소프트웨어공학 Homework #2

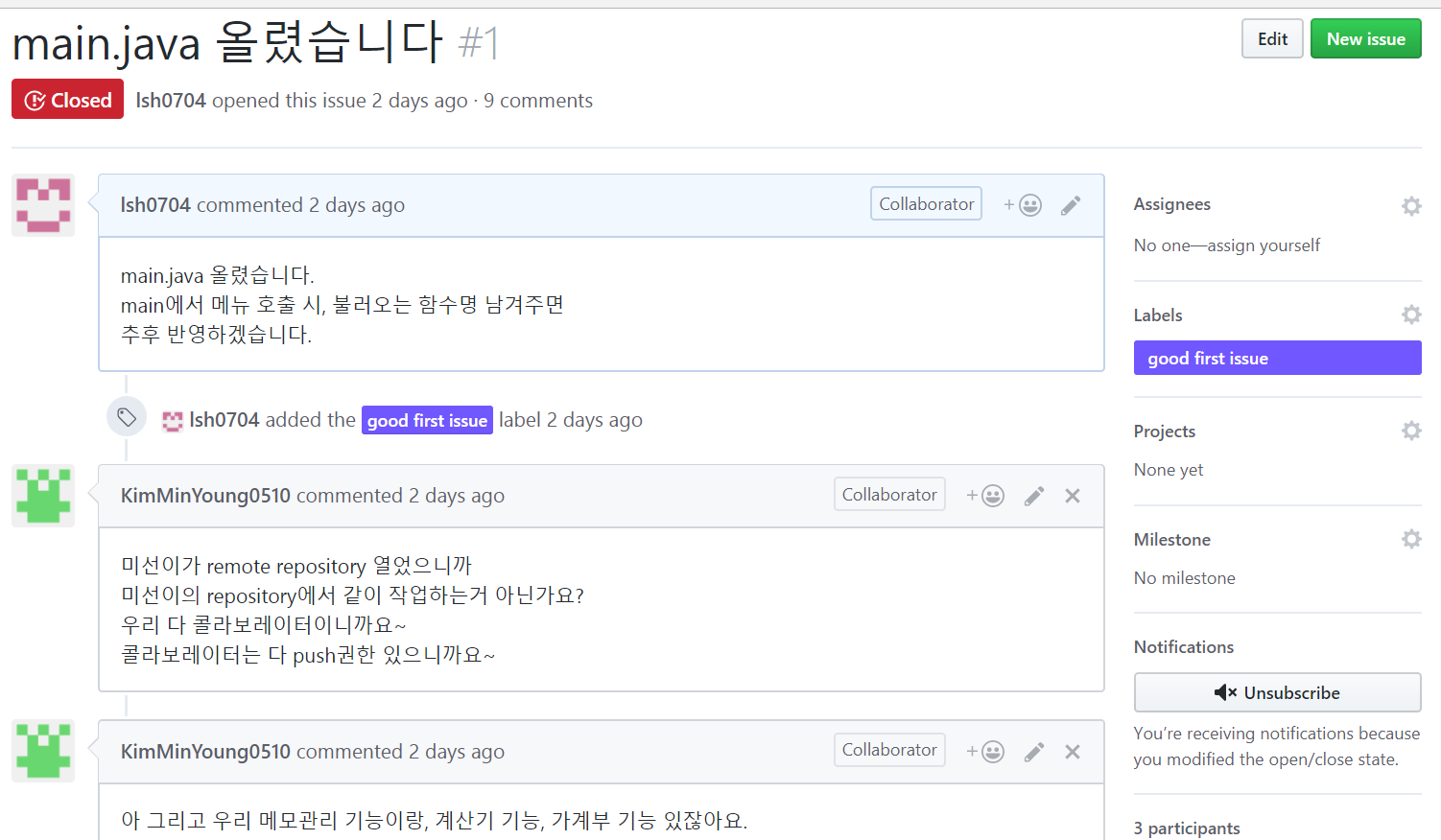
2. Issue Tracker

<team’s GitHub Remote URL>

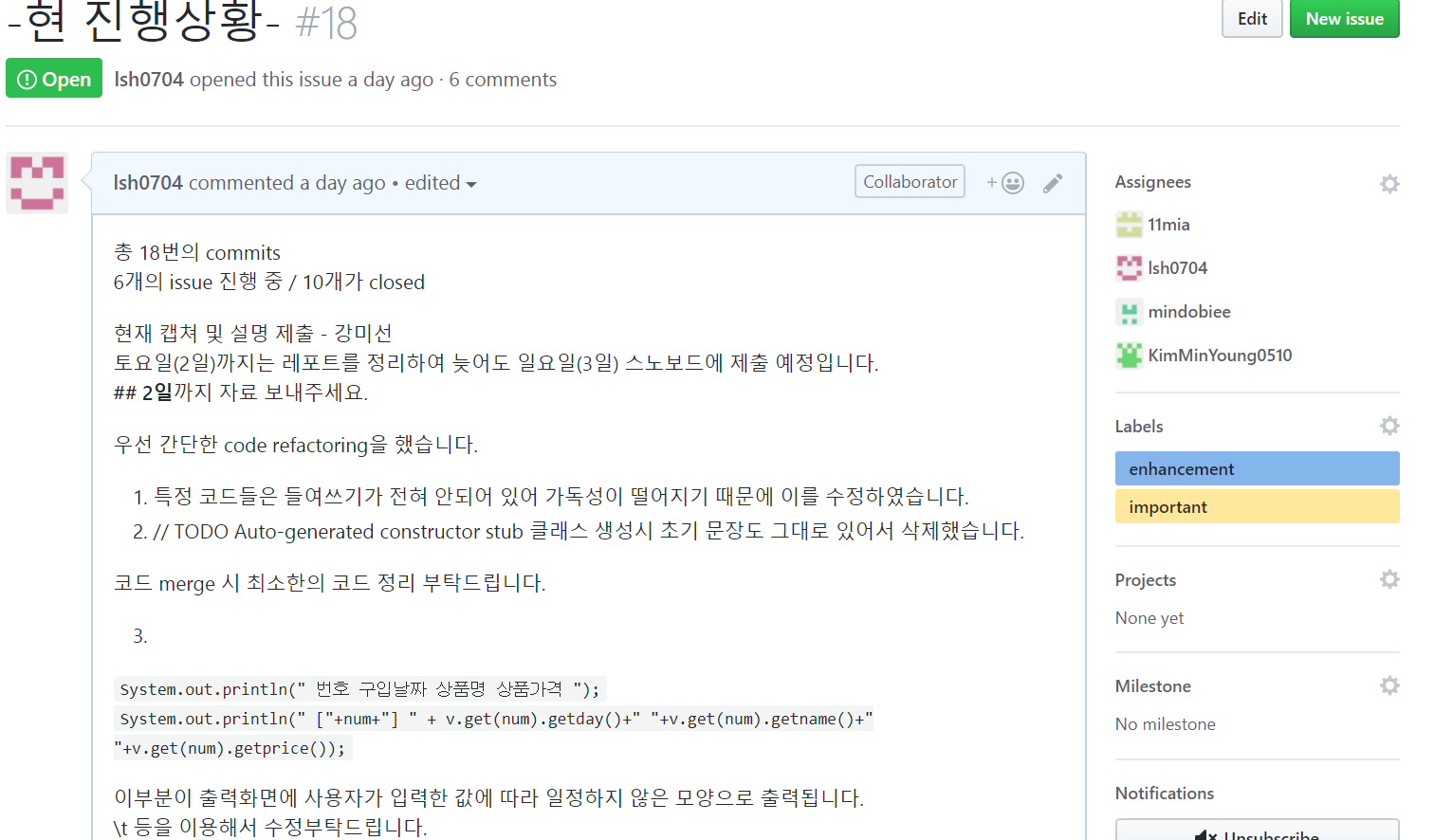
-<https://github.com/11mia/GiveUsManJeom.git>

<4> 임소희 (1515655)

① create/open a new issue



전 기능을 사용하기 위한 메인 코드를 팀원들이 테스트 할 수 있도록 commit하였고, 이를 이슈 트래커에 알렸습니다. 이 때 메인을 올린 것이 제 개인 브랜치 였는데, 이슈 트래커에서의 의견 공유를 통해 main.java를 master로 merge 하자는 결과를 얻을 수 있었습니다.

