

**Github**

# Git/GitHub 고급 (1)

- **협업을 위한 Git/GitHub 기능과 실습**
  - **협업자 등록 및 협업 환경 설정**
  - **Branch (분기)**
  - **Merge (통합)**
  - **Resolve Conflict (충돌 해결)**

강사 양석환



**협업을 위한 Git/GitHub 기능**

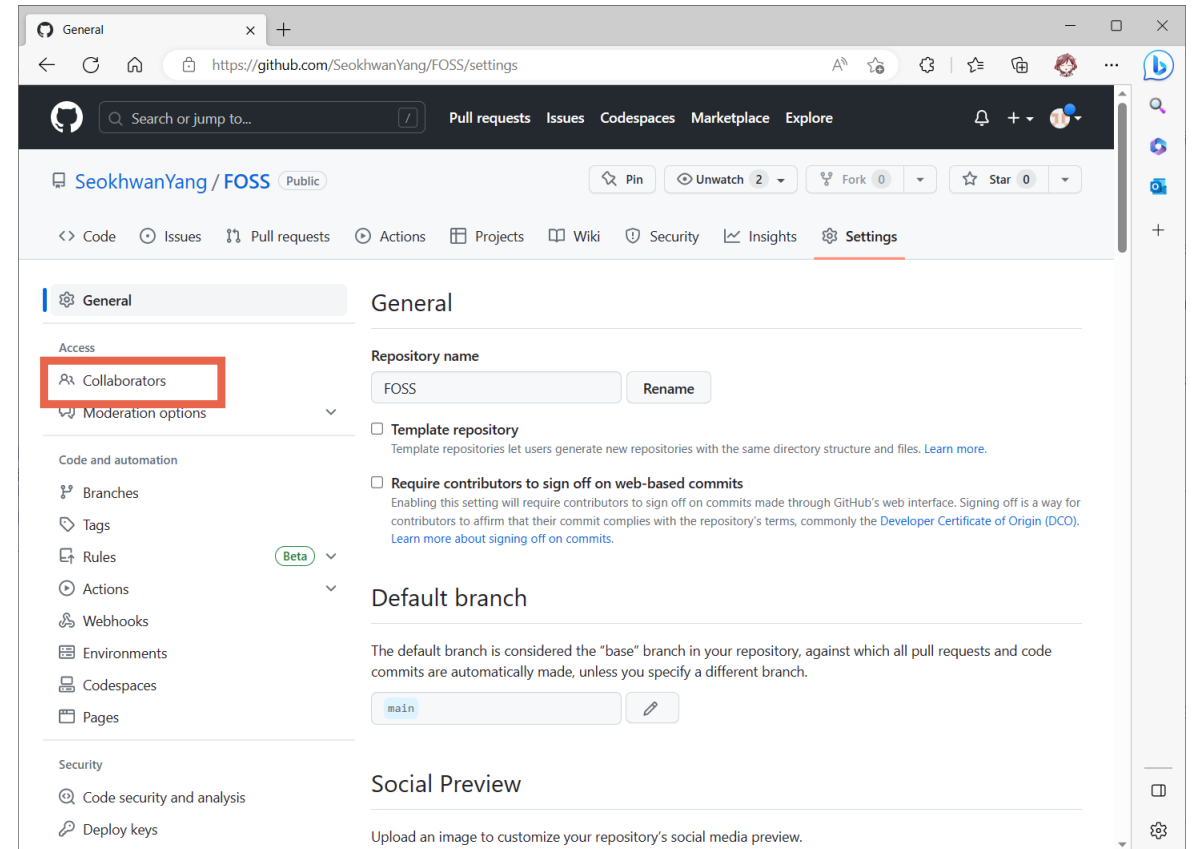
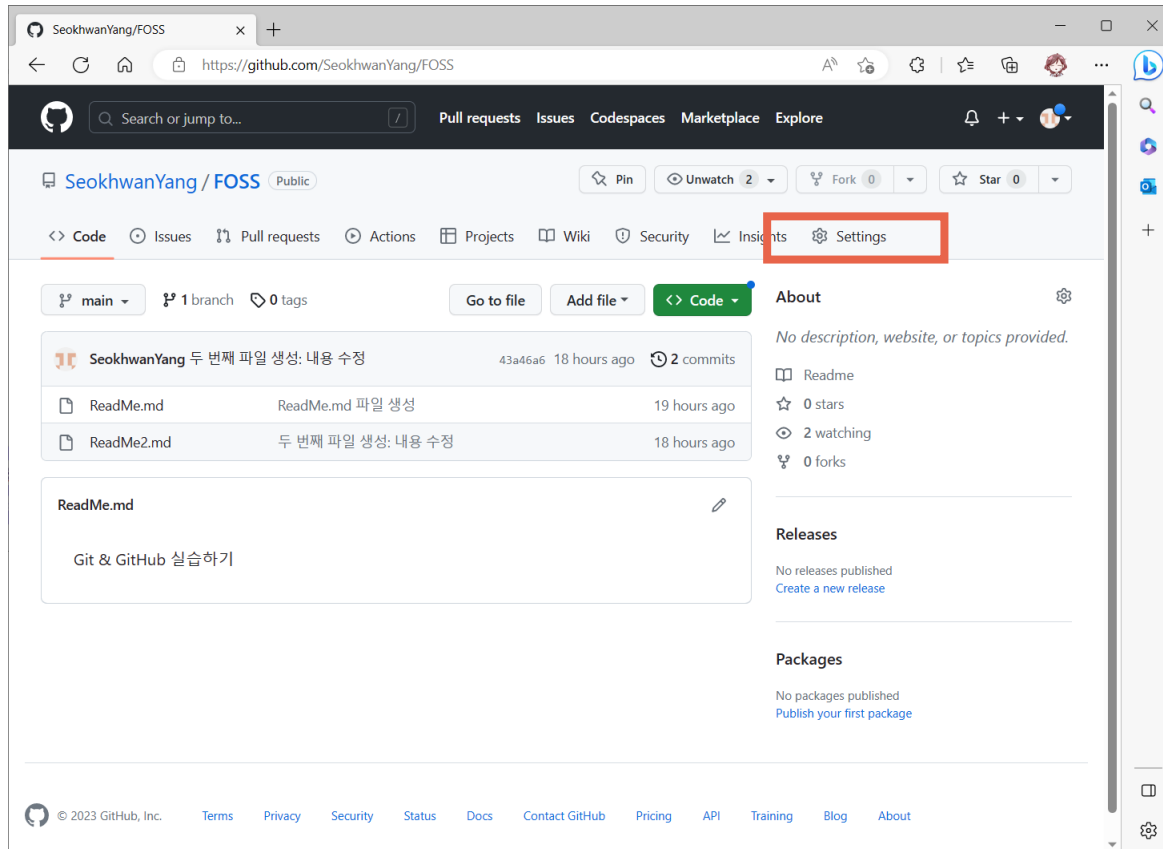
**협업자 등록 및 협업 환경 설정**

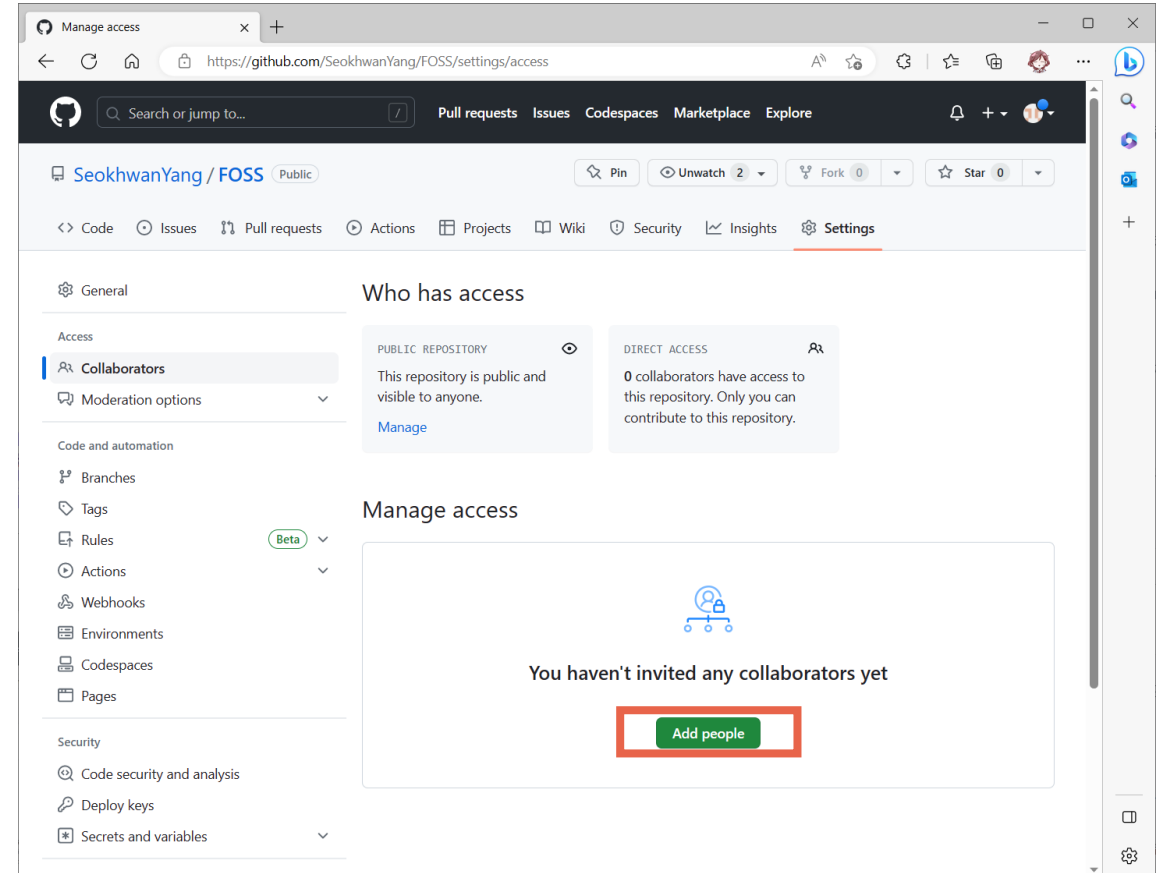
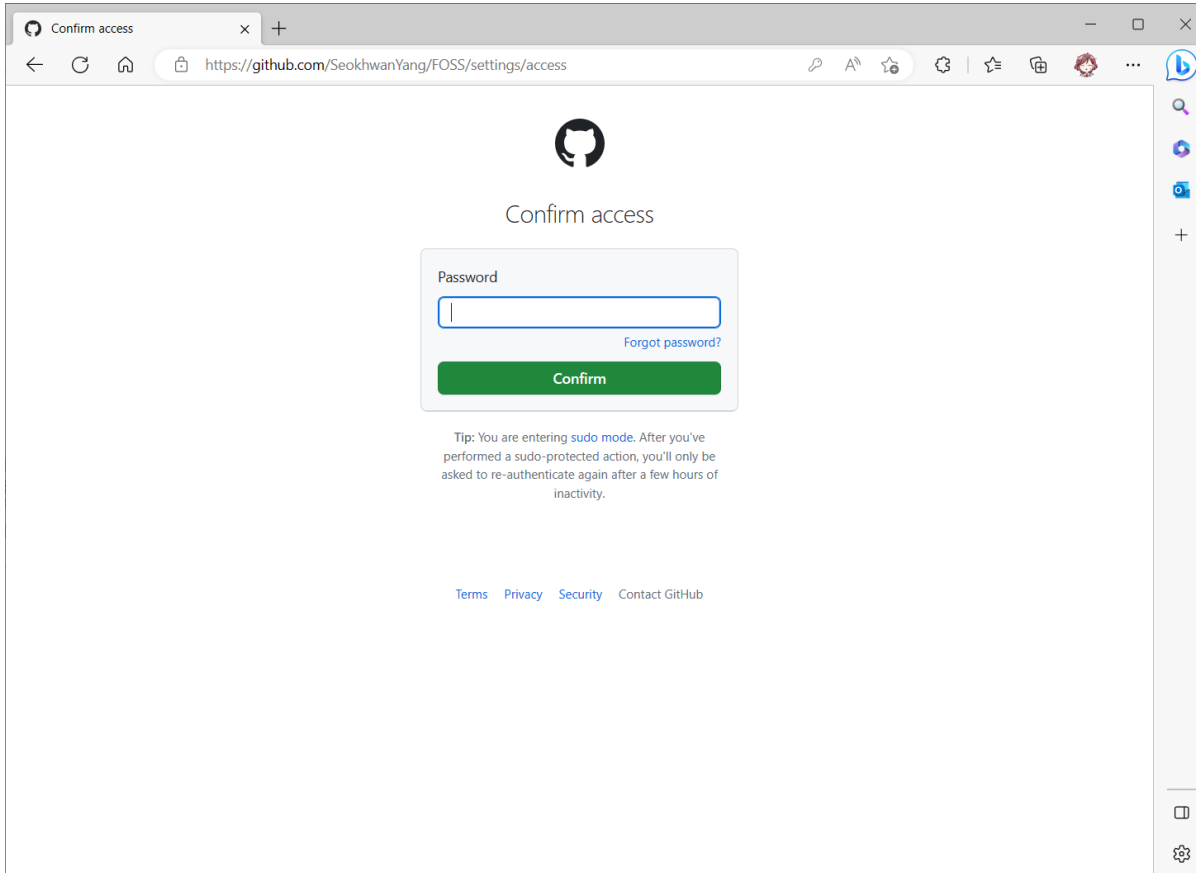
## • 등록 순서

- GitHub 사이트에서 해당 저장소로 이동
- 저장소 상단 메뉴에서 [Settings] 선택
- [Settings] 화면 좌측 메뉴바에서 [Collaborator] 선택
  - 비밀번호를 물어볼 때도 있음
- [Add people] 버튼 클릭
- 추가하려는 사용자 검색
  - GitHub 사용자 이름 또는 이메일로 검색 가능
- 검색 후 추천 명단에서 해당 사용자 선택 후 [추가 버튼]을 눌러서 초대

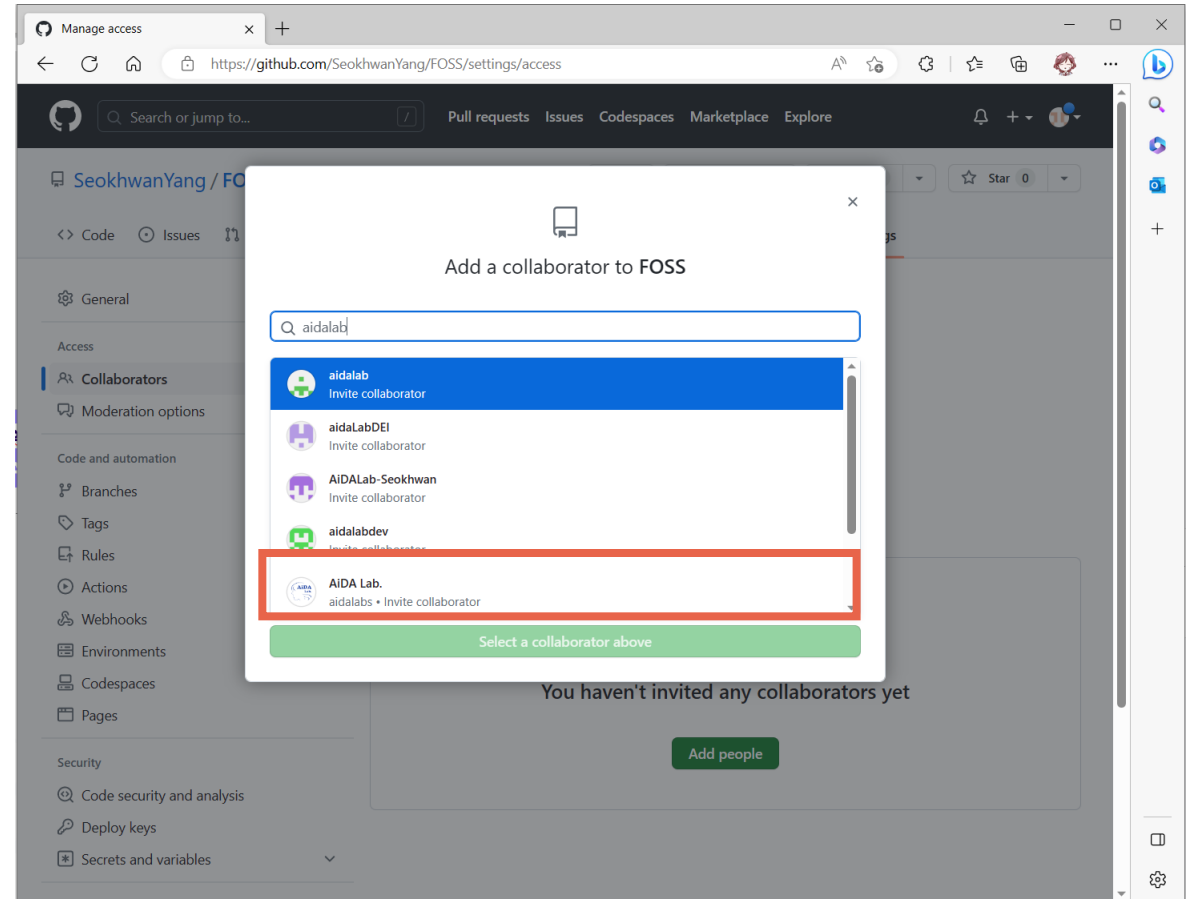
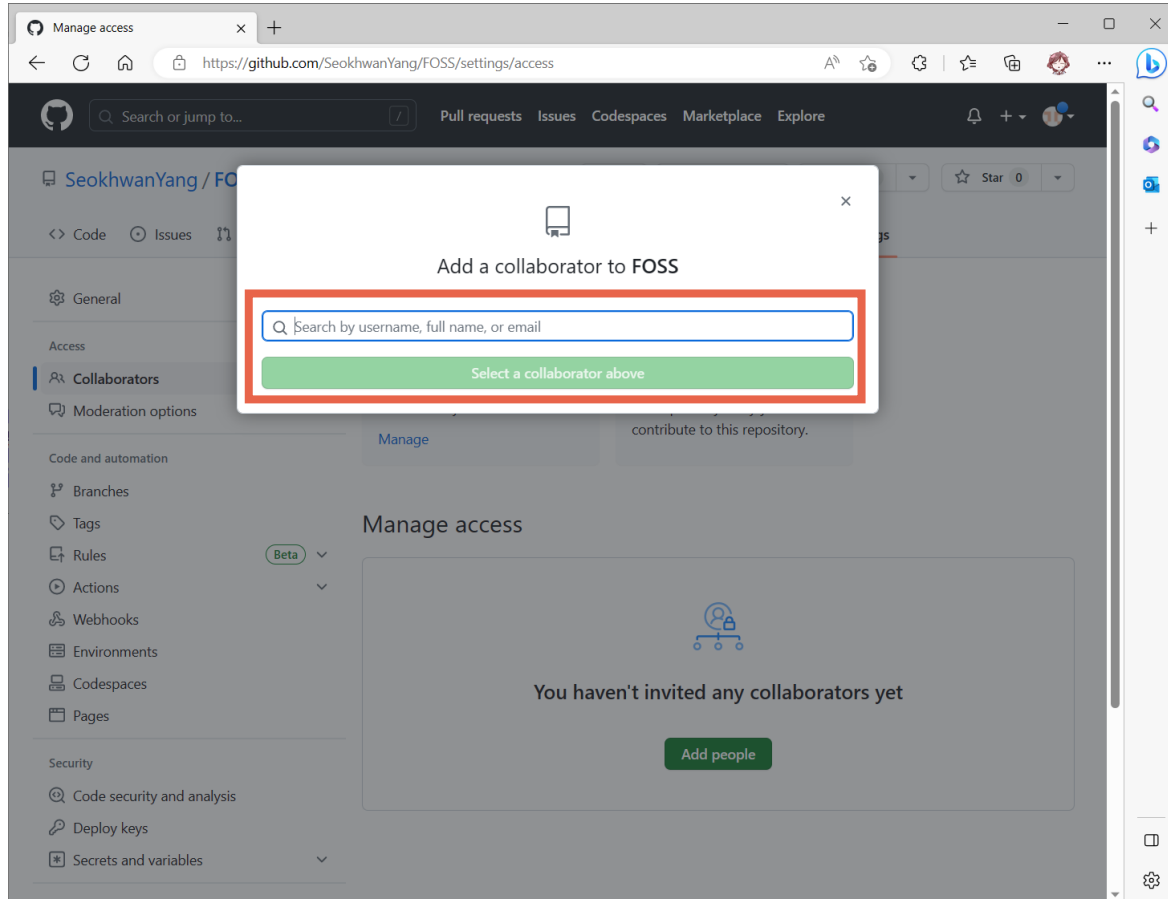


# 협업자(Collaborator) 등록하기

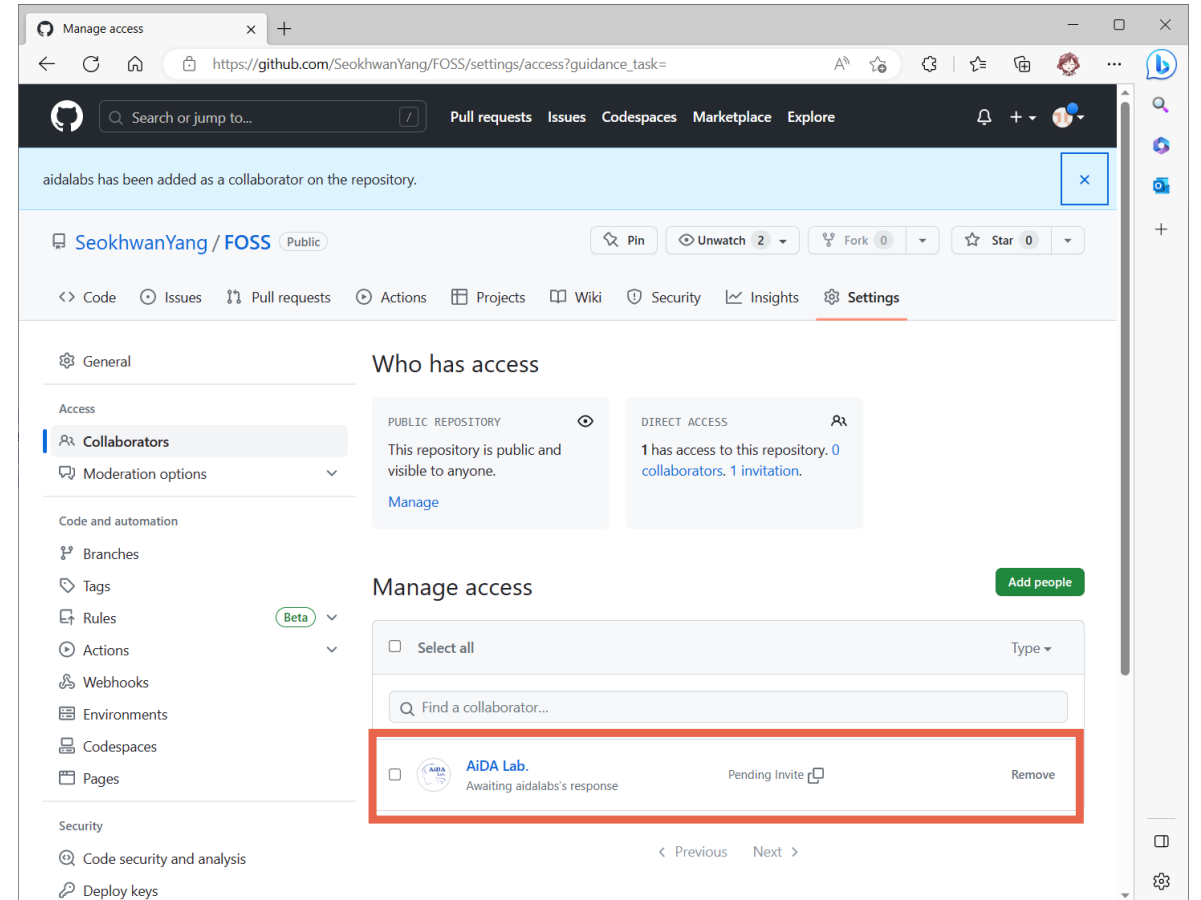
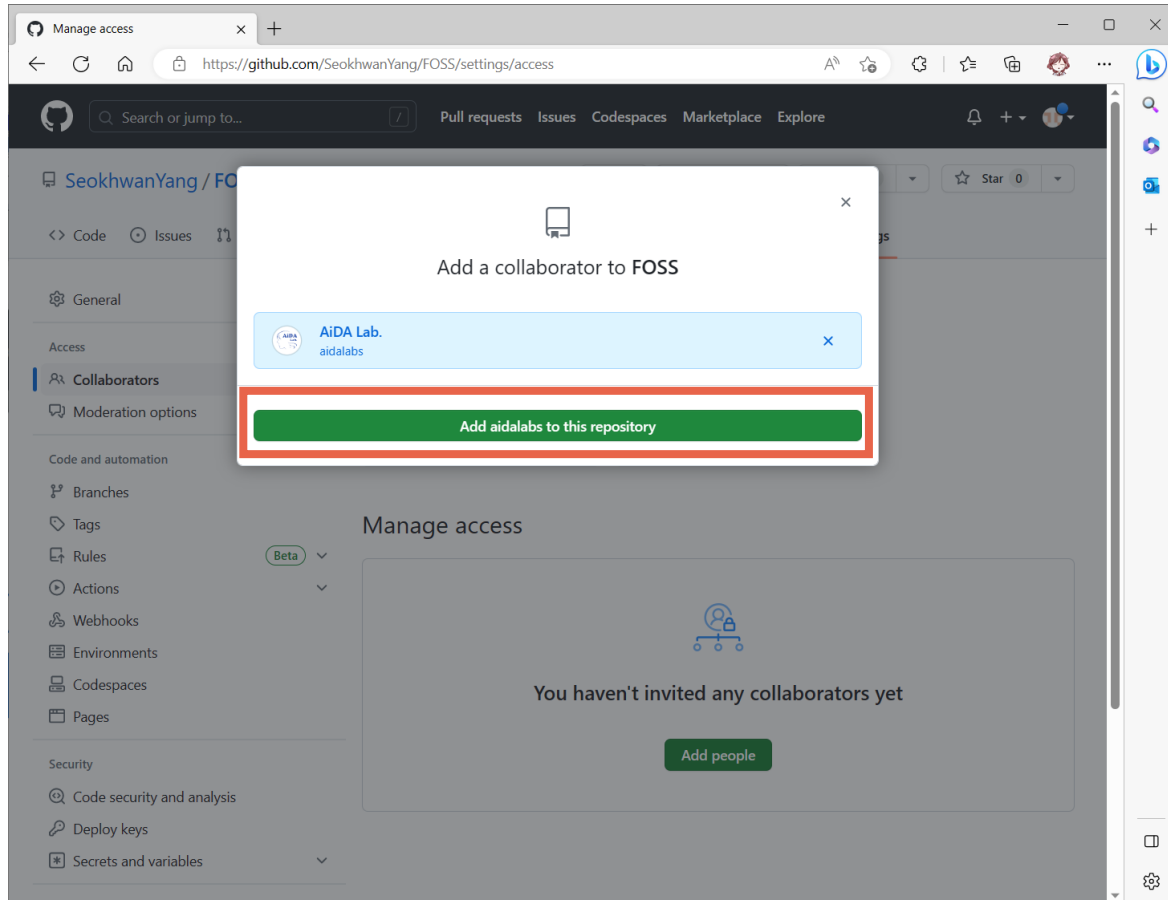




