



SMU UMC

GIT 과제제출 안내



0. 사전 준비

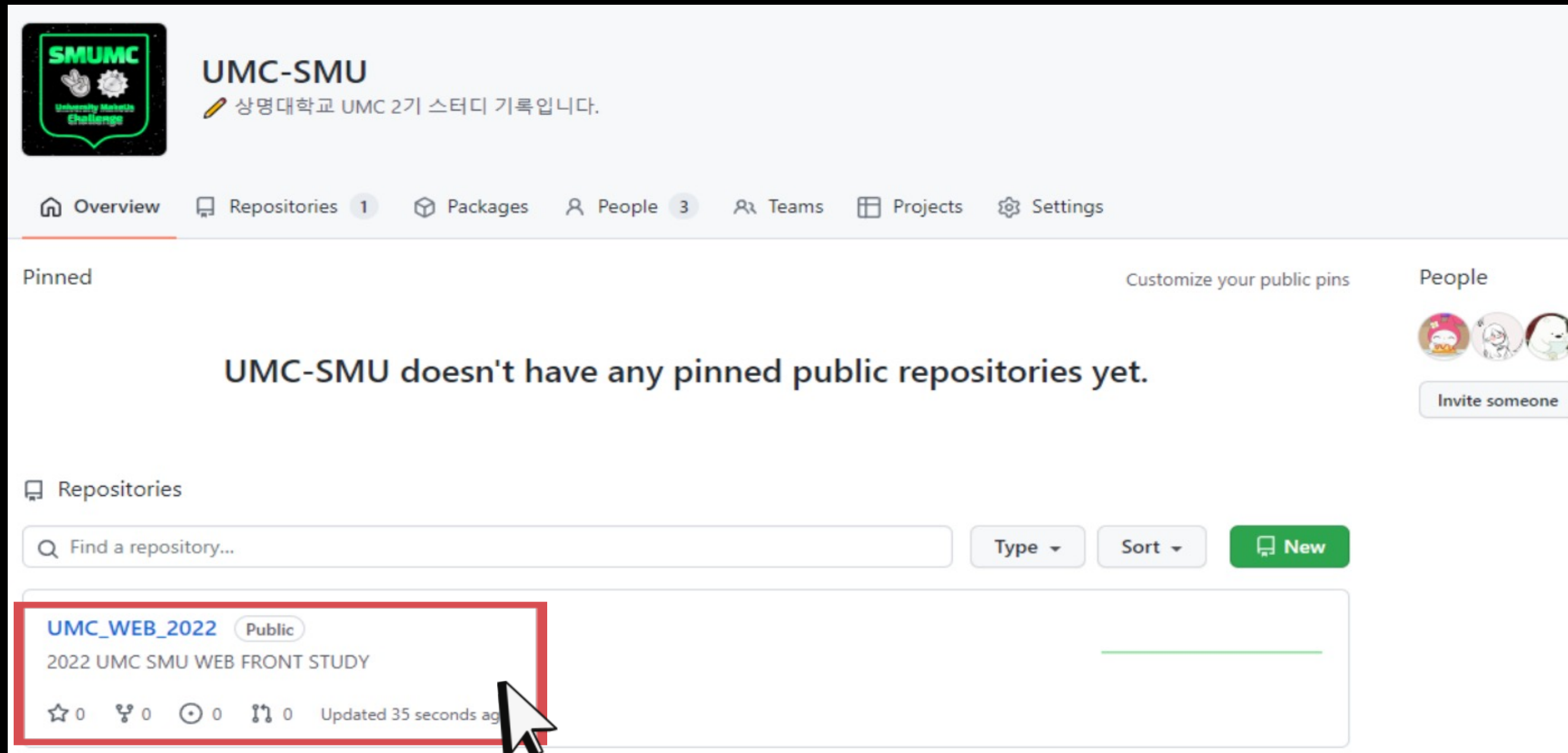
이메일로 받은 **Join @UMC-SMU** 클릭

! 각 스터디의 리더는 미니 or 루피에게 깃허브 이메일을 받은 후, 공유해주세요.