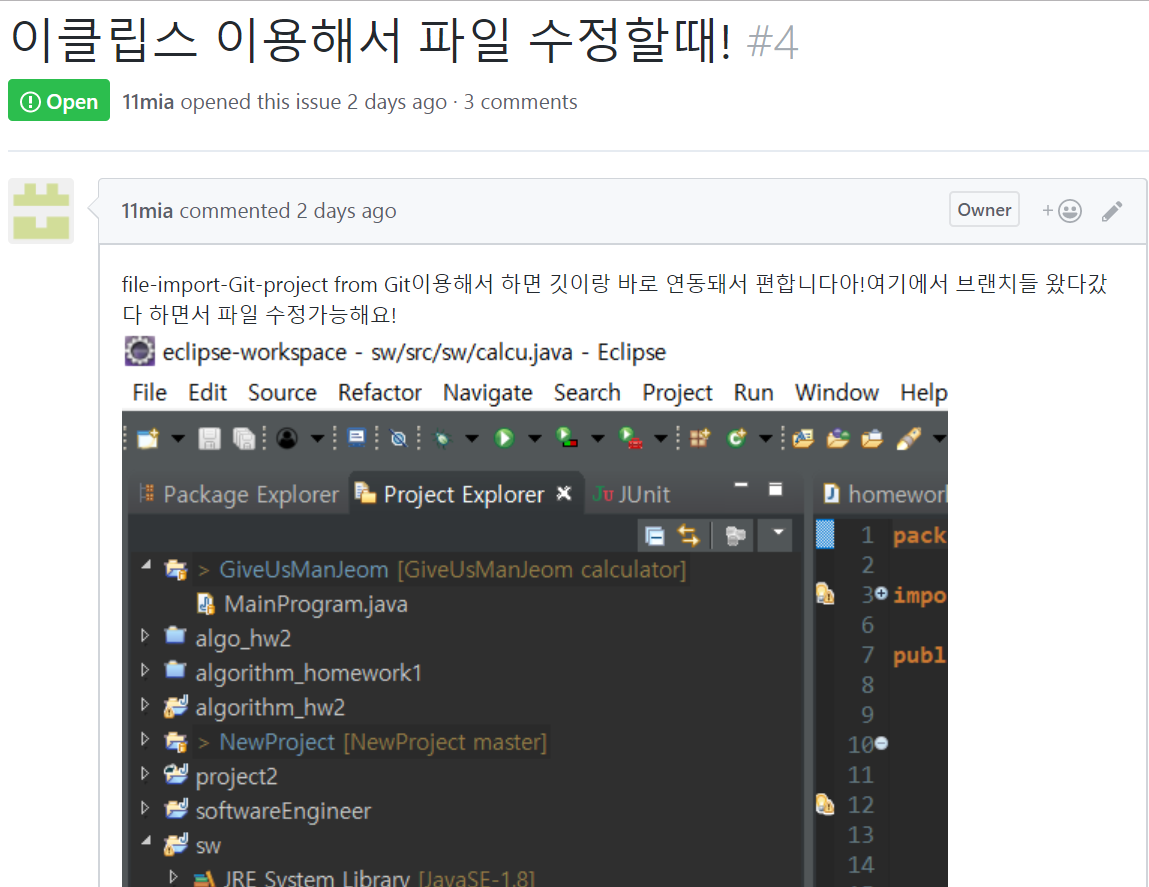


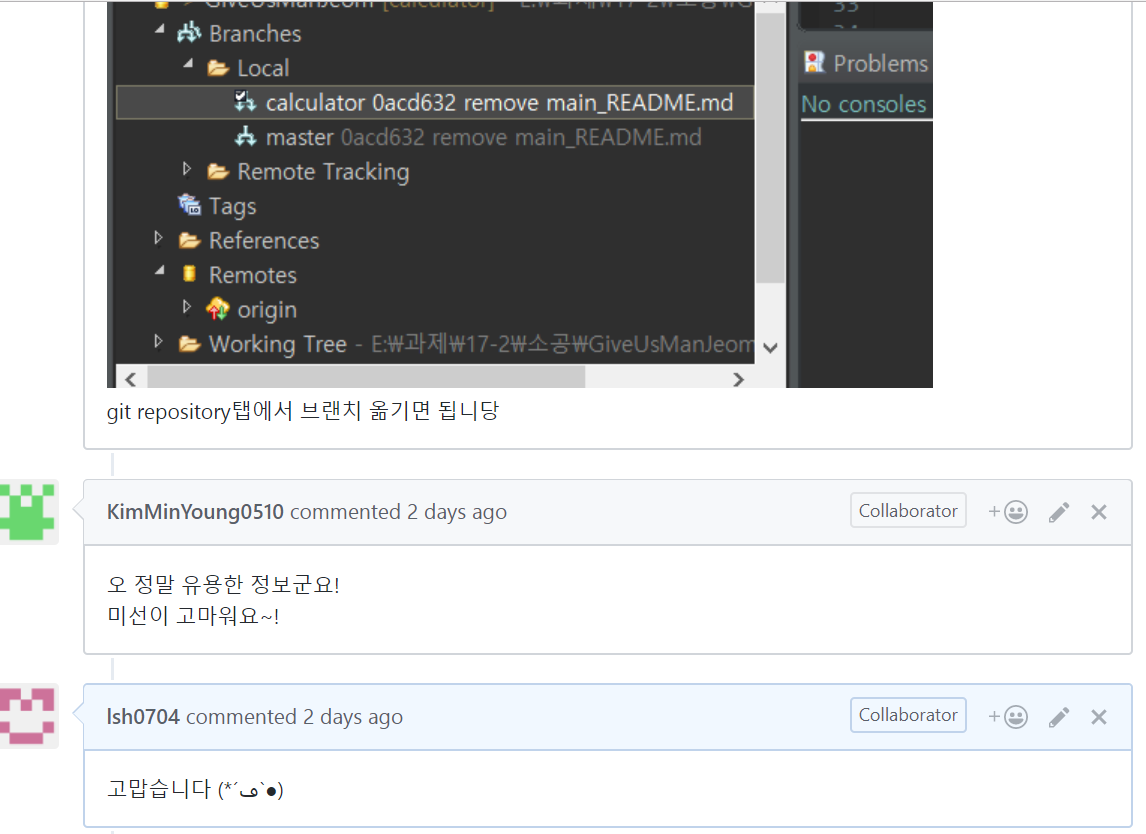
현재 진행상황과 레포트를 작성하는 데 있어 기한을 설정하기 위해 이슈를 작성하였습니다. code refactoring한 결과를 보고 하고, 팀원에게 수정할 사항을 요청한 결과 수정하기로 피드백 받는 결과를 얻었습니다.



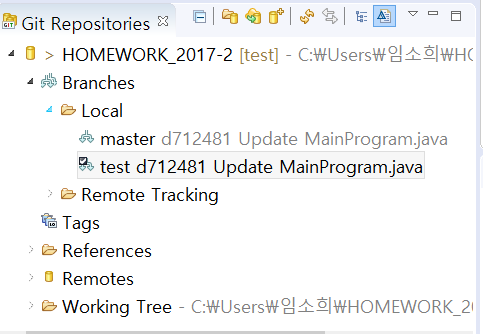
2차 code refactoring을 위해 각 기능 클래스의 순서를 정돈하였고, 이해하기 쉽도록 각 기능마다 주석을 달았음을 알리는 이슈를 작성하였습니다. 기타 디자인적 수정사항을 팀원에게 요청하였고, 팀원들이 수정하여 git에 commit 하였습니다.