# 협업자(Collaborator) 등록하기

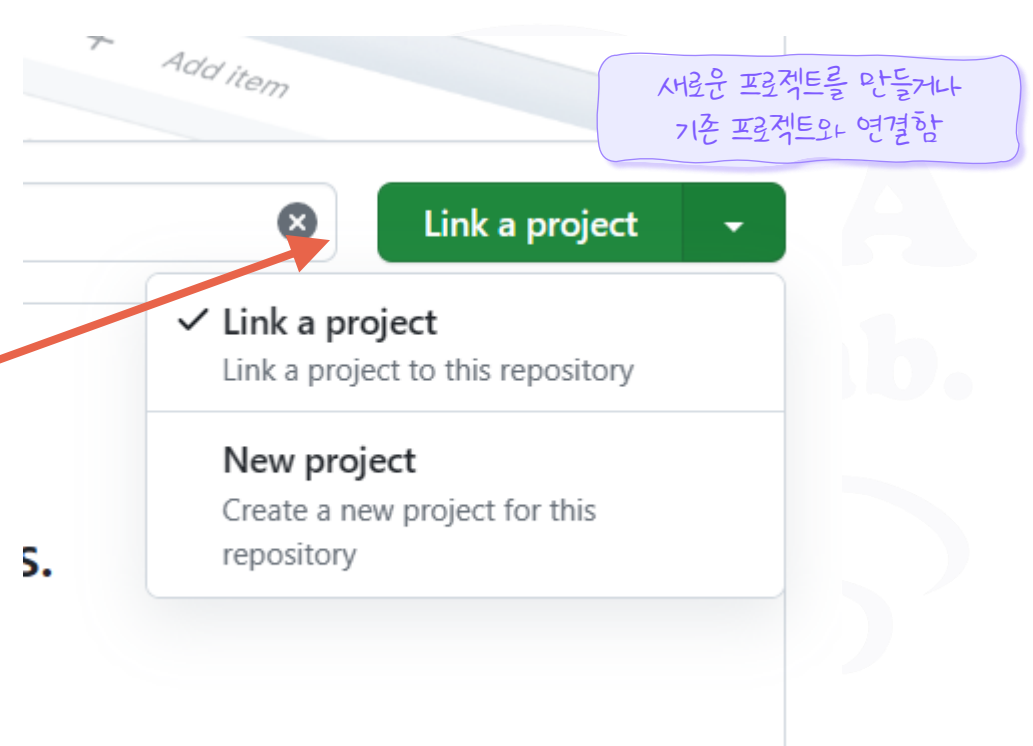
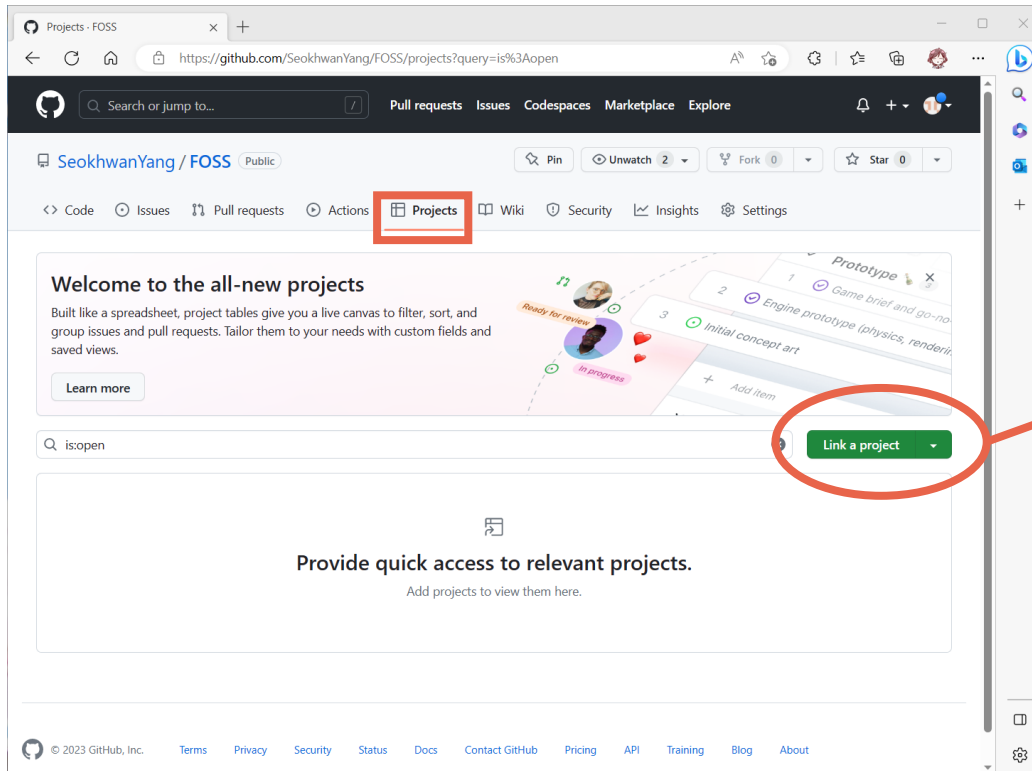


# 협업자(Collaborator) 등록하기

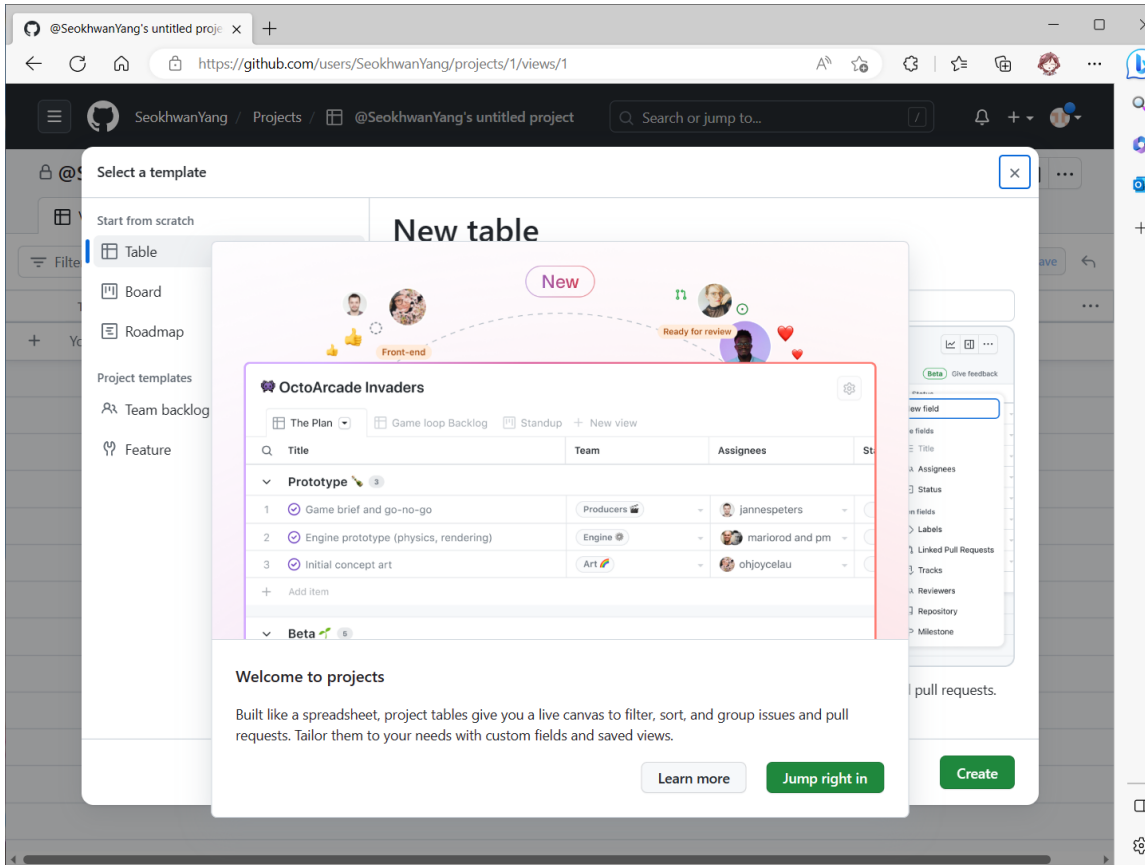


## • 프로젝트 도구

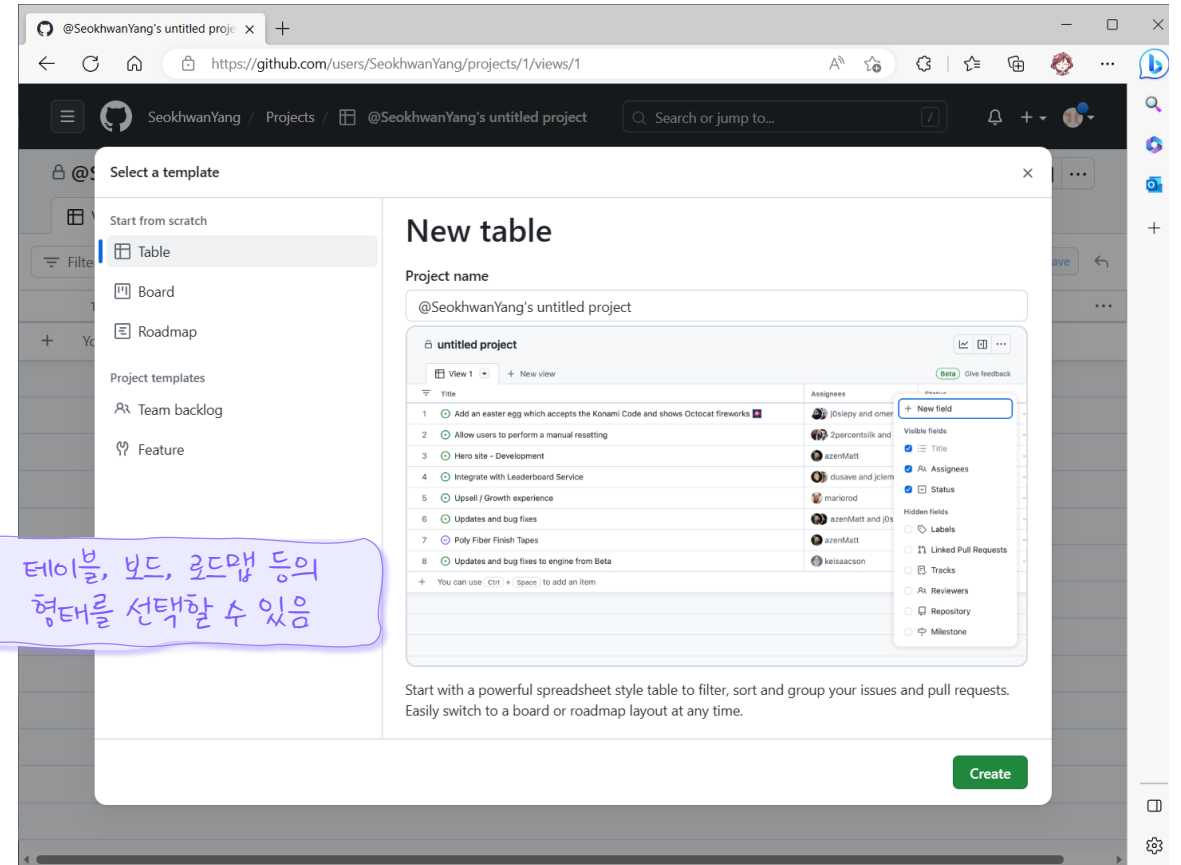
- 프로젝트의 진행에 대한 작업 및 우선순위를 돕는 도구
- 현재 진행 중인 프로젝트의 성격에 맞게 관리할 수 있도록 다양한 유형의 프로젝트 보드를 제공함



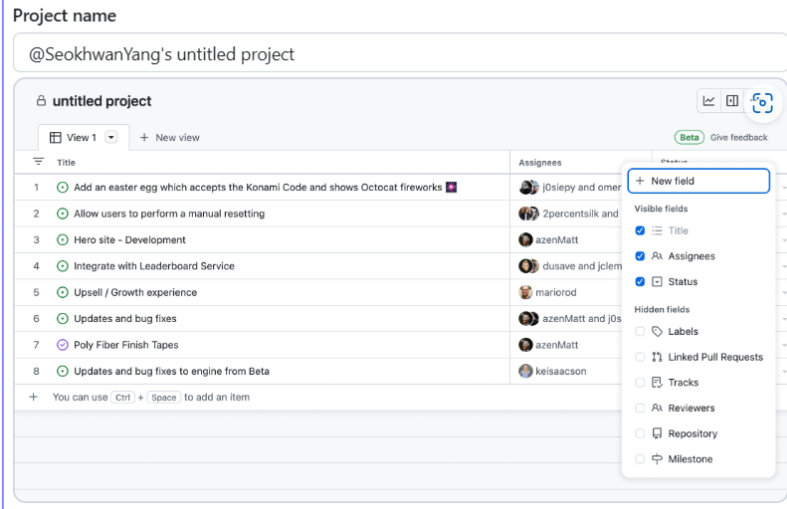
# 프로젝트(Projects)



테이블, 보드, 로드맵 등의 형태를 선택할 수 있음

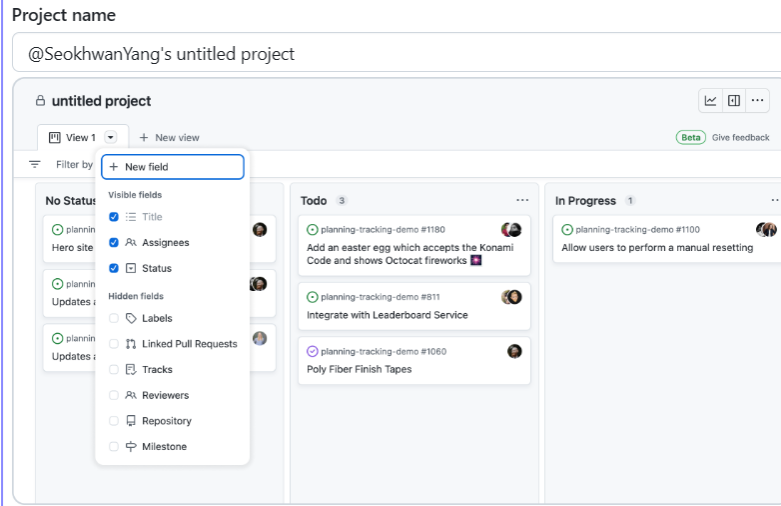


## New table



그냥 표 형태

## New board

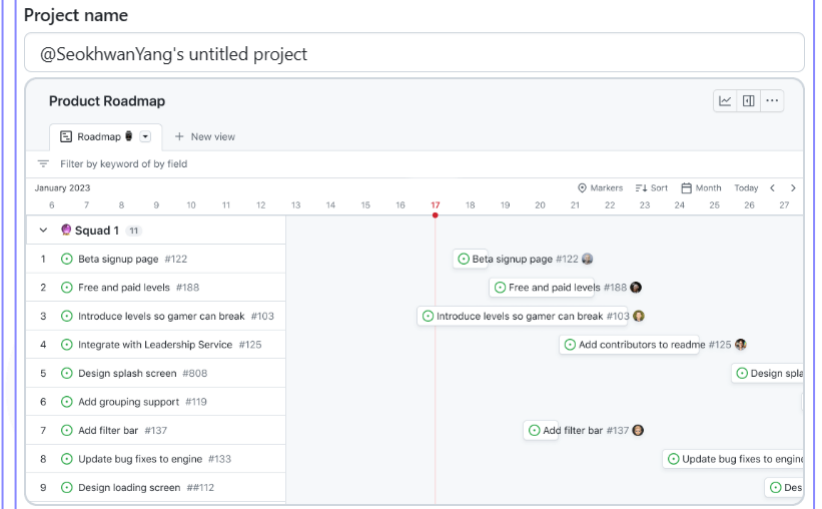


칸반 보드 형태

(이슈기반) 프로젝트 관리에서 많이 사용되는 Atlassian의 JIRA와 같은 형태

수업에서는 이슈를 중심으로 할 것이므로 칸반 보드를 사용

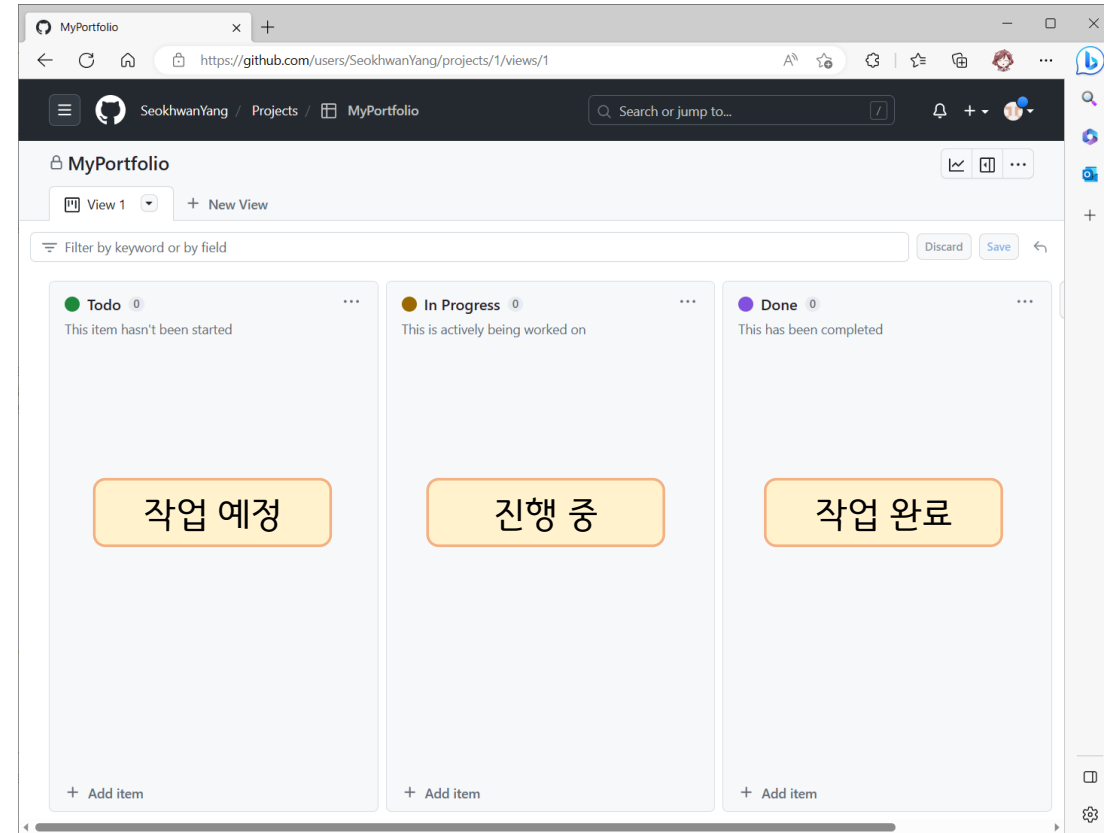
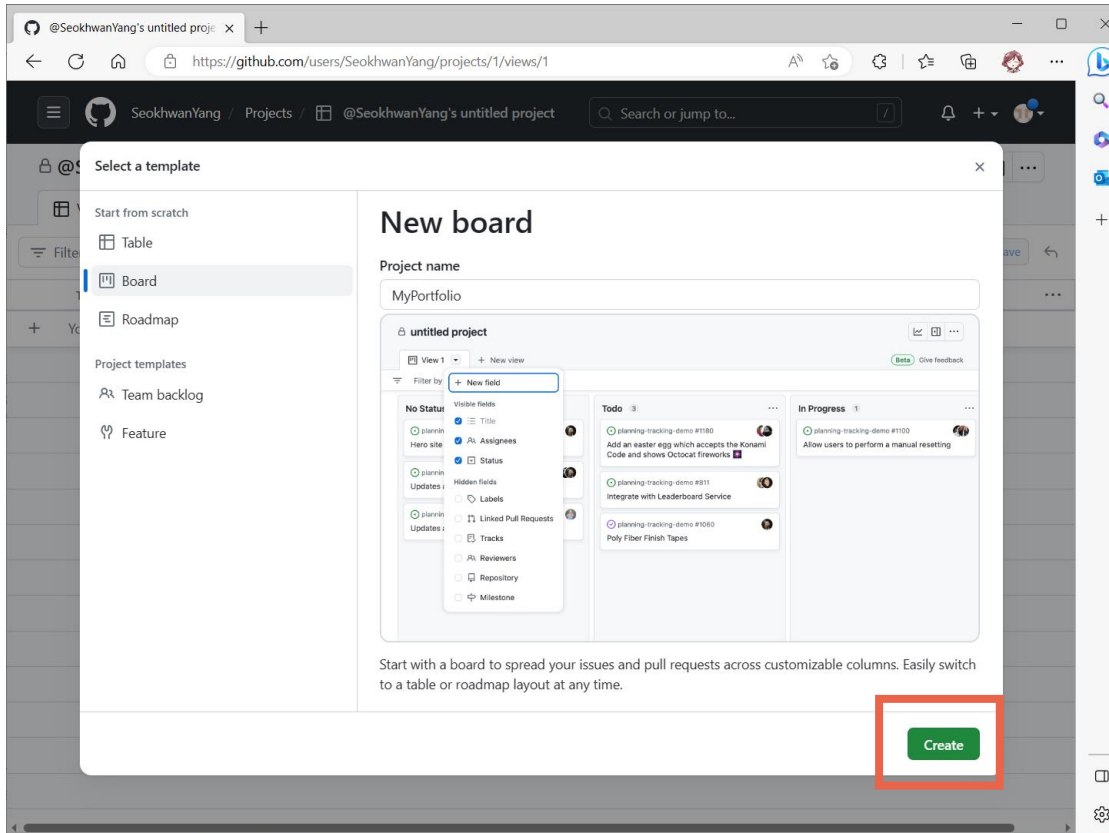
## Roadmap

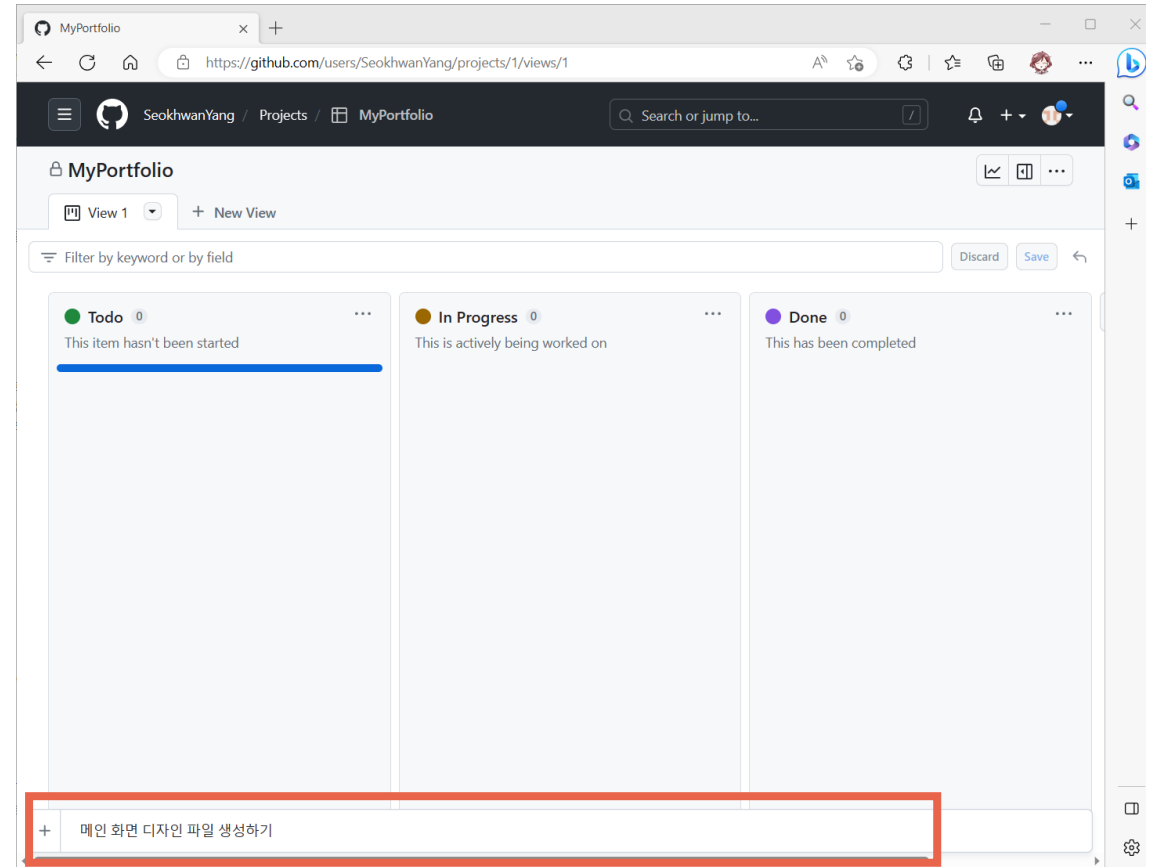
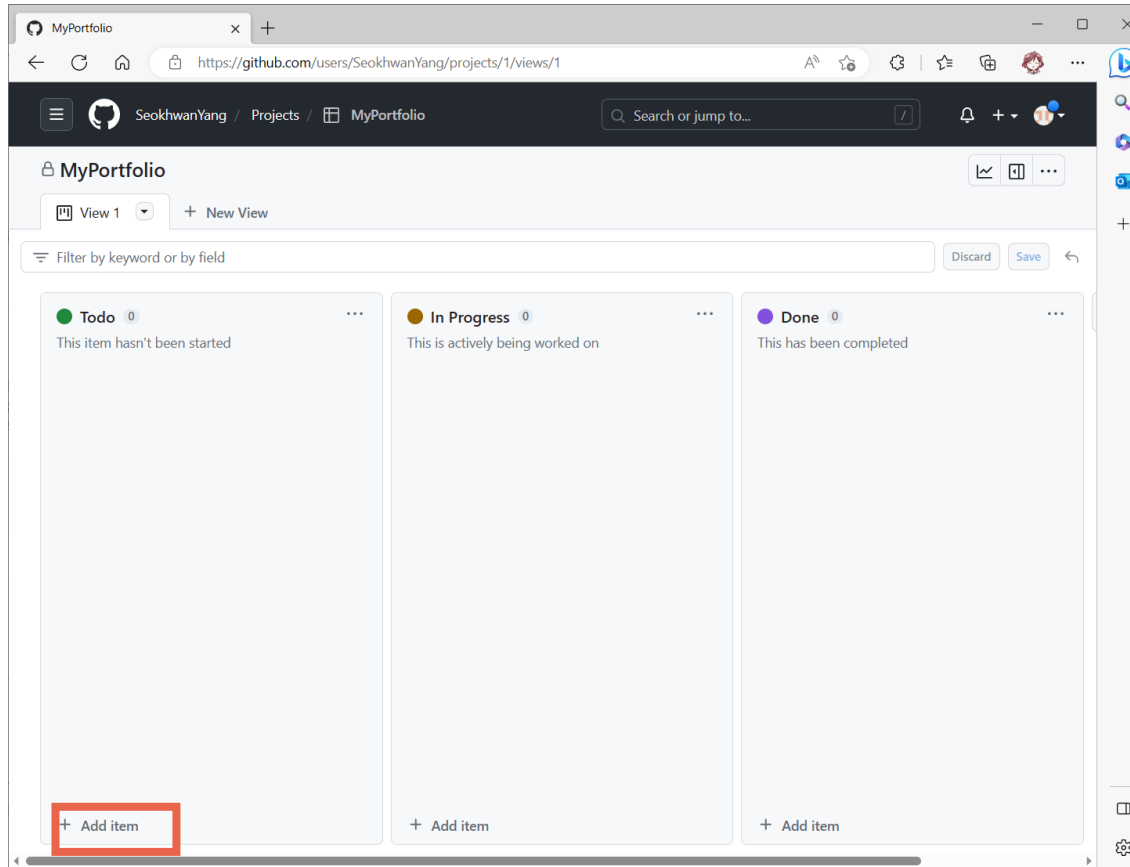


