdf](#)

- html 을 JSX 로 표현하기 : Ex210.jsx (style 정의하는 부분)

3. (6 주) [Ex82useState.jsx](#)

- style 을 useState 로 처리하는 법
- incDec 관련 파일들 : [IncDec.jsx](#) [incDec.html](#) [incDec3.jsx](#)
- [incDec-lam.jsx](#) [incDec2.jsx](#) [incDecSepComp.jsx](#)

4. (7 주) [ArrayKey.jsx](#),

- 키를 갖는 배열로 list 만들기
[facUseState.jsx](#)
- NumberList 컴포넌트:
- `a.map((k, i) => [k, b[i]]);`
- `const indexArr = [...arr.keys()]`

5. (9 주) [Conditional.jsx](#)

```
6.      {toggle && <p>hello</p>} {/* if - then 형태 */}
7.      {toggle || <p>안녕</p>}   {/* if - then 형태 */}
8.      {toggle ? <p>hello</p> : <p>안녕</p>}  {/* if - then - else 형태 */}
```

[hw-9-unmountRemove.pdf](#) 의 과제 이해하기

Mount.jsx

```
const Mount = () => {
  const [toggle, setToggle] = useState(true)
  return (
    <div>
      {toggle && <Comp1 />}
      <Comp2 />
      <button onClick={() => setToggle(!toggle)}>
        {toggle ? 'Unmount Comp1' : 'Mount Comp1'}
      </button>
    </div>
  );
};

const Comp1 = () => <h3>Comp1 is mounted</h3>
const Comp2 = () => <h3>Comp2 is mounted</h3>
```

mount.html

```
button.onclick = () => {
  if (button.innerText === "Unmount Comp1") {
    body.removeChild(h31);
    button.innerText = 'Mount Comp1';
  } else {
    body.prepend(h31);
    button.innerText = 'Unmount Comp1';
  }
}
```

6. (11 주) [Ex82useReducer.jsx](#)

- Ex82useState.jsx 를 useReducer 로 바꿔 표현하기

7. (12 주) [UseContextEx1.jsx](#)

- useContext 사용 방법

createContext

Provider 로 감싸기

useContext

8. (13 주) Router : 파일 [HomePage.jsx](#)

Router 구조 이해하기

useRef : 파일 [UseRef1.jsx](#)

9. (14 주) restAPI :

[TodoRestApi.jsx](#),
[restApiDemo.html](#)
[hw-todoRestApi.pdf](#) 과제 내용 이해하기