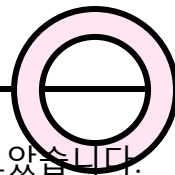


최수진 소감



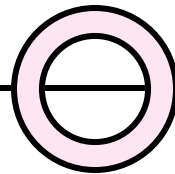
- redux가 아직 익숙지 않아서 좀 더 발전시키고 싶은 마음에 redux로 구현해 보았습니다.
- 처음에는 위치에 따른 지역 대기질을 구현하려 했지만, 두 개의 api를 사용해야 하는 부분이 두 명에서 하기에는 시간이 걸리겠다는 생각이 들었습니다. 따라서 서울이라는 한정적인 지역으로 추려서 구현하게 되었습니다.
- 액션함수를 디스패치할 때, 필요한 데이터만 추려서 상태값에 반영하고 싶었습니다. 하지만 받아온 payload 데이터 중에 필요한 items는 몇 겹 내부에 존재했기 때문인지 가져온 데이터를 쓸 수는 있어도 다시 추려서 받아오는 계획은 실패했습니다.

```
▼ {response: {body: {t
  ▼ response: {body: {
    ▼ body: {totalCoun
      ▼ items: [{so2Gr
        ▶ 0: {so2Grade:
        ▶ 1: {so2Grade:
        ▶ 2: {so2Grade:
```

- 차선택으로 삼항연산자와 함수로 구현하였고 이 부분이 처음에 계획과는 달라져서 아쉽습니다.
- 대기 지수에 따라서 이모지를 표현하였는데 등급은 4단계였지만 출력된 값은 1,2단계뿐이어서 살짝 아쉽기도 하고 서울공기가 이렇게 좋았나? 싶었습니다.
- 클론코딩과는 달리 원하는 바를 상상하면서 로직을 짜는 과정이 재밌게 느껴졌습니다.
- 소스코드를 팀원과 공유하면서 코딩한 적이 처음이었는데 장단점이 있다고 생각되었습니다.
- 장점은 고민되는 부분을 의논할 수 있다는 점과 작성 코드가 줄어든다는 점이었습니다.
- 단점은 git사용이 아직 서툴러서 pull push할 때 반드시 상대방에게 얘기를 해야 하는 점이 번거로웠습니다.



박 세 영 소 감



- 처음 어떤 방향으로 나아갈지는 수월하게 논의되고 결정이 되었습니다. 서로 응원하고 다독이며 했고 완성했기에 많이 뿌듯합니다.
- 어떻게 각자의 영역을 나누고 시작할지 많이 고민을 했습니다. 아직 코드를 짜는 것이 서툰 저를 위해 수진님께서 코드 뼈대를 작성해주시고 제가 그 위의 스타일을 입히는 방식으로 했는데 수진님의 코드를 보며 좋은 공부가 되었습니다.
- 수진님의 코드 위에 제가 CSS를 덧붙이며 대기질의 상태에 따라 배경색을 바꾸는 것을 하고싶었고 최종적으로 useRef를 이용하였습니다.
useRef가 바로 생각이 나지않아 많은 시간이 걸렸고 useRef를 적용하고 코드를 작성할때 수진님께서 작성해주신 Grader함수들을 이용해 적용하고 싶었지만 실패했고 제가 따로 코드를 작성하여 추가해 대기질의 상태에 따라 배경색을 바꾸도록 구현했습니다.
- 대기질의 상태에 따라 배경색을 바꾸도록 구현했지만 서울의 대기질이 전부 보통으로 나와 구현 결과 스크린샷이 전부 보통일때 뿐인게 아쉽습니다..
- 수진님을 보며 저의 문제점을 발견했습니다. 같이 하는 팀원이 있기에 발견할수 있었던 문제점이었고 문제점을 개선할 기회를 얻었다는것이 매우 좋다고 생각합니다.

