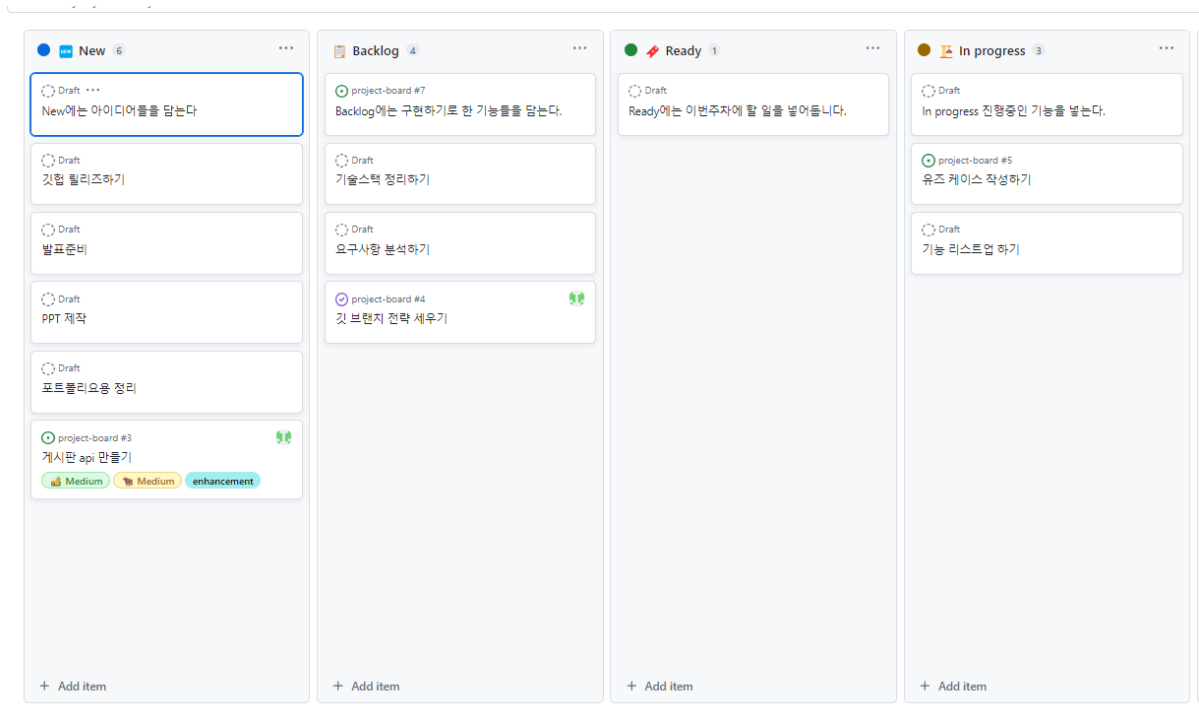




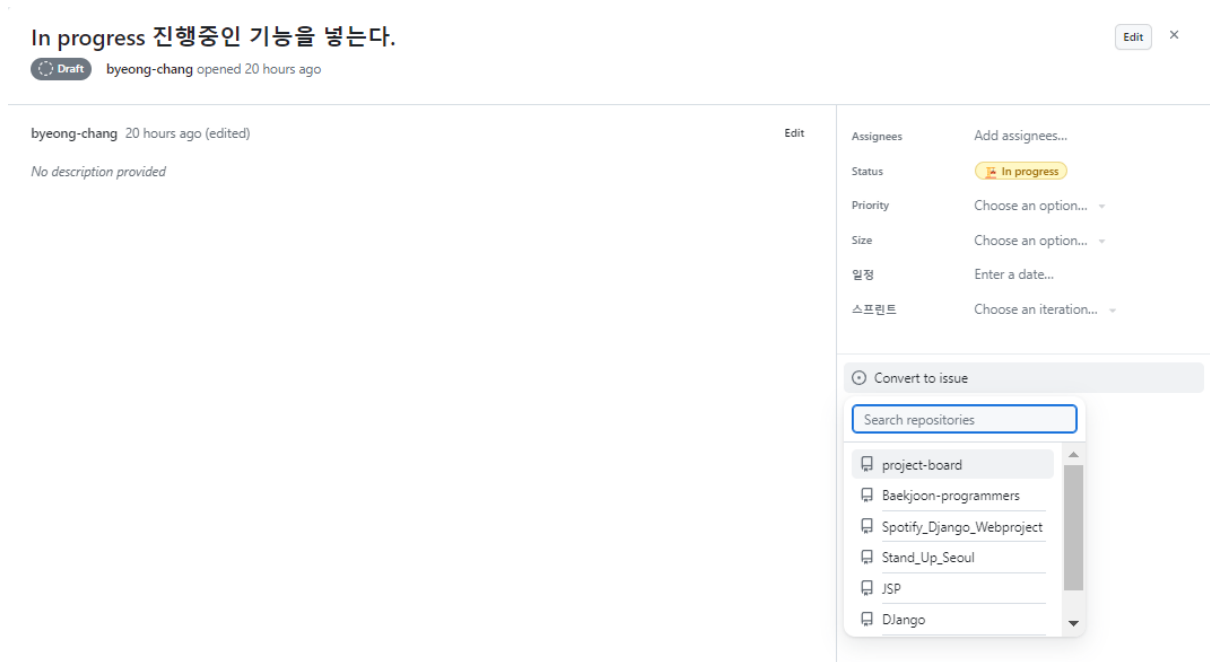
# 깃 브랜치 전략

칸반보드를 기준으로 시작합니다.