② read/view an issue





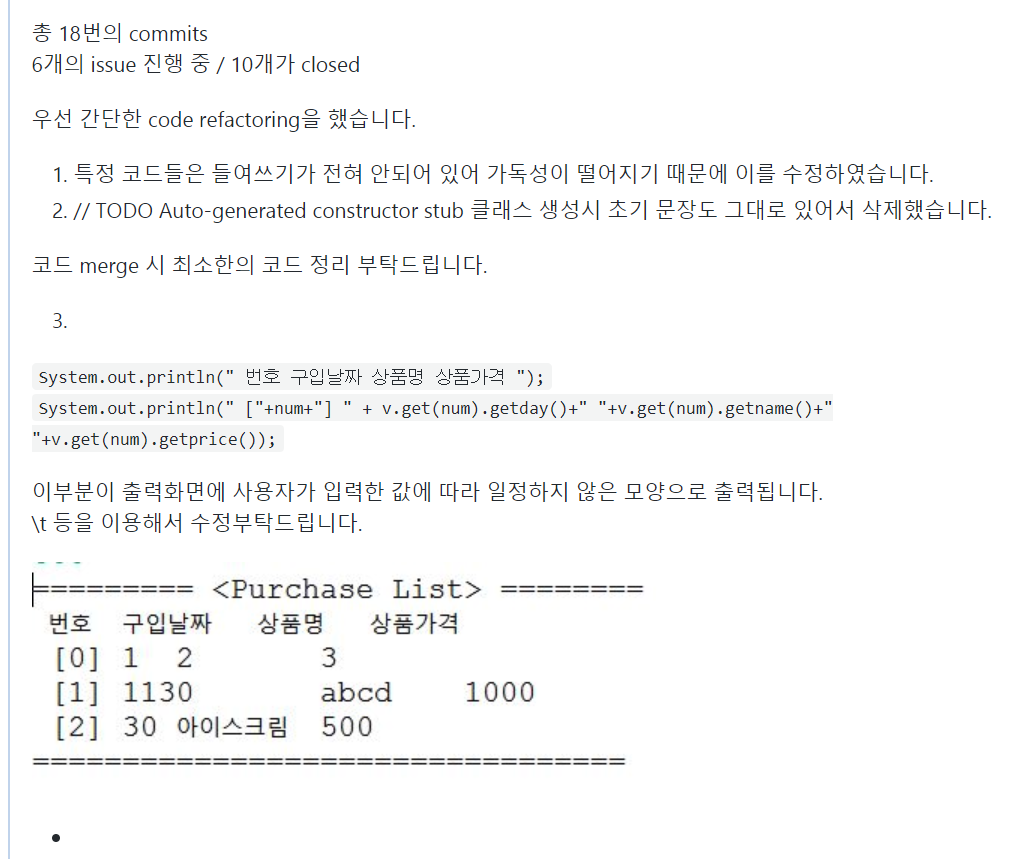
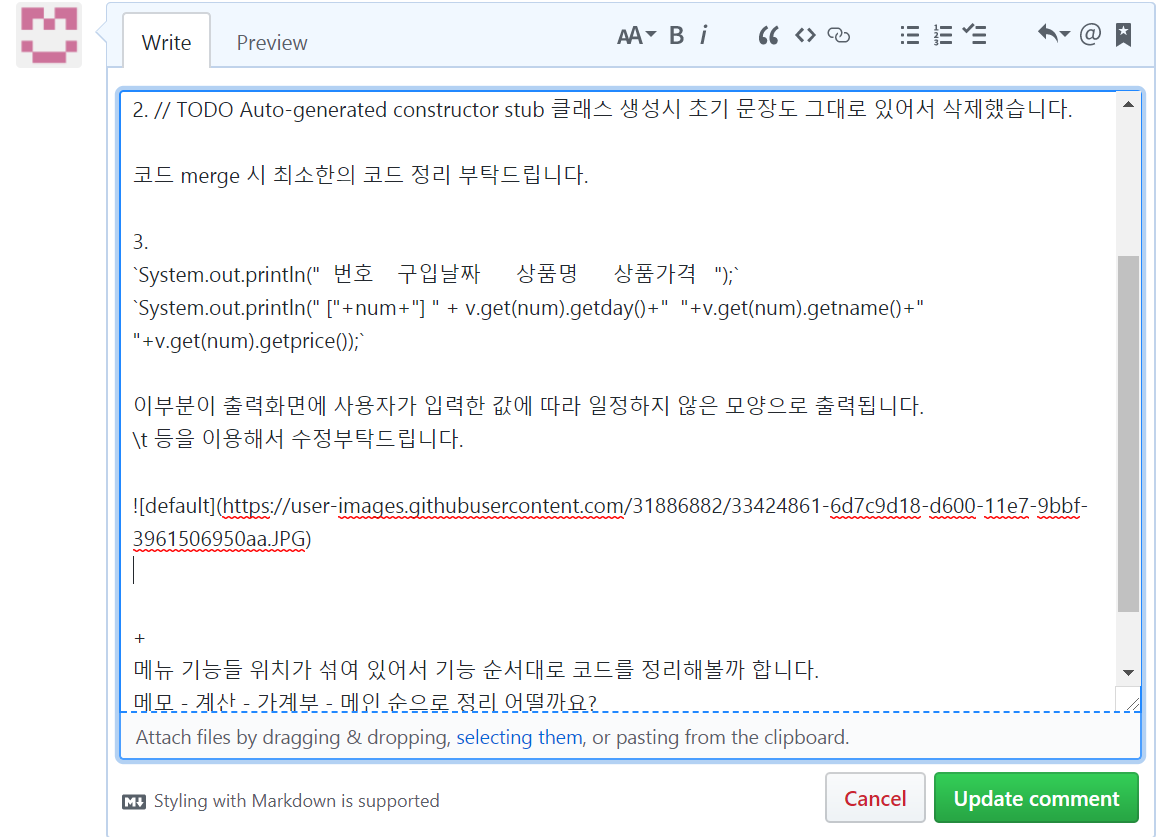
이슈 트래커에 팀원이 이클립스 내에서 git 연동 사용법을 올렸습니다. 이를 읽고 실제로 코드 작업에 활용하였습니다.