간트 차트 형태

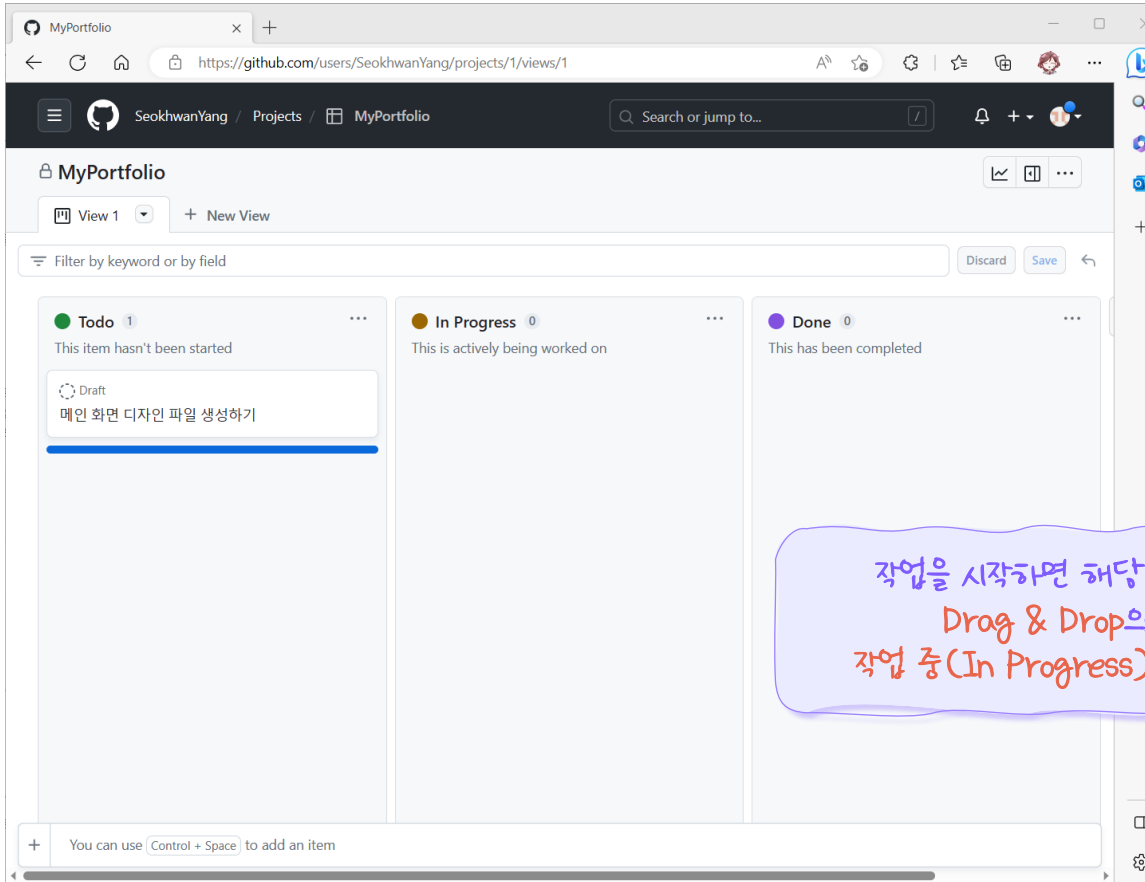
유명한 프로젝트 관리 소프트웨어인 MS Project와 같은 형태

취향 또는 프로젝트의 운영 형태에 따라 선택하면 됨

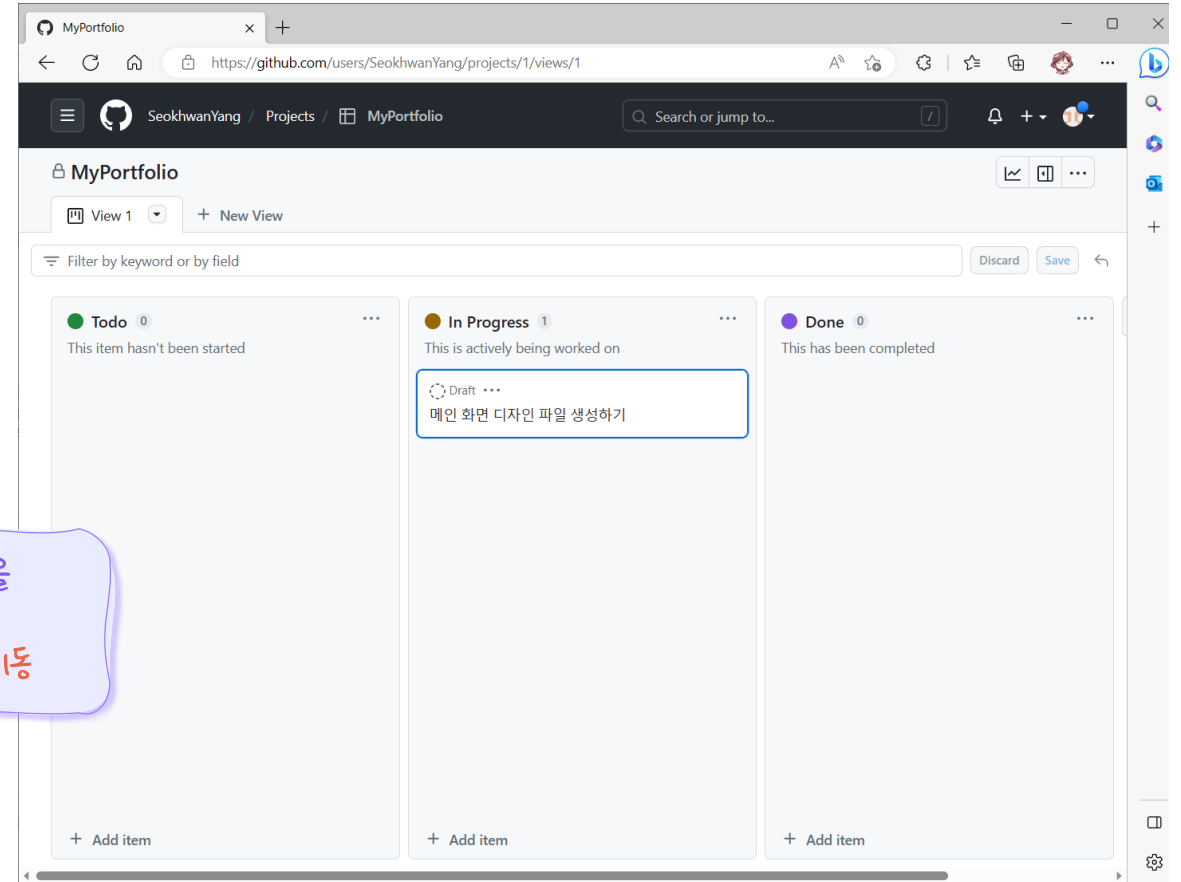




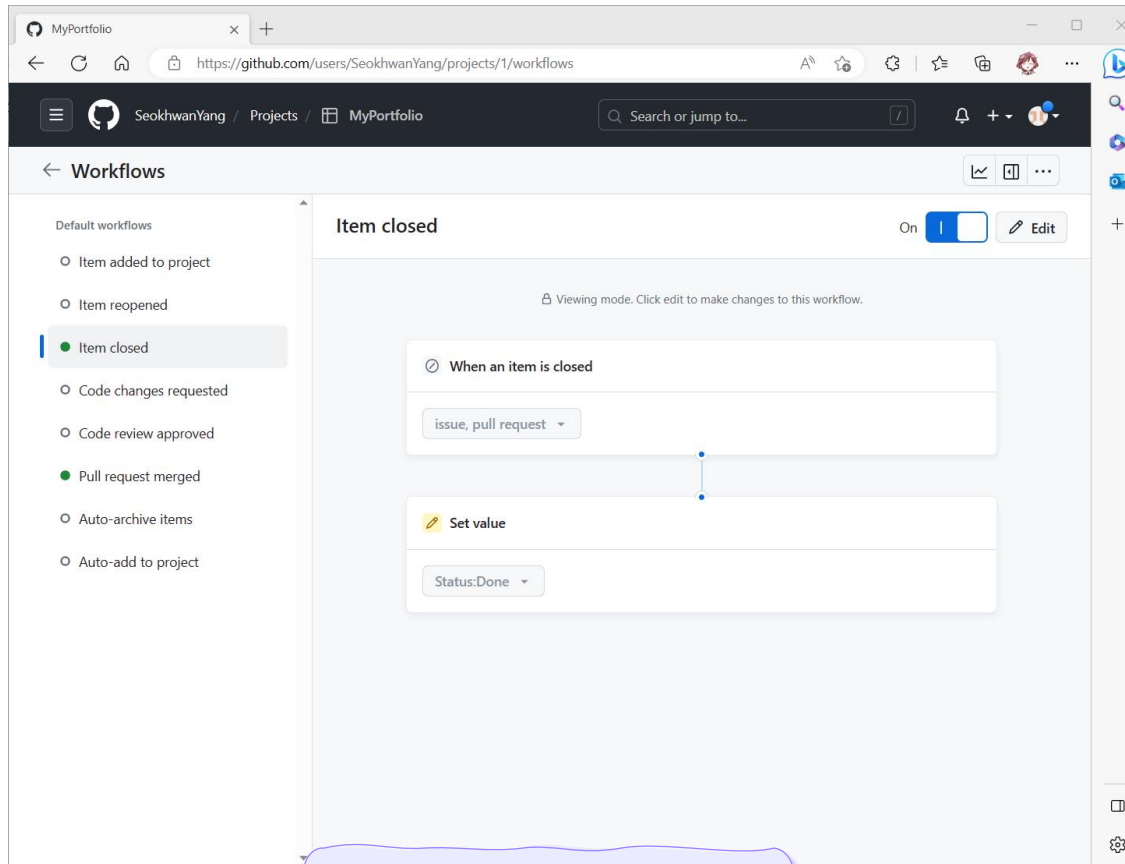
작업 항목을 추가함



작업을 시작하면 해당 항목을  
Drag & Drop으로  
작업 중(In Progress)으로 이동



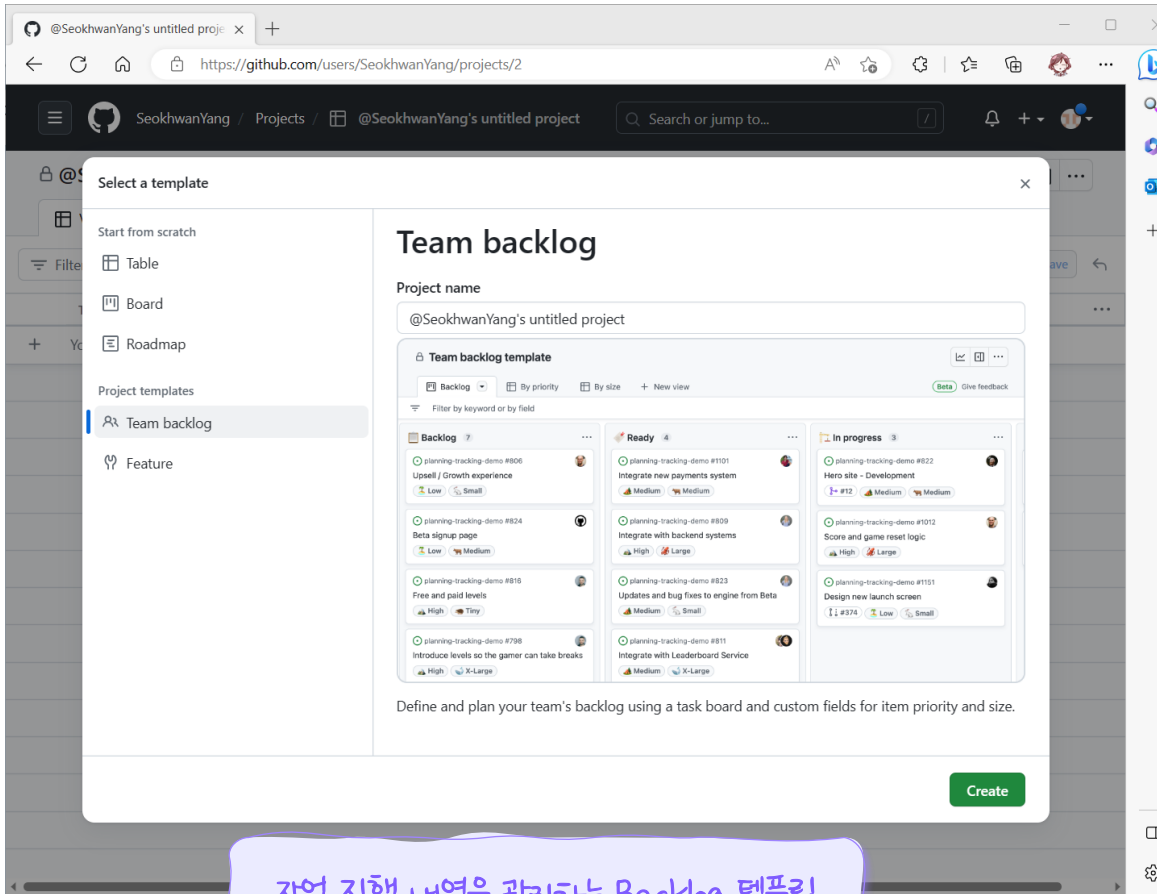
마찬가지로 작업이 끝나면  
작업 완료(Done)로 이동



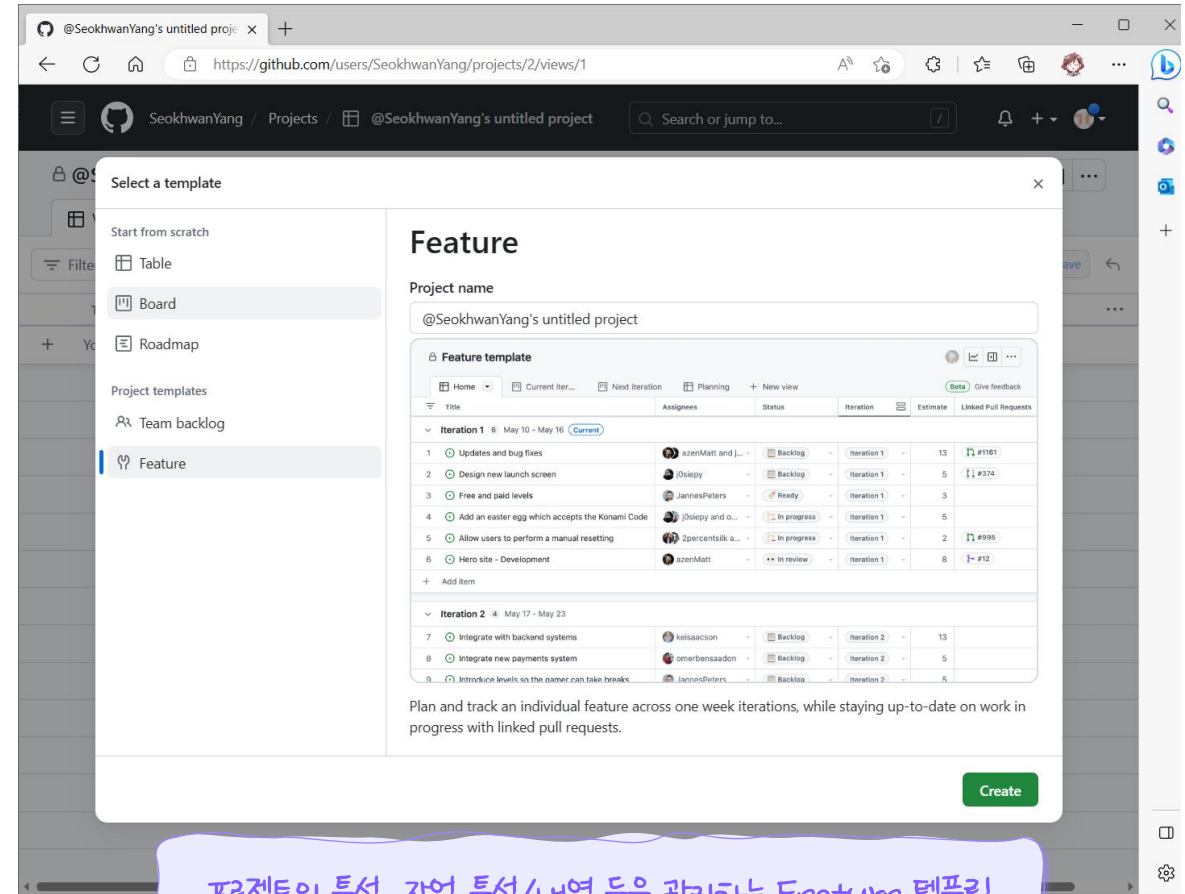
Workflow 등의 운영도 가능



- 프로젝트 생성 시, 템플릿을 적용할 수도 있음



작업 진행 내역을 관리하는 Backlog 템플릿



프로젝트의 특성, 작업 특성/내역 등을 관리하는 Feature 템플릿

## • 이슈(Issues)

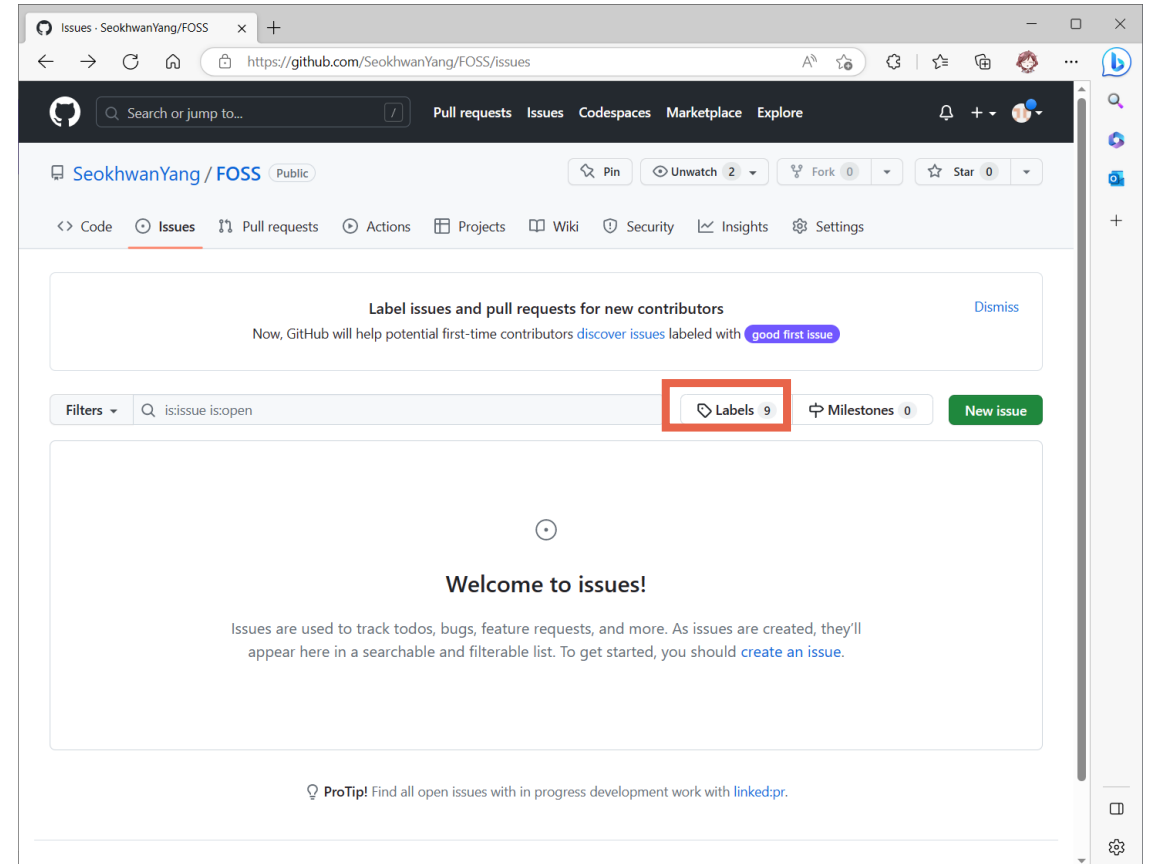
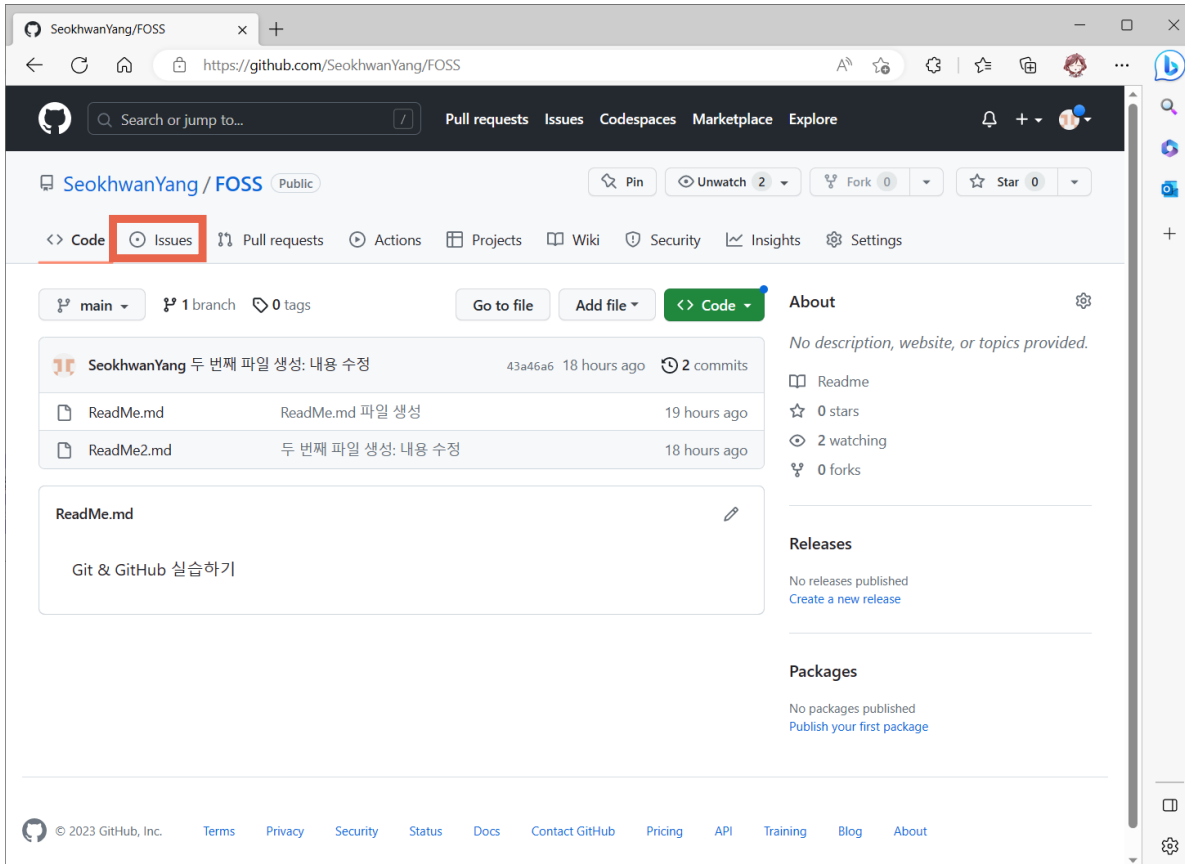
- 프로젝트 작업, 개선 사항, 오류 추적 기능을 제공하기 위한 도구
- 실제 프로젝트에서 수시로 밀려드는 새로운 기능, 기존 기능 개선, 오류 해결 등의 작업을 **이슈 단위**로 작성하고 관리할 수 있음
- 이슈는 각 저장소마다 존재함

## • 이슈 라벨(Labels)

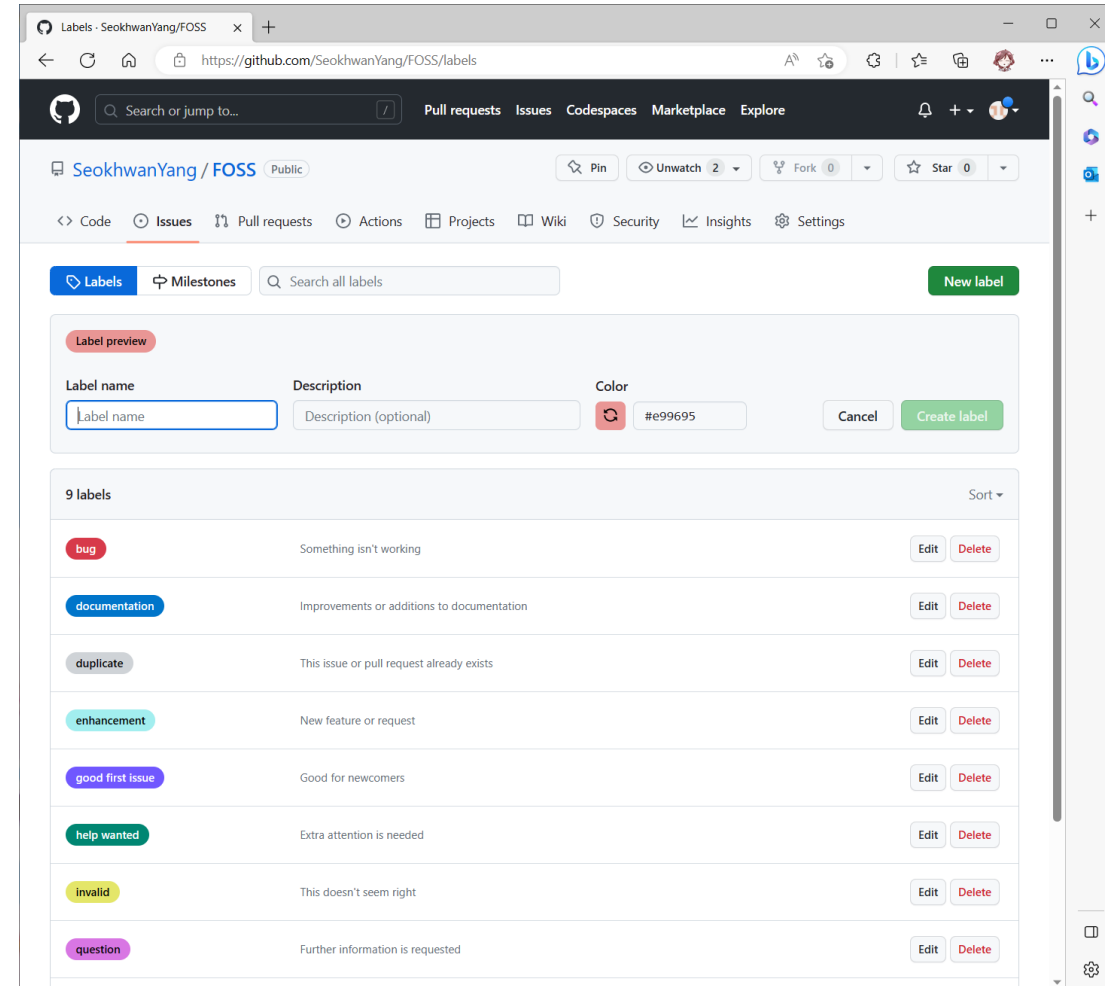
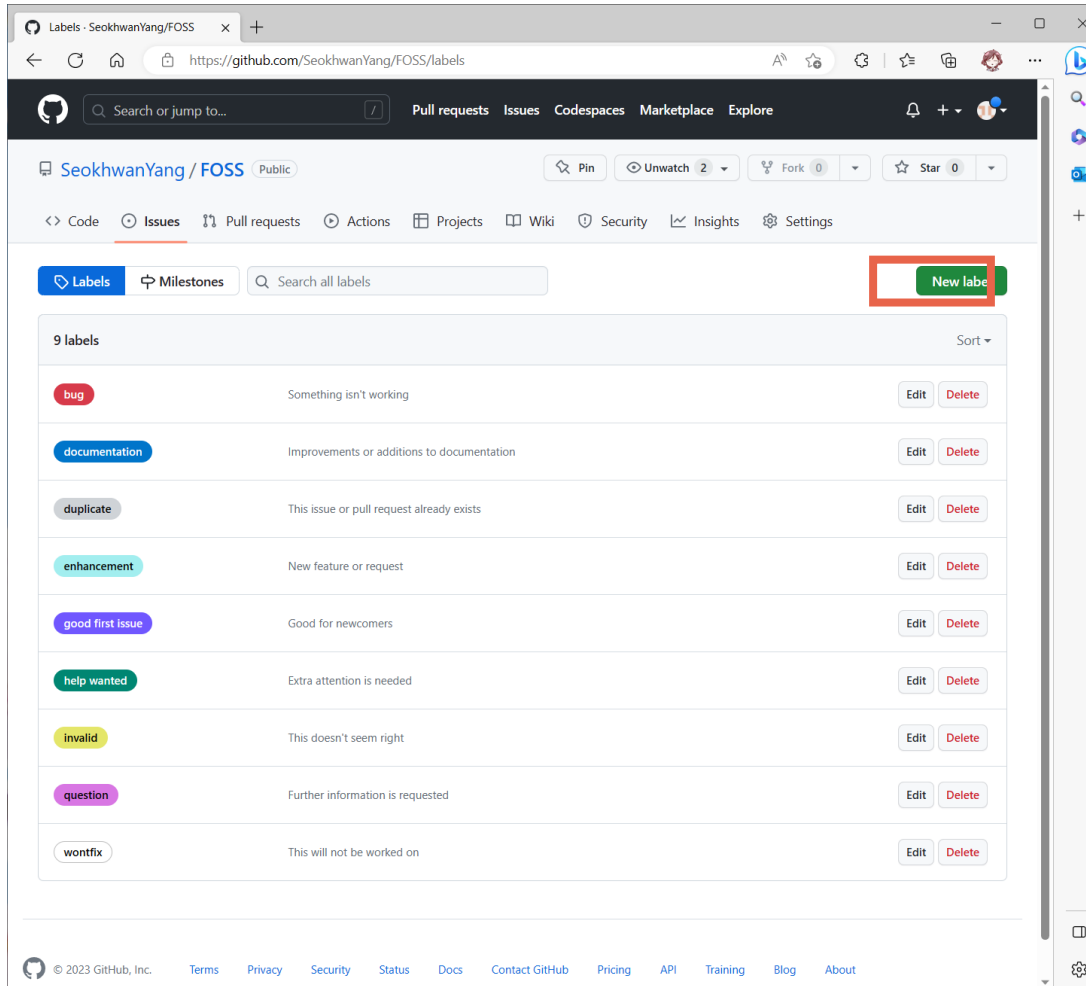
- 이슈의 성격을 구분 짓는 도구
- 기본적으로 bug, documentation, duplicate, enhancement, good first issue, help wanted, invalid, question, wontfix의 9가지가 존재함
- 그 외에 필요한 라벨은 직접 생성 가능하며, 기존 라벨의 수정도 가능함



