

**GIT의 가장 큰 목적.
협업!**

Github의 가장 큰 장점!

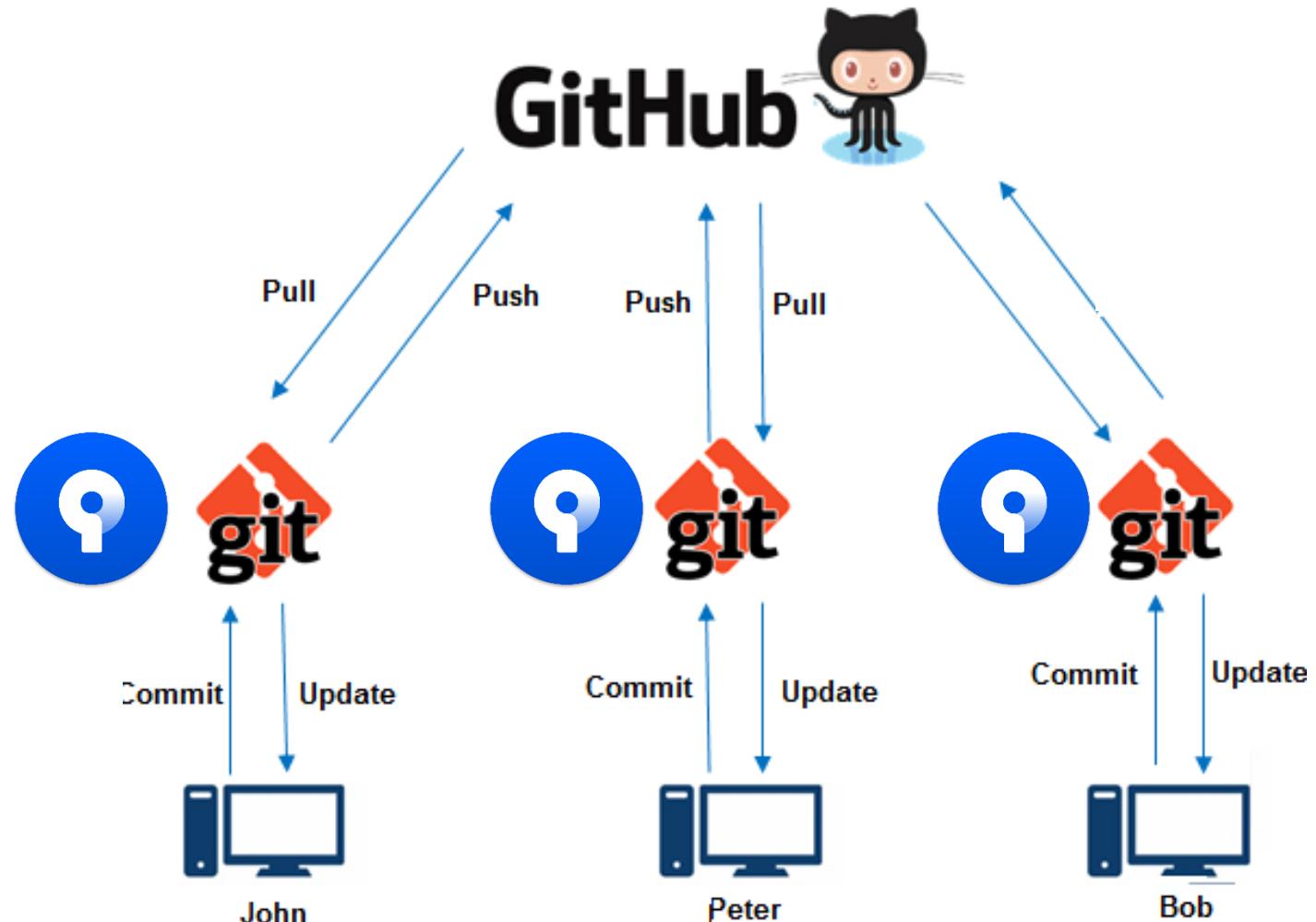
- 하나의 프로젝트를 여럿이서 관리할 수 있음!
- 그래, 조별과제는 그럴 수 있어!
- 하지만 회사라면?!



Github의 가장 큰 장점!

- 하나의 프로젝트를 여럿이서 관리할 수 있음!
- 그래, 조별과제는 그럴 수 있어!
- 하지만 회사라면?!
- 여럿이서 한 프로젝트를 진행해야함!