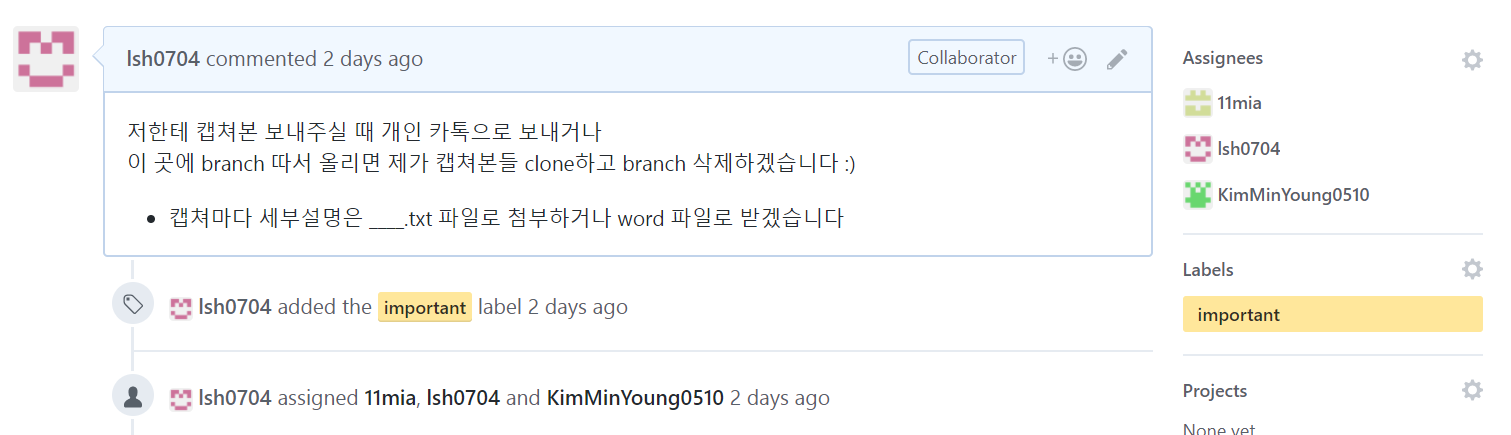
③ update an issue



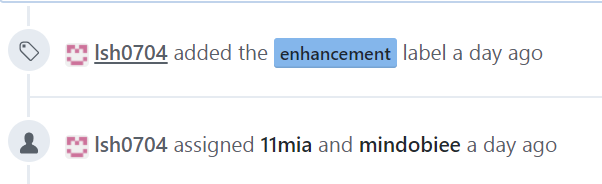
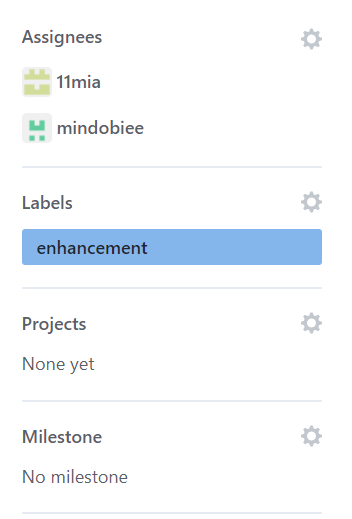
기존에 작성한 이슈입니다.



수정할 때 캡처를 삽입하여, 출력 오류가 어떻게 나는지 한눈에 볼 수 있게 하였습니다.

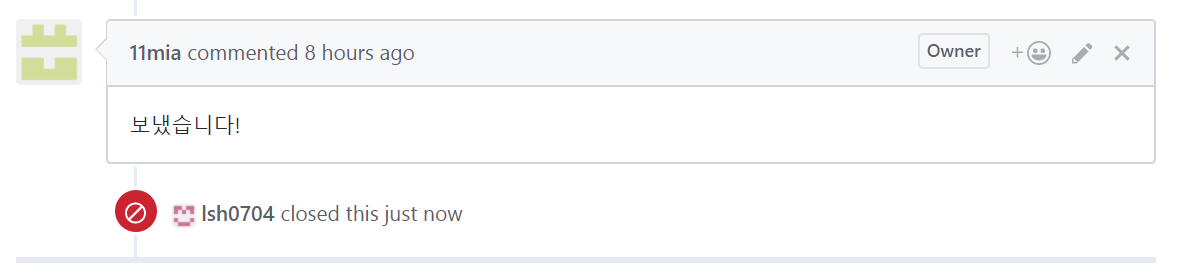
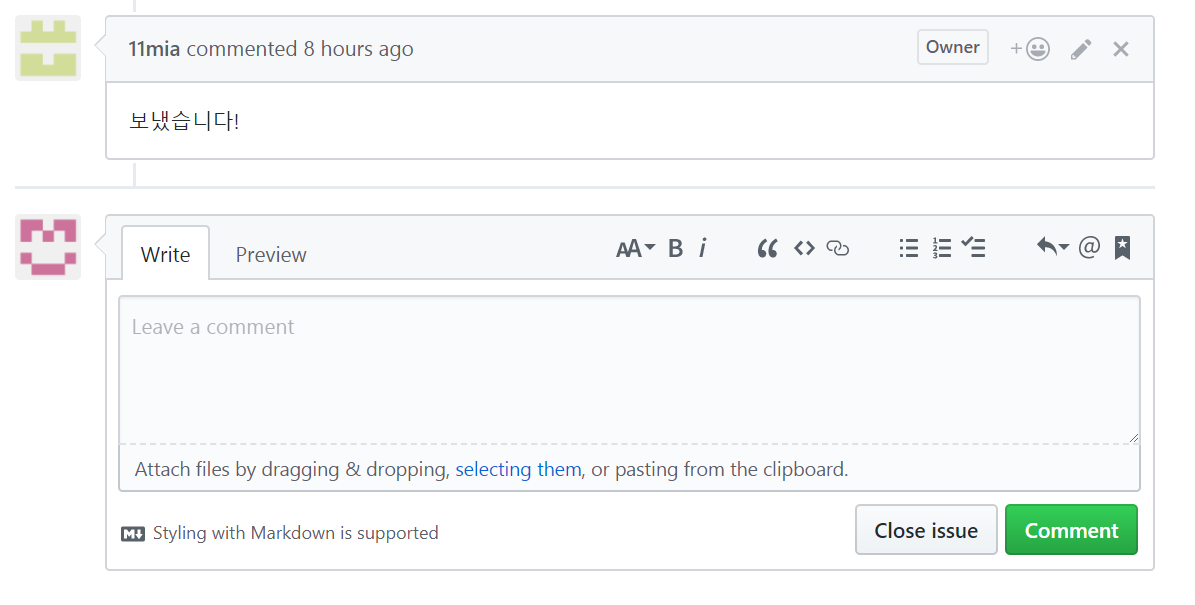


