

useState 복습 문제





간단한 블로그 글 추가/삭제가 되도록 React를 작성 하시오.

조건

- ① 블로그의 글 추가 / 글 삭제 / 좋아요 클릭하면 따봉 숫자 1씩 증가되도록 작성하시오.
- ② 함수이름은 `Blog()`로 작성하시오.
- ③ 아래 주어진 `useState`를 이용하여 작성하시오.

```
const [posts, setPosts] = useState(['남자코트 추천', '강남 우동맛집', '파이썬독학']);  
const [likes, setLikes] = useState([0, 0, 0]);  
const [modalOpen, setModalOpen] = useState(false);  
const [selected, setSelected] = useState(0);  
const [inputValue, setInputValue] = useState('');
```

- ① `props`를 이용해 작성하시오.
- ② `useState()` 를 이용해 작성하시오.
- ③ 배열 같은 복사 이용해 작성하시오.
- ④ 배열이름.`unshift()` 이용하면 값이 위로 추가된다.
- ⑤ `src` 폴더안에 -> `stateComponent` 폴더 생성 -> `Blog.jsx` 파일을 생성하여 작성하시오.
- ⑥ `src` 폴더안에 -> `App.jsx`에 작성한 `Blog.jsx`를 import하여 작성하시오

제목을 추가하려면 클릭하십시오.



<초기 출력 화면>

남자코트 추천 0

11월 1일 발생

삭제

강남 우동맛집 0

11월 1일 발생

삭제

파이썬독학 0

11월 1일 발생

삭제

글발행

<글 추가 화면>

코딩 맛집 0

11월 1일 발생

삭제

남자코트 추천 0

11월 1일 발생

삭제

강남 우동맛집 0

11월 1일 발생

삭제

파이썬독학 0

11월 1일 발생

삭제

코딩 맛집

글발행

<글 삭제 화면>

코딩 맛집 0

11월 1일 발생

삭제

남자코트 추천 0

11월 1일 발생

삭제

강남 우동맛집 0

11월 1일 발생

삭제

코딩 맛집

글발행



간단한 영화 리뷰 관리 앱 만들기

조건

- ① 나만의 영화 리뷰 관리 앱을 만드세요.
- ② 영화 제목과 한줄평을 입력하고 등록 버튼을 누르면 목록에 추가된다.
- ③ 각 리뷰 옆에는 “” 버튼이 있고, 누를 때마다 좋아요 수가 1씩 증가한다.
- ④ 각 리뷰 옆에는 삭제 버튼도 있어서 삭제 가능하다.
- ⑤ 최신 리뷰가 맨 위로 올라오게 한다.
- ⑥ 아래와 같이 오브젝트 배열을 이용해 작성한다.

<아래>

```
const [reviews, setReviews] = useState([
  { id:0, title: '인셉션', comment: '꿈속의 꿈! 최고였어요.', likes: 0 },
  { id:1, title: '인터스텔라', comment: '음악이 너무 웅장해요.', likes: 0 },
]);
```

- ① src 폴더 안에 -> MovieReview.jsx 로 작성한다.
- ② Src 폴더 안에 -> MovieReview.css 작성한다.
- ③ MovieReview.jsx파일에 MovieReview.css를 import하여 사용한다.

```
import './MovieReview.css';
```
- ④ src 폴더안에 -> App.jsx에 작성한 MovieReview.jsx 를 import하여 실행 하시오.



<초기 화면>

나의 영화 리뷰

영화 제목 한줄평

등록

인셉션
꿈속의 꿈! 최고였어요.

삭제

👍 0

인터스텔라
음악이 너무 웅장해요.

삭제

👍 0

<영화 리뷰 추가 화면>

나의 영화 리뷰

영화 제목 한줄평

등록

우리들의 블루스
인생영화

👍 0

삭제

인셉션
꿈속의 꿈! 최고였어요.

삭제

👍 0

인터스텔라
음악이 너무 웅장해요.

삭제

👍 0

<좋아요 증가>

나의 영화 리뷰

영화 제목 한줄평

등록

인셉션
꿈속의 꿈! 최고였어요.

삭제

👍 3

인터스텔라
음악이 너무 웅장해요.