Distributed Version Control System



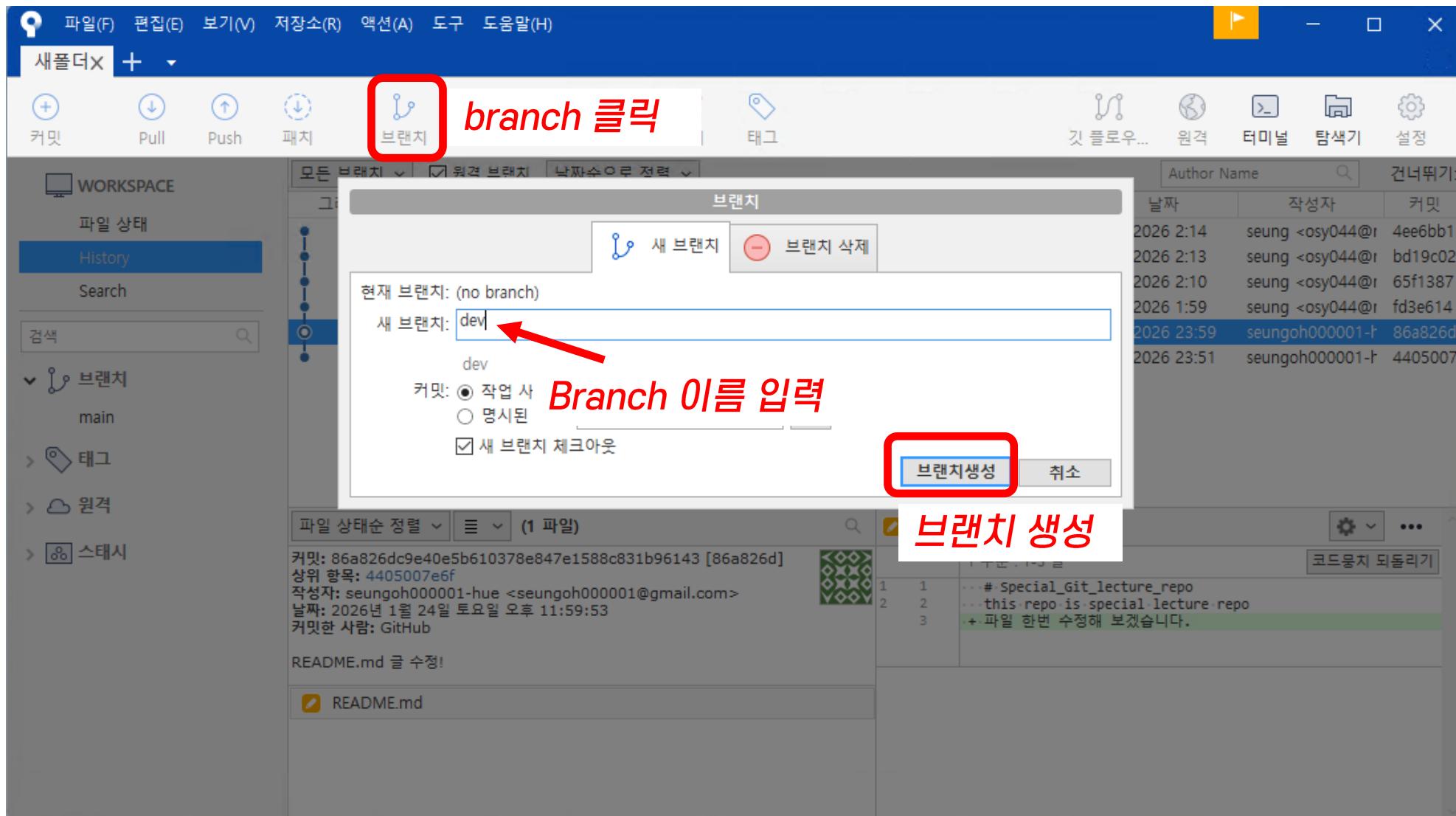
Branch & Merge

Branch

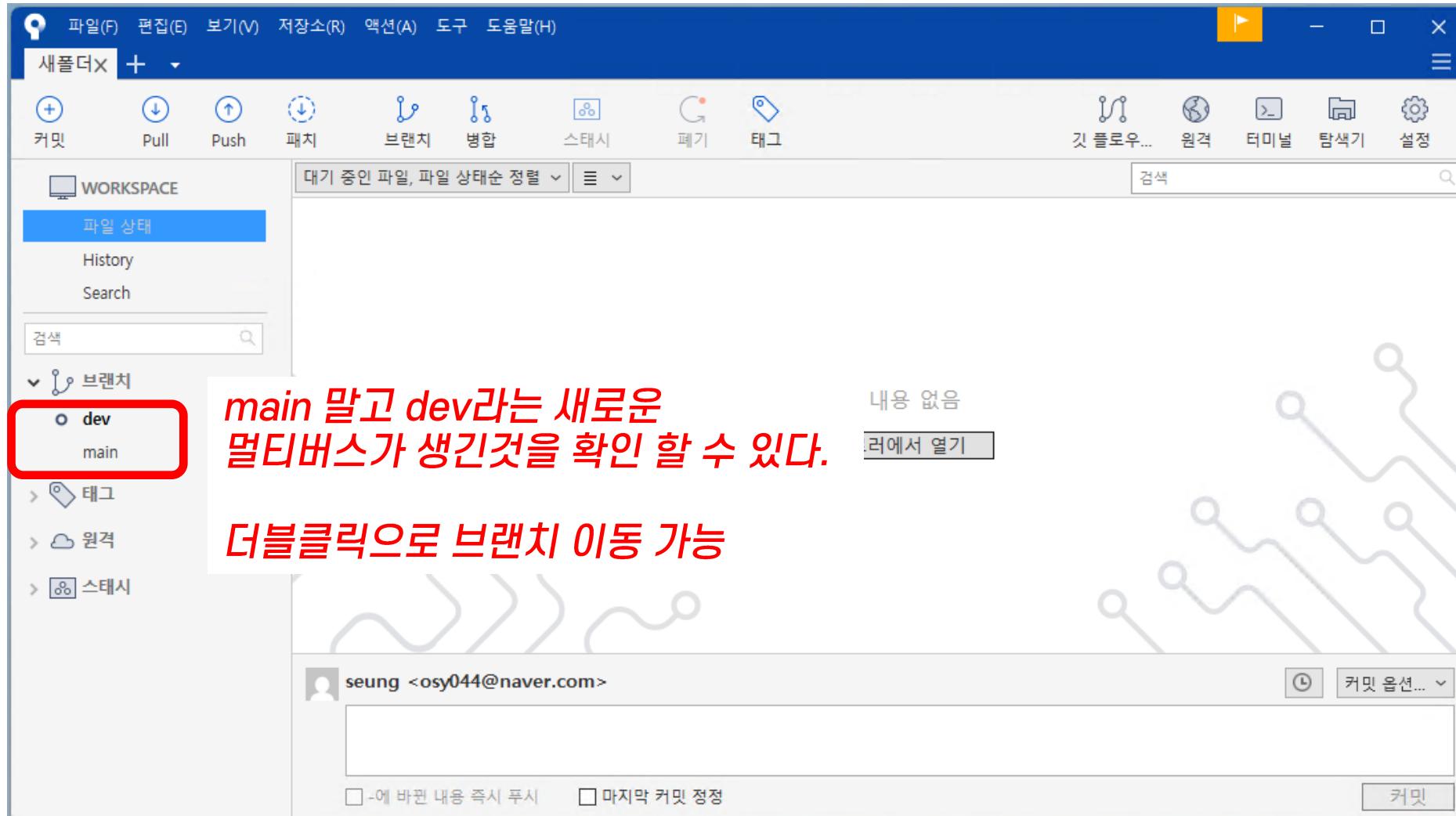
- Branch라고 하는것은 분기점임
- 과거로 돌아가서 새로운 버전을 만들고 이걸 최신 버전과 합쳐서 새로운 버전을 만들 수 있음
- 즉, 하나의 프로젝트 내에서 멀티버스를 만들 수 있음