레포트 작성 시, 레포트 제출 담당인 팀원에게 파일 보내는 방법을 공지하므로 Label을 important로 붙이고, 팀원 전부를 assignee 하였습니다.



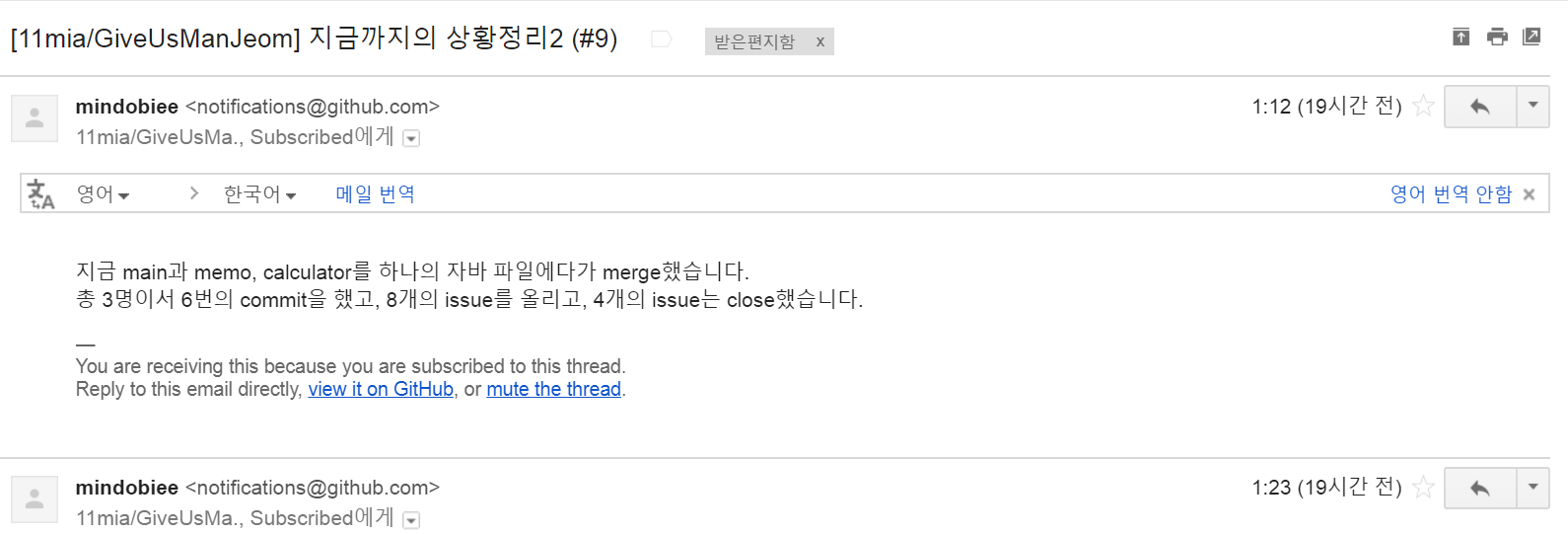
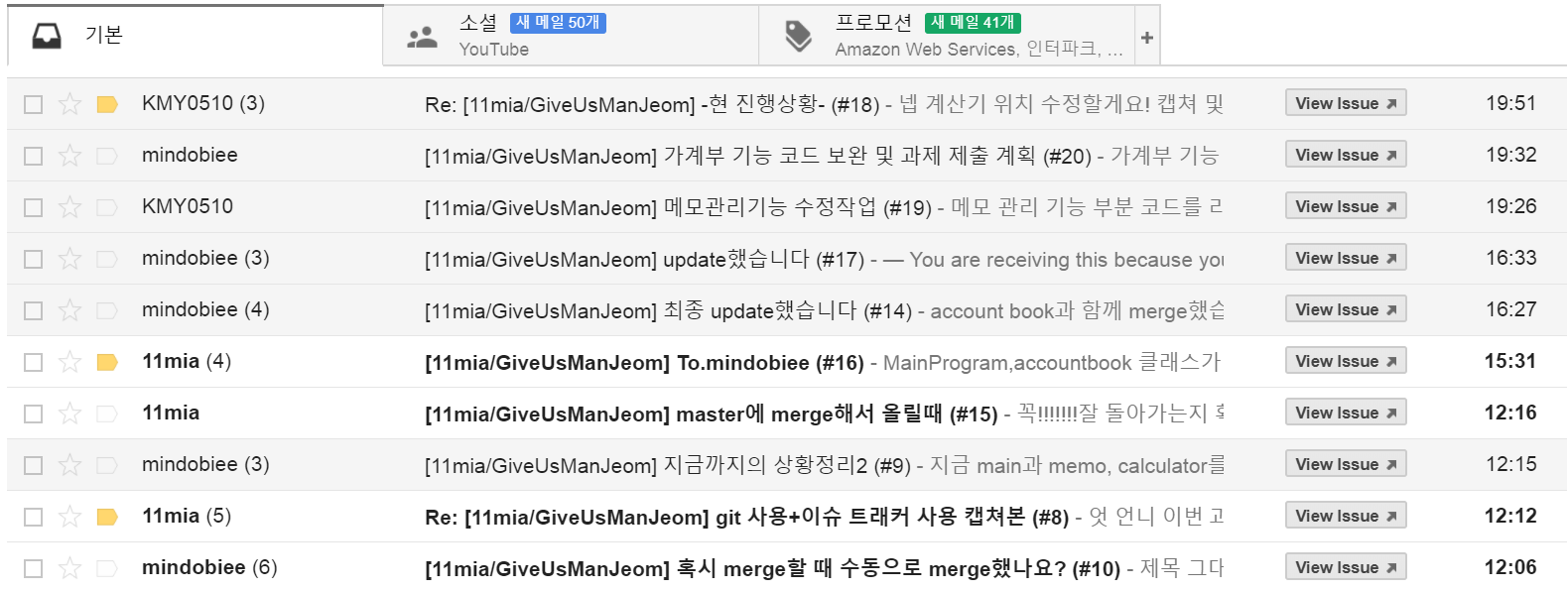
코드 수정 요청을 위해 해당하는 기능을 담당하는 팀원을 assignee하고, 요청 사항이 디자인 측면에서의 수정이므로 enhancement label을 붙였습니다.

