**#네비게이션관련 커스텀 훅**

**커스텀 훅**

리액트의 컴포넌트는 화면을 재구성하는 것이다. 즉 화면을 재사용할 수 있는 단위가 컴포넌트이다.

이 컴포넌트들을 만들 때 똑같은 로직이 들어가는 경우가 있다. 그것도 훅을 쓰는 로직이.

훅을 안쓰는 로직은 import를 통해 사용하면 된다. useState/useNavigate와 같은 훅을 사용하는 로직을 재사용하는 단위가 커스텀 훅이다.

JSP의 include와 비슷하다.

**커스텀 훅을 위해 디렉토리 hooks를 생성하고 지금부터 useNavigate를 사용하는 커스텀 훅을 만들어보자**

텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명

페이지 이동 시에 쿼리스트링이 따라가야하는 경우가 많다. 예를 들어,

조회 페이지에서 목록페이지로 돌아갈 때 쿼리스트링 따라가야하는데 그런 코드들을 매 컴포넌트 만들 때마다 작성해준다는 것은 굉장히 번거롭다.

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

getNum 은 param 값 없으면 그냥 default값을 반환하도록 하는 함수.

텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명

useSearchParams를 사용하여 쿼리스트링 파라미터를 가져와 만든 getNum함수로 파라미터값 검증 및 초기화를 하고

createSearchParams라는 함수를 사용해서 쿼리스트링을 만들어준다음.

쿼리를 가지고 list로 이동하는 navigate함수를 실행하는 moveToList라는 함수를 만들어 리턴해준다.

이렇게 moveToList라는 쿼리스트링을 구성한 채 페이지이동을 하는 함수를 반환하는 useCustom훅을 사용해보자.

텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명

이렇게 쿼리스트링에 대한 페이징 처리를 커스텀 훅을 통해 처리했다.

List버튼을 추가하였고 해당 버튼을 누르게 되면 지정한 쿼리스트링 갖고 List페이지로 이동하게된다.

텍스트, 스크린샷, 라인, 폰트이(가) 표시된 사진

자동 생성된 설명

현재는 리스트페이지 구성을 아직 안해놓았으니까

임의로 url에 쿼리 스트링을 작성하여 readPage에 접근한 뒤에 List 버튼을 누르게 되면

텍스트, 스크린샷, 번호, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

이렇게 쿼리스트링을 가지고 페이지 이동을 하게 된다.

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

여기서 아까 만든 커스텀 훅을 살짝 수정해주면 아래와 같다.

텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명 텍스트, 스크린샷, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

**#수정/삭제 페이지 개발**

useCustomMove에 moveToList와 같이 moveToModify함수를 만들어 쿼리스트링과 navigate처리를 담당해준다.

텍스트, 스크린샷, 소프트웨어, 디스플레이이(가) 표시된 사진

자동 생성된 설명

ReadPage; 조회페이지에 moveToModify도 받아와서

텍스트, 스크린샷, 폰트, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

ReadPage; 조회페이지에 수정버튼을 추가해준다.

텍스트, 스크린샷, 소프트웨어, 디스플레이이(가) 표시된 사진

자동 생성된 설명

테스트

스크린샷, 텍스트, 직사각형, 폰트이(가) 표시된 사진

자동 생성된 설명

텍스트, 스크린샷, 폰트, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

**#목록 페이지 개발**

텍스트, 스크린샷, 폰트, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

목록 리스트를 백엔드 서버에서 가져올 때 페이지 구성에 필요한 모든 것들을 다 가져오도록 사전에 구성해두었기 때문에

프론트단에서는 단순 보여주는 식으로 좀 편리하게 작업할 수가 있다.

writer null 뜨는 건 writer컬럼을 나중에 추가해서 그렇다.

텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

자동 생성된 설명

텍스트, 스크린샷, 번호, 폰트이(가) 표시된 사진

자동 생성된 설명텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명

텍스트, 스크린샷, 디스플레이, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

리스트 페이지에 리스트 컴포넌트 추가해주기.

텍스트, 스크린샷, 소프트웨어, 디스플레이이(가) 표시된 사진

자동 생성된 설명 **->** 텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명

이렇게 코드는 간단해진다. 컴포넌트의 사용으로 인해서 쿼리스트링 관련된 처리 없어져도 됨.

쿼리스트링 관련 처리하는 커스텀 훅 -> 커스텀 훅을 사용하는 컴포넌트 -> 페이지에 컴포넌트 사용

마찬가지로 readPage에도 더이상 쿼리스트링 관련 처리 없어져도 된다.

텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명**->** 텍스트, 스크린샷, 소프트웨어, 디스플레이이(가) 표시된 사진

자동 생성된 설명

테스트

텍스트, 소프트웨어, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

리스트 화면에 뿌려주는 코드 추가

텍스트, 스크린샷, 소프트웨어, 디스플레이이(가) 표시된 사진

자동 생성된 설명

다시 테스트

텍스트, 스크린샷, 소프트웨어, 번호이(가) 표시된 사진

자동 생성된 설명

여기까지 커스텀 훅과 컴포넌트를 활용하여 조회/수정/목록 페이지를 개선해보았다.

api에 이미 페이징처리와 관련된 것들을 보내주도록 해두었기 때문에 페이지관련 버튼들을 컴포넌트로 보다 편리하게 구성할 수 있겠다.