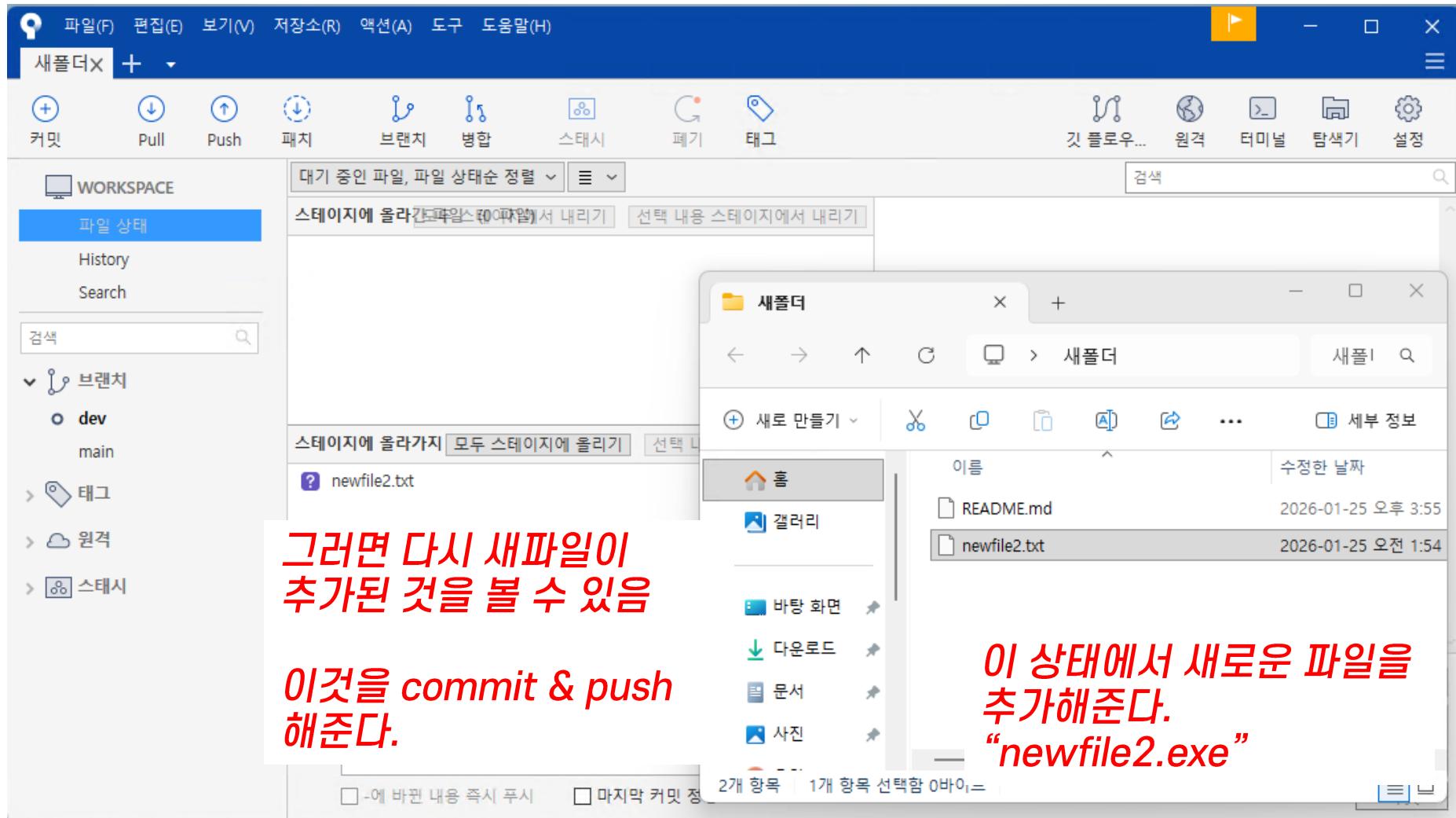
History



Branch



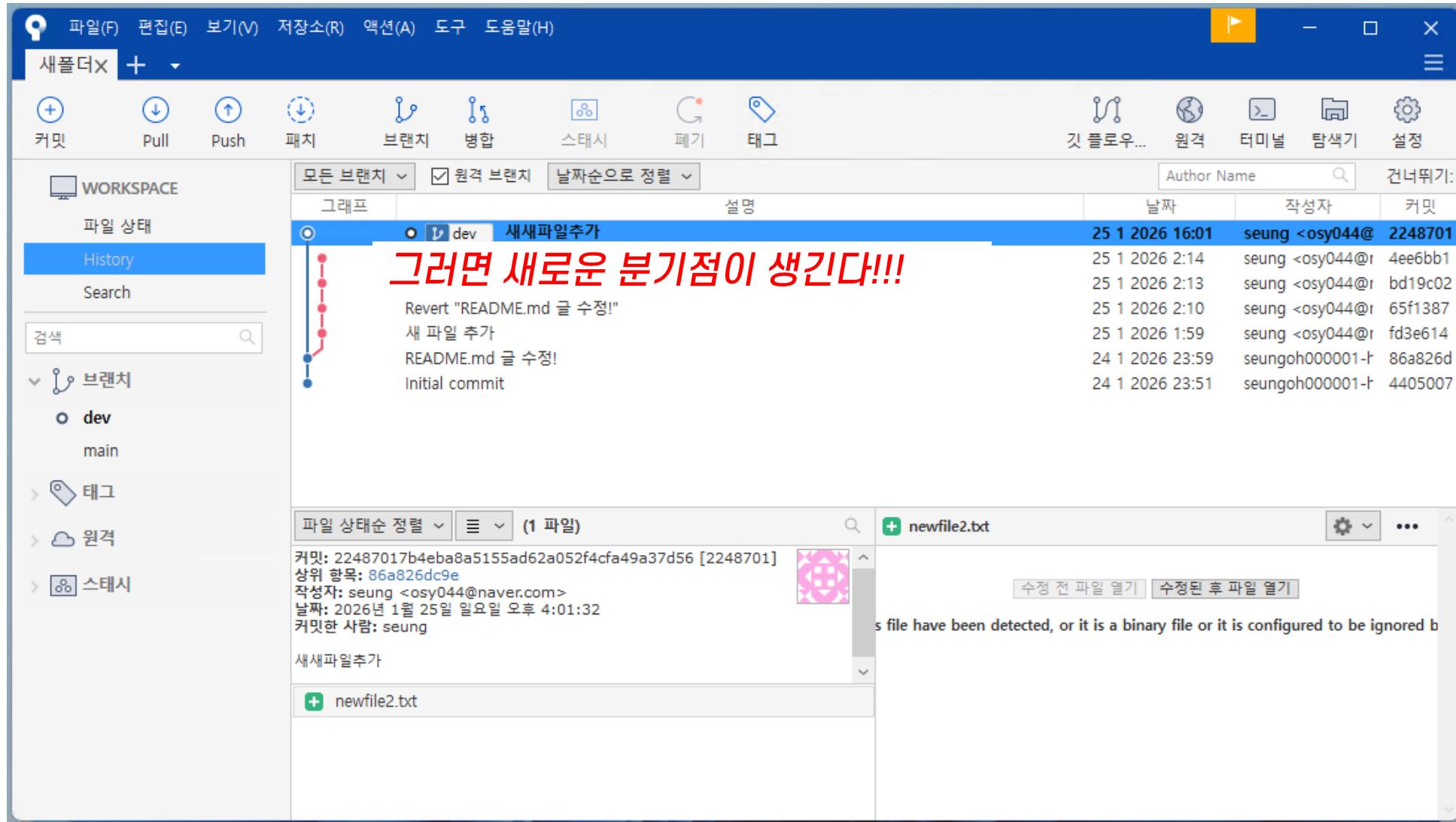
Branch



Branch

- 그리고 commit 해주세요!