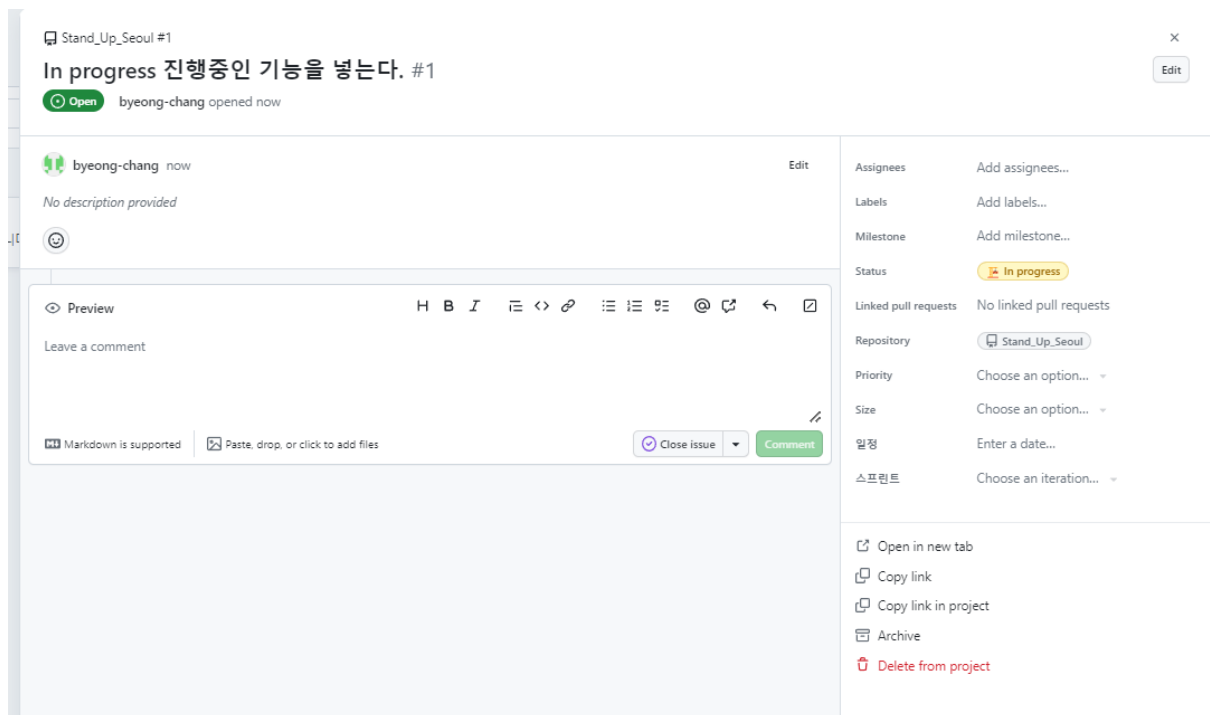
- Add items로 원하는 작업을 만들 수 있고, 현재 진행중인 작업을 In Progress에 넣어줍니다.