0. 사전 준비

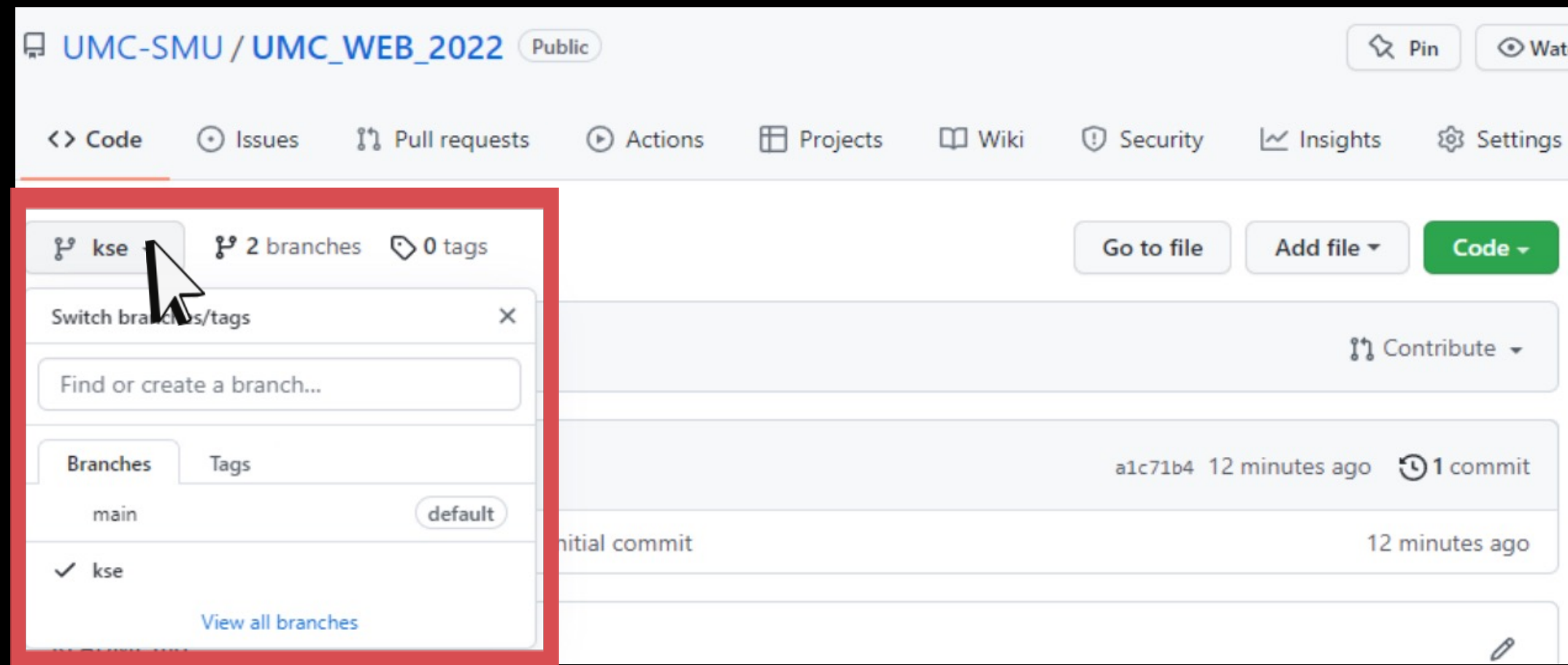
초대받은 **UMC-SMU organization**에서
각 **파트**의 repository 클릭



1. 제출할 본인 Branch 확인

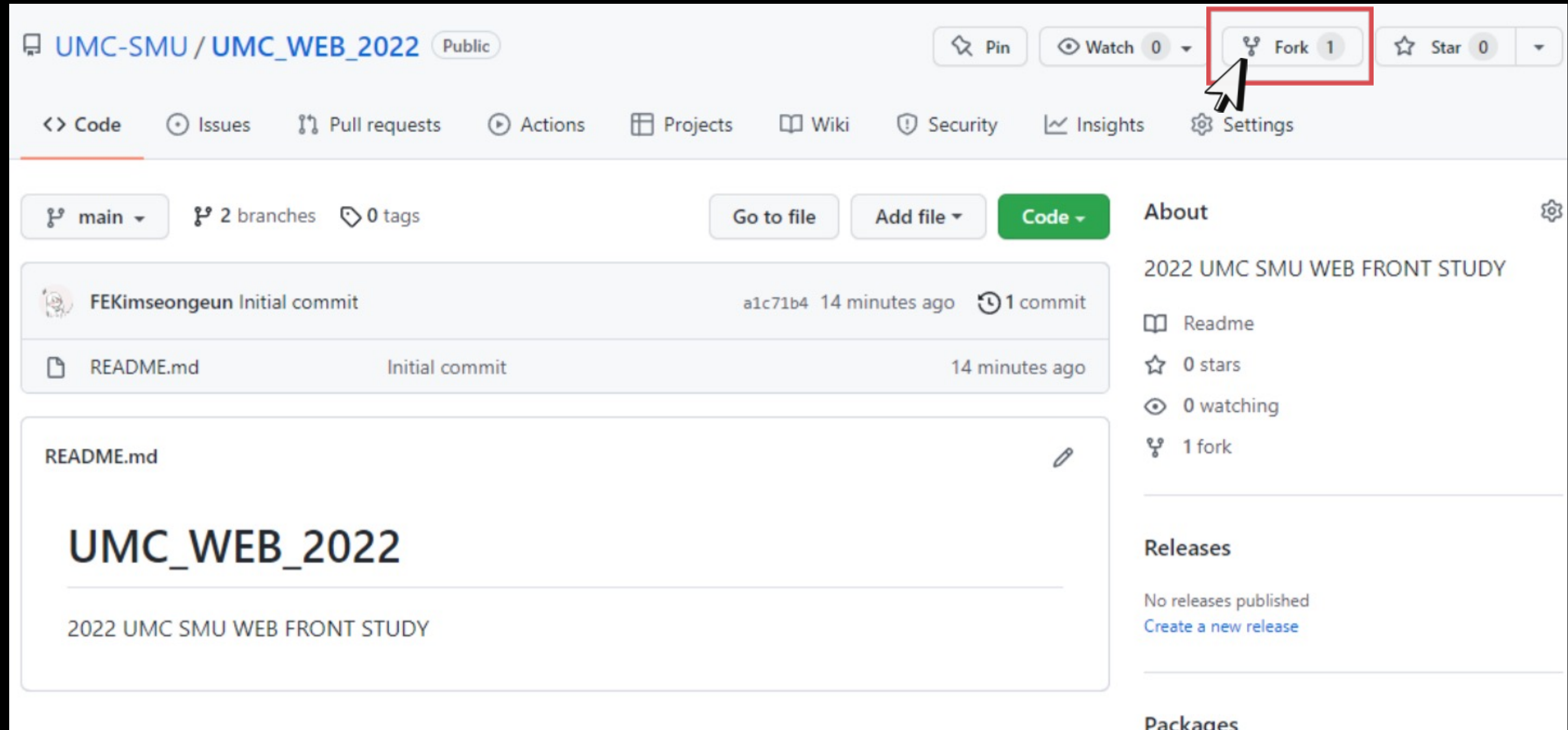
Branch 목록에 본인의 Branch가 없다면
Find or create a branch 란에 **닉네임**을 작성하고 **create branch**

! Branch명은 본인의 닉네임으로 설정 (예시 - minnie)