Branch



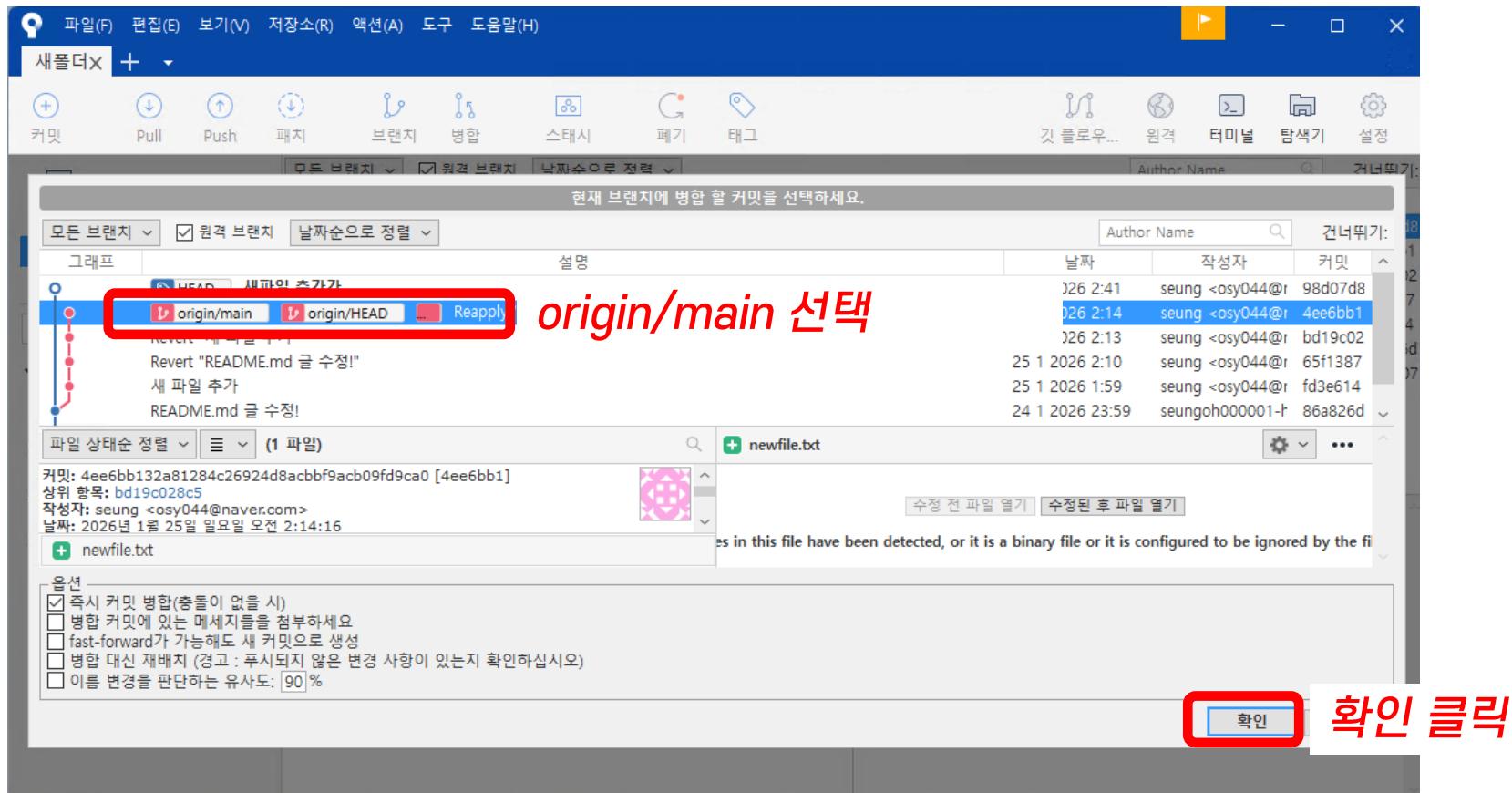
Merge

- 이때 이제 merge(병합)를 하면 두개의 브랜치를 하나로 합칠 수 있음!