- In Progress에 있는 작업 중 원하는 작업을 클릭한 후 Convert to issue 버튼을 누릅니다.
- 그 경우 어떤 레포지토리에 이슈로 올릴 것인지 선택이 가능하고, Stand\_Up\_Seoul을 클릭해줍니다.
- Assignees는 누가 이 일을 하고 있는지 할당해주는 칸입니다. 본인과 같이 작업한 사람을 넣어주세요
- priority,size,일정은 알아서 골라주시고 스프린트의 경우 설정해두면 주차별로 어떤일을 했는지 테이블형태로 확인 가능합니다. 넣어주시면 좋아요



- 그 결과 아래와 같이 이슈번호(#1)이 붙게 되고, 해당 이슈번호로 commit을 추가할 경우 commit 메시지가 이슈별로 칸반보드 및 깃허브 이슈에 깔끔하게 정리됩니다.
- 나중에 이슈별로 commit이력들을 조회하기 수월하기 때문에 각 이슈별 기능을 만들때 있던 일들을 남겨주시면 됩니다.
- Edit버튼을 눌러 어떤 일을 하는 이슈인지 정리해주면 좋을듯 합니다.(아래 사진이 예시임)



## 깃헙 프로젝트와 이슈 정리하기 #2



djkeh just now (edited)

Edit

깃헙 프로젝트를 세팅하고, 카드를 만들어 정리하자.