2. 본인의 Repository로 Fork

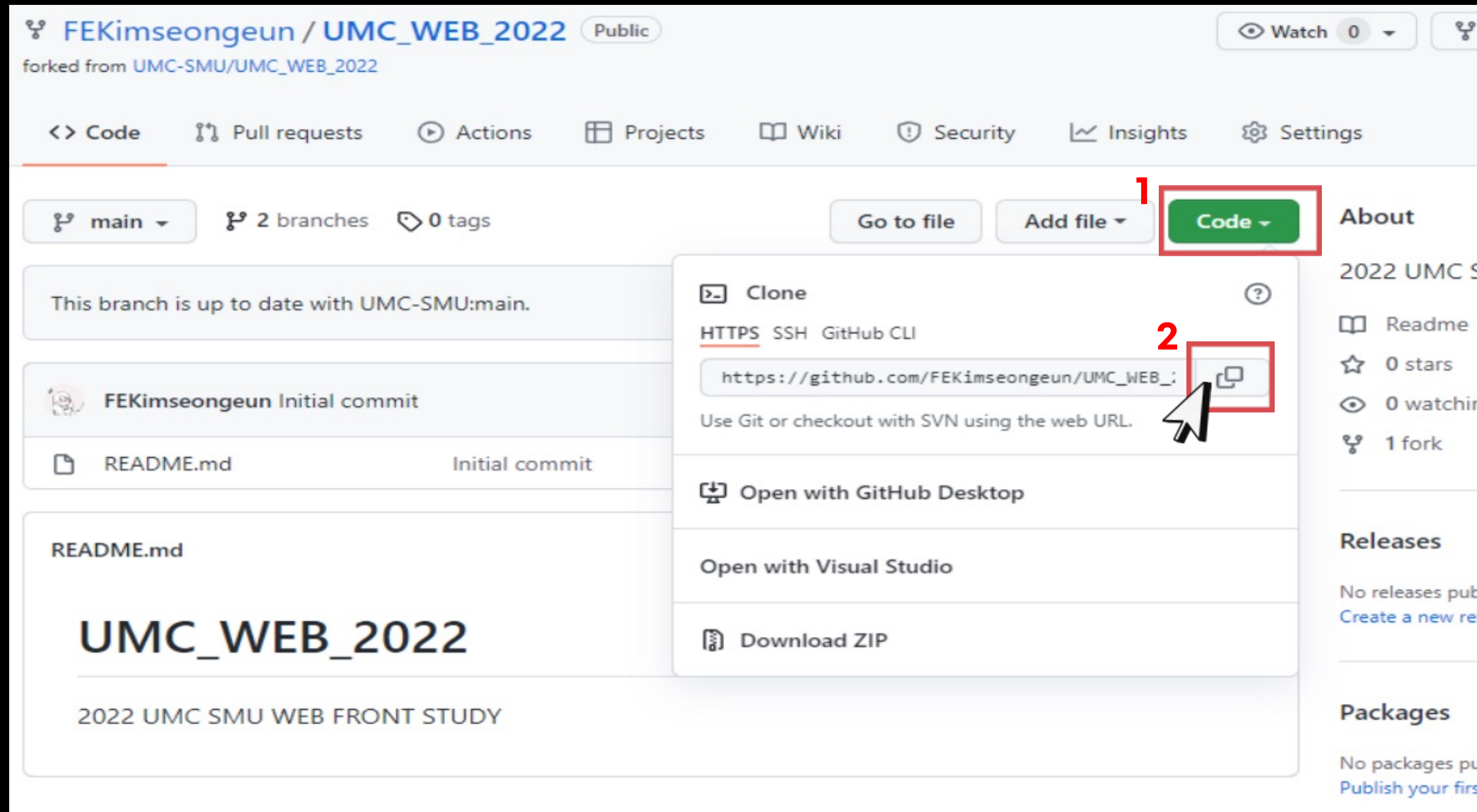
Fork를 눌러 해당 repository를 **자신의 repository**로 복제



3. Git clone

본인의 레포지토리 목록으로 이동 후, UMC repository 클릭

Code > 주소 복사 버튼 클릭



3. Git clone

cmd 를 열고, 파일을 clone 받을 **폴더**로 이동

`cd` [이동할 경로]

이동 후, 다음과 같이 명령어 실행

`git clone` [복사한 주소]

```
C:\Users\BLUE\code>git clone https://github.com/FEKimseongeun/UMC_WEB_2022.git
Cloning into 'UMC_WEB_2022' ...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.

C:\Users\BLUE\code>
```

! clone 주소에 **본인 레포**인지 반드시 확인

1. https://github.com/UMC-SMU/UMC_WEB_2022.git
2. https://github.com/FEKimseongeun/UMC_WEB_2022.git

1번이 아닌 2번임을 확인해주세요.