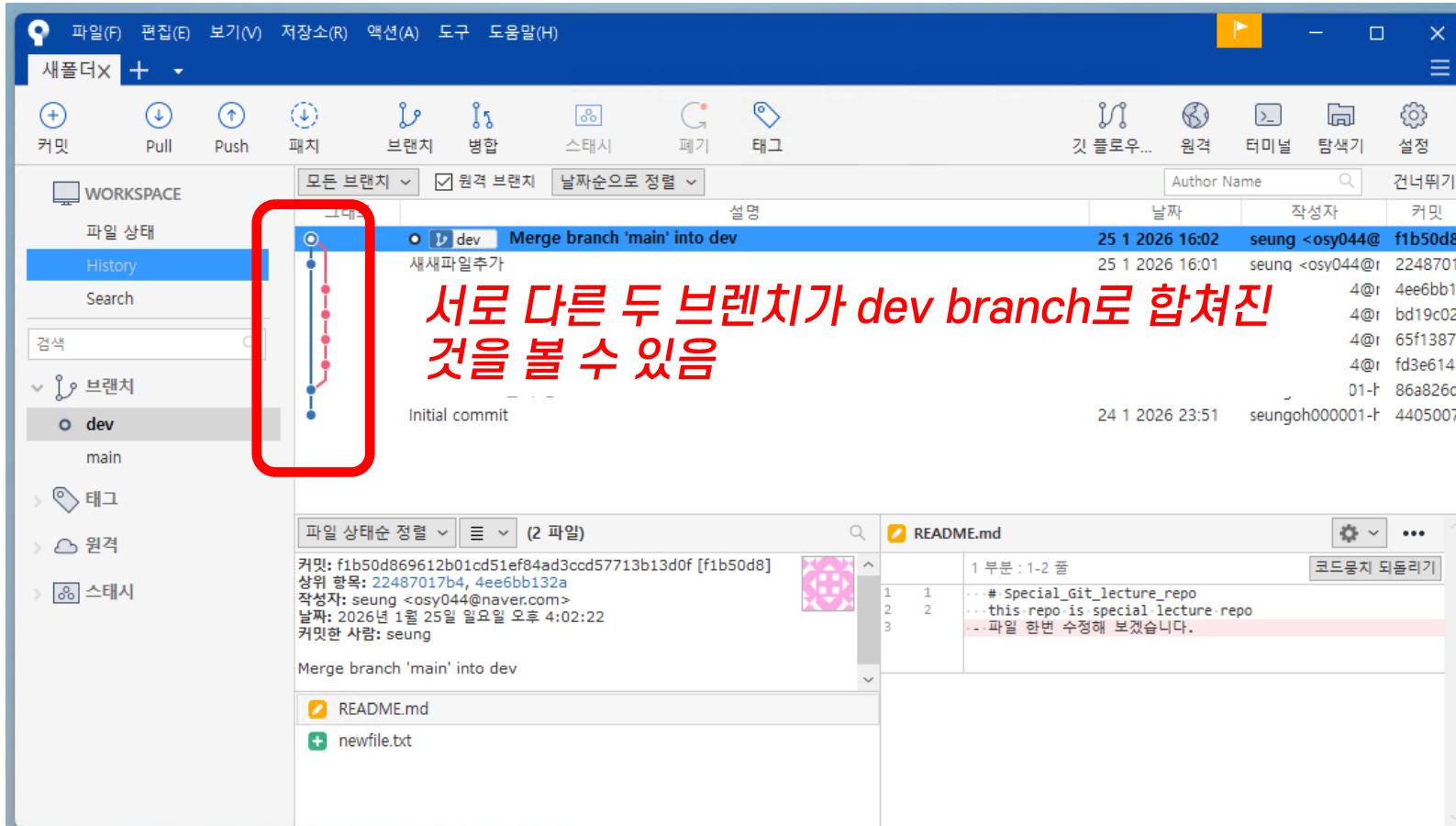
Merge

- 병합(merge)를 누르고 origin/main 과 합쳐보자



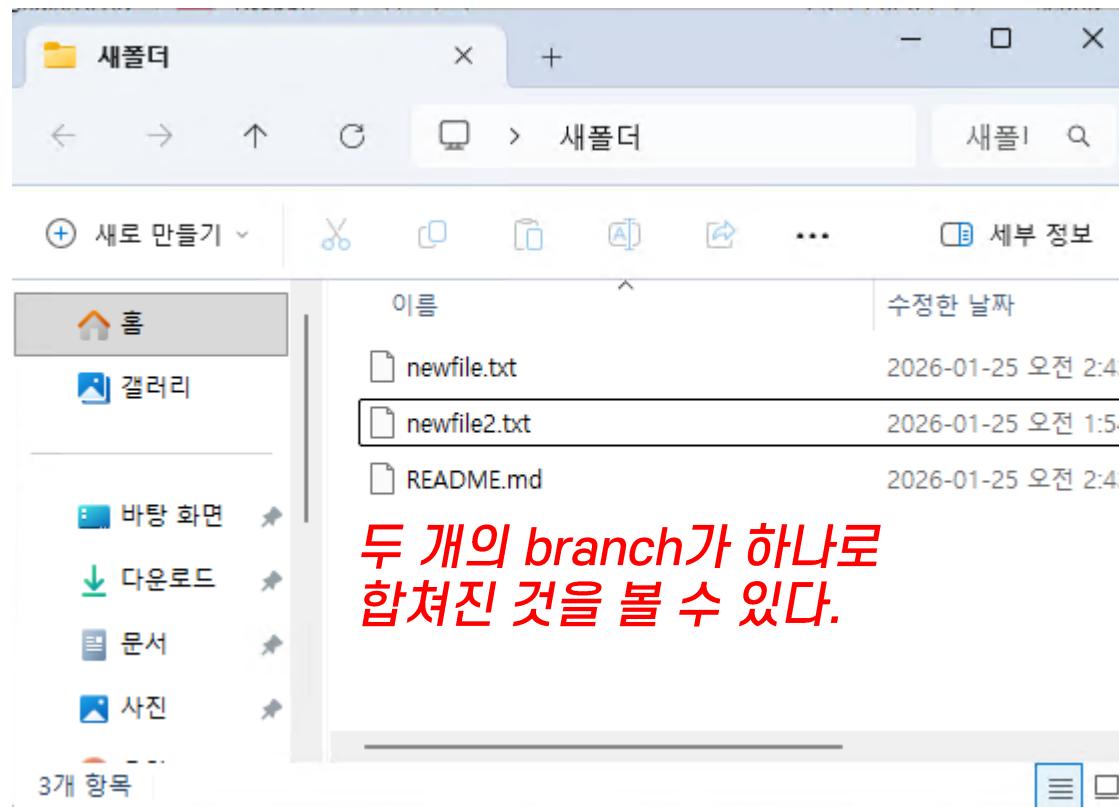
Merge

- 병합(merge)를 누르고 origin/main 과 합쳐보자



Merge

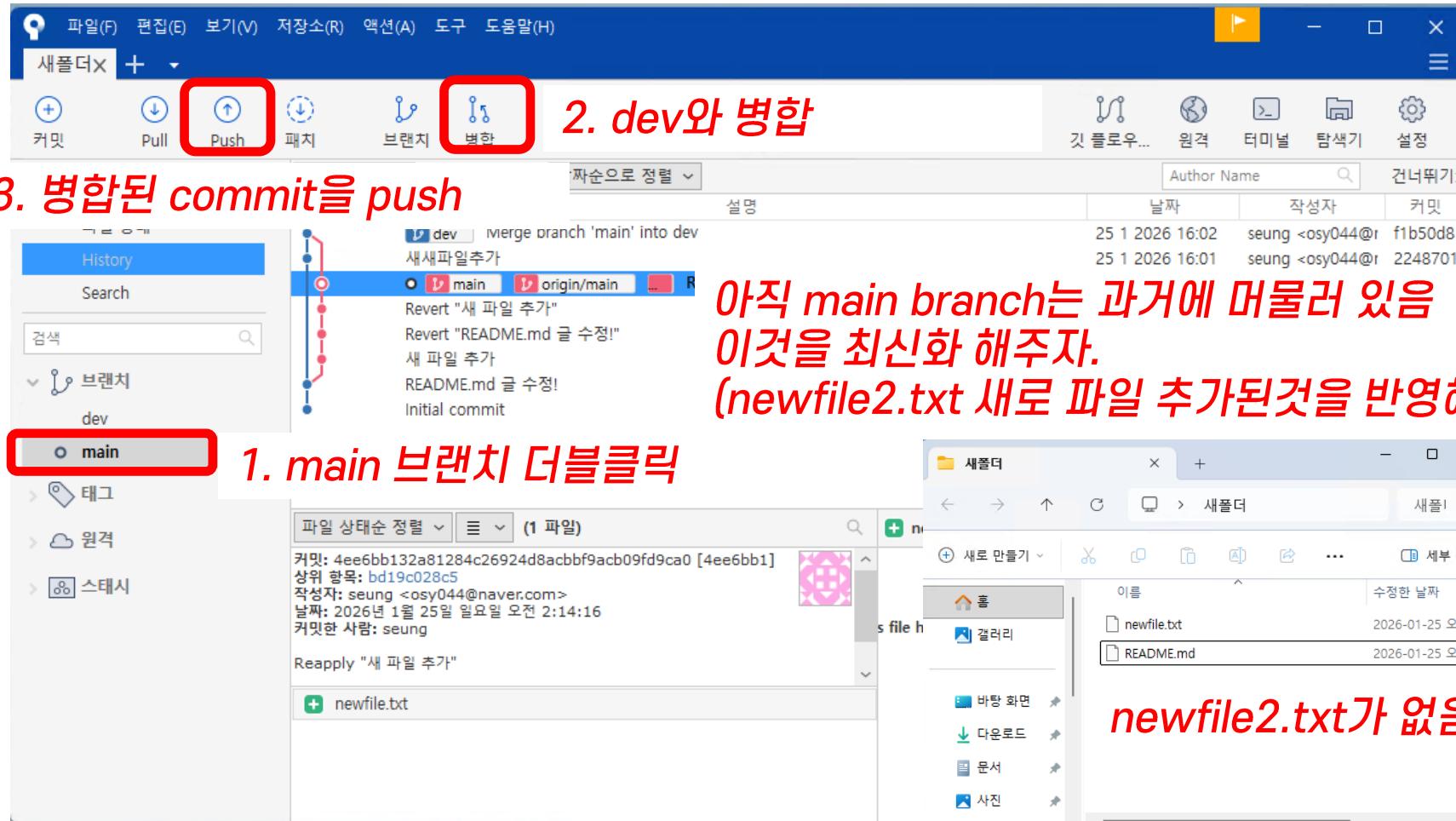
- 그럼 과거의 newfile1.txt와 newfile2.txt 모두 있는게 보임
- 이 둘다 dev 브랜치로 합쳐졌음.



로키가 멀티버스의 시간선을 merge 하고 있다.