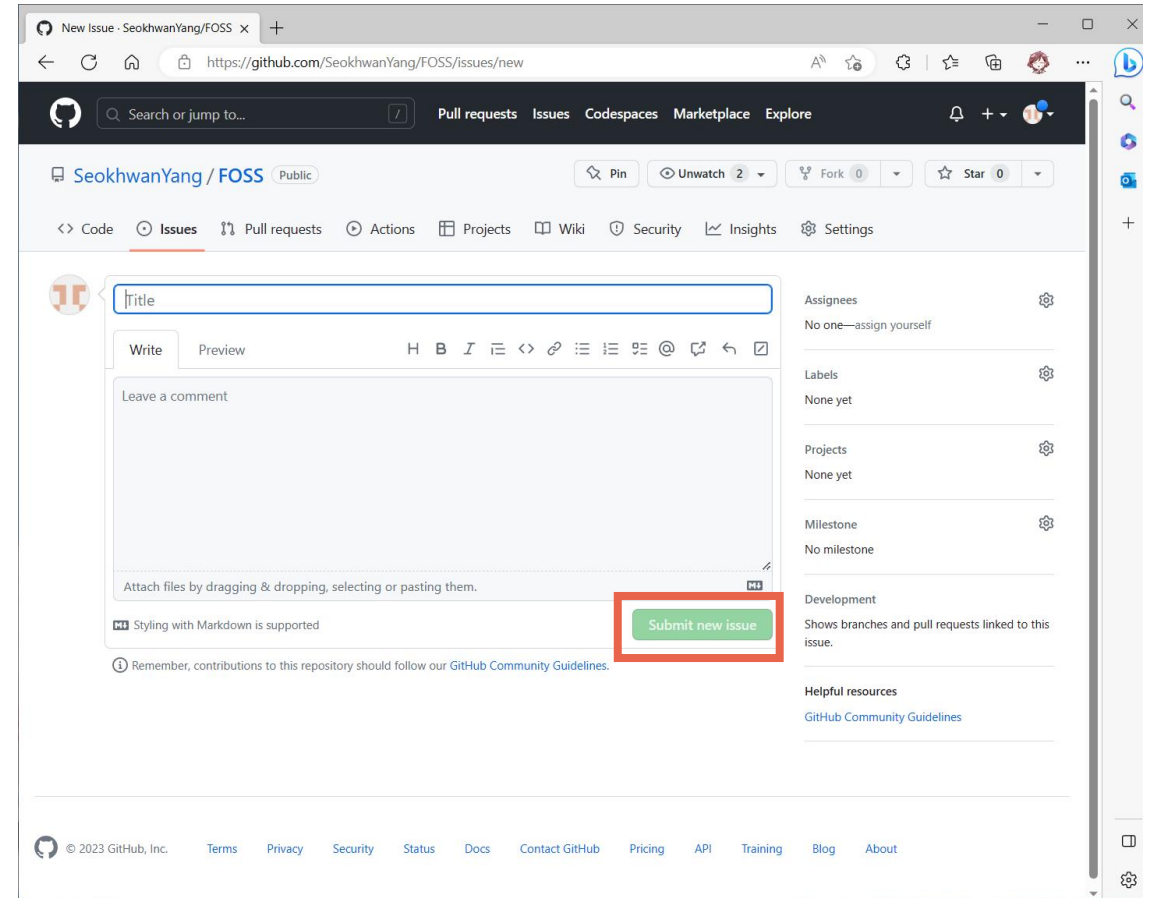
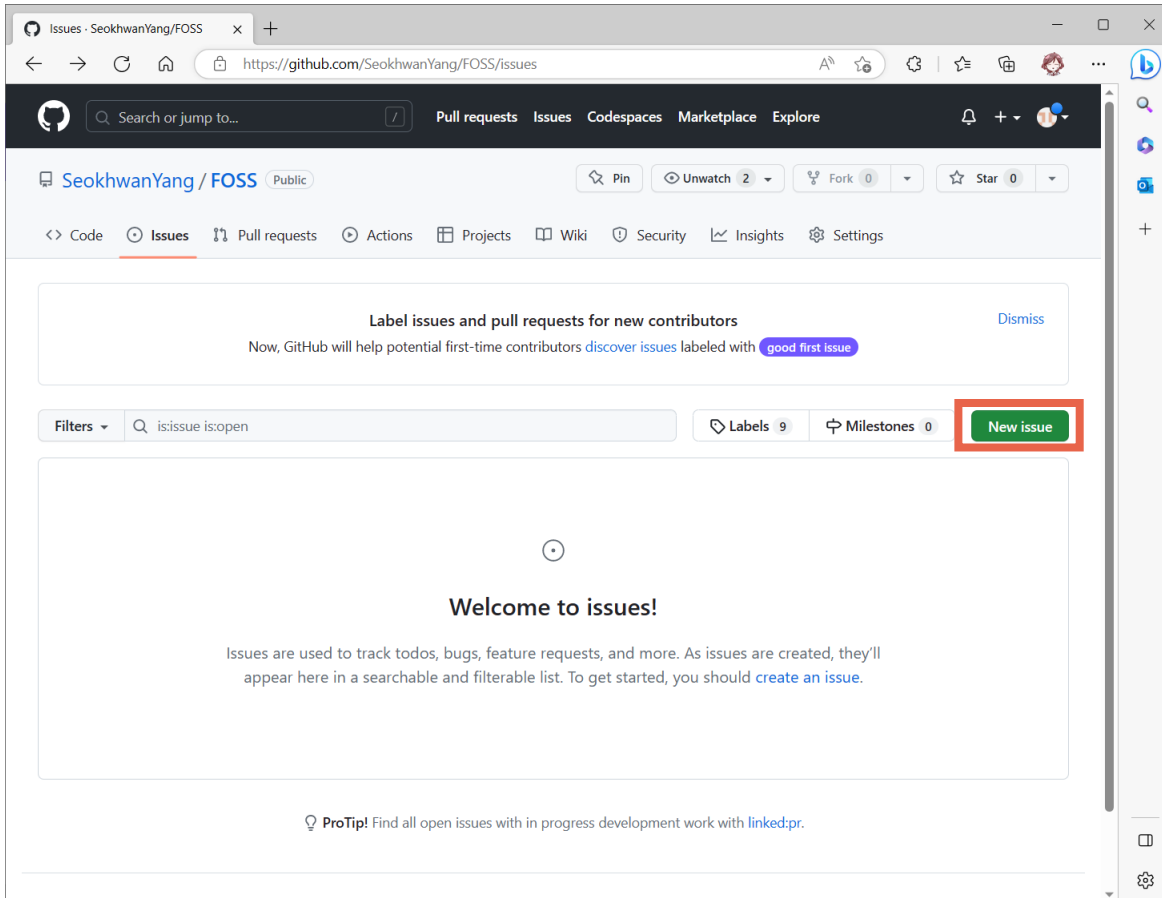
## • 이슈 라벨 만들기



## • 이슈 라벨 만들기



## • 이슈 만들기



## • 이슈 만들기

The screenshot shows the GitHub issue creation interface. At the top left is a profile icon. Below it is a text input field for the issue title, with the placeholder text 'Title' and an annotation box labeled '이슈 제목'. Below the title field are two tabs: 'Write' (selected) and 'Preview'. To the right of the tabs is a rich text editor toolbar with icons for bold (H), italic (I), link (🔗), code (⟨ >), list (☰), ordered list (☰), quote (≡), mention (@), share (🔗), undo (↶), and redo (↷). Below the toolbar is a large text area for the issue content, with the placeholder text 'Leave a comment' and an annotation box labeled '이슈 상세 내용'. At the bottom of the text area is a dashed line and the text 'Attach files by dragging & dropping, selecting or pasting them.' with a plus icon. Below the text area is a note: 'Styling with Markdown is supported'. At the bottom right of the form is a green button labeled 'Submit new issue'.

Remember, contributions to this repository should follow our [GitHub Community Guidelines](#).

Assignees

이슈를 해결할 사람 지정

No one—assign yourself

Labels

이슈 종류 지정



None yet

Projects

이슈가 포함될 프로젝트 선택

None yet

Milestone

이슈가 포함될 마일스톤 선택

No milestone

Development

Shows branches and pull requests linked to this issue.

Helpful resources

[GitHub Community Guidelines](#)

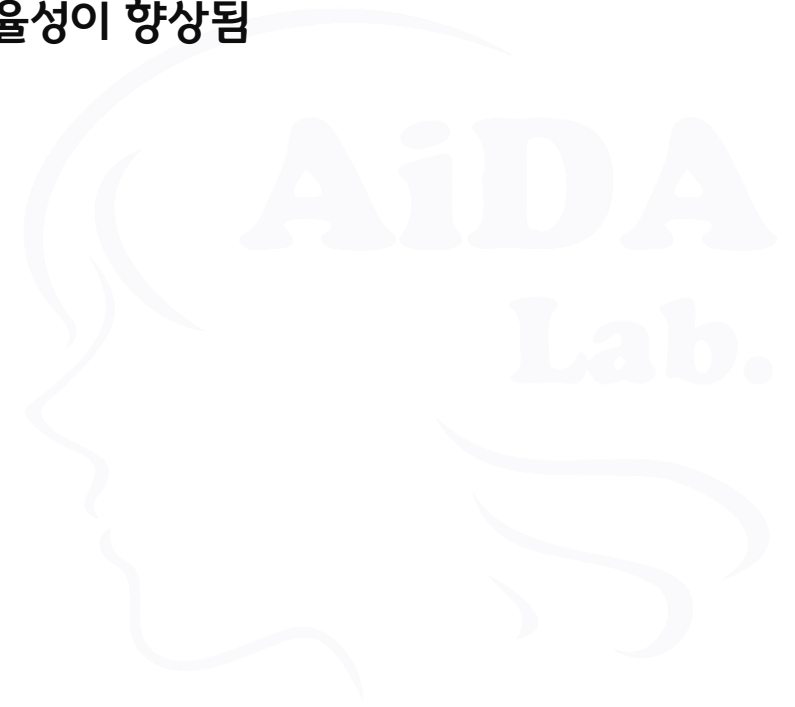
## • 프로젝트와 이슈의 연계

### • 이슈

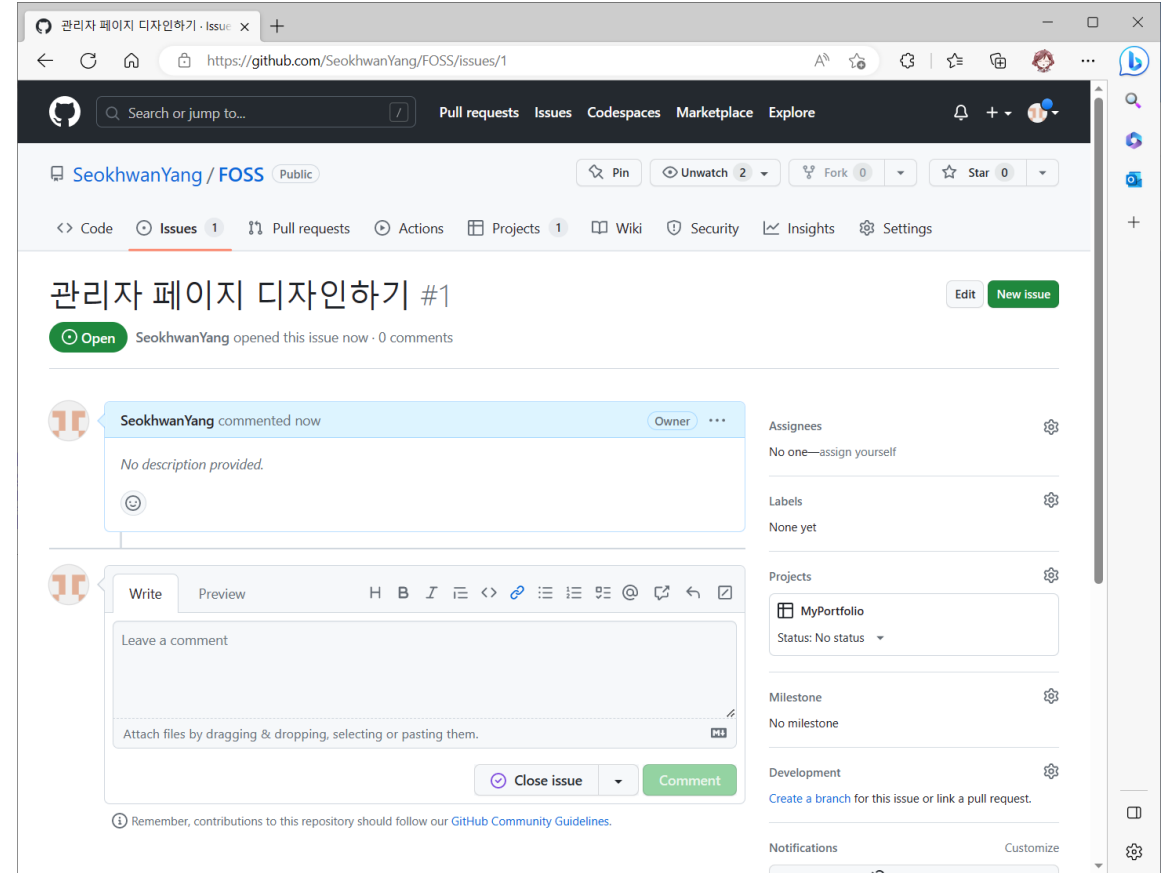
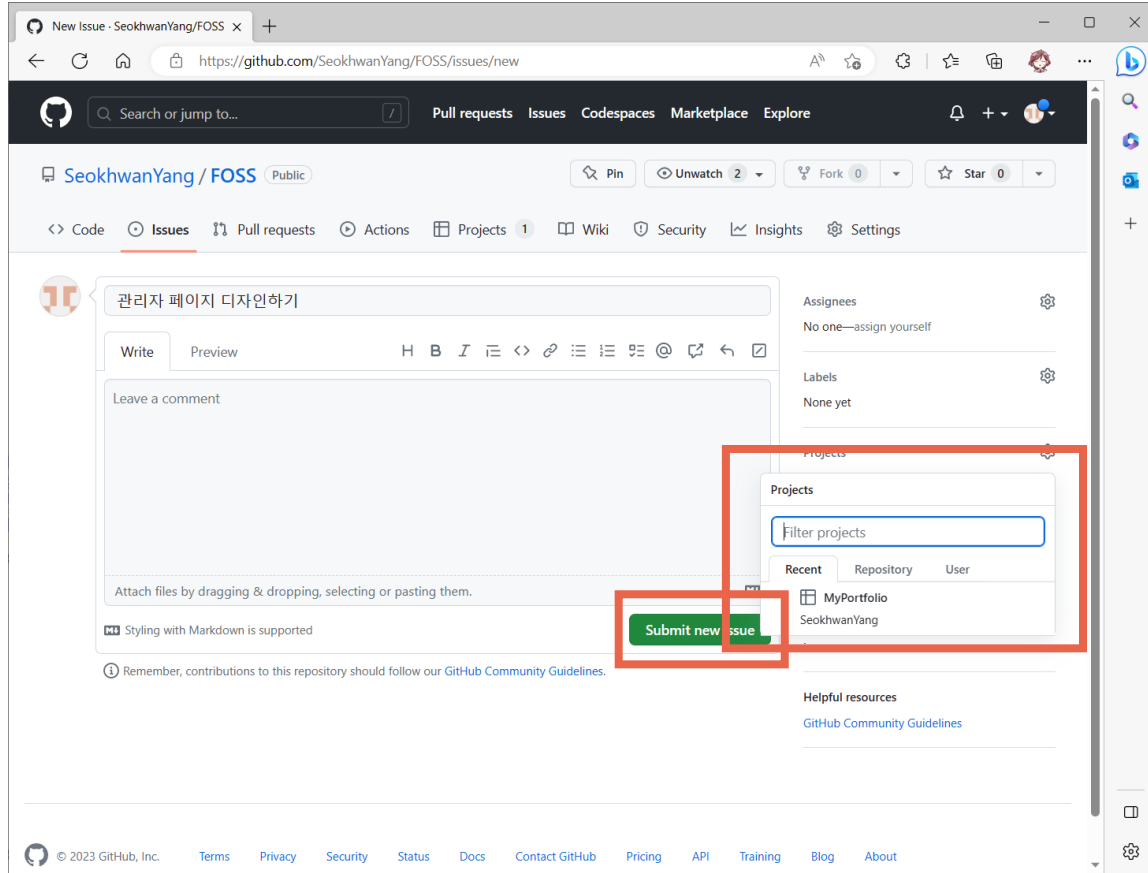
- 프로젝트 작업, 개선 사항, 오류 추적 기능을 제공하기 위한 도구
- 자체적으로 운영할 수도 있지만 프로젝트 도구와 연계하여 사용하면 효율성이 향상됨

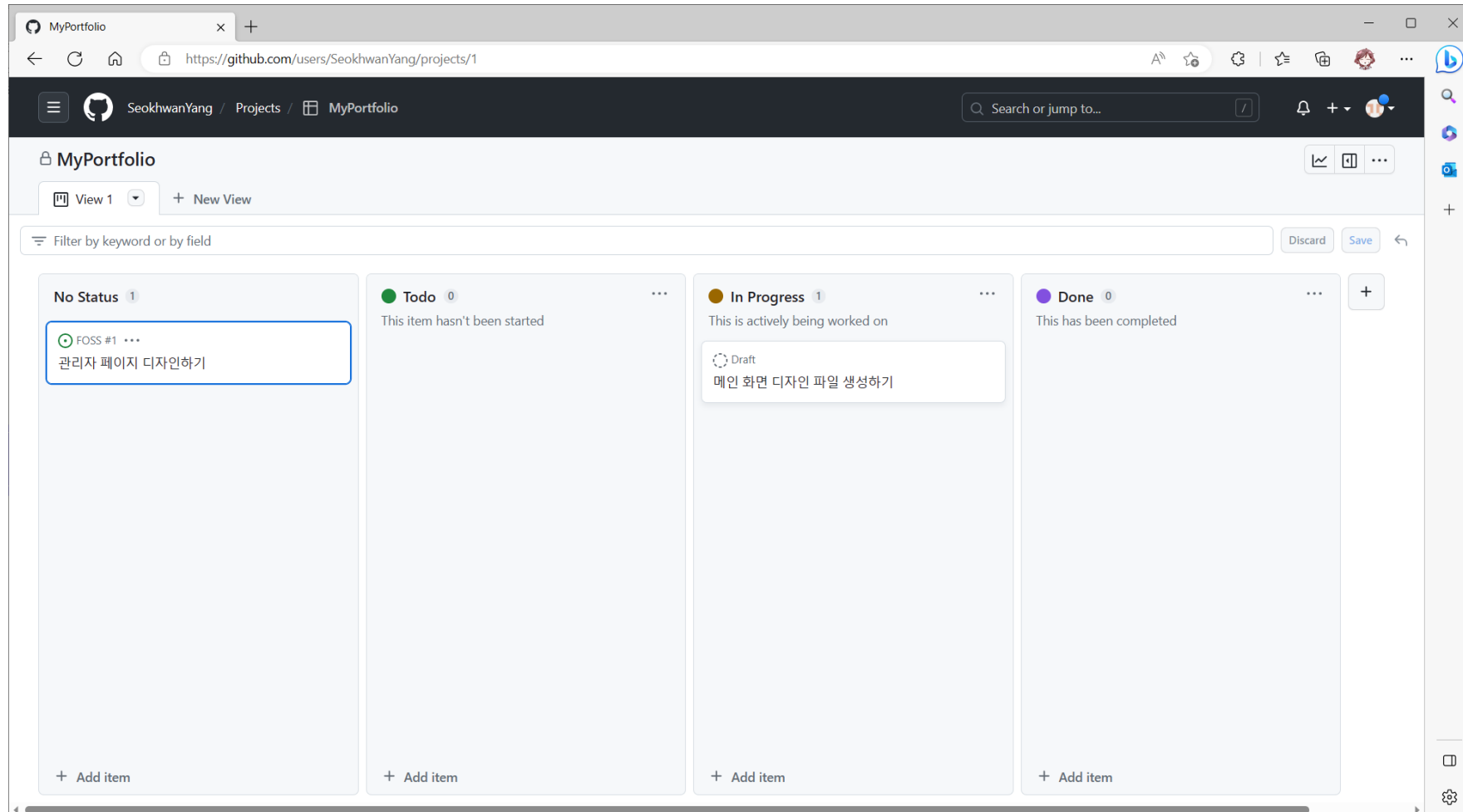
### • 사용법

- 이슈를 생성할 때, 해당 프로젝트를 선택함
- 프로젝트 페이지에 새로 추가된 이슈가 반영됨
- 프로젝트의 운영을 통해 이슈를 처리함



## • 프로젝트와 이슈의 연계





# 협업을 위한 Git/GitHub 기능

## Branch (분기)

- 브랜치(Branch, 분기)란?