- ☒ 프로젝트 베타 만들기
- ☒ 카드 목록 만들기 - 강의 커리큘럼 참고
- ☒ 이슈로 적절히 바꾸기

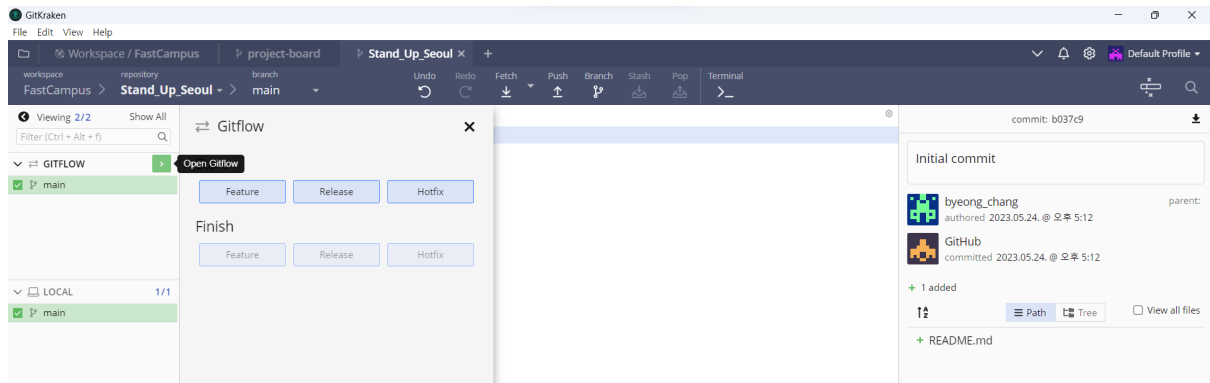
## GitKraken에서 레포 연동

- 크라켄을 설치하지 않았다면 아래 page로 이동하여 설치해주세요

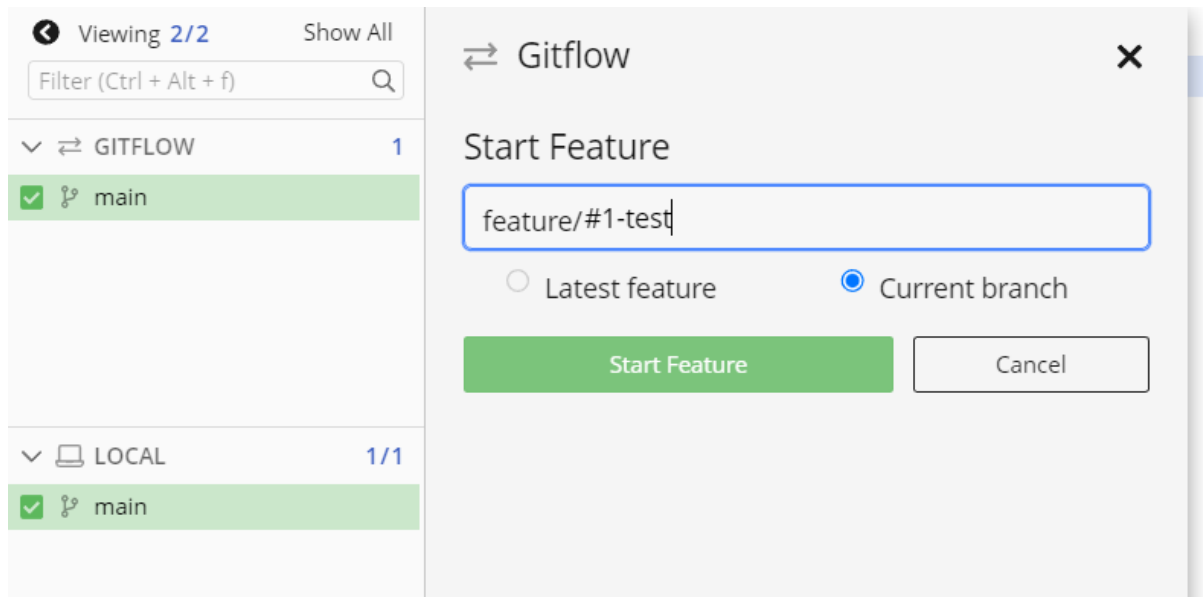


### 깃허브 깃크라켄 연동 및 gitflow 설정

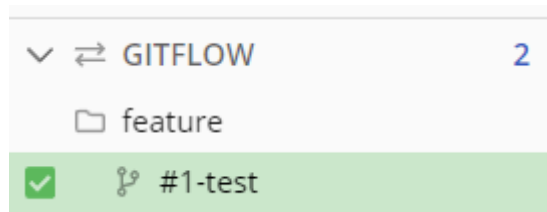
- 아래 그림의 open gitflow 화살표 클릭 후 feature 클릭



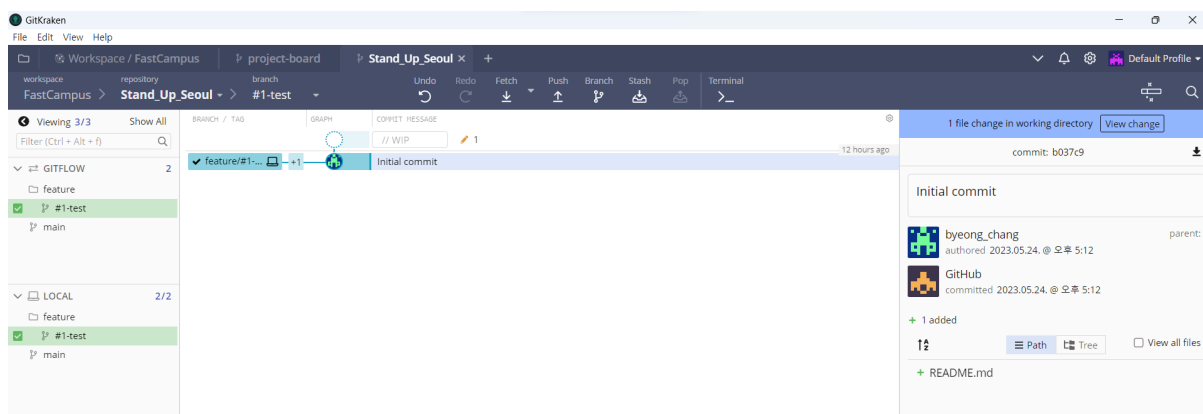
- 우리가 만든 이슈번호가 #1이기 때문에 아래와 같이 이름을 짓고(test 대신 원하는 부연 설명 넣어주면 된다) Start Feature 클릭
- 이럴경우 feature라는 디렉토리 아래에 #1-test 브랜치가 생성되게 되는데, 우리는 backend, data, frontend로 나누어서 디렉토리명을 설정하고 브랜치를 생성할지 얘기를 해보면 좋겠습니다.