반대로 main에서도 merge 해줘야함

- 방금전에 main을 dev로 merge 했는데, main에서는 dev가 merge가 안됐음



Merge

- 즉, 여럿이서 작업할때, branch를 여러 개 만들어준다.
 - 그리고 새로운 브랜치에서 팀원들이 작업하게 하고,
 - 팀장이 해당 브랜치 관리 및 merge 관리를 함.
-
- 즉, 팀장: branch 관리, merge 관리
 - 팀원: commit, pull request (제 변경사항 적용시켜주세요!)

Merge

- 주의사항
- github의 수많은 문제점은 merge에서 나온다.
- github의 아킬레스건…
- 만약 여러 브렌치에서 같은 파일을 수정했다면, merge가 굉장히 어려워짐
- 이 경우, “충돌”난다고 함. 즉, merge가 안됨
- 이 때는, 직접 충돌난 파일을 수정해서 다시 commit 해야함. (야근이 시작됨…)
- 따라서 처음에 계획을 잘 세워야함. 서로 같은 파일 수정하지 않도록

함께 하는 Github pull request

다른 사람의 Repo를 수정해보자!

- 보통 팀장이 Repo 관리를 함
- 나머지 직원은 이 Repo를 clone해서 수정사항을 업데이트 해야함
- 팀장은 먼저, 새로운 브랜치를 만든다. (팀원들 수정사항 관리용 브랜치)
- 다른 사람의 repo를 수정하는 방법은 2가지가 있음
- 첫번째: collaborator (공동 소유)
 - Repo 설정에서 collaborator 초대를 하면 수정권한을 받게 됨
- 두번째: Pull request (수정 요청)
 - 그 사람의 repo를 clone한 다음에 수정하고 pull request를 날림!

Pull Request

- 하.. 팀장 힘들다..



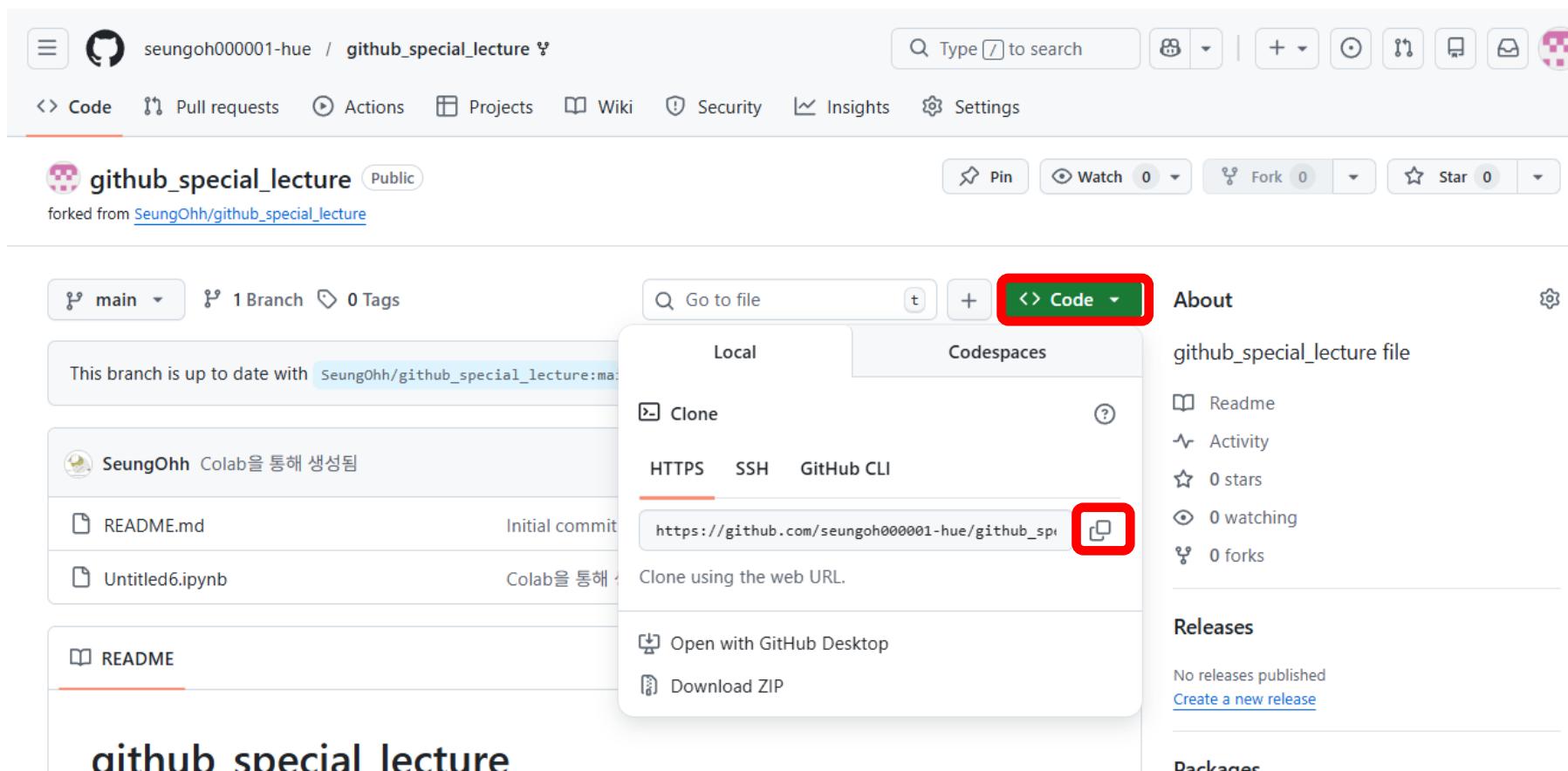
Pull Request

- 모두 팀원과 팀장이 되어보자!
- 이제 옆사람의 git repo에 들어가서, fork를 해보자
- fork를 하면 내 github에 해당 레포가 복제됨.

The screenshot shows a GitHub repository page for 'github_special_lecture'. The top navigation bar includes 'Code', 'Issues', 'Pull requests', 'Actions', 'Projects', 'Security', and 'Insights'. Below the navigation is the repository name 'github_special_lecture' and its status as 'Public'. A 'Watch' button and a 'Fork' button are visible. The 'Fork' button is highlighted with a red box. A modal window titled 'Existing forks' displays the message 'You don't have any forks of this repository.' with a red box around the '+ Create a new fork' button. To the right of the modal, the text '새 fork 만들기' is written in red. The main repository area shows two commits: 'Initial commit' by 'SeungOhh' and 'Untitled6.ipynb' by 'Colab'. The README file is also visible. On the right side of the page, there are statistics: 0 stars, 0 watching, and 0 forks. The 'Releases' section indicates 'No releases published'.