3. Git clone

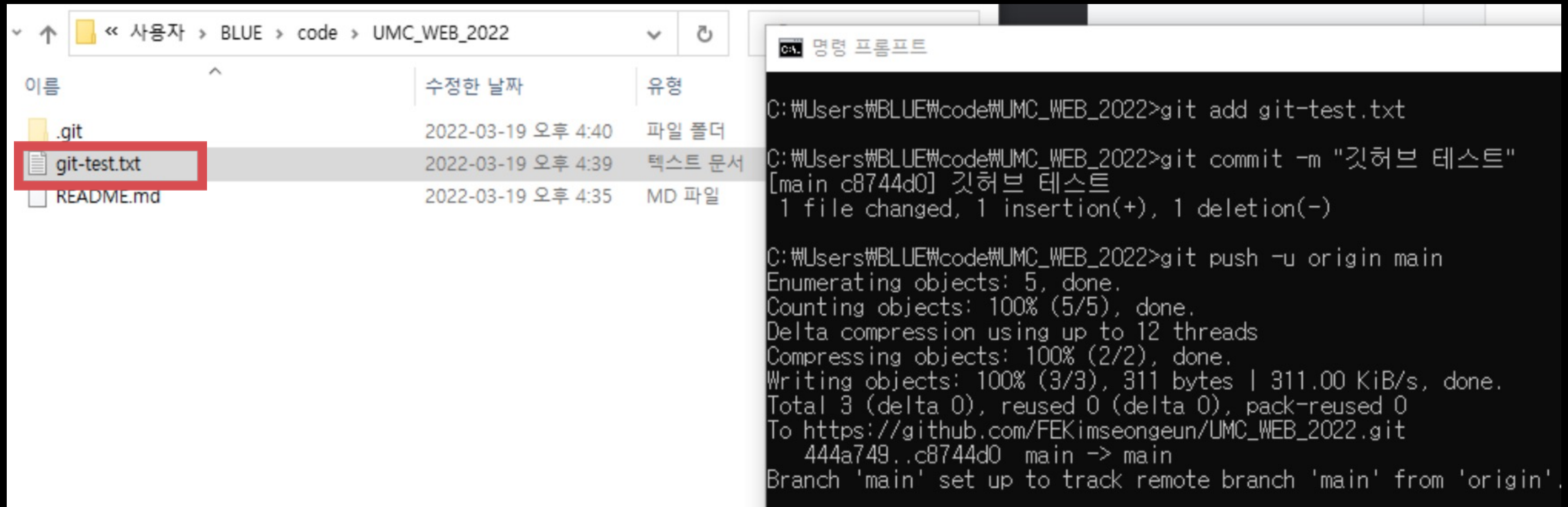
해당 폴더 안에 repository 안의 파일들이 모두 있는지 확인

The image shows two side-by-side screenshots. The left screenshot is a GitHub repository page for 'FEKimseongeun / UMC_WEB_2022'. It shows the 'main' branch is up to date with 'UMC-SMU:main'. The commit history shows an 'Initial commit' by 'FEKimseongeun' with a file named 'README.md' highlighted by a red box. The right screenshot is a Windows File Explorer window showing the contents of the 'UMC_WEB_2022' folder. The address bar shows the path: << 사용자 > BLUE > code > UMC_WEB_2022. The file list shows a folder named '.git' and a file named 'README.md', both with a modification date of '2022-03-19 오후 4:28'. The 'README.md' file is highlighted by a red box.

이름	수정한 날짜	유형
.git	2022-03-19 오후 4:28	파일 폴더
README.md	2022-03-19 오후 4:28	MD 파일

4. git add/ commit / push

과제를 폴더에 저장한 후 해당 경로에서 **add, commit, push** 명령 실행



The image shows a Windows file explorer window on the left and a command prompt window on the right. The file explorer is displaying the contents of the folder 'C:\Users\BLUE\code\UMC_WEB_2022'. It contains a '.git' folder, a 'git-test.txt' file (highlighted with a red box), and a 'README.md' file. The command prompt window shows the following commands and their output:

```
C:\Users\BLUE\code\UMC_WEB_2022>git add git-test.txt

C:\Users\BLUE\code\UMC_WEB_2022>git commit -m "깃허브 테스트"
[main c8744d0] 깃허브 테스트
1 file changed, 1 insertion(+), 1 deletion(-)

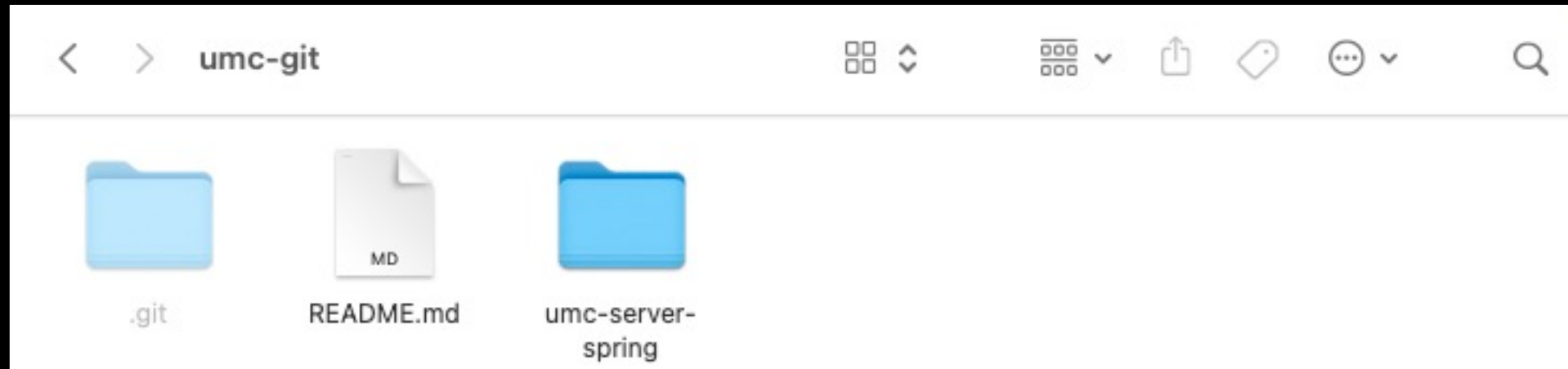
C:\Users\BLUE\code\UMC_WEB_2022>git push -u origin main
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 311 bytes | 311.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/FEKimseongeun/UMC_WEB_2022.git
 444a749..c8744d0  main -> main
Branch 'main' set up to track remote branch 'main' from 'origin'.
```

💡 프로젝트 파일을 커밋할 시 다음 페이지를 참고해주세요.

4. git add/ commit / push

💡 이전 페이지와 동일한 내용이며, 이 페이지는 폴더 파일 전체를 커밋하는 경우입니다.