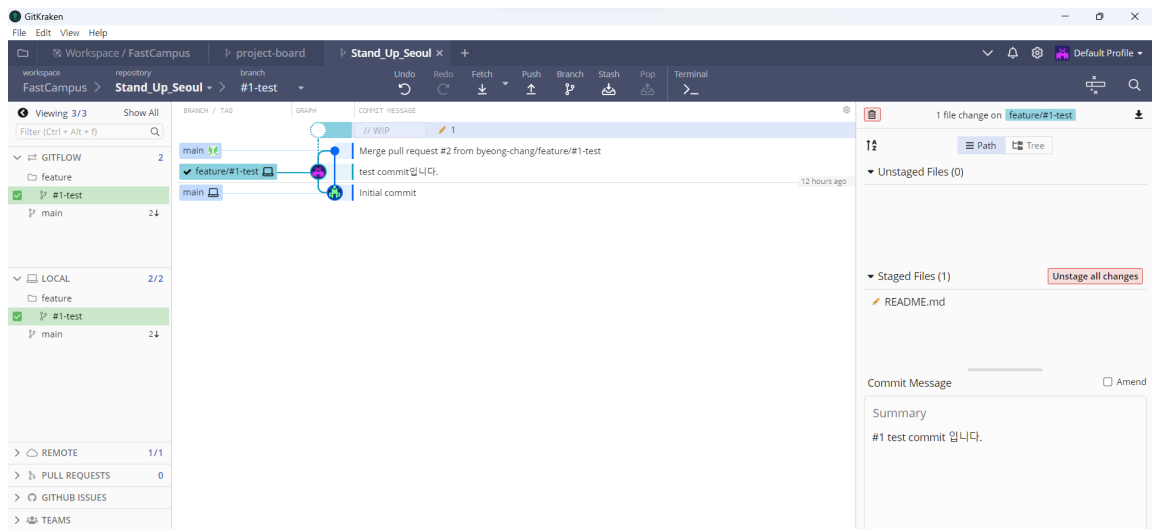
- 그러면 아래와 같이 feature 디렉토리 아래에 브랜치가 생성된다.
- 여기서 수정하여 일어난 일은 Issue 기준으로 commit 되도록 하기 위해서 이름을 지어 줄 때 반드시 #이슈번호 로 시작해야한다.



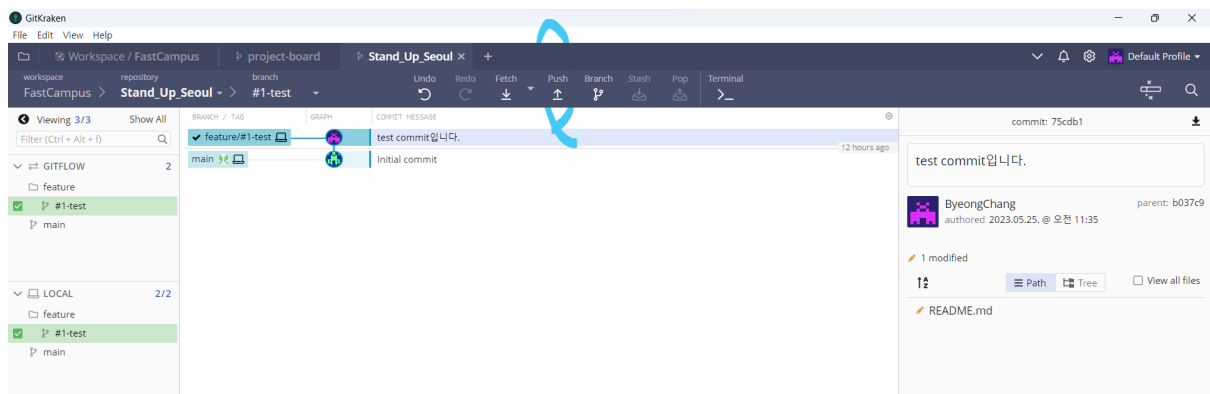
- 간단히 README.md 파일을 수정하고 gitkraken을 확인하면 오른쪽 상단 변경이력을 볼 수 있다.
- View change 클릭