- 프로젝트 기준 코드인 main 브랜치로부터 독립적인 작업 공간을 만들어주는 기능
- 여러 개발자가 서로 다른 버전의 코드를 만들 때 서로의 작업에 영향을 주고받지 않기 위하여 사용됨

```
Windows PowerShell
PS C:\AiDALab\Workspace> git status
On branch main
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    FOSS_Clone/
    PersonalAccessToken_20230522_90Days.txt

nothing added to commit but untracked files present (use "git add" to track)
PS C:\AiDALab\Workspace> |
```

지금까지 main 브랜치에서 작업하고 커밋을 생성했었음

- 커밋과 main 브랜치

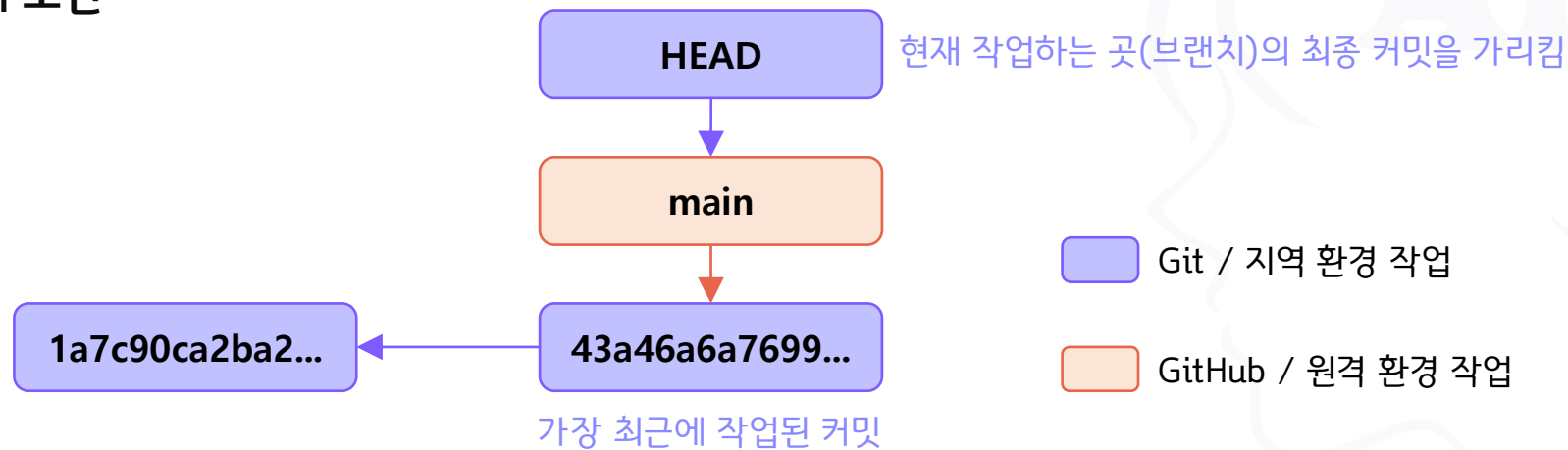
- 현 시점에서 main 브랜치는 Commit 3을 바라보고 있으며,  
main 브랜치의 커밋 내역에는 Commit 1, Commit 2, Commit 3이 포함되어 있음



- 현재 브랜치의 상태 확인해 보기

```
Windows PowerShell
PS C:\AiDALab\Workspace> git log --pretty=oneline --graph
* 43a46a6a7699d0b67ccc5cceb6c71783ef8ddd9f (HEAD -> main, origin/main) 두 번째 파일 생성: 내용 수정
* 1a7c90ca5ba22bafef4fe7902aab59eb20b2eee88 README.md 파일 생성
PS C:\AiDALab\Workspace>
```

- 도식화 해 보면

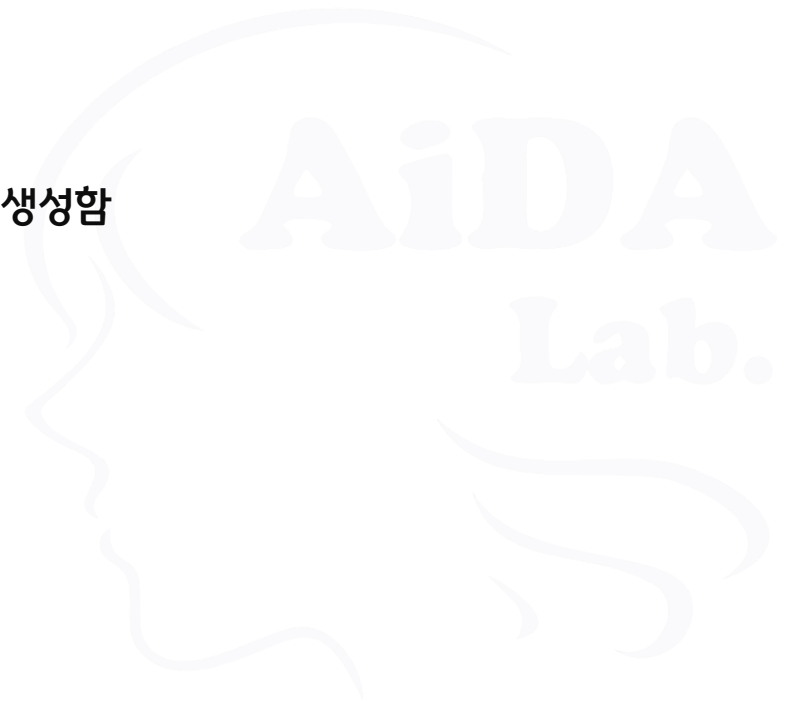


HEAD 포인터는 main 브랜치에서 작업 중이며, main 브랜치는 가장 최근의 커밋을 바라보고 있음

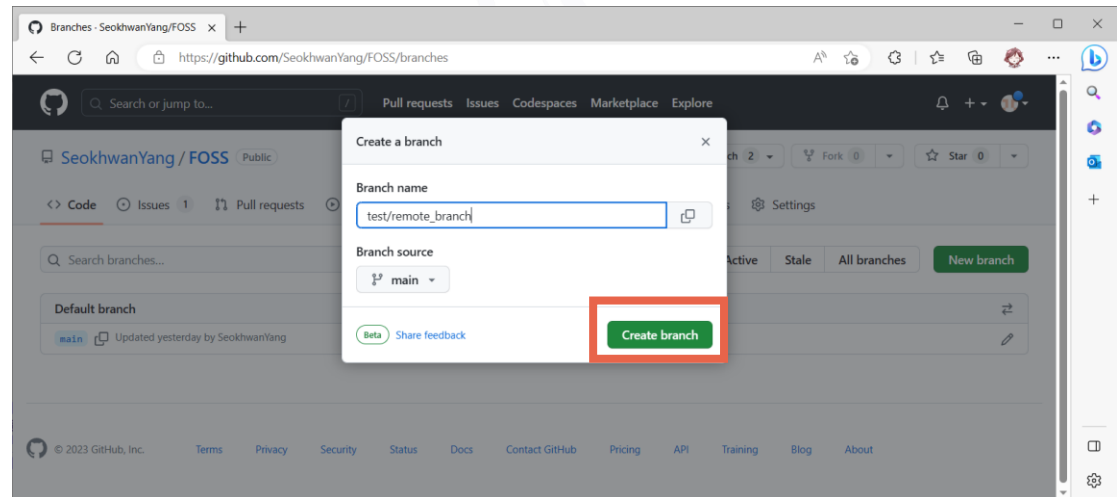
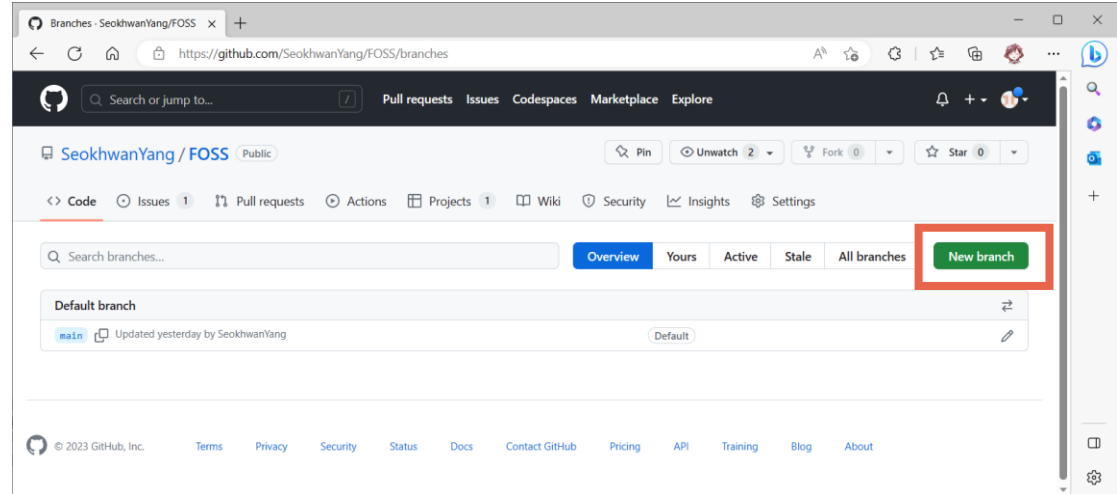
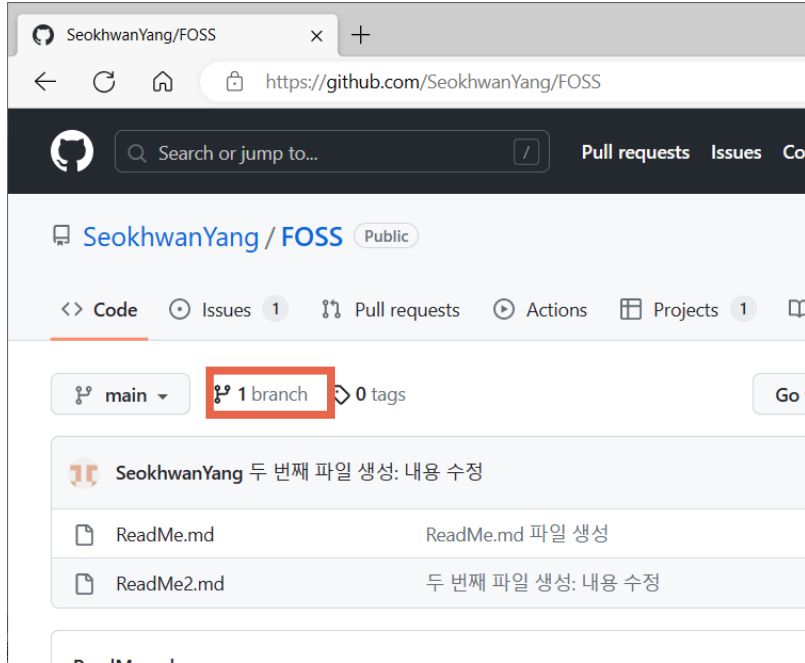
## • 브랜치 생성하기

### • 방법

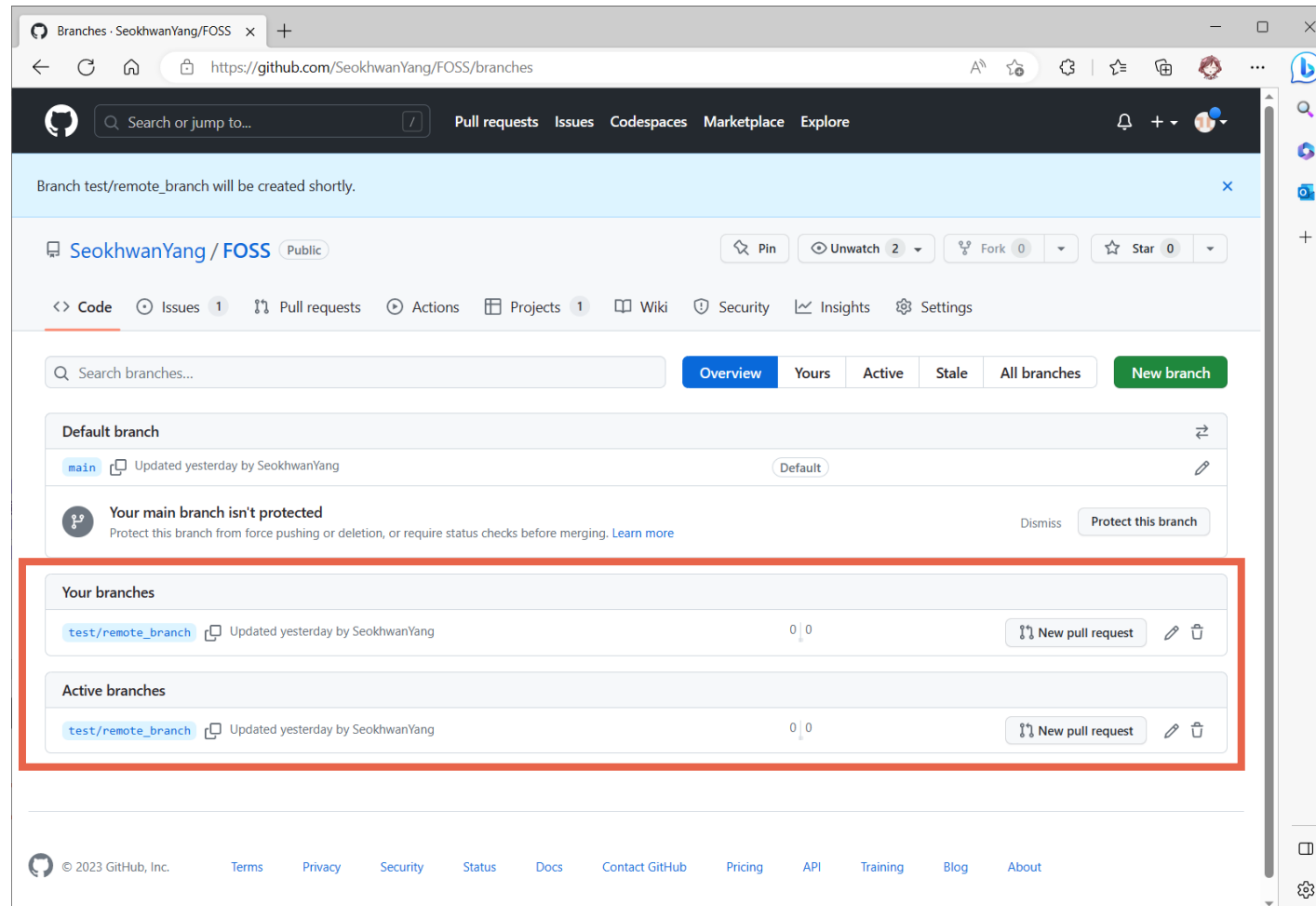
- GitHub 원격 저장소에서 생성 후, 지역 저장소로 가져오기
  - GitHub 사이트에서 해당 메뉴를 이용하여 생성함
- 지역 저장소에서 생성 후, 원격 저장소에 반영하기
  - 로컬 시스템 상에서 Git 명령어를 사용하거나(CLI), GUI도구를 이용하여 생성함



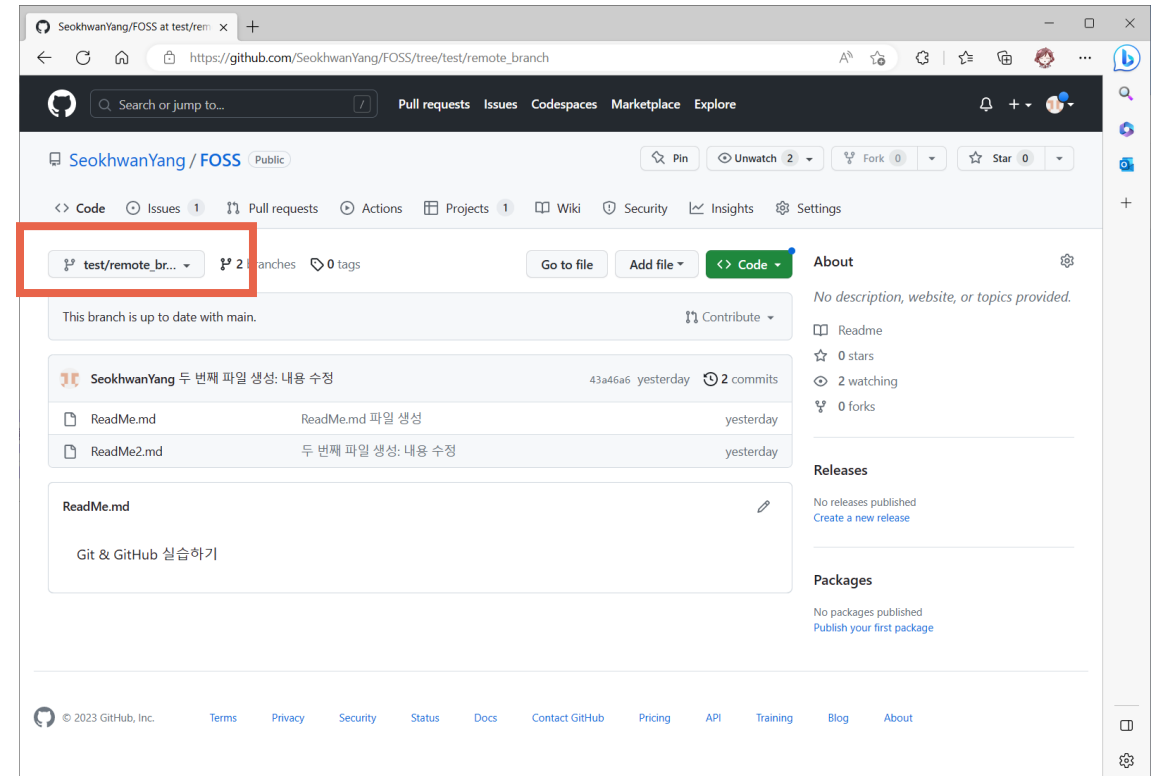
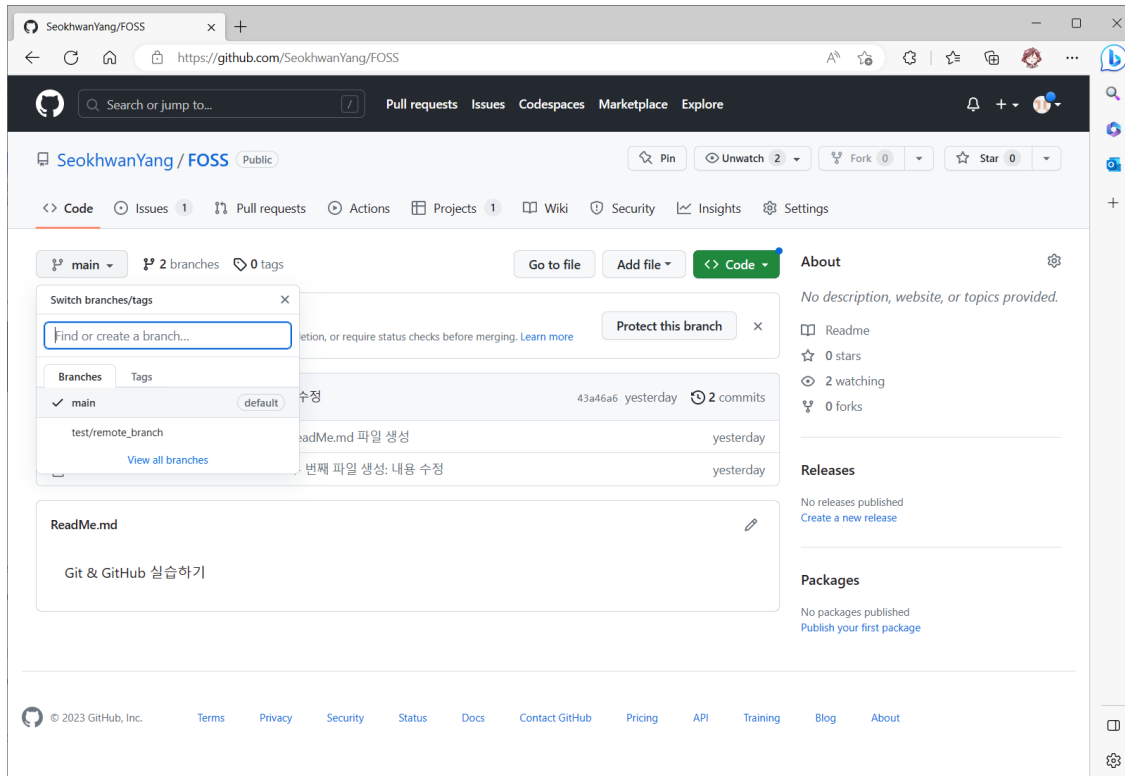
- GitHub 원격 저장소에서 생성 후, 지역 저장소로 가져오기



- GitHub 원격 저장소에서 생성 후, 지역 저장소로 가져오기



- GitHub 원격 저장소에서 생성 후, 지역 저장소로 가져오기



- GitHub 원격 저장소에서 생성 후, 지역 저장소로 가져오기

```
Windows PowerShell
PS C:\AiDALab\Workspace> git remote update
From https://github.com/SeokhwanYang/FOSS
* [new branch]      test/remote_branch -> origin/test/remote_branch
PS C:\AiDALab\Workspace> git branch -a
* main
remotes/origin/main
remotes/origin/test/remote_branch
PS C:\AiDALab\Workspace> git checkout -t origin/test/remote_branch
Switched to a new branch 'test/remote_branch'
branch 'test/remote_branch' set up to track 'origin/test/remote_branch'
PS C:\AiDALab\Workspace> git branch -a
main
* test/remote_branch
remotes/origin/main
remotes/origin/test/remote_branch
PS C:\AiDALab\Workspace>
```

```
PS C:\AiDALab\Workspace> git remote update
From https://github.com/SeokhwanYang/FOSS
* [new branch]      test/remote_branch -> origin/test/remote_branch
```

```
PS C:\AiDALab\Workspace> git branch -a
* main
remotes/origin/main
remotes/origin/test/remote_branch
```

```
PS C:\AiDALab\Workspace> git checkout -t origin/test/remote_branch
Switched to a new branch 'test/remote_branch'
branch 'test/remote_branch' set up to track 'origin/test/remote_branch'.
```

```
PS C:\AiDALab\Workspace> git branch -a
main
* test/remote_branch
remotes/origin/main
remotes/origin/test/remote_branch
```

- **GitHub 원격 저장소에서 생성 후, 지역 저장소로 가져오기**

- **Git Checkout 명령어 주요 옵션**

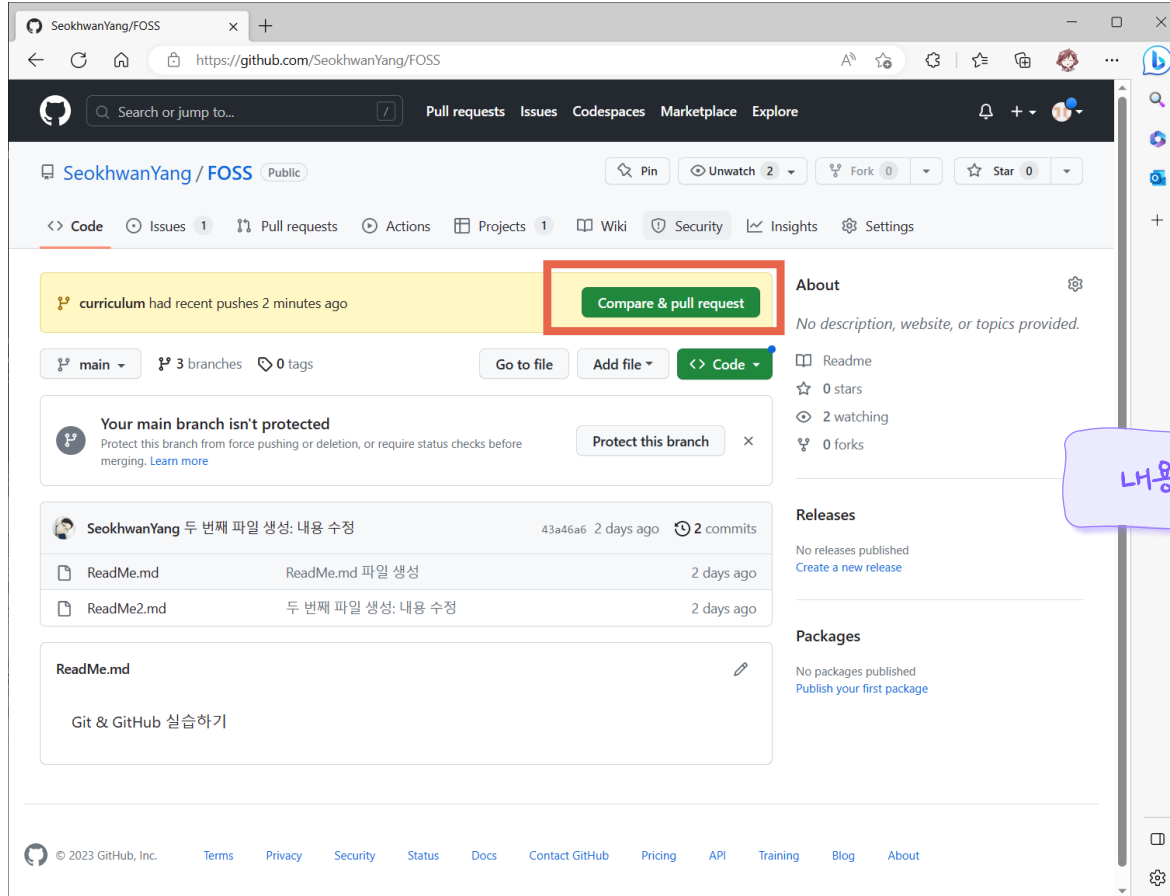
옵션	설명	명령문 예
	사용할 브랜치를 지정함	git checkout
-b	브랜치를 생성하고 사용할 브랜치로 지정함	git checkout -b
-t	원격 저장소에서 생성한 브랜치를 지역 저장소에서 사용할 브랜치로 지정함	git checkout -t

- **Git branch 명령어 주요 옵션**

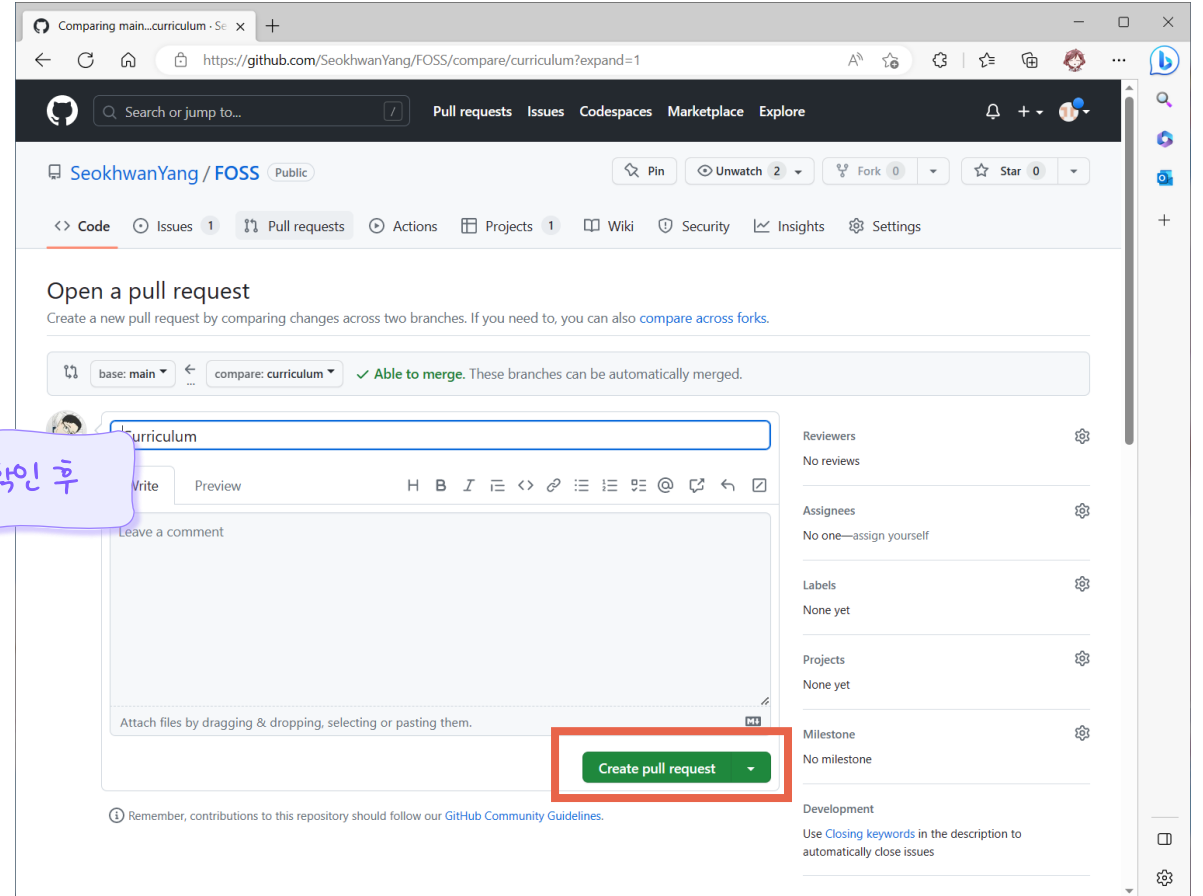
옵션	설명	명령문 예
-a	지역 저장소와 원격 저장소의 브랜치 정보를 함께 보여줌	git branch -a
-d	브랜치 삭제	git branch -d test/delete
-l	지역 저장소의 브랜치 정보를 보여줌 -l 옵션은 생략 가능하며 git branch 명령어만 실행해도 같은 결과를 얻을 수 있음	git branch -l
-r	원격 저장소의 브랜치 정보를 보여줌	git branch -r
-v	지역 저장소의 브랜치 정보를 최신 커밋 내역과 함께 보여줌	git branch -v