.git이 있는 경로에 과제 폴더를 저장한 후 add, commit, push 명령 실행



```
jimin@imjimin-ui-MacBookAir ~/Desktop/4-1/umc main git add .
jimin@imjimin-ui-MacBookAir ~/Desktop/4-1/umc main + git commit -m "test"
[main 7556d54] test
10 files changed, 415 insertions(+)
create mode 100644 demo 2/.gitignore
create mode 100644 demo 2/build.gradle
create mode 100644 demo 2/gradle/wrapper/gradle-wrapper.jar
create mode 100644 demo 2/gradle/wrapper/gradle-wrapper.properties
create mode 100755 demo 2/gradlew
create mode 100644 demo 2/gradlew.bat
create mode 100644 demo 2/settings.gradle
create mode 100644 demo 2/src/main/java/com/example/demo/UmcGitApplication.java
create mode 100644 demo 2/src/main/resources/application.properties
create mode 100644 demo 2/src/test/java/com/example/demo/UmcGitApplicationTests.java
jimin@imjimin-ui-MacBookAir ~/Desktop/4-1/umc main git push origin main
```

! add 시 “.” 도 포함입니다.

4. git add/ commit / push

- git add

git add [파일명] : 커밋할 파일명 지정

git add . : 전체 파일 선택

- git commit -m “커밋 메시지”