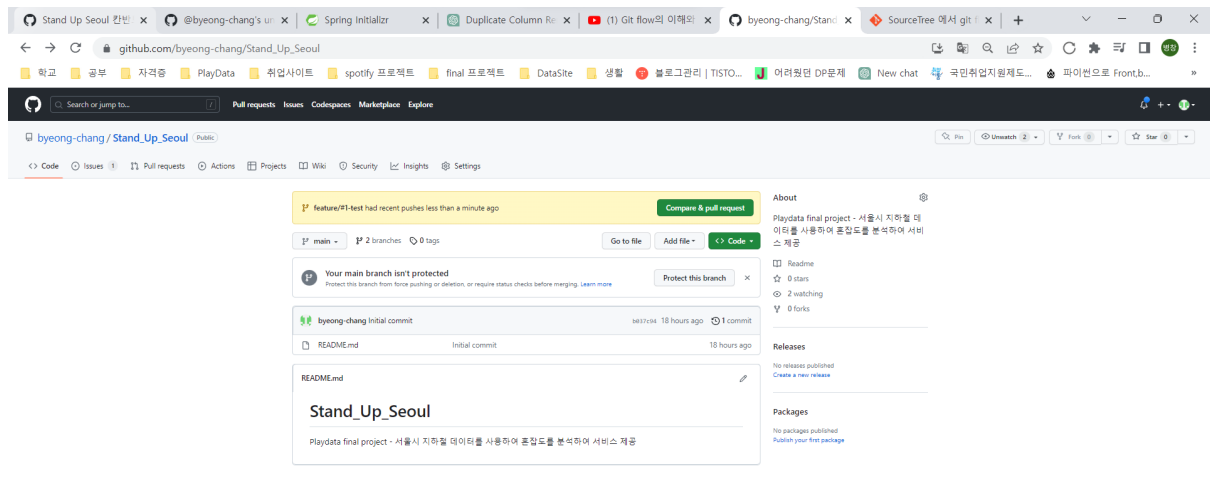


- 아래 그림에서 초록색 Stage all changes를 눌러주고(또는 우리 github에 올릴 파일만 추가해 주고 - 기능별로 커밋을 여러번 하는것이 좋기 때문에 Stage all을 누르기 보단 하나씩 직접 추가한 후 commit 메시지를 작성하는게 좋습니다.) commit 메시지를 작성한 후 오른쪽 하단의 Stage files/ changes to commit 클릭
  - 아래 사진처럼 commit 메시지 시작부분에 #1 test commit 입니다 처럼 #이슈번호를 넣어주어야 합니다.