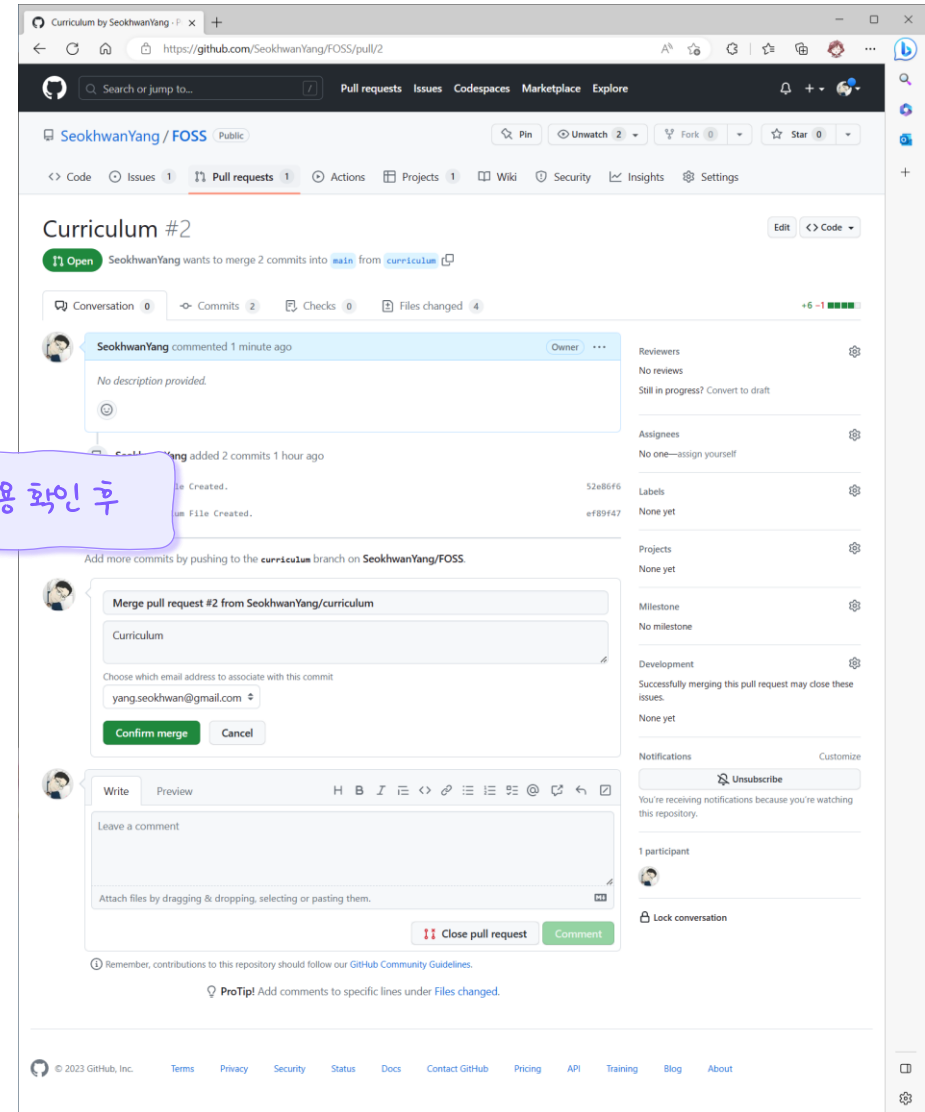
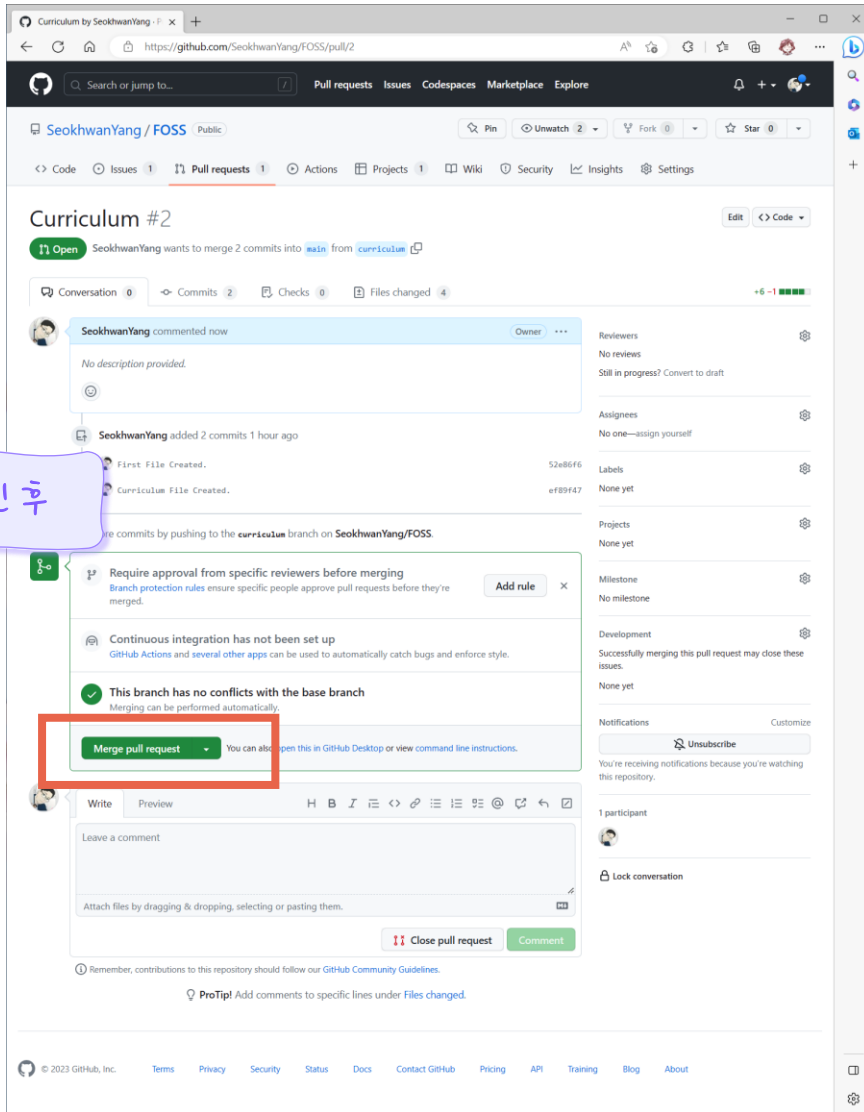
- 지역 저장소에서 생성 후, 원격 저장소에 반영하기
  - 현재의 브랜치 확인하기
    - 지역 저장소: `git branch -l`
    - 원격 저장소: `git branch -r`
    - 지역/원격 모두: `git branch -a`
  - 브랜치 생성하기: `git branch {새로운 브랜치 명}`
  - 브랜치 변경하기: `git checkout {변경할 브랜치 명}`
  - 생성된 브랜치를 원격 저장소에 반영하기
    - `git push origin {새로운 브랜치 명}`

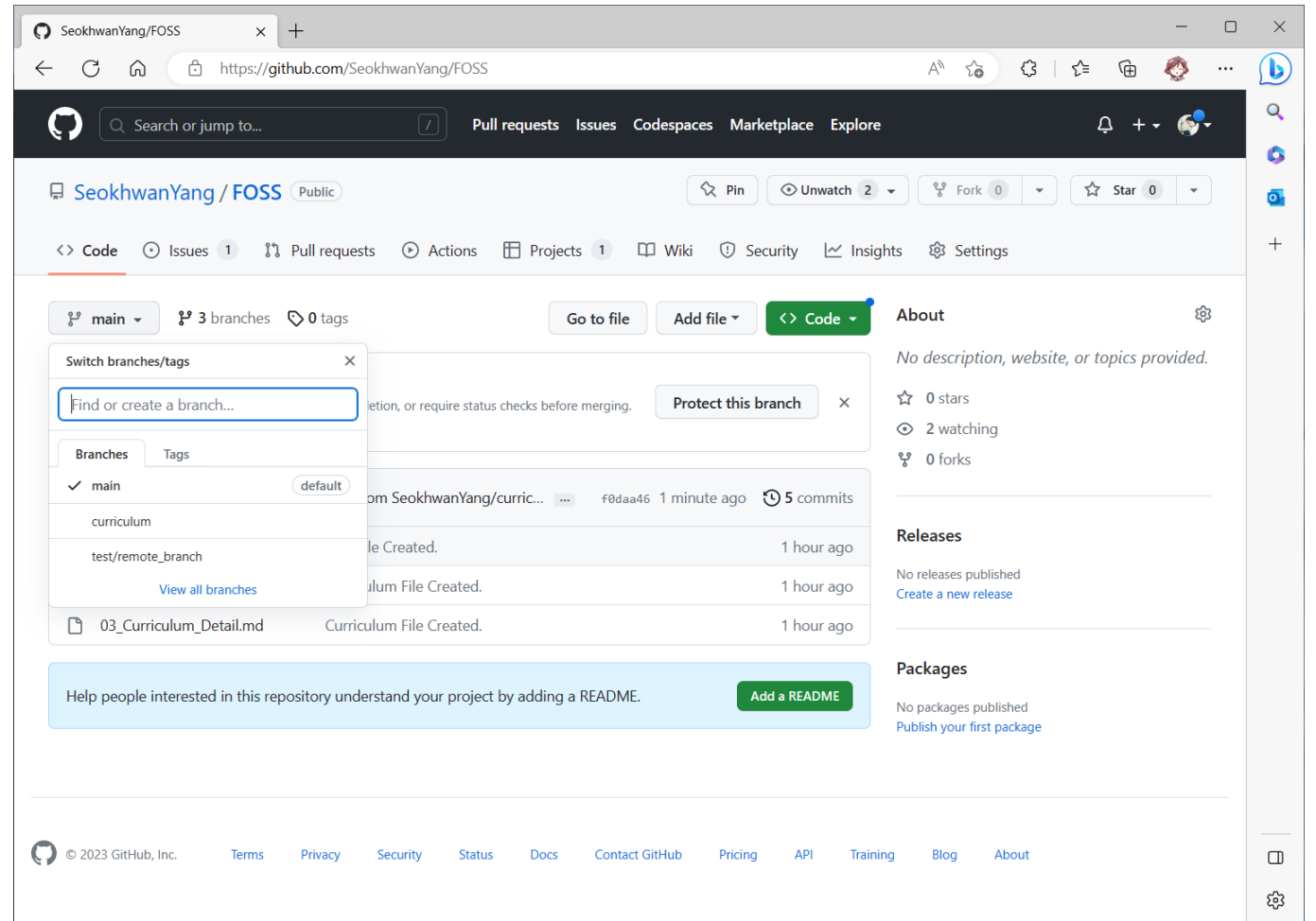
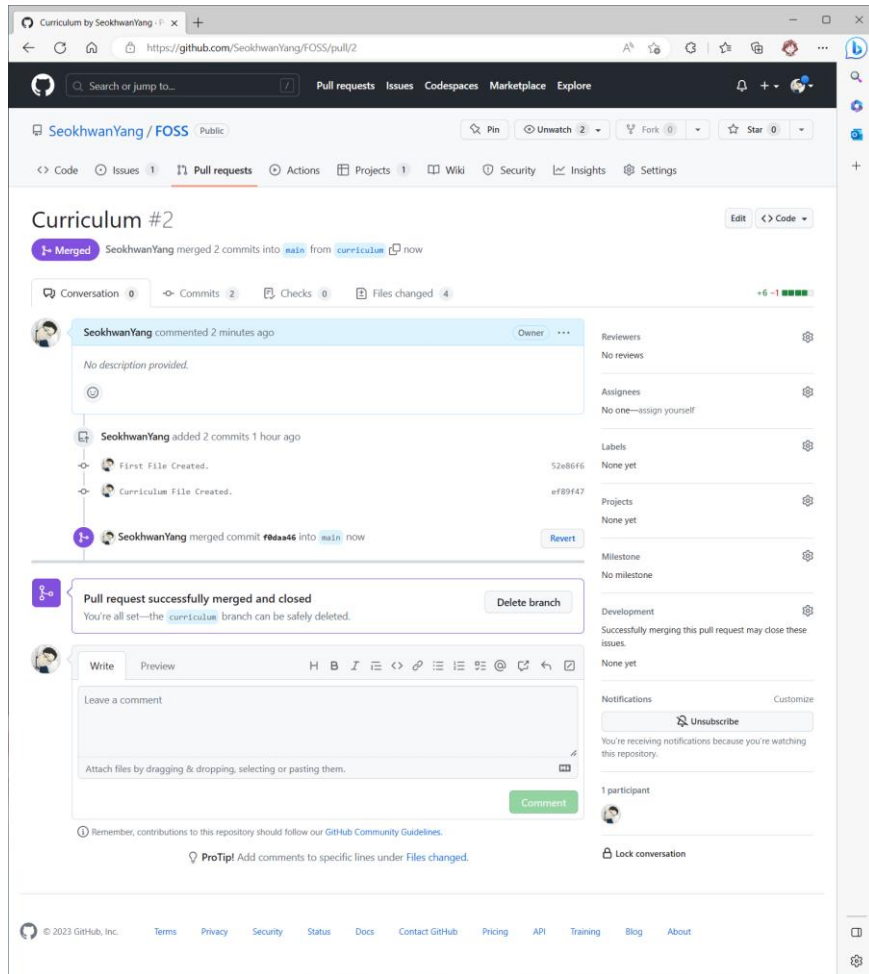
```
Windows PowerShell
PS C:\AiDALab\Workspace\FOSS> git branch -l
* main
PS C:\AiDALab\Workspace\FOSS> git branch -r
origin/main
origin/test/remote_branch
PS C:\AiDALab\Workspace\FOSS> git branch -a
* main
remotes/origin/main
remotes/origin/test/remote_branch
PS C:\AiDALab\Workspace\FOSS> git branch curriculum
PS C:\AiDALab\Workspace\FOSS> git branch -l
curriculum
* main
PS C:\AiDALab\Workspace\FOSS> git checkout curriculum
Switched to branch 'curriculum'
PS C:\AiDALab\Workspace\FOSS> git branch -l
* curriculum
main
PS C:\AiDALab\Workspace\FOSS> git branch -a
* curriculum
main
remotes/origin/main
remotes/origin/test/remote_branch
PS C:\AiDALab\Workspace\FOSS> git push origin curriculum
warning: git-credential-manager-core was renamed to git-credential-manager
warning: see https://aka.ms/gcm/rename for more information
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 8 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (5/5), 702 bytes | 351.00 KiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'curriculum' on GitHub by visiting:
remote: https://github.com/SeokhwanYang/FOSS/pull/new/curriculum
remote:
To https://github.com/SeokhwanYang/FOSS.git
 * [new branch] curriculum -> curriculum
PS C:\AiDALab\Workspace\FOSS> git branch -a
* curriculum
main
remotes/origin/curriculum
remotes/origin/main
remotes/origin/test/remote_branch
PS C:\AiDALab\Workspace\FOSS> |
```



내용 확인 후







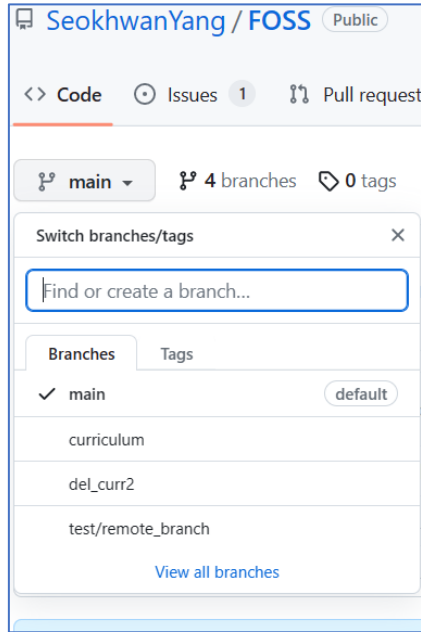
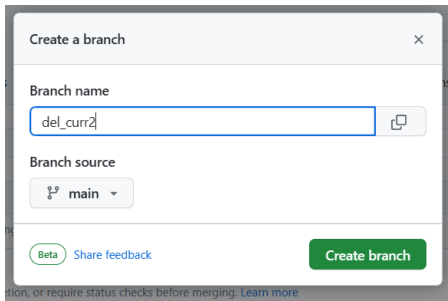
- 브랜치 삭제하기

- 지역 저장소: `git branch -d {삭제할 브랜치 명}`

```
Windows PowerShell
PS C:\AiDALab\Workspace\FOSS> git branch del_curr
PS C:\AiDALab\Workspace\FOSS> git branch -a
* curriculum
  del_curr
  main
  remotes/origin/curriculum
  remotes/origin/main
  remotes/origin/test/remote_branch
PS C:\AiDALab\Workspace\FOSS> git branch -d del_curr
Deleted branch del_curr (was ef89f47).
PS C:\AiDALab\Workspace\FOSS> git branch -a
* curriculum
  main
  remotes/origin/curriculum
  remotes/origin/main
  remotes/origin/test/remote_branch
PS C:\AiDALab\Workspace\FOSS> |
```

## • 브랜치 삭제하기

- 원격 저장소: `git push origin -d {삭제할 브랜치 명}`



```
Windows PowerShell
PS C:\AiDALab\Workspace\FOSS> git branch -a
* curriculum
  main
  remotes/origin/main
  remotes/origin/test/remote_branch
PS C:\AiDALab\Workspace\FOSS> git pull origin
remote: Enumerating objects: 1, done.
remote: Counting objects: 100% (1/1), done.
remote: Total 1 (delta 0), reused 0 (delta 0), pack-reused
Unpacking objects: 100% (1/1), 618 bytes | 206.00 KiB/s, done.
From https://github.com/SeokhwanYang/FOSS
 * [new branch]      del_curr2 -> origin/del_curr2
  43a46a6..f0daa46  main      -> origin/main
You asked to pull from the remote 'origin', but did not specify
a branch. Because this is not the default configured remote
for your current branch, you must specify a branch on the command line.
PS C:\AiDALab\Workspace\FOSS> git branch -a
* curriculum
  main
  remotes/origin/curriculum
  remotes/origin/del_curr2
  remotes/origin/main
  remotes/origin/test/remote_branch
PS C:\AiDALab\Workspace\FOSS> |
```

```
Windows PowerShell
PS C:\AiDALab\Workspace\FOSS> git push origin -d del_curr2
warning: git-credential-manager-core was renamed to git-credential-manager
warning: see https://aka.ms/gcm/rename for more information
To https://github.com/SeokhwanYang/FOSS.git
 - [deleted]      del_curr2
PS C:\AiDALab\Workspace\FOSS> |
```

**협업을 위한 Git/GitHub 기능**

**Merge (병합)**

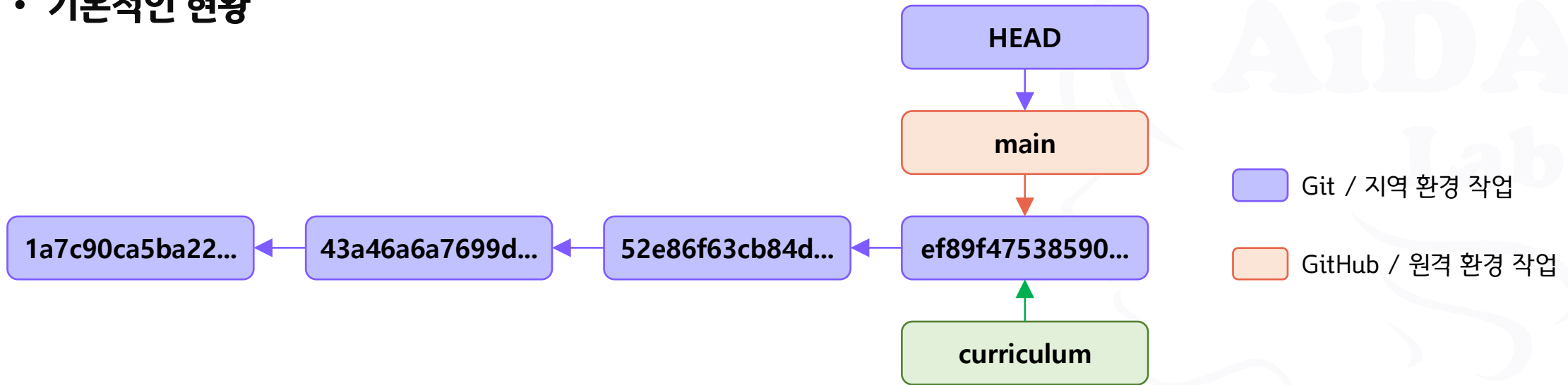
## • 브랜치 병합하기: Merge

- 각 작업자 별, 작업 내용 별로 브랜치를 나눈다면 당연히 각 브랜치를 병합하는 과정도 필요함
- 두 브랜치의 변경 내용을 비교하여 병합 여부 결정 및 수행
- 실습 내용
  - 브랜치 변경: main → curriculum
  - curriculum 브랜치에서 파일 수정 후 Commit & Push
  - main 브랜치와 curriculum 브랜치 내용 비교(파일 변경 상황을 비교)
  - curriculum 브랜치를 main 브랜치에 병합: `git merge {병합할 브랜치}`
  - main 브랜치 내용 확인



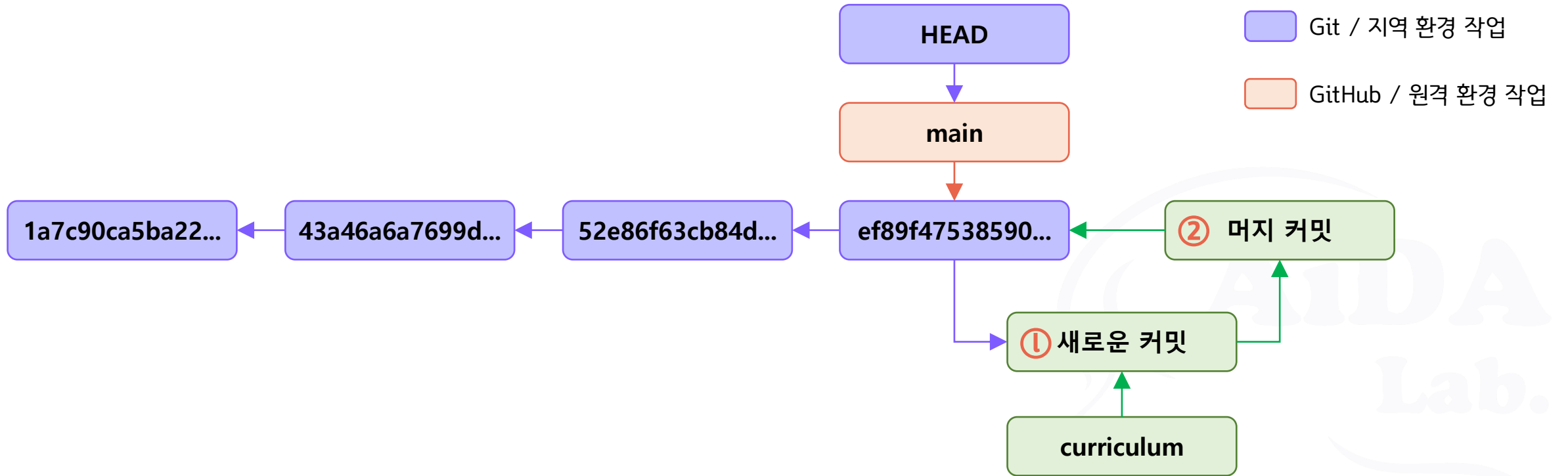
```
Windows PowerShell
PS C:\AiDALab\Workspace\FOSS> git checkout main
Switched to branch 'main'
PS C:\AiDALab\Workspace\FOSS> git log --pretty=oneline --graph
* ef89f475385901efd59cf1191b5ae6aa501cab77 (HEAD -> main, origin/curriculum, curriculum) Curriculum File Created.
* 52e86f63cb84d4172ce1d0240eb5d6aa71cb8bb3 First File Created.
* 43a46a6a7699d0b67ccc5cceb6c71783ef8ddd9f (origin/test/remote_branch) 두 번째 파일 생성: 내용 수정
* 1a7c90ca5ba22bafef4fe7902aab59eb20b2eee88 ReadMe.md 파일 생성
PS C:\AiDALab\Workspace\FOSS> |
```

## • 기본적인 현황



지역 브랜치와 원격 브랜치가 동일한 커밋을 바라보고 있음

- 작업 브랜치에서 새로운 커밋 생성 후 병합 예시



머지 커밋에는 curriculum 브랜치에서 작업한 내용이 반영되어 있음

## • 브랜치 병합 해 보기

```
Windows PowerShell
PS C:\AiDALab\Workspace\FOSS> git branch
  curriculum
* main
PS C:\AiDALab\Workspace\FOSS> git checkout curriculum
Switched to branch 'curriculum'
PS C:\AiDALab\Workspace\FOSS> |
```

curriculum 브랜치에서 새 파일 만들기

```
04_index.html U x
04_index.html > html > footer > p
1 <html>
2
3 <head>
4   <title>강의계획서 ver.1</title>
5   <link rel="stylesheet" href="/stylesheets/style.css">
6 </head>
7
8 <body>
9   <h1>강의계획서</h1>
10  <p>글로벌 오픈소스 프로젝트 참여</p>
11 </body>
12
13 <footer>
14   <p>연락처: 1111-2222</p>
15 </footer>
16
17 </html>
```

```
Windows PowerShell
PS C:\AiDALab\Workspace\FOSS> git status
On branch curriculum
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    04_index.html

nothing added to commit but untracked files present (use "git add" to track)
PS C:\AiDALab\Workspace\FOSS> git add .
PS C:\AiDALab\Workspace\FOSS> git commit -m "Create 04_index.html"
[curriculum baa4009] Create 04_index.html
1 file changed, 17 insertions(+)
 create mode 100644 04_index.html
PS C:\AiDALab\Workspace\FOSS> git log --pretty=oneline --graph
* baa40098ca885a1e8d8bef8c43c63e6e16c4c588 (HEAD -> curriculum) Create 04_index.html
* ef89f475385901efd59cf1191b5ae6aa501cab77 (origin/curriculum, main) Curriculum File Created.
* 52e86f63cb84d4172ce1d0240eb5d6aa71cb8bb3 First File Created.
* 43a46a6a7699d0b67ccc5cceb6c71783ef8ddd9f (origin/test/remote_branch) 두 번째 파일 생성: 내용 수정
* 1a7c90ca5ba22baf4fe7902aab59eb20b2eee88 ReadMe.md 파일 생성
PS C:\AiDALab\Workspace\FOSS> git checkout main
Switched to branch 'main'
PS C:\AiDALab\Workspace\FOSS> git log --pretty=oneline --graph
* ef89f475385901efd59cf1191b5ae6aa501cab77 (HEAD -> main, origin/curriculum) Curriculum File Created.
* 52e86f63cb84d4172ce1d0240eb5d6aa71cb8bb3 First File Created.
* 43a46a6a7699d0b67ccc5cceb6c71783ef8ddd9f (origin/test/remote_branch) 두 번째 파일 생성: 내용 수정
* 1a7c90ca5ba22baf4fe7902aab59eb20b2eee88 ReadMe.md 파일 생성
PS C:\AiDALab\Workspace\FOSS> git merge curriculum
Updating ef89f47..baa4009
Fast-forward
 04_index.html | 17 ++++++
 1 file changed, 17 insertions(+)
 create mode 100644 04_index.html
PS C:\AiDALab\Workspace\FOSS> git log --pretty=oneline --graph
* baa40098ca885a1e8d8bef8c43c63e6e16c4c588 (HEAD -> main, curriculum) Create 04_index.html
* ef89f475385901efd59cf1191b5ae6aa501cab77 (origin/curriculum) Curriculum File Created.
* 52e86f63cb84d4172ce1d0240eb5d6aa71cb8bb3 First File Created.
* 43a46a6a7699d0b67ccc5cceb6c71783ef8ddd9f (origin/test/remote_branch) 두 번째 파일 생성: 내용 수정
* 1a7c90ca5ba22baf4fe7902aab59eb20b2eee88 ReadMe.md 파일 생성
PS C:\AiDALab\Workspace\FOSS>
```

curriculum 브랜치에서 "Create 04\_index.html" 커밋 내용 확인

main 브랜치에서 "Create 04\_index.html" 커밋이 없음을 확인

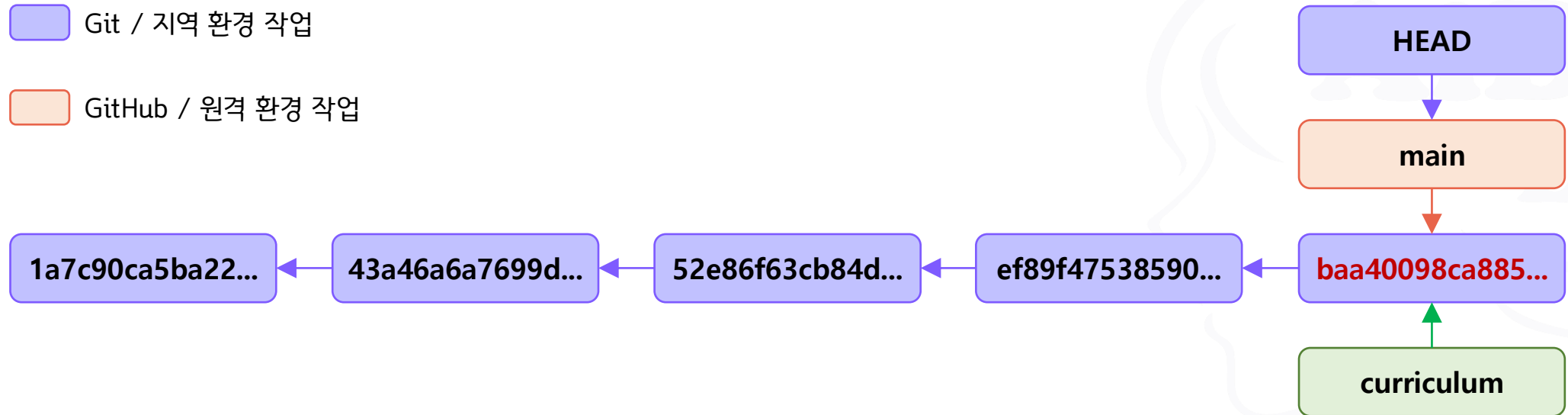
브랜치 병합 (merge) 수행

main 브랜치에서 "Create 04\_index.html" 커밋 내용 확인

```
Windows PowerShell
PS C:\AiDALab\Workspace\FOSS> git log --pretty=oneline --graph
* baa40098ca885a1e8d8bef8c43c63e6e16c4c588 (HEAD -> main, curriculum) Create 04_index.html
* ef89f475385901efd59cf1191b5ae6aa501cab77 (origin/curriculum) Curriculum File Created.
* 52e86f63cb84d4172ce1d0240eb5d6aa71cb8bb3 First File Created.
* 43a46a6a7699d0b67ccc5cceb6c71783ef8ddd9f (origin/test/remote_branch) 두 번째 파일 생성: 내용 수정
* 1a7c90ca5ba22baf4fe7902aab59eb20b2eee88 ReadMe.md 파일 생성
PS C:\AiDALab\Workspace\FOSS> |
```