“커밋 메시지” 는 의미없는 내용보다는 작업한 내용 중심으로 작성하는 것을 권장

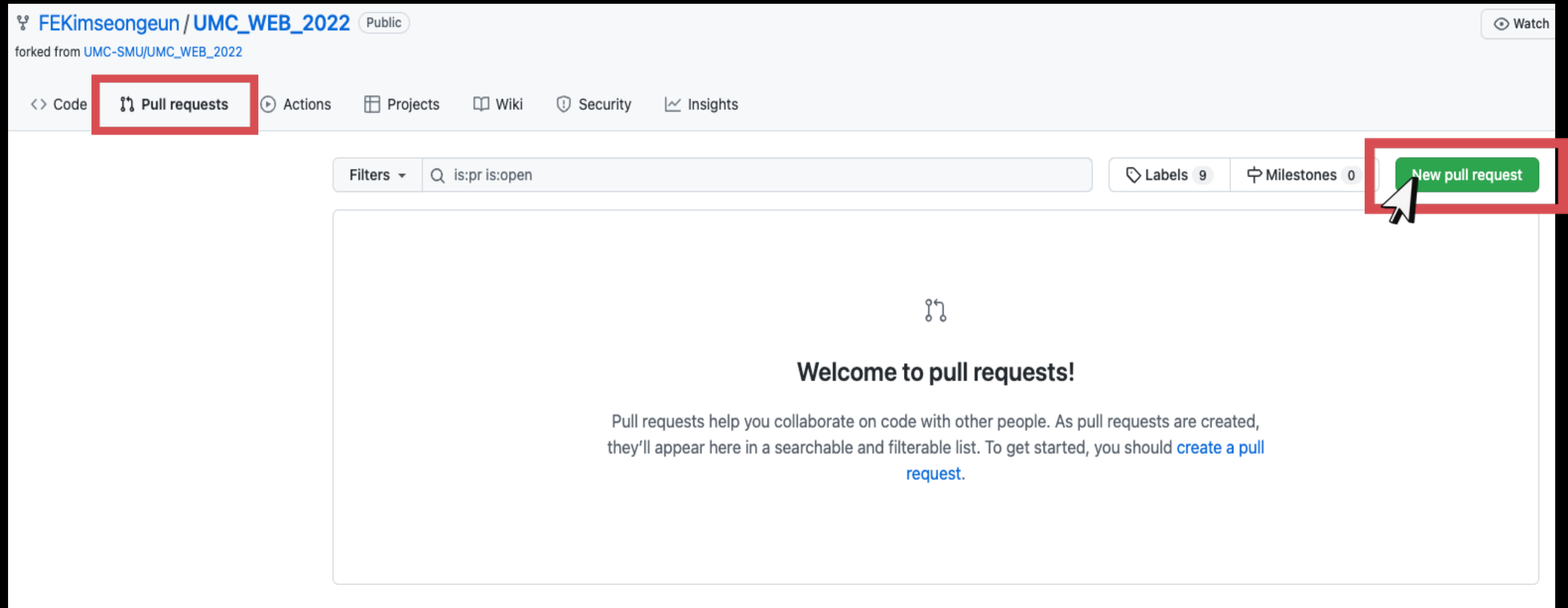
참고 자료: <https://richone.tistory.com/26>

- git push -u origin main

커밋한 내용을 원격 저장소의 main 브랜치에 push

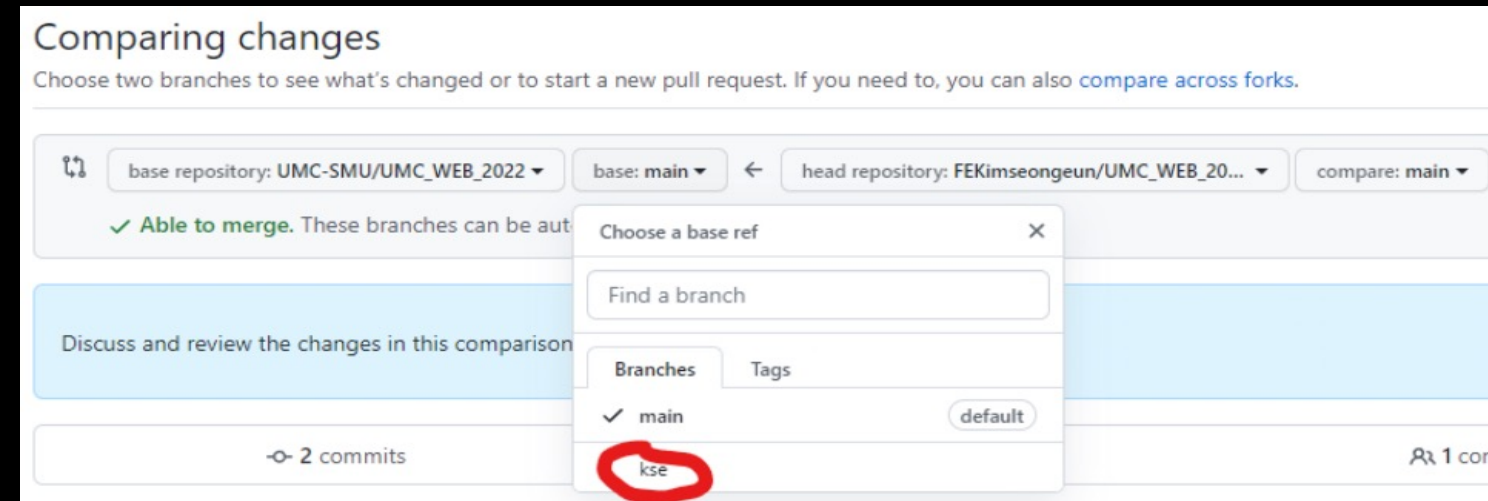
5. Pull request

자신의 repository > Pull requests > New pull request로 이동

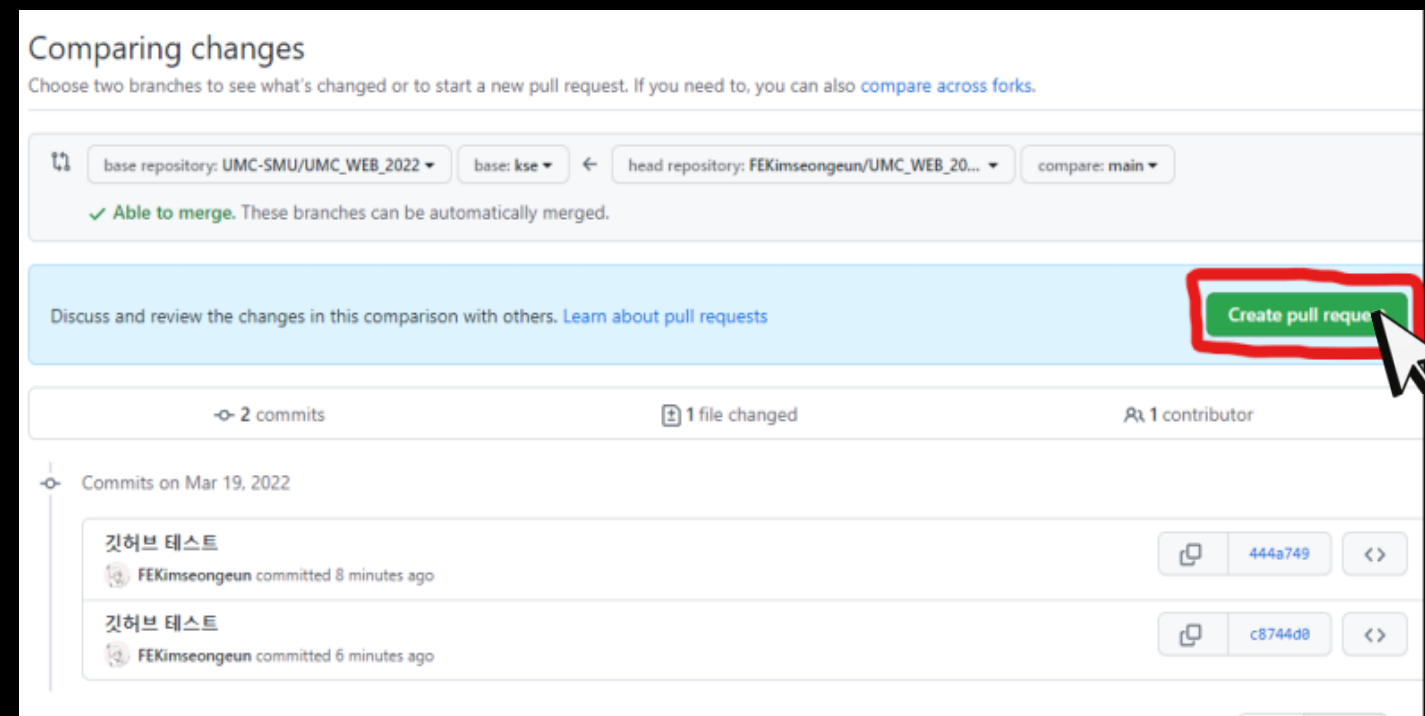