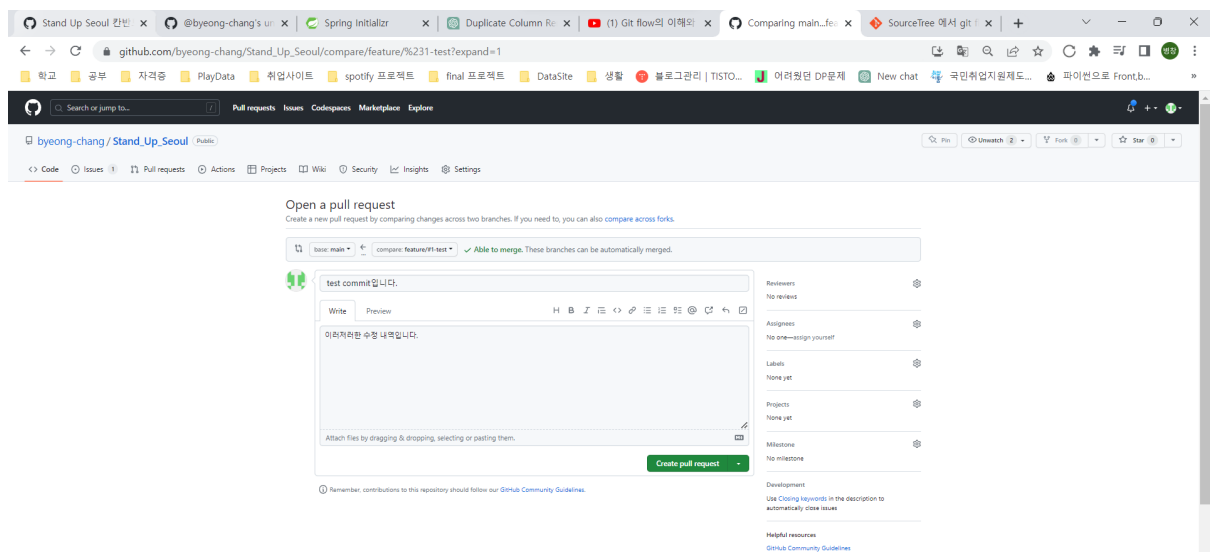


- push 버튼 클릭



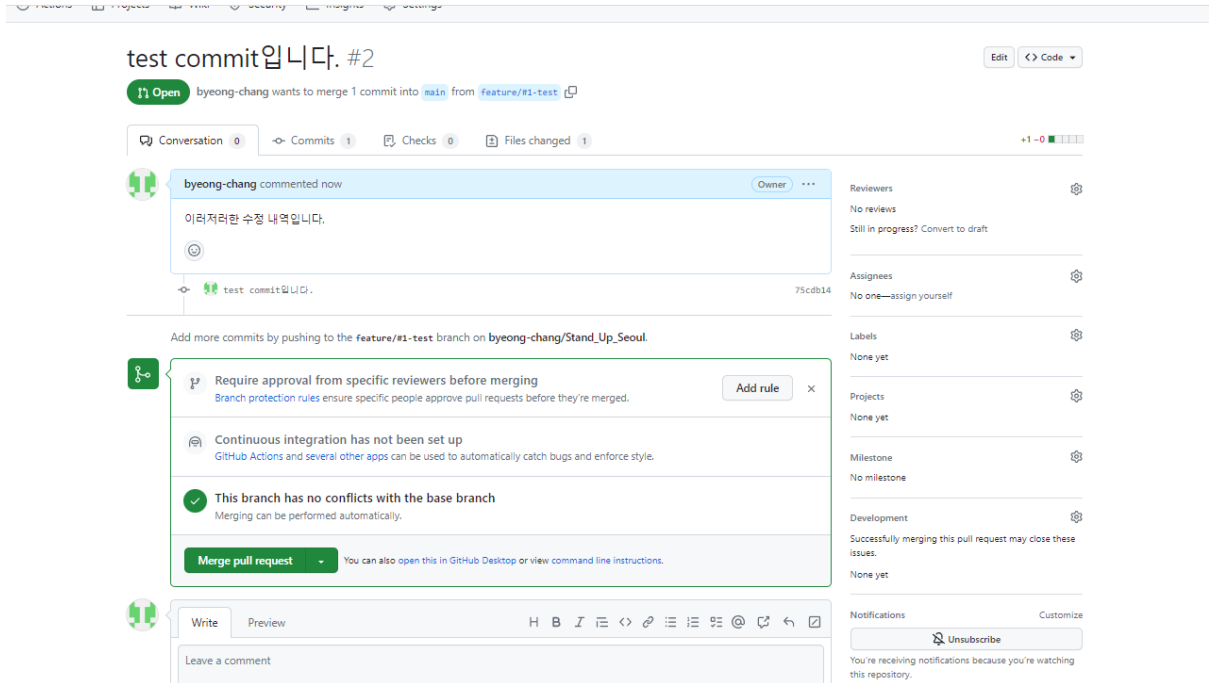


- 어떤 기능을 추가했는지 메시지를 남겨주고 create pull request 해주면 제가 확인해서 merge 할 겁니다.(사실 merge를 제한하지 않았기에 본인이 합칠 수 있습니다.)
- merge 할 경우 자동 branch 삭제 옵션을 사용하도록 설정해두었기 때문에 gitkraken에 는 남아 있지만 branch 자체는 사라지게 됩니다.
  - 원한다면 branch는 복구가 가능하다고 합니다.
- 그 외에 오른쪽 옵션으로 누가 이 일을 했고(Assignees)
- 누가 이 글을 읽으면 좋겠고(reviewers) 등등 설정을 해주는게 보기 좋긴 합니다.
- 아래 사진 기준으로 적지는 않았지만 이러저러한 수정 내용입니다 부분 아래에 This close #issue number(이 경우엔 #1)를 넣어주어야 합니다. This close를 사용해주면 이슈가 종료되면서 칸반보드에서 자동으로 Done 필드로 이동이 됩니다.



- merge 해줌

- merge하는 경우에도 이슈가 생성됩니다(#2가 생성됨을 볼 수 있음)
- 그래서 merge를 남용할 경우 issue가 너무 많이 생기는 문제가 생기기 때문에 주의바랍니다.



- #1의 issue를 확인해 보면 commit 이력들이 다 남아있어서 보기 좋게 정리됨을 볼 수 있습니다.