④ close/resolve an issue



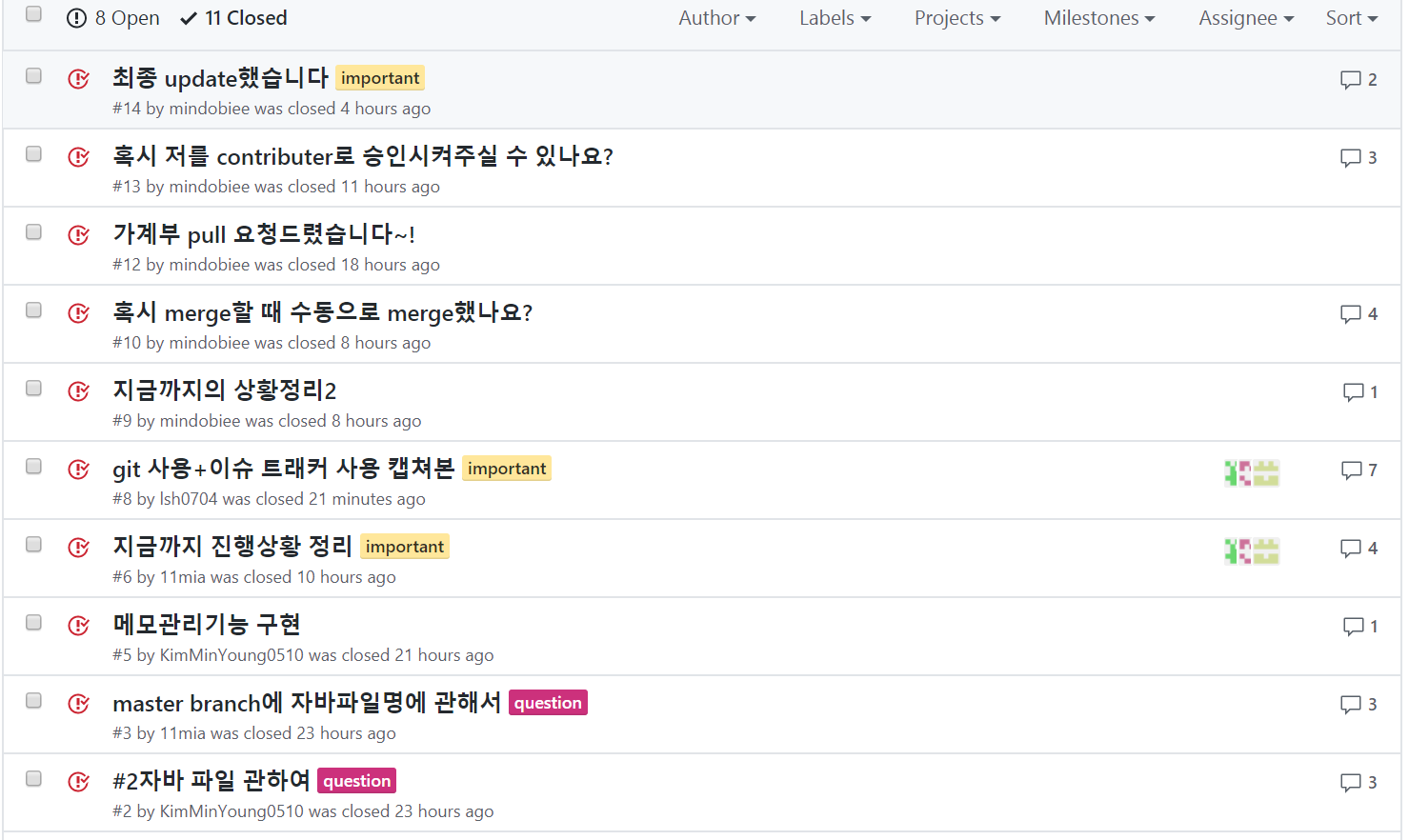
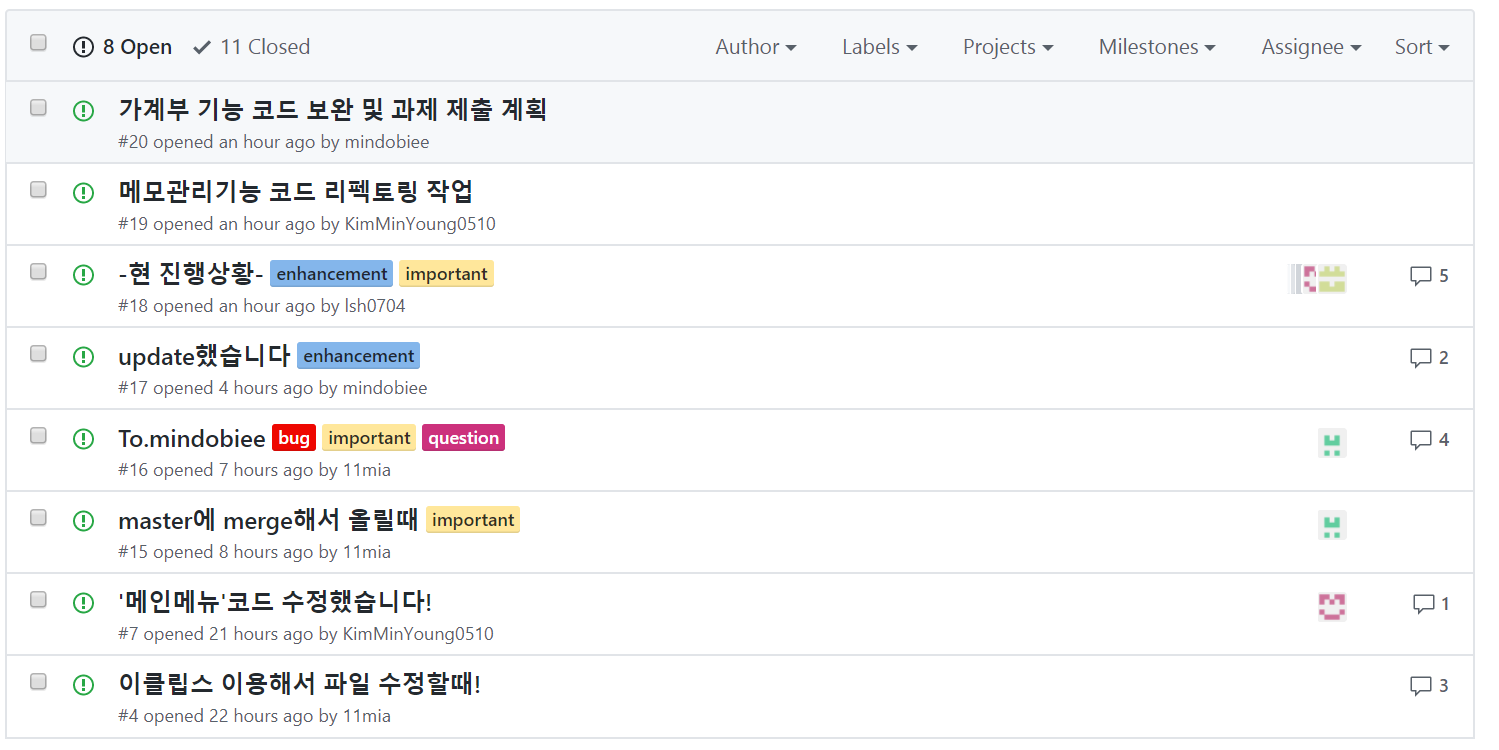
위에 작성한 이슈(레포트 파일 보내는 방법 공지)에 대해 팀원 전부가 확인하였고, 레포트 파일 제출 기한 공지를 새로 이슈로 작성하기 위해 closed 했습니다. 공지한 결과, 팀원 한 명은 파일을 제출 해주었습니다.