Git / 지역 환경 작업

GitHub / 원격 환경 작업

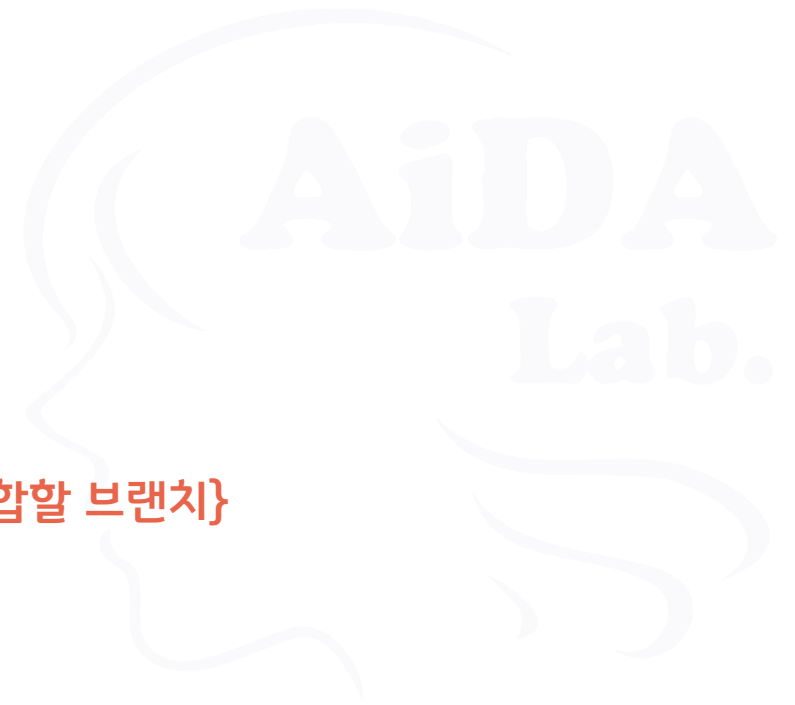


- 병합 커밋 생성: merge commit

- 기준 브랜치에 다른 변경 사항이 존재하는 경우, 새로운 브랜치의 작업 내용을 기준 브랜치에 병합하려면?
- 기준 브랜치와 새로운 작업 브랜치의 변경 내용을 하나로 합치는 작업

- 실습 내용

- main 브랜치에서 04\_index.html 파일 수정
- 브랜치 변경: main → curriculum
- curriculum 브랜치에서 04\_index.html 파일 수정
- main 브랜치와 curriculum 브랜치 내용 비교(파일 변경 상황을 비교)
- curriculum 브랜치를 main 브랜치에 병합: `git merge commit {병합할 브랜치}`
- main 브랜치 내용 확인



# Merge (병합)

```
Windows PowerShell
PS C:\AiDALab\Workspace\FOSS> git branch
  curriculum
* main
PS C:\AiDALab\Workspace\FOSS>
```



```
Visual Studio Code
04_index.html M
<html>
  <head>
    <title>강의계획서 ver.1</title>
    <link rel="stylesheet" href="/stylesheets/style.css">
  </head>
  <body>
    <h1>강의계획서</h1>
    <p>글로벌 오픈소스 프로젝트 참여</p>
    <p>Main Branch 수정</p>
  </body>
  <footer>
    <p>연락처: 1111-2222</p>
  </footer>
</html>
```

```
Windows PowerShell
PS C:\AiDALab\Workspace\FOSS> git status
On branch main
Changes not staged for commit:
  (use "git add <file>.." to update what will be committed)
  (use "git restore <file>.." to discard changes in working directory)
    modified:   04_index.html

no changes added to commit (use "git add" and/or "git commit -a")
PS C:\AiDALab\Workspace\FOSS> git add .
PS C:\AiDALab\Workspace\FOSS> git commit -m "Change Body"
[main 6f89bf0] Change Body
 1 file changed, 1 insertion(+)
PS C:\AiDALab\Workspace\FOSS>
```

```
Windows PowerShell
PS C:\AiDALab\Workspace\FOSS> git checkout curriculum
Switched to branch 'curriculum'
PS C:\AiDALab\Workspace\FOSS> git checkout main
Switched to branch 'main'
PS C:\AiDALab\Workspace\FOSS> git log --pretty=oneline --graph --all
* 6f89bf05864b19e078e243a2a5467999dfb9a886 (HEAD -> main) Change Body
* baa40098ca885a1e8d8bef8c43c63e6e16c4c588 (curriculum) Create 04_index.html
| * f0daa4678688e543c0151cacef5cf48c7f731c30 (origin/main) Merge pull request #2 from SeokhwanYang/curriculum
| |
| | \
| | /
| | /
| | \
* | ef89f475385901efd59cf1191b5ae6aa501cab77 (origin/curriculum) Curriculum File Created.
* | 52e86f63cb84d4172ce1d0240eb5d6aa71cb8bb3 First File Created.
| |
| | /
* | 43a46a6a7699d0b67ccc5cceb6c71783ef8ddd9f (origin/test/remote_branch) 두 번째 파일 생성: 내용 수정
* | 1a7c90ca5ba22bafef4fe7902aab59eb20b2eee88 ReadMe.md 파일 생성
PS C:\AiDALab\Workspace\FOSS> |
```

```
Windows PowerShell
PS C:\AiDALab\Workspace\FOSS> git checkout curriculum
Switched to branch 'curriculum'
M       04_index.html
PS C:\AiDALab\Workspace\FOSS> git branch
* curriculum
  main
PS C:\AiDALab\Workspace\FOSS> |
```

04\_index.html > html

```
1 <html>
2   You, 1시간 전 • Create 04_index.html ...
3 <head>
4   <title>강의계획서 ver.1</title>
5   <link rel="stylesheet" href="/stylesheets/style.css">
6 </head>
7
8 <body>
9   <h1>강의계획서</h1>
10  <p>글로벌 오픈소스 프로젝트 참여</p>
11 </body>
12
13 <footer>
14   <p>연락처: 1111-2222</p>
15 </footer>
16
17 </html>
```

main 브랜치에서 변경된 내용이 없음

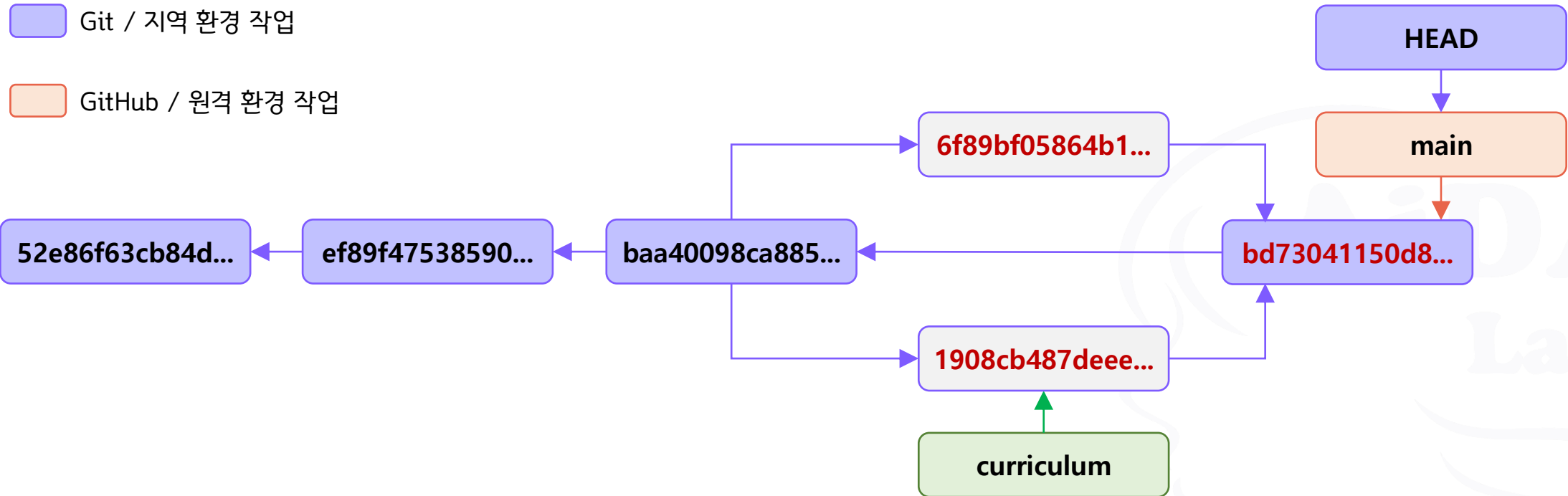
```
04_index.html M X
04_index.html > html
You, 1초 전 | 1 author (You)
1 <html>
2
3 <head>
4   <title>강의계획서 ver.1</title>
5   <link rel="stylesheet" href="/stylesheets/style.css">
6 </head>
7
8 <body>
9   <h1>강의계획서</h1>
10  <p>글로벌 오픈소스 프로젝트 참여</p>
11 </body>
12 You, 1시간 전 • Create 04_index.html ...
13 <footer>
14   <p>연락처: 1111-2222</p>
15   <p>주소: 경기도 하남시 XXXX XXXX</p>
16 </footer>
17
18 </html>
```

```
Windows PowerShell
PS C:\AiDALab\Workspace\FOSS> git status
On branch curriculum
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   04_index.html

no changes added to commit (use "git add" and/or "git commit -a")
PS C:\AiDALab\Workspace\FOSS> git add .
PS C:\AiDALab\Workspace\FOSS> git commit -m "Change Footer"
[curriculum 1908cb4] Change Footer
 1 file changed, 1 insertion(+)
PS C:\AiDALab\Workspace\FOSS> |
```



```
Windows PowerShell
PS C:\AiDALab\Workspace\FOSS> git checkout main
Switched to branch 'main'
PS C:\AiDALab\Workspace\FOSS> git merge curriculum
Auto-merging 04_index.html
Merge made by the 'ort' strategy.
 04_index.html | 1 +
 1 file changed, 1 insertion(+)
PS C:\AiDALab\Workspace\FOSS> git log --pretty=oneline --graph --all
*   bd73041150d87c121ca0fb0c227a867133acab43 (HEAD -> main) Merge branch 'curriculum'
|
| * 1908cb487deee385bed2b2b624639dfb408bf4e2 (curriculum) Change Footer ←
| * 6f89bf05864b19e078e243a2a5467999dfb9a886 Change Body ←
|/
* baa40098ca885a1e8d8bef8c43c63e6e16c4c588 Create 04_index.html
| * f0daa4678688e543c0151cacef5cf48c7f731c30 (origin/main) Merge pull request #2 from SeokhwanYang/curriculum
|/
|
| * ef89f475385901efd59cf1191b5ae6aa501cab77 (origin/curriculum) Curriculum File Created. ←
| * 52e86f63cb84d4172ce1d0240eb5d6aa71cb8bb3 First File Created.
|/
* 43a46a6a7699d0b67ccc5cceb6c71783ef8ddd9f (origin/test/remote_branch) 두 번째 파일 생성: 내용 수정
* 1a7c90ca5ba22baf4fe7902aab59eb20b2eee88 README.md 파일 생성
PS C:\AiDALab\Workspace\FOSS> |
```

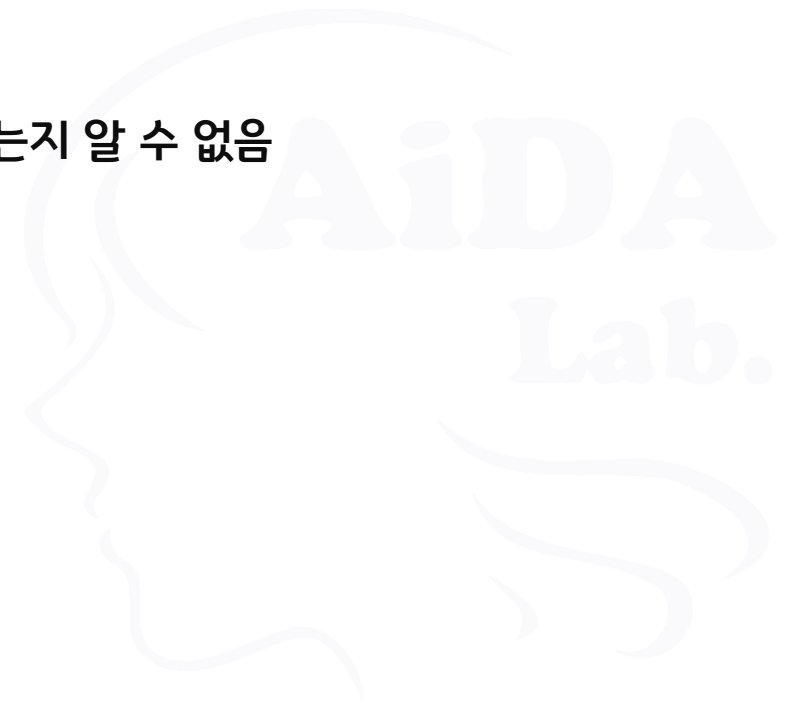


**협업을 위한 Git/GitHub 기능**

**Resolve Conflict (충돌 해결)**

## • 충돌 해결하기

- 여러 개발자가 프로젝트를 진행하다 보면 동일한 코드를 제각각 수정하는 경우가 발생함
- 이런 경우에 Merge(병합) 시 충돌(Conflict) 발생
- 충돌이란 Git이 자동으로 Merge를 수행할 수 없는 상황을 말함
  - 동일한 코드가 수정되었기 때문에 어떤 내용을 최종적으로 반영해야 하는지 알 수 없음
- 이 경우에는 사람이 직접 조정해 주어야 함



# Resolve Conflict (충돌 해결)

```
Windows PowerShell
PS C:\AiDALab\Workspace\FOSS> git branch
  curriculum
* main
PS C:\AiDALab\Workspace\FOSS> |
```

```
04_index.html
04_index.html > html > footer > p
You, 1초 전 | 1 author (You)
1 <html>
2
3 <head>
4   <title>강의계획서 ver.1</title>
5   <link rel="stylesheet" href="/stylesheets/style.css">
6 </head>
7
8 <body>
9   <h1>강의계획서</h1>
10  <p>글로벌 오픈소스 프로젝트 참여</p>
11  <p>Main Branch 수정</p>
12 </body>
13
14 <footer>
15   <p>연락처: 3333-4444</p>
16   <p>주소: 경기도 하남시 XXXX XXXX</p>
17 </footer>
18
19 </html>
```

<p>연락처: 1111-2222</p> → <p>연락처: 3333-4444</p> 로 수정

```
Windows PowerShell
PS C:\AiDALab\Workspace\FOSS> git status
On branch main
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   04_index.html

no changes added to commit (use "git add" and/or "git commit -a")
PS C:\AiDALab\Workspace\FOSS> git add .
PS C:\AiDALab\Workspace\FOSS> git commit -m "연락처를 3333-4444로 변경"
[main 710cf83] 연락처를 3333-4444로 변경
 1 file changed, 2 insertions(+), 1 deletion(-)
PS C:\AiDALab\Workspace\FOSS> |
```

# Resolve Conflict (충돌 해결)

```
Windows PowerShell
PS C:\AiDALab\Workspace\FOSS> git checkout curriculum
Switched to branch 'curriculum'
PS C:\AiDALab\Workspace\FOSS> |
```

```
04_index.html
1 <html>
2
3 <head>
4   <title>강의계획서 ver.1</title>
5   <link rel="stylesheet" href="/stylesheets/style.css">
6 </head>
7
8 <body>
9   <h1>강의계획서</h1>
10  <p>글로벌 오픈소스 프로젝트 참여</p>
11 </body>
12
13 <footer>
14   <p>연락처: 5555-6666</p>
15   <p>주소: 경기도 하남시 XXXX XXXX</p>
16 </footer>
17
18 </html>
```

<p>연락처: 1111-2222</p> → <p>연락처: 5555-6666</p> 로 수정

```
Windows PowerShell
PS C:\AiDALab\Workspace\FOSS> git status
On branch curriculum
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   04_index.html

no changes added to commit (use "git add" and/or "git commit -a")
PS C:\AiDALab\Workspace\FOSS> git add .
PS C:\AiDALab\Workspace\FOSS> git commit -m "연락처를 5555-6666으로 변경"
[curriculum 820e5d7] 연락처를 5555-6666으로 변경
1 file changed, 1 insertion(+), 1 deletion(-)
PS C:\AiDALab\Workspace\FOSS> |
```

```
Windows PowerShell
PS C:\AiDALab\Workspace\FOSS> git checkout main
Switched to branch 'main'
PS C:\AiDALab\Workspace\FOSS> git merge curriculum
Auto-merging 04_index.html
CONFLICT (content): Merge conflict in 04_index.html
Automatic merge failed; fix conflicts and then commit the result.
PS C:\AiDALab\Workspace\FOSS> |
```

충돌 발생 -> 충돌로 인한 Merge(병합) 실패

# Resolve Conflict (충돌 해결)

Visual Studio Code interface showing a merge conflict in the file `04_index.html`. The editor displays conflicting changes from the current branch (HEAD) and the incoming branch (curriculum).

**Conflict Content:**

```
1 <html>
2
3 <head>
4   <title>강의계획서 ver.1</title>
5   <link rel="stylesheet" href="/stylesheets/style.css">
6 </head>
7
8 <body>
9   <h1>강의계획서</h1>
10  <p>글로벌 오픈소스 프로젝트 참여</p>
11  <p>Main Branch 수정</p>
12 </body>
13
14 <footer>
15 <<<<<< HEAD (현재 변경 사항)
16   <p>연락처: 3333-4444</p>
17 =====
18   <p>연락처: 5555-6666</p>
19 >>>>>> curriculum (수신 변경 사항)
20   <p>주소: 경기도 하남시 XXXX XXXX</p>
21 </footer>
22
23 </html>
```

**Handwritten Notes:**

- 파일을 열어보면 충돌 내용이 기록되어 있음 어떻게 할 것인지 결정하여야 함
- 최종적으로 반영할 부분만 남기고 나머지 부분을 삭제한 후 저장

**Buttons:**

- 병합 편집기에서 확인

```
<html>
<head>
  <title>강의계획서 ver.1</title>
  <link rel="stylesheet" href="/stylesheets/style.css">
</head>
<body>
  <h1>강의계획서</h1>
  <p>글로벌 오픈소스 프로젝트 참여</p>
  <p>Main Branch 수정</p>
</body>
<footer>
  <p>연락처: 3333-4444</p>
  <p>주소: 경기도 하남시 XXXX XXXX</p>
</footer>
</html>
```

최종적으로 반영할 부분만 남기고 나머지 부분을 삭제한 후 저장

저장 후 다시 커밋 생성

병합 편집기에서 확인

```
Windows PowerShell
PS C:\AiDALab\Workspace\FOSS> git status
On branch main
You have unmerged paths.
  (fix conflicts and run "git commit")
  (use "git merge --abort" to abort the merge)

Unmerged paths:
  (use "git add <file>..." to mark resolution)
    both modified:   04_index.html

no changes added to commit (use "git add" and/or "git commit -a")
PS C:\AiDALab\Workspace\FOSS> git add .
PS C:\AiDALab\Workspace\FOSS> git commit -m "충돌 해결"
[main ef5bf60] 충돌 해결
PS C:\AiDALab\Workspace\FOSS> git log --pretty=oneline --graph
*   ef5bf60b615b72493fb6bca2b430bb964895cbc0 (HEAD -> main) 충돌 해결
|
| * 820e5d78034141e442662c0190aa685d04ab5f0b (curriculum) 연락처를 5555-6666으로 변경
| * 1908cb487deee385bed2b2b624639dfb408bf4e2 Change Footer
| * | 710cf83d90f66b7bbd2a94049139d5675bb7f606 연락처를 3333-4444로 변경
| * | 6f89bf05864b19e078e243a2a5467999dfb9a886 Change Body
|/
|
| * baa40098ca885a1e8d8bef8c43c63e6e16c4c588 Create 04_index.html
| * ef89f475385901efd59cf1191b5ae6aa501cab77 (origin/curriculum) Curriculum File Created.
| * 52e86f63cb84d4172ce1d0240eb5d6aa71cb8bb3 First File Created.
| * 43a46a6a7699d0b67ccc5cceb6c71783ef8ddd9f (origin/test/remote_branch) 두 번째 파일 생성: 내용 수정
| * 1a7c90ca5ba22baf4fe7902aab59eb20b2eee88 ReadMe.md 파일 생성
PS C:\AiDALab\Workspace\FOSS> |
```

충돌로 인해 병합되지 못했던 내용을 알려줌

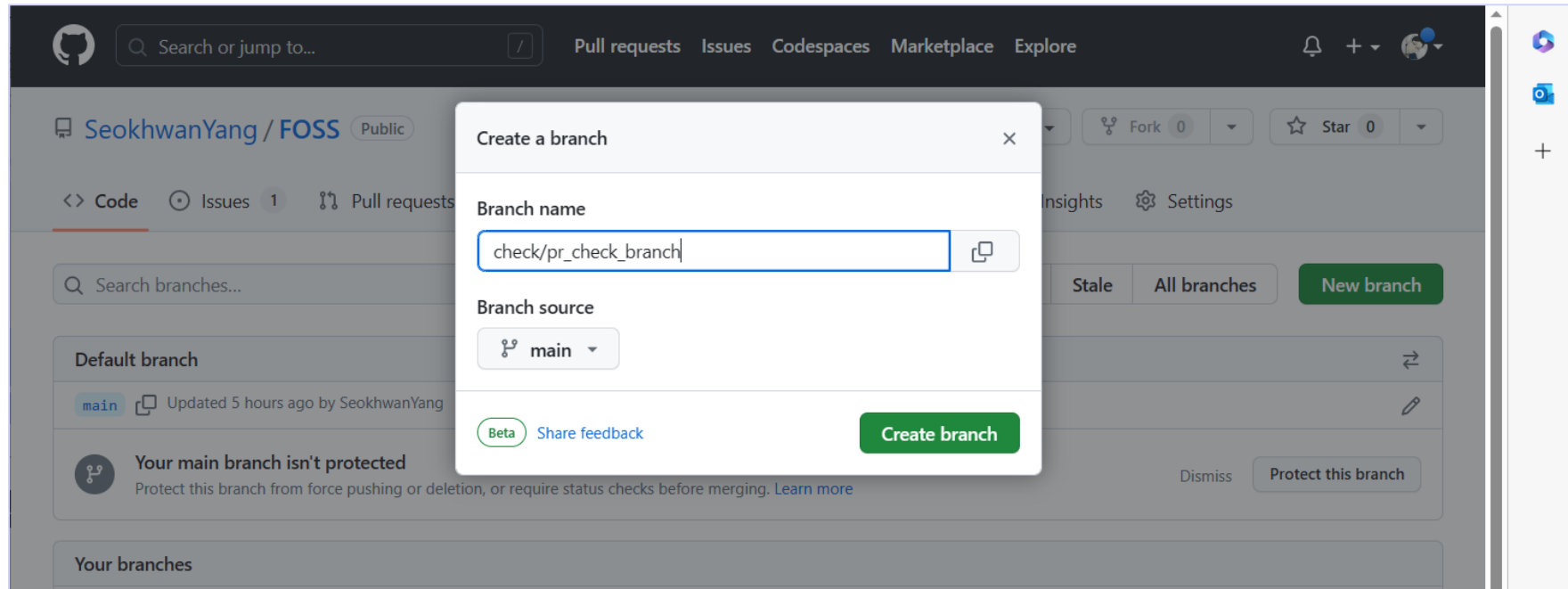
충돌 해결 후 병합된 결과를 보여줌

**협업을 위한 Git/GitHub 기능**

**Pull Requests 요청 및 처리**

## • Pull Requests

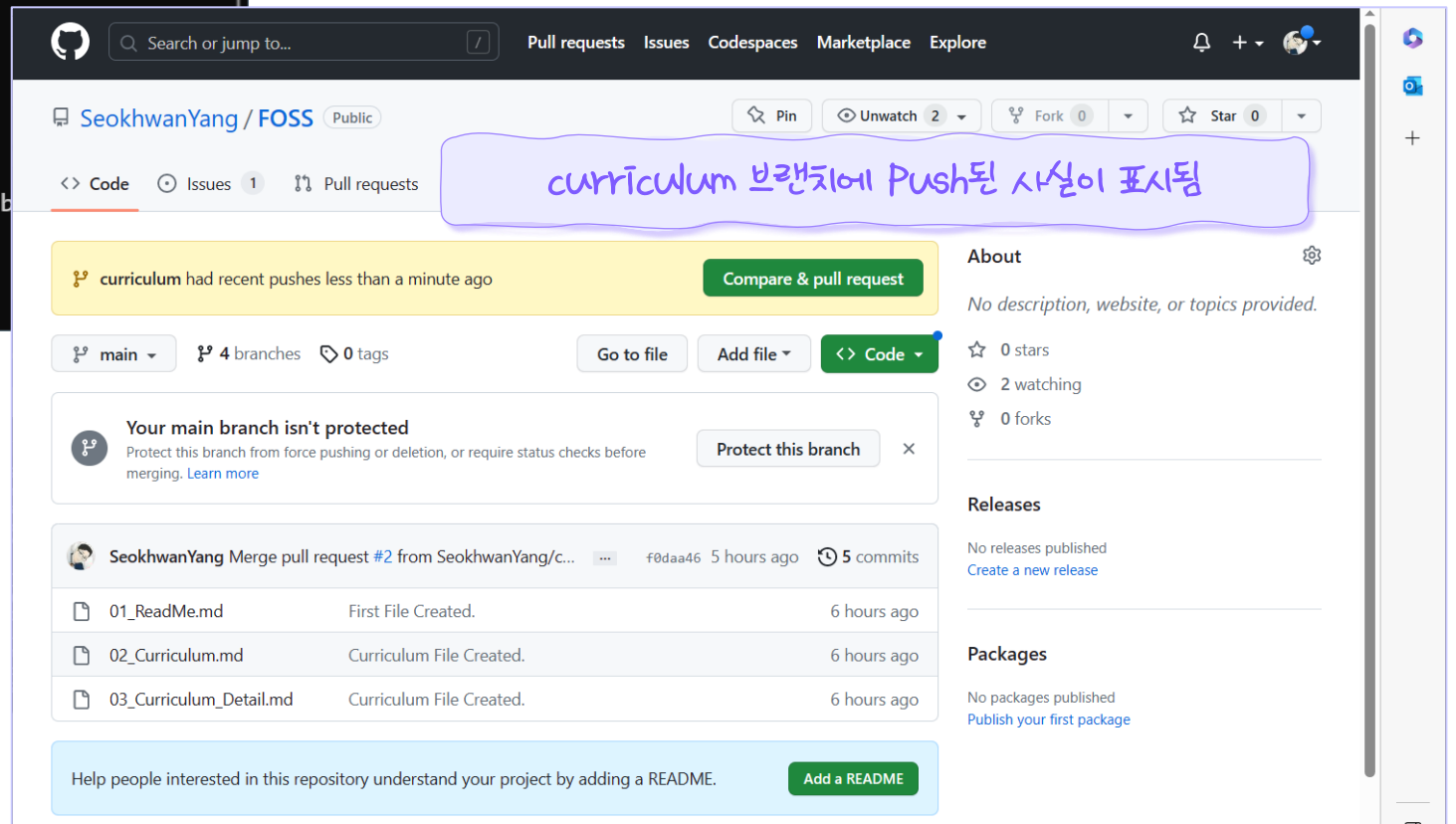
- 병합을 하기 전, 작업 브랜치의 변경 내용을 협업 개발자들에게 공유하고 리뷰를 받는 과정
  - 하나의 프로젝트에서 여러 개발자가 함께 작업을 하고 있는데
  - 내가 코드를 변경했다고 그대로 병합해버리면 매우 위험하므로 (모든 작업이 다 망가질 수 있음)
  - 내가 변경한 내용이 다른 개발자들의 작업 내용에 어떻게 영향을 주는지 빠짐없이 검토해 달라는 요청임
- 실습 내용
  - GitHub에 별도의 브랜치 생성
  - 내가 작업한 내용을 GitHub에서 내가 사용하는 브랜치에 push
  - Pull Requests 작성
  - Pull Requests 검토 후 반영



Your branches			
check/pr_check_branch	Updated 5 hours ago by SeokhwanYang	0   0	New pull request
curriculum	Updated 6 hours ago by SeokhwanYang	1   0	#2 Merged
test/remote_branch	Updated 2 days ago by SeokhwanYang	3   0	New pull request

- GitHub에 Push

```
Windows PowerShell
PS C:\AiDALab\Workspace\FOSS> git push origin curriculum
warning: git-credential-manager-core was renamed to git-credential-manager
warning: see https://aka.ms/gcm/rename for more information
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 8 threads
Compressing objects: 100% (9/9), done.
Writing objects: 100% (9/9), 1.01 KiB | 1.01 MiB/s, done.
Total 9 (delta 5), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (5/5), completed with 1 local object.
To https://github.com:SeokhwanYang/FOSS.git
ef89f47..820e5d7 curriculum -> curriculum
PS C:\AiDALab\Workspace\FOSS>
```



- 잘 반영되었나 확인

The screenshot displays the GitHub interface for the repository 'SeokhwanYang / FOSS'. At the top, there is a search bar and navigation links for Pull requests, Issues, Codespaces, Marketplace, and Explore. Below this, the repository name and public status are shown, along with buttons for Pin, Unwatch (2), Fork (0), and Star (0). The main navigation bar includes Code, Issues (1), Pull requests, Actions, Projects (1), Wiki, Security, Insights, and Settings.