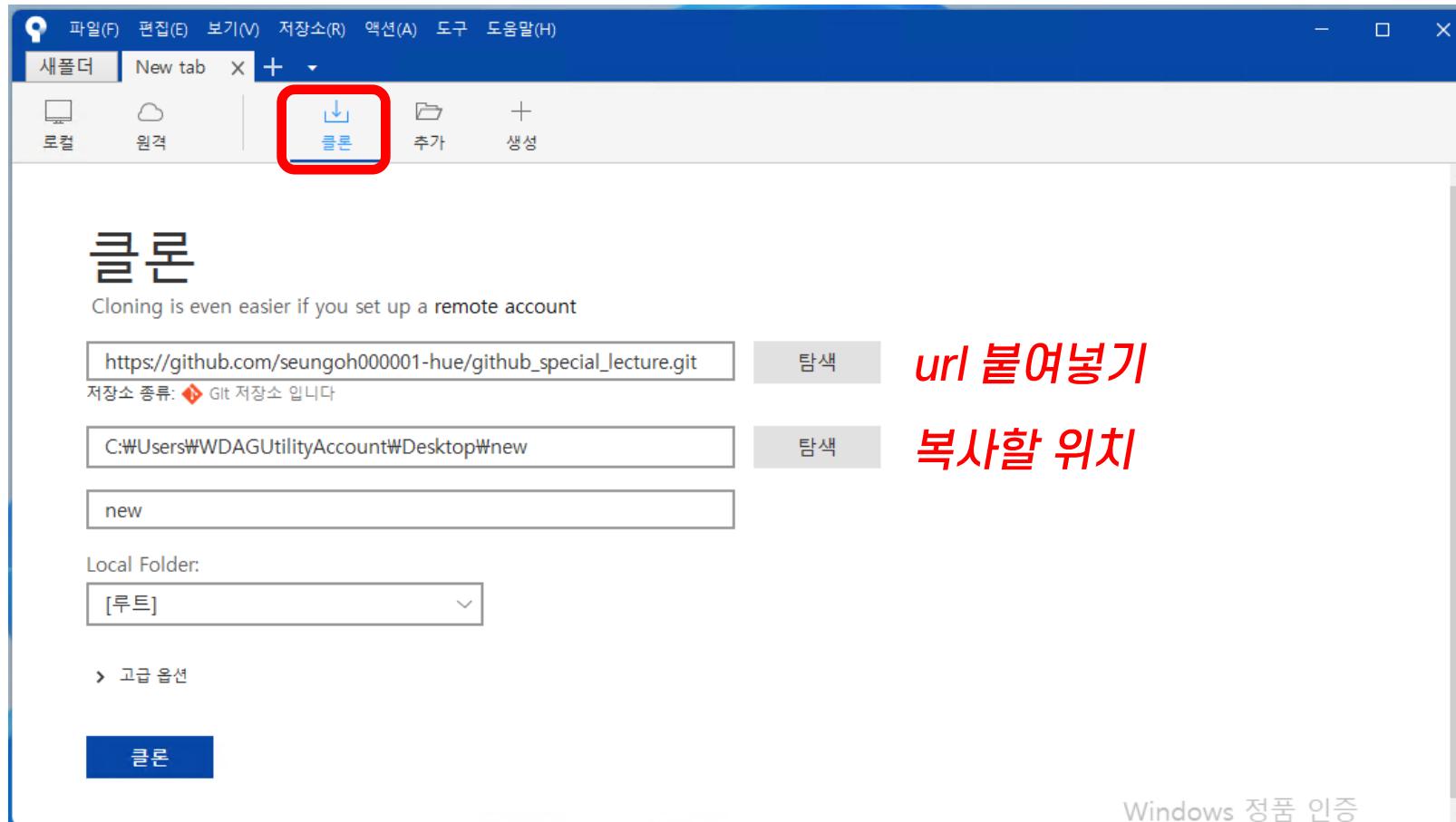
Pull Request

- 그럼 내 github repo에서 해당 url를 복사해서 sourcetree에서 클론 함



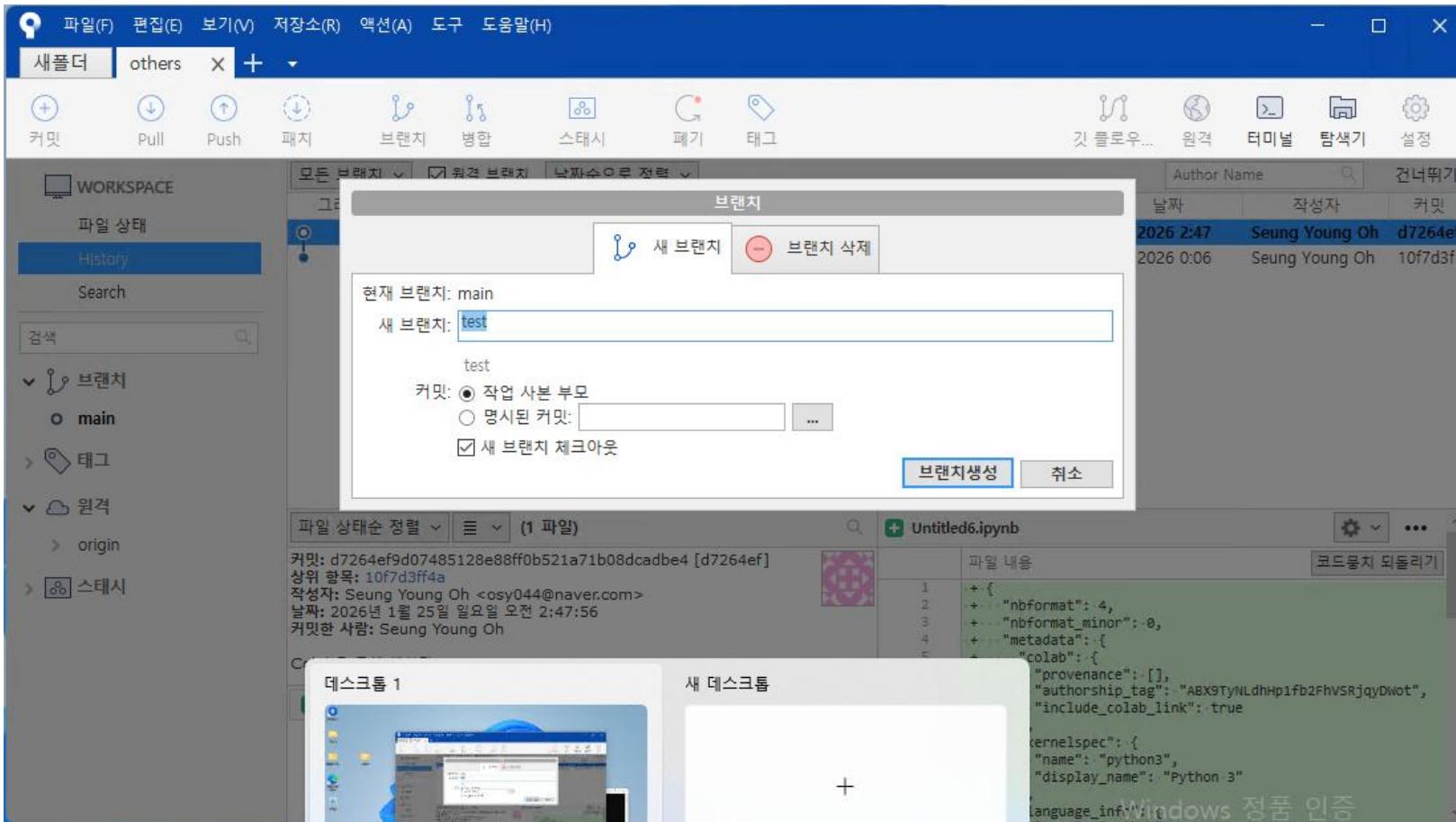
Pull Request

- 모두 팀원과 팀장이 되어보자!
- 이제 옆사람의 git repo를 클론해보자.



