**과거의 SSR 방식 (PRG 패턴)**  
Server side rendering -> post방식으로 전송 후 redirect 하여 get 방식으로 처리(prg 방식)

client에서 post 방식으로 호출이 일어나고 서버에서 처리 후 Http헤더 중 location header를 줘서 해당 주소를 get방식으로 호출한다.

자바스크립트는 싱글 스레드이기 때문에 경고창을 띄운다는 것은 굉장히 위험하다.

자바스크립트의 엔진을 완전히 멈추겠다는 것이기 때문이다. 따라서 안쓰는 것이 좋다. 모달 창을 활용하는 것이 좋다.

**#모달 컴포넌트**

**모달 창을 만들어보자.**

**모달은 공통 컴포넌트로 사용할 것이다.**

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

자동 생성된 설명

위치 고정으로 백그라운드 페이지 위에 하나의 페이지를 만든다고 생각하면 된다.

공통 컴포넌트로 작성 후에

AddPage를 구성하는 AddComponent에 추가시키고 페이지에 접근해보면

아래와 같이 기존 페이지위에 위치 고정으로 추가시킨 bg-opacity-20으로 투명도를 준 모달 div가 페이지 전체위에 덮어져 나온다.

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

자동 생성된 설명

텍스트, 스크린샷, 소프트웨어, 컴퓨터 아이콘이(가) 표시된 사진

자동 생성된 설명

위에 커다란 페이지 위에서 작은 태그를 만들어서 위치를 지정하는 식으로 모달을 만드는 것이다.

다음 작업의 결과물부터 보자면 아래와 같다.

텍스트, 스크린샷, 소프트웨어, 컴퓨터 아이콘이(가) 표시된 사진

자동 생성된 설명

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

자동 생성된 설명

이런 식으로 absolute를 통해서 절대값으로 위치 지정하여 실행 결과처럼 나오는 모달 창을 위치 시킨 것.

**#Add 버튼 동작 추가하고 테스트 해보기**

**add페이지의 add버튼 테스트를 위해 add컴포넌트의 모달을 잠시 지우고**

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

자동 생성된 설명

이전에 add 버튼에 onclick으로 달아놓은 함수를 api호출하도록 수정하였다.

postAdd 메서드는 이전에 todoApi.js 에 작성해둔 post방식으로 api를 호출하는 메서드다.

해당 호출을 서버에서는 튜플 생성 후 기본키 반환으로 처리해줬기 때문에 반환되는 기본키를 확인할 수 있다.

텍스트, 멀티미디어 소프트웨어, 소프트웨어, 폰트이(가) 표시된 사진

자동 생성된 설명

텍스트, 스크린샷, 소프트웨어, 컴퓨터 아이콘이(가) 표시된 사진

자동 생성된 설명

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

텍스트, 스크린샷, 소프트웨어, 컴퓨터 아이콘이(가) 표시된 사진

자동 생성된 설명

**모달 창의 구성과 기능 개발하기**

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

자동 생성된 설명

Result모달 창을 add버튼의 결과에 맞게 구성해준다.

모달창 밖을 클릭하거나 close를 클릭했을 때 사용할 callback함수도 등록해준다.

텍스트, 스크린샷, 소프트웨어, 디스플레이이(가) 표시된 사진

자동 생성된 설명

Api 호출의 결과를 받을 result를 상태관리해준다.

호출 결과로 result를 구성한다.

add하였으니 작성했던 칸을 비운다.

close버튼을 클릭하면 리스트로 이동시킨다.

텍스트, 스크린샷, 디스플레이, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

result모달 컴포넌트에 close시 사용함수와 api 호출 결과 데이터를 넘겨준다.

텍스트, 스크린샷, 폰트, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

텍스트, 스크린샷, 소프트웨어, 번호이(가) 표시된 사진

자동 생성된 설명