⑤ notify issues to project members

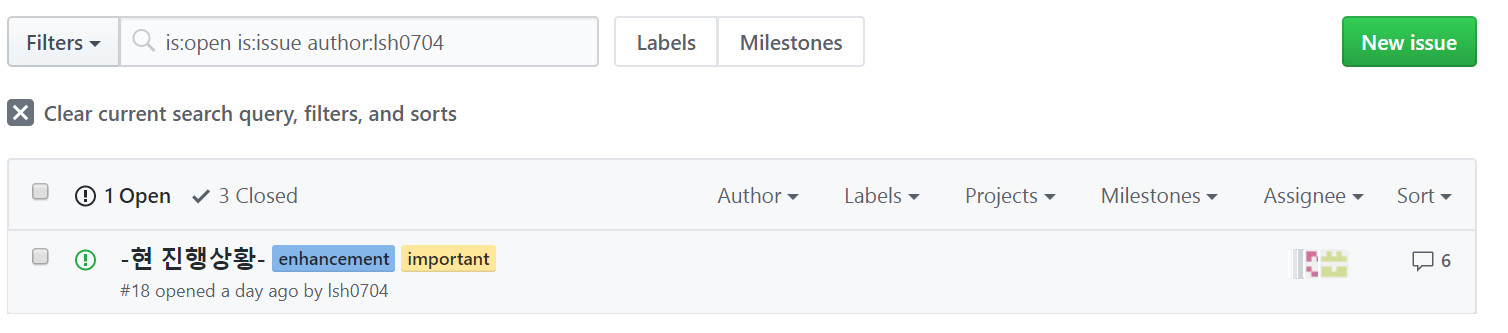


메일을 통해 이슈가 작성되었음을 실시간으로 알람 받을 수 있었습니다. 이 이슈를 통해 저희의 프로젝트 진행 상황을 확인할 수 있었습니다.

⑥ list all issues



Open, Closed를 클릭하여 Open된 이슈 리스트 및 개수, Closed된 이슈 리스트 및 개수를 확인할 수 있었습니다.



필터를 이용하면 좀 더 세분화 하여 이슈를 리스트 할 수 있습니다. 이 경우에는 Open된 이슈 중, 제가 작성한 이슈만을 리스트한 결과입니다.