Pull Request

- branch 생성 → 새파일 만들기 → commit → push를 해보자



Pull Request

- push까지 하면 pull request가 생김

The screenshot shows a GitHub repository page for 'github_special_lecture'. The repository is public and was forked from 'SeungOhh/github_special_lecture'. A yellow banner at the top indicates that the 'test' branch has recent pushes 10 seconds ago. The main navigation bar includes links for Code, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. Below the banner, there's a summary card showing 2 Branches and 0 Tags, and a message stating the branch is up to date with the original repository's main branch. The repository's activity section lists two commits: an initial commit by 'SeungOhh' and a commit for 'Untitled6.ipynb'. On the right side, there's an 'About' section with details like the file name, Readme, Activity, stars, watching, forks, and releases. The 'About' section also includes a gear icon for settings.

seungoh00001-hue / **github_special_lecture** Public

forked from [SeungOhh/github_special_lecture](#)

test had recent pushes 10 seconds ago

Compare & pull request

main ▾ 2 Branches 0 Tags

Go to file + Code

This branch is up to date with [SeungOhh/github_special_lecture:main](#).

Contribute Sync fork

SeungOhh Colab을 통해 생성됨 d7264ef · 15 hours ago 2 Commits

README.md Initial commit 17 hours ago

Untitled6.ipynb Colab을 통해 생성됨 15 hours ago

About

github_special_lecture file

Readme

Activity

0 stars

0 watching

0 forks

Releases

No releases published

Pull Request

- push까지 하면 pull request가 생김
- 그럼 내용과 함께 pull request를 날릴 수 있음.

The screenshot shows a user interface for creating a pull request. At the top left is a profile icon with a pink and white checkered pattern. Next to it is a title input field labeled "Add a title" containing the text "testcommit". To the right are two small tabs: "He" (highlighted) and "Git". Below the title is a "Add a description" section with a "Write" tab (selected) and a "Preview" tab. A rich text editor toolbar is above the preview area, featuring icons for bold, italic, underline, and other text styles. The preview area contains the text "아 코드 수정했으니가 한 번 확인해주세요!". At the bottom of the description section, there are two buttons: "Markdown is supported" and "Paste, drop, or click to add files". At the very bottom is a green "Create pull request" button with a dropdown arrow to its right. To the left of the "Create" button is a checkbox labeled "Allow edits by maintainers" with a question mark icon.

Pull Request

- 그럼 팀장은 본인 repo에 pull request가 생긴것을 볼 수 있음

The screenshot shows a GitHub repository interface for 'SeungOhh / github_special_lecture'. The top navigation bar includes links for Code, Issues, Pull requests (with 1 item), Actions, Projects, Wiki, Security, Insights, and Settings. A search bar is also present.

A modal window titled 'Label issues and pull requests for new contributors' is displayed, stating: 'Now, GitHub will help potential first-time contributors [discover issues](#) labeled with [good first issue](#)'. There is a 'Dismiss' button in the top right corner of the modal.

The main content area shows a list of pull requests. A filter bar at the top allows searching by 'is:pr is:open' and includes buttons for Labels (9), Milestones (0), and a 'New pull request' button. Below the filters, there are two filter options: '1 Open' and 'testcommit'. The 'testcommit' filter is selected, showing one open pull request:

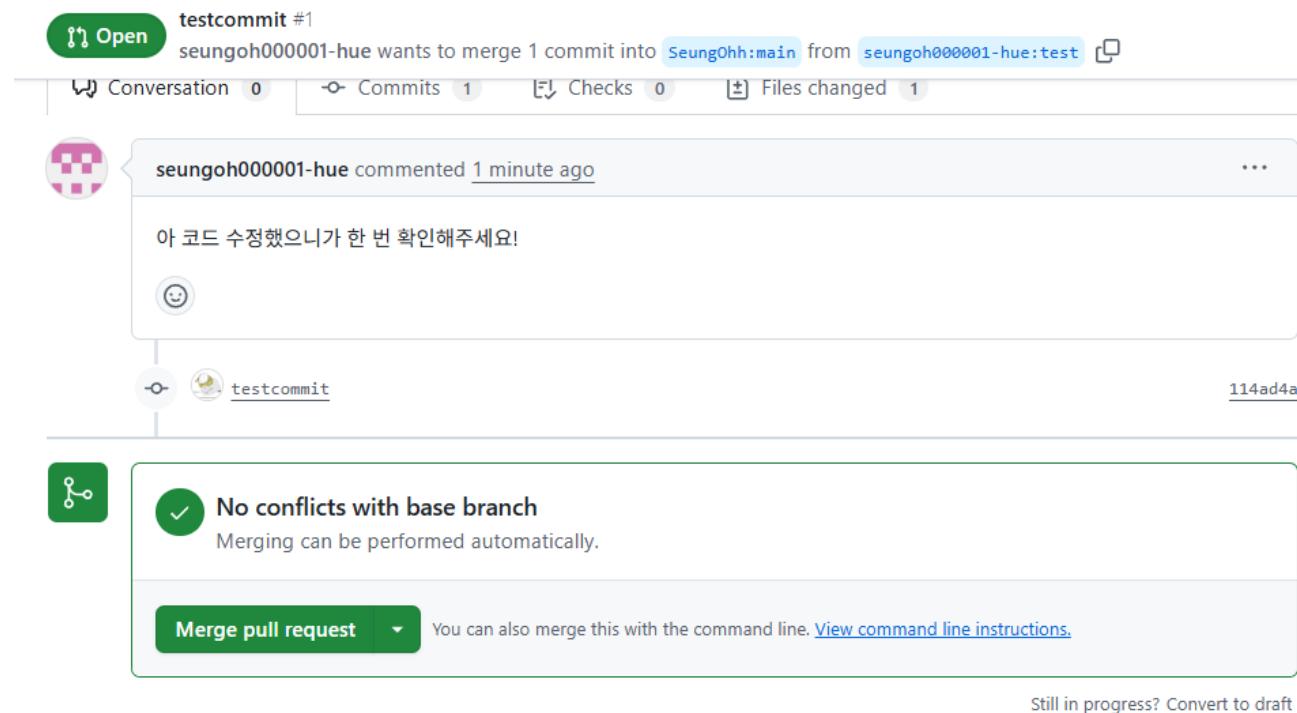
- #1 opened 1 minute ago by seungoh000001-hue

A 'ProTip!' message at the bottom suggests: 'Mix and match filters to narrow down what you're looking for.'



Pull Request

- 그럼 팀장은 본인 repo에 pull request가 생긴것을 볼 수 있음
- 마지막으로 팀장이 merge를 하면 끝!!



미션3

옆사람과 pull request 주고 받기