5. Pull request

Base repository의 branch를 자신의 branch로 변경

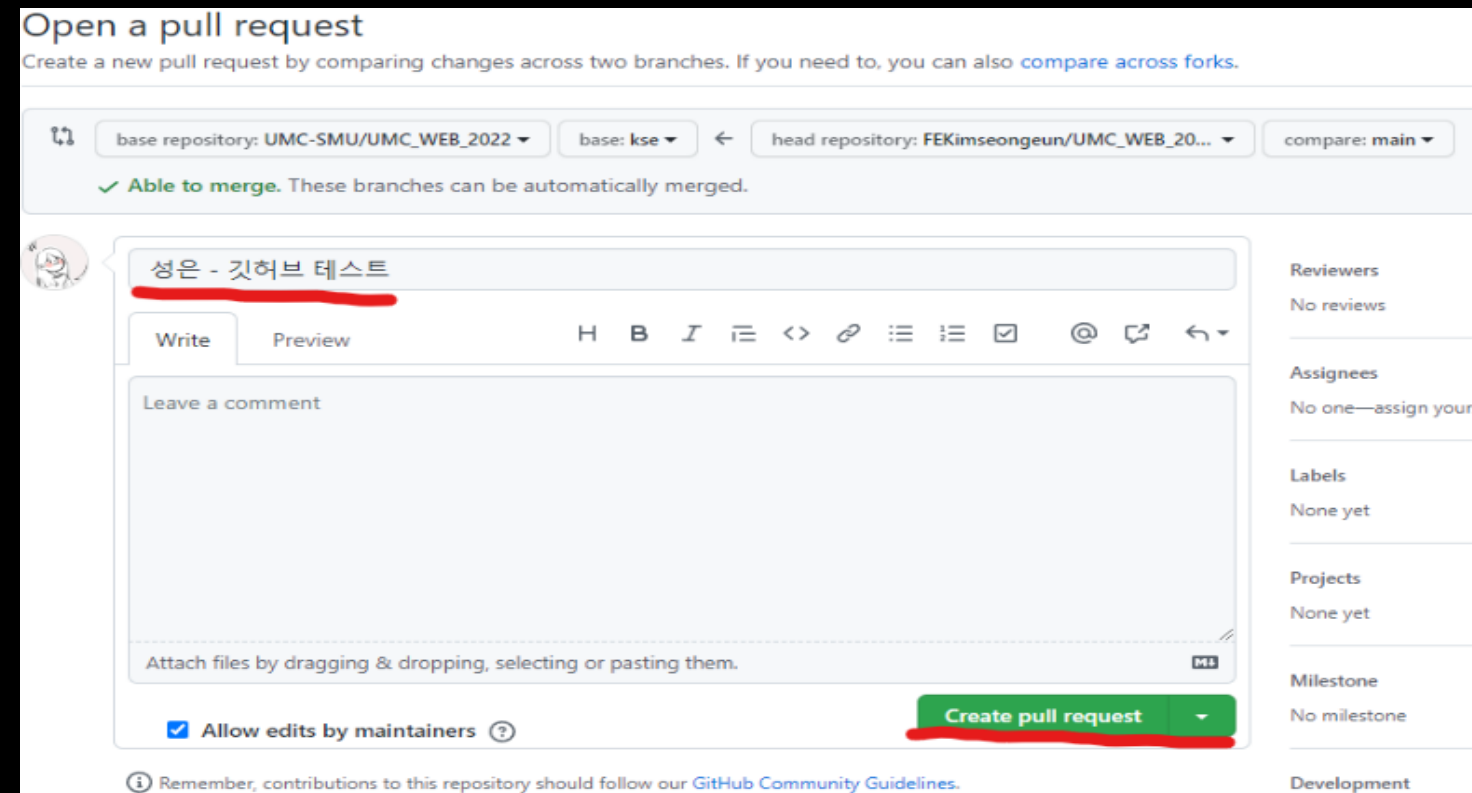


Create pull request 클릭



5. Pull request

메세지 작성 후 **Create pull request** 클릭



Title: [n주차 과제] 본인 닉네임

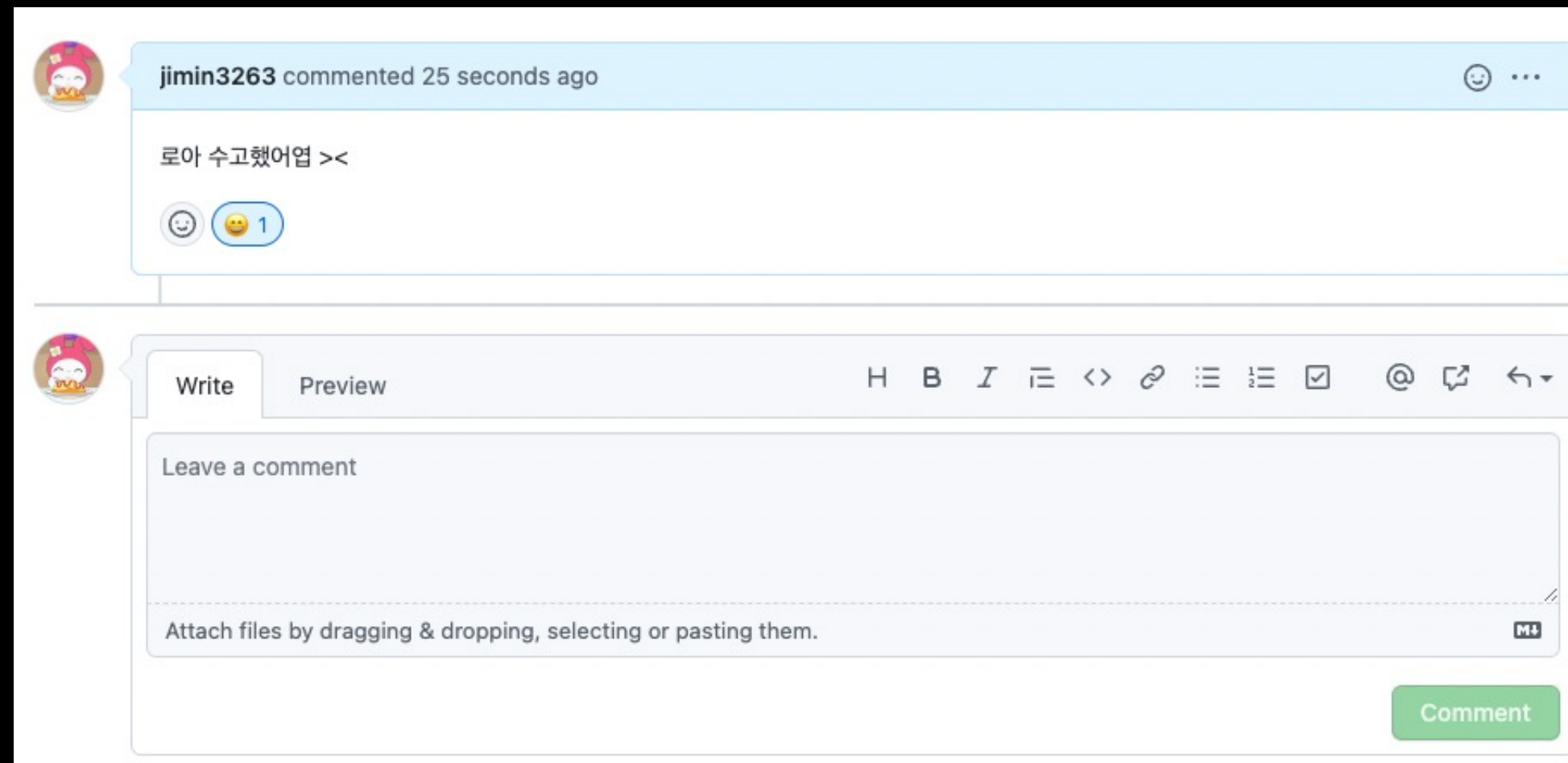
예시 - [1주차 과제] minnie

Comment: 구현 내용, 구현하지 못한 내용 등 자유롭게 작성