삭제

👍 0



간단한 쇼핑리스트 관리 앱 만들기

조건

- ① 나만의 쇼핑 목록을 관리하는 앱을 만드세요.
- ② 물건 이름을 입력하고 추가 버튼을 누르면 목록에 추가된다.
- ③ 각 항목에는 구매 완료 버튼이 있고, 클릭 시 "구매 완료"로 표시된다.
- ④ 각 항목에는 삭제 버튼이 있어 삭제도 가능하다.
- ⑤ 아래와 같이 오브젝트 배열을 이용해 작성한다.

<아래>

```
const [items, setItems] = useState([  
  { id:0, name: '우유', done: false },  
  { id:1, name: '계란', done: false },  
]);
```

- ① src 폴더 안에 -> ShoppingList.jsx 로 작성한다.
- ② Src 폴더 안에 -> ShoppingList.css 작성한다.
- ③ ShoppingList.jsx파일에 ShoppingList.css를 import하여 사용한다.
`import './ShoppingList.css';`
- ④ src 폴더안에 -> App.jsx에 작성한 ShoppingList.jsx 를 import하여 실행 하시오.

출력 화면 결과

<초기 화면>

🛒 쇼핑 리스트

물건 이름 입력 추가

우유	구매 완료	삭제
계란	구매 완료	삭제

<쇼핑 목록 추가>

🛒 쇼핑 리스트

물건 이름 입력 추가

과자	구매 완료	삭제
우유	구매 완료	삭제
계란	구매 완료	삭제

<쇼핑 목록 구매완료 버튼 클릭 시 [취소]버튼 추가 >

🛒 쇼핑 리스트

물건 이름 입력 추가

과자	취소	삭제
우유	구매 완료	삭제
계란	구매 완료	삭제

<쇼핑 목록 삭제>

🛒 쇼핑 리스트

물건 이름 입력 추가

과자	취소	삭제
우유	구매 완료	삭제



간단한 댓글 게시판 (부모 → 자식 props 활용) 만들기

조건

- ① 댓글 컴포넌트를 props로 분리해봅시다.
- ② 부모 컴포넌트에서 댓글 목록과 입력값을 관리한다.
- ③ 댓글 하나를 CommentItem이라는 자식 컴포넌트로 분리한다.
- ④ 자식은 props로 전달받은 댓글 내용과 좋아요 수를 표시한다.
- ⑤ 좋아요 버튼을 누르면 해당 댓글의 좋아요 수가 1 증가한다.

<아래>

```
const [comments, setComments] = useState([
  { id:0, text: '리액트 재밌어요!', like: 0 },
  { id:1, text: 'useState 너무 신기해요', like: 0 },
]);
```

- ① src 폴더 안에 -> CommentBoard.jsx로 작성한다.
- ② Src 폴더 안에 -> CommentBoard.css 작성한다.
- ③ CommentBoard.jsx파일에 CommentBoard.css를 import하여 사용한다.

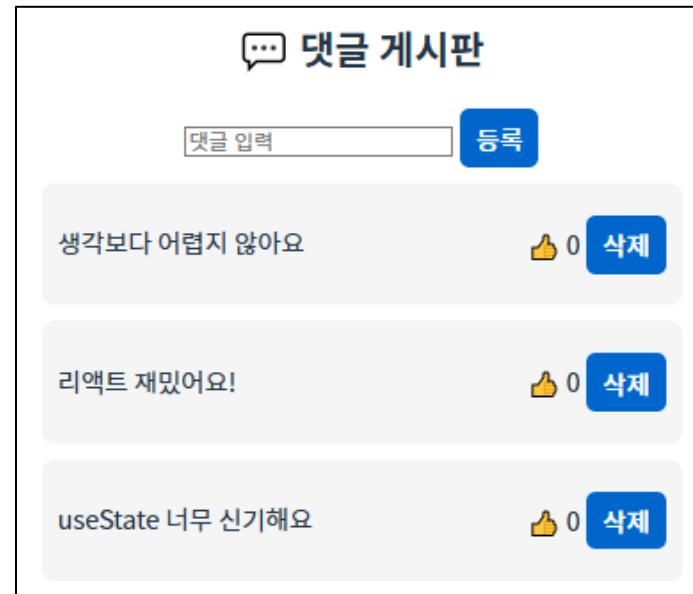
```
import './CommentBoard.css';
```
- ④ src 폴더안에 -> App.jsx에 작성한 CommentBoard.jsx 를 import하여 실행 하시오.

출력 화면 결과

<초기 화면>



<댓글 추가>



<댓글 삭제>



<댓글 좋아요 증가>