5. Pull request

다른 스터디원들의 PR 내역 확인

💡 Comment, 이모지 추가도 가능합니다.

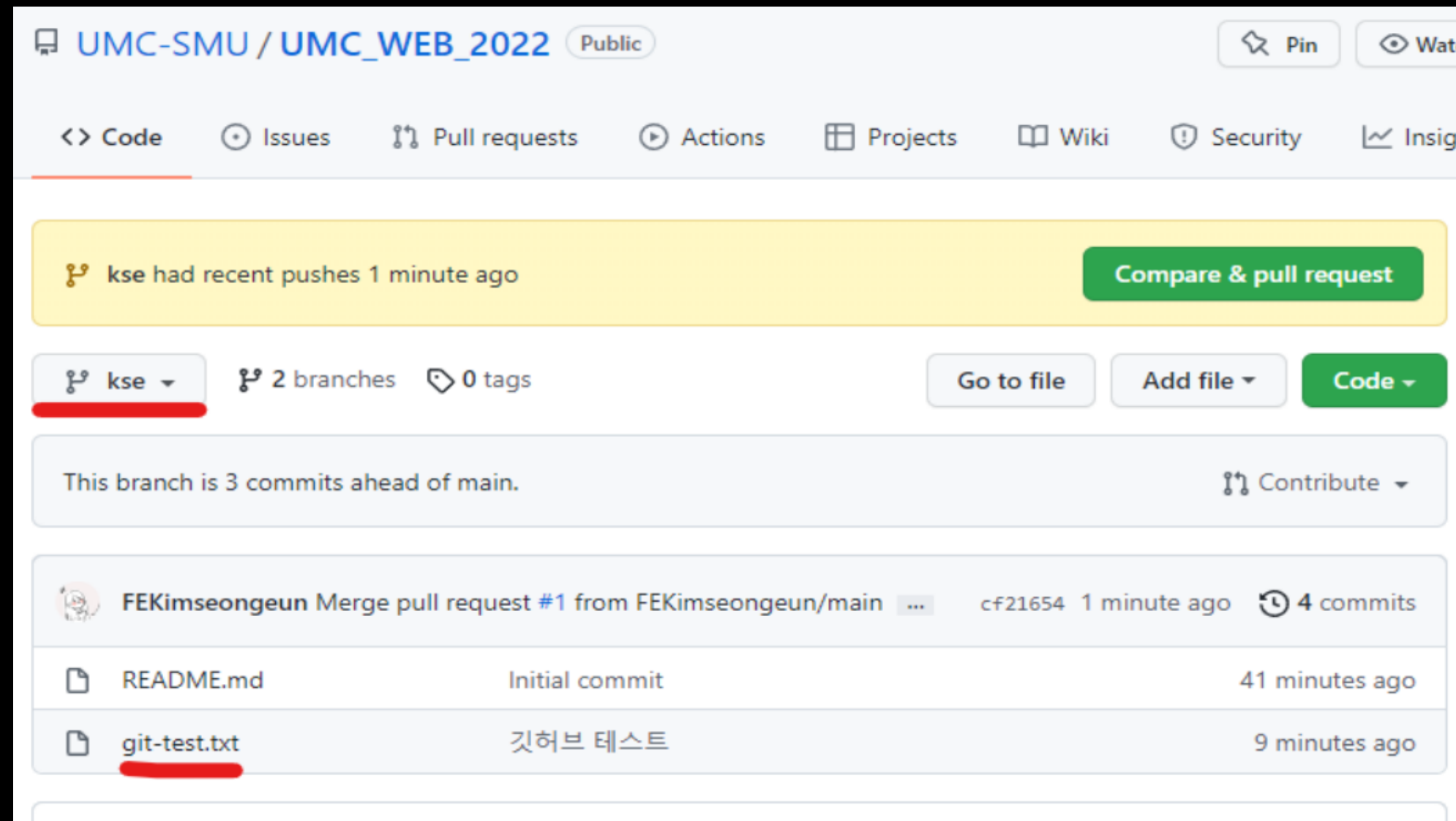


6. Merge

merge 권한을 가진 **스터디 리더** 들은 과제 확인 후 **merge**

! 각자의 브랜치가 맞는지 확인 후 merge 부탁드립니다.

Fork한 repository와 **일치**하는지 확인





감사합니다.

Git 관련 질문이 있다면 **로아 or 민니**에게 문의 부탁드립니다.

! git 관련 질문 시 오류 캡처, 실행 명령 등 구체적으로 상황을 설명해주면 더 좋습니다.