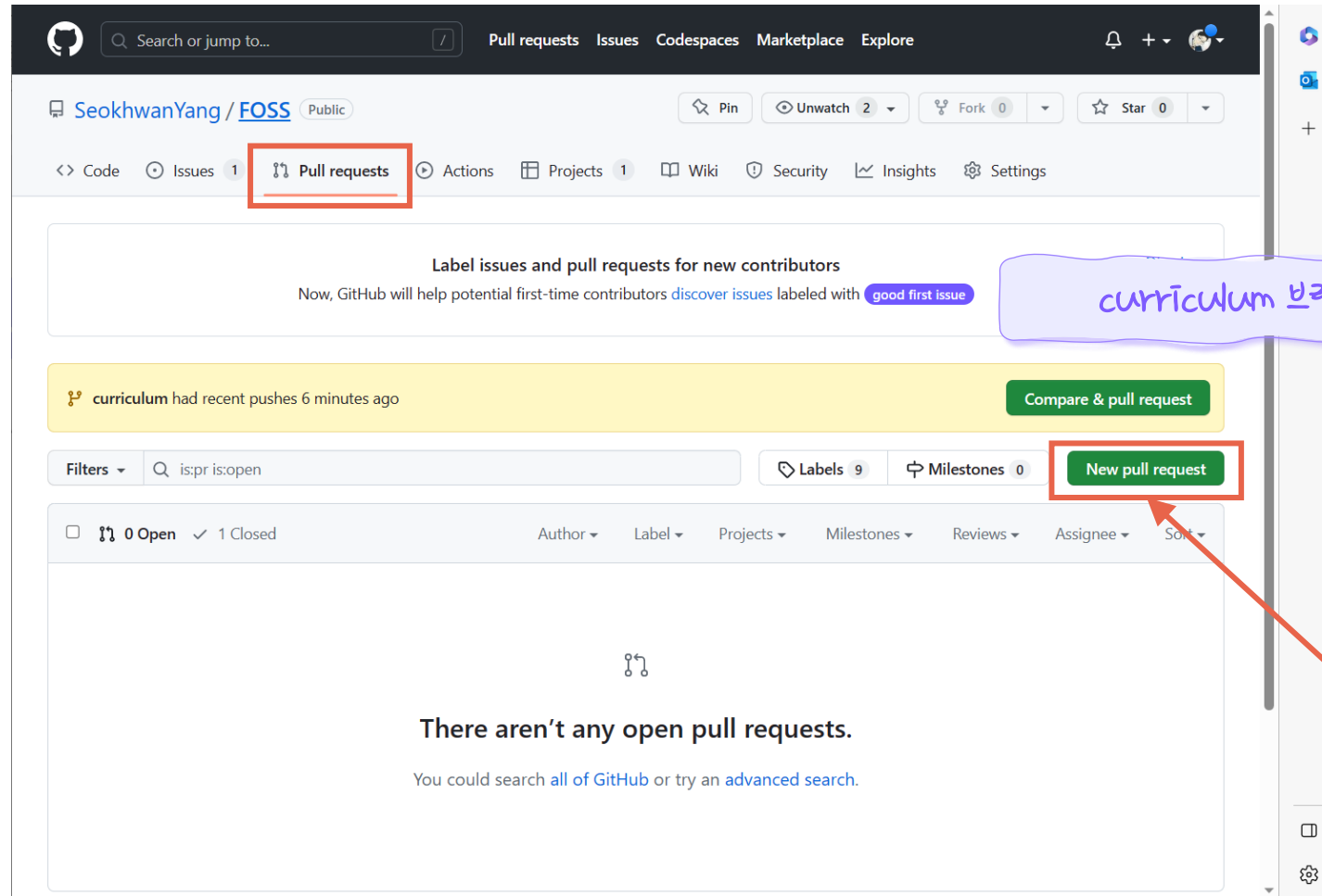
A yellow notification bar indicates that 'curriculum' had recent pushes 3 minutes ago, with a 'Compare & pull request' button. Below this, the repository name 'SeokhwanYang / FOSS' and the branch 'curriculum' are highlighted with a red box. The branch status shows '4 branches' and '0 tags', with buttons for 'Go to file', 'Add file', and 'Code'. A status bar indicates 'This branch is 3 commits ahead, 1 commit behind main.' with a 'Contribute' button.

The commit history table shows the following entries:

Commit	Message	Time
820e5d7	연락처를 5555-6666으로 변경	42 minutes ago (7 commits)
	01_ReadMe.md	First File Created. (6 hours ago)
	02_Curriculum.md	Curriculum File Created. (6 hours ago)
	03_Curriculum_Detail.md	Curriculum File Created. (6 hours ago)
	04_index.html	연락처를 5555-6666으로 변경 (42 minutes ago)

The '04\_index.html' row is highlighted with a red box. The right sidebar contains sections for 'About' (No description, website, or topics provided), 'Releases' (No releases published), and 'Packages' (No packages published).

- Pull Request 생성



curriculum 브랜치에 Push된 사실이 표시됨



여기를 눌러서 Pull Request 생성

The screenshot shows the GitHub 'Compare changes' interface for the repository 'SeokhwanYang / FOSS'. The 'base' branch is set to 'main' and the 'compare' branch is also set to 'main'. A dropdown menu is open for 'Choose a head ref', showing a search bar and a list of branches: 'main' (selected), 'check/pr\_check\_branch', 'curriculum', and 'test/remote\_branch'. A purple callout box contains the Korean text: '기준 브랜치는 main, 비교할 브랜치는 curriculum 브랜치로 설정' (Set the base branch to main and the branch to compare to curriculum). Below the dropdown is an 'Example comparisons' table.

Example comparisons	
curriculum	52 minutes ago
check/pr_check_branch	5 hours ago
test/remote_branch	2 days ago
main@{1day}...main	24 hours ago

Comparing changes

Choose two branches to see what's changed or to start a new pull request. If you need to, you can also [compare across forks](#).

base: main ← compare: curriculum ✓ Able to merge. These branches can be automatically merged.

Discuss and review the changes in this comparison with others. [Learn about pull requests](#) [Create pull request](#)

3 commits 1 file changed 1 contributor

Commits on May 24, 2023

- Create 04\_index.html  
SeokhwanYang committed 2 hours ago
- Change Footer  
SeokhwanYang committed 1 hour ago
- 연락처를 5555-6666으로 변경  
SeokhwanYang committed 1 hour ago

Showing 1 changed file with 18 additions and 0 deletions. Split Unified

```
18 04_index.html
...  ...  @@ -0,0 +1,18 @@
1  + <html>
2  +
3  + <head>
4  +   <title>강의계획서 ver.1</title>
5  +   <link rel="stylesheet" href="/stylesheets/style.css">
6  + </head>
7  +
8  + <body>
9  +   <h1>강의계획서</h1>
10 +   <p>글로벌 오픈소스 프로젝트 참여</p>
11 + </body>
12 +
13 + <footer>
14 +   <p>연락처: 5555-6666</p>
15 +   <p>주소: 경기도 하남시 XXXX XXXX</p>
16 + </footer>
17 +
18 + </html>
```

기준 브랜치인 main과 다른 내용(변경 내역)을 모두 표시함

Comparing changes

Choose two branches to see what's changed or to start a new pull request. If you need to, you can also [compare across forks](#).

base: main ← compare: curriculum ✓ Able to merge. These branches can be automatically merged.

Discuss and review the changes in this comparison with others. [Learn about pull requests](#) **Create pull request**

3 commits 1 file changed 1 contributor

Commits on May 24, 2023

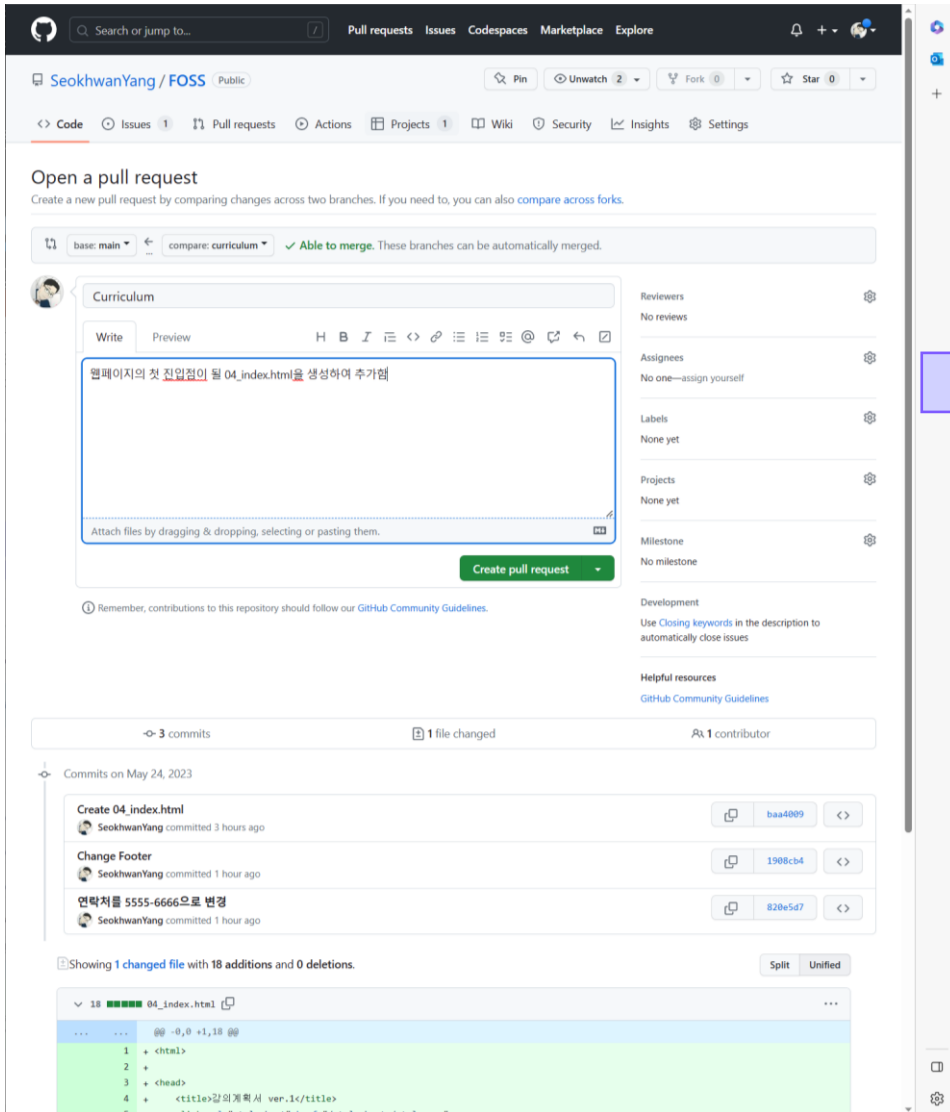
- Create 04\_index.html  
SeokhwanYang committed 2 hours ago
- Change Footer  
SeokhwanYang committed 1 hour ago
- 연락처를 5555-6666으로 변경  
SeokhwanYang committed 1 hour ago

Showing 1 changed file with 18 additions and 0 deletions. Split Unified

```
18 04_index.html
@@ -0,0 +1,18 @@
1 + <html>
2 +
3 + <head>
4 +   <title>강의계획서 ver.1</title>
5 +   <link rel="stylesheet" href="/stylesheets/style.css">
6 + </head>
```

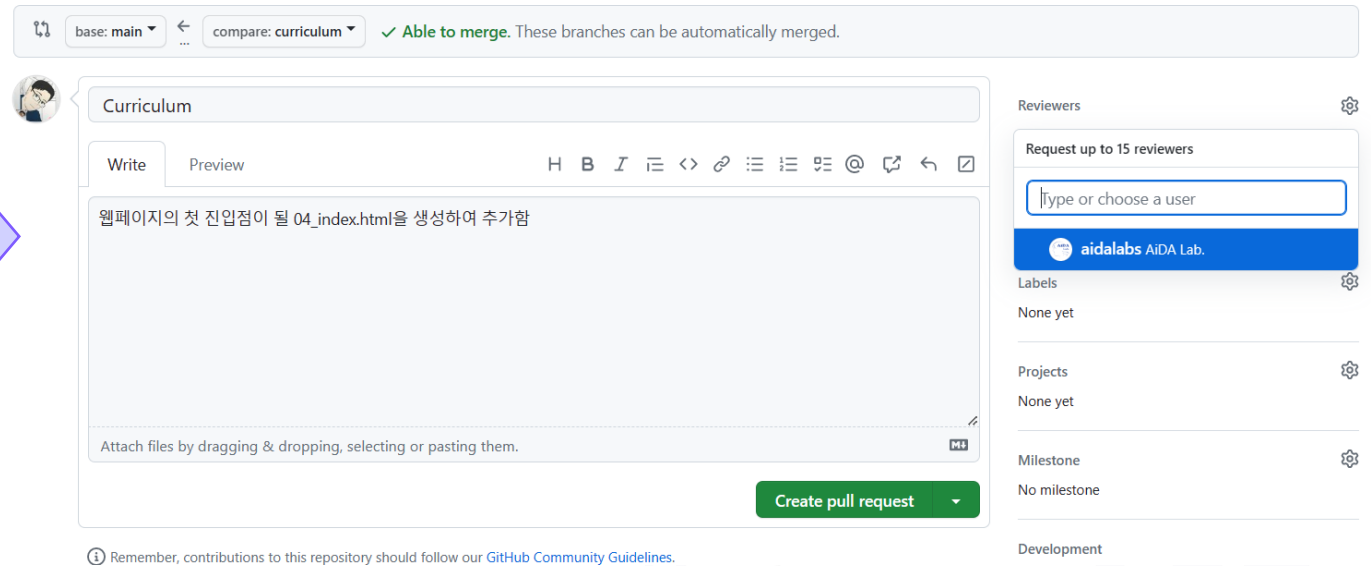
변경 내용 확인 후 Pull Request를 생성함





## Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#).



Pull Request의 내용을 작성하고 누가 Review를 할 것인지를 지정한 후 생성을 완료함

원격저장소에 Collaborator로 등록된 사람만 지정 가능

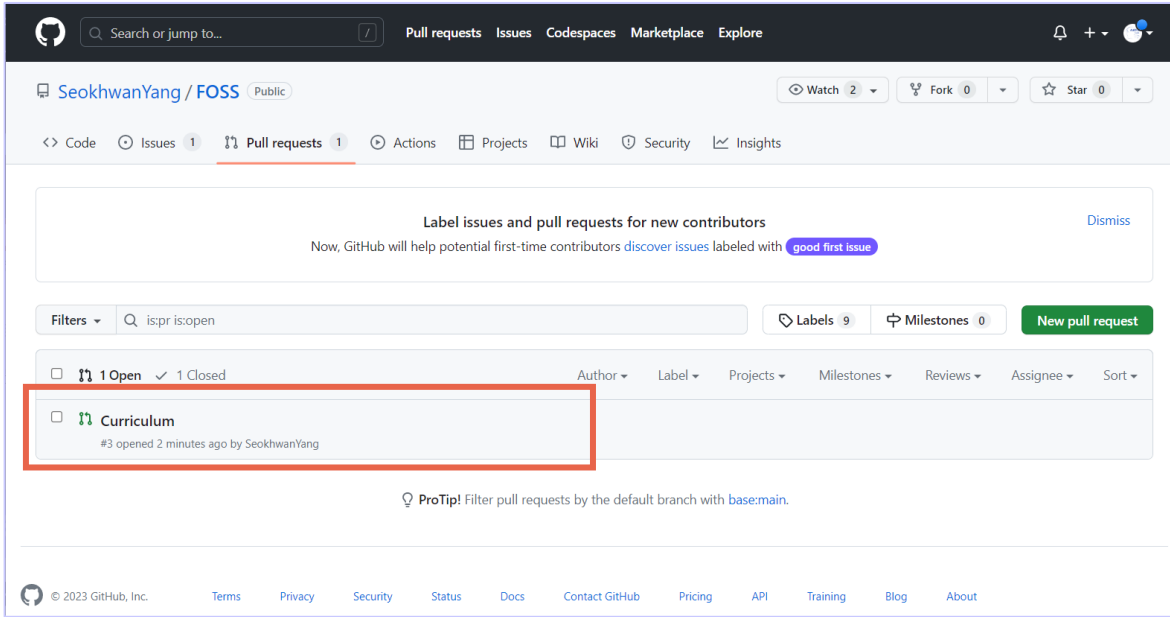
# Pull Requests 요청 및 처리

The screenshot shows a GitHub Pull Request for the repository 'SeokhwanYang / FOSS'. The pull request is titled 'Curriculum #3' and is currently open. The main content area shows a commit history with three commits: 'Create 04\_index.html', 'Change Footer', and '연락처를 5555-6666으로 변경'. A comment from the author 'SeokhwanYang' is visible, stating '웹페이지의 첫 진입점이 될 04\_index.html을 생성하여 추가함'. The right sidebar shows the 'Reviewers' section with 'aidalabs' listed as a reviewer. The bottom section shows a 'Review requested' status and a 'Merge pull request' button.

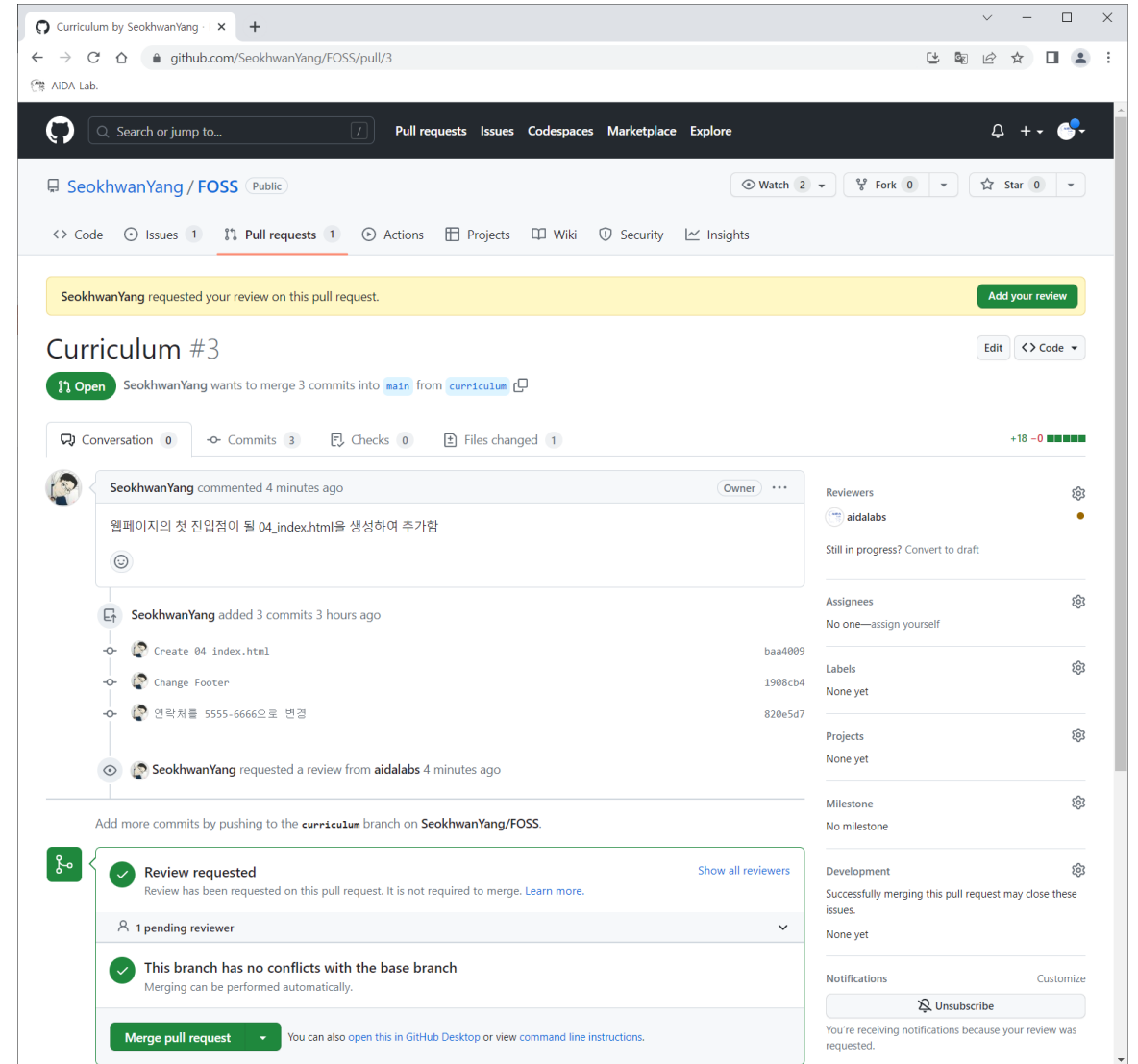
The screenshot shows the GitHub Pull Requests list page for the repository 'SeokhwanYang / FOSS'. The 'Pull requests' tab is highlighted with a red box. Below the navigation bar, there is a message: 'Label issues and pull requests for new contributors. Now, GitHub will help potential first-time contributors discover issues labeled with good first issue'. A red box highlights the filters section, which includes a search bar with 'is:pr is:open', 'Labels 9', 'Milestones 0', and a 'New pull request' button. Below the filters, there is a list of pull requests, with one entry for 'Curriculum' (#3) opened 2 minutes ago by SeokhwanYang. A 'ProTip!' message at the bottom suggests filtering pull requests by the default branch with 'base:main'.

리뷰어로 지정된 사람의 계정에서 내용 확인이 가능함

# Pull Requests 요청 및 처리



Pull Request 내용을 검토한 후 Request 승인/반려 결정



The screenshot shows a GitHub Pull Request interface. At the top, the repository is identified as 'SeokhwanYang / FOSS' (Public). The pull request title is 'Curriculum #3'. A green 'Open' button indicates the PR is ready for review. The PR description states 'SeokhwanYang wants to merge 3 commits into main from curriculum'. Below this, navigation links for Conversation (0), Commits (3), Checks (0), and Files changed (1) are shown. The 'Files changed' link is highlighted with a red box. A blue callout bubble with Korean text '파일 단위로도 볼 수 있음' (Can also be viewed by file) points to the 'Files changed' link. The main content area shows a diff for '04\_index.html', with line 18 highlighted in red. The diff shows HTML code with several additions (+) and deletions (-). At the bottom right, there are buttons for 'Review in codespace' and 'Review changes'.

Search or jump to... Pull requests Issues Codespaces Marketplace Explore

SeokhwanYang / FOSS Public Watch 2 Fork 0 Star 0

Code Issues 1 Pull requests 1 Actions Projects Wiki Security Insights

### Curriculum #3

Open SeokhwanYang wants to merge 3 commits into main from curriculum

Conversation 0 Commits 3 Checks 0 Files changed 1 +18 -0

Changes from all commits File filter Conversations Jump to

0 / 1 files viewed Review in codespace Review changes

```
1 + <html>
2 +
3 + <head>
4 +   <title>강의계획서 ver.1</title>
5 +   <link rel="stylesheet" href="/stylesheets/style.css">
6 + </head>
7 +
8 + <body>
9 +   <h1>강의 계획서</h1>
10 +   <p>글로벌 오픈소스 프로젝트 참여</p>
11 + </body>
12 +
13 + <footer>
14 +   <p>연락처 : 5555-6666</p>
15 +   <p>주소 : 경기도 하남시 XXXX XXXX</p>
16 + </footer>
17 +
18 + </html>
```

Finish your review

Write Preview H B I ≡ <> @ ↩

Leave a comment

Attach files by dragging & dropping, selecting or pasting them.

- Comment  
Submit general feedback without explicit approval.
- Approve  
Submit feedback and approve merging these changes.
- Request changes  
Submit feedback that must be addressed before merging.

Submit review

검토가 끝나면 Review changes 버튼을 누르고  
코멘트를 작성한 후 적용함

## Curriculum #3

Edit <> Code

Open SeokhwanYang wants to merge 3 commits into main from curriculum

Conversation 0 Commits 3 Checks 0 Files changed 1 +18 -0

Changes from all commits File filter Conversations Jump to 0 / 1 files viewed Review in codespace Review changes

18 04\_index.html

```
@@ -0,0 +1,18 @@
1 + <html>
2 +
3 + <head>
4 +   <title>강의계획서 ver.1</title>
5 +   <link rel="stylesheet" href="/stylesheets/style.css">
6 + </head>
7 +
8 + <body>
9 +   <h1>강의계획서</h1>
10 +   <p>글로벌 오픈소스 프로젝트 참여</p>
11 + </body>
12 +
13 + <footer>
14 +   <p>연락처 : 5555-6666</p>
15 +   <p>주소 : 경기도 하남시 XXXX XXXX</p>
16 + </footer>
17 +
18 + </html>
```

Finish your review

Write Preview H B I

수정 내용 확인 완료했습니다.

Attach files by dragging & dropping, selecting or pasting them.

- Comment  
Submit general feedback without explicit approval.
- Approve**  
Submit feedback and approve merging these changes.
- Request changes  
Submit feedback that must be addressed before merging.

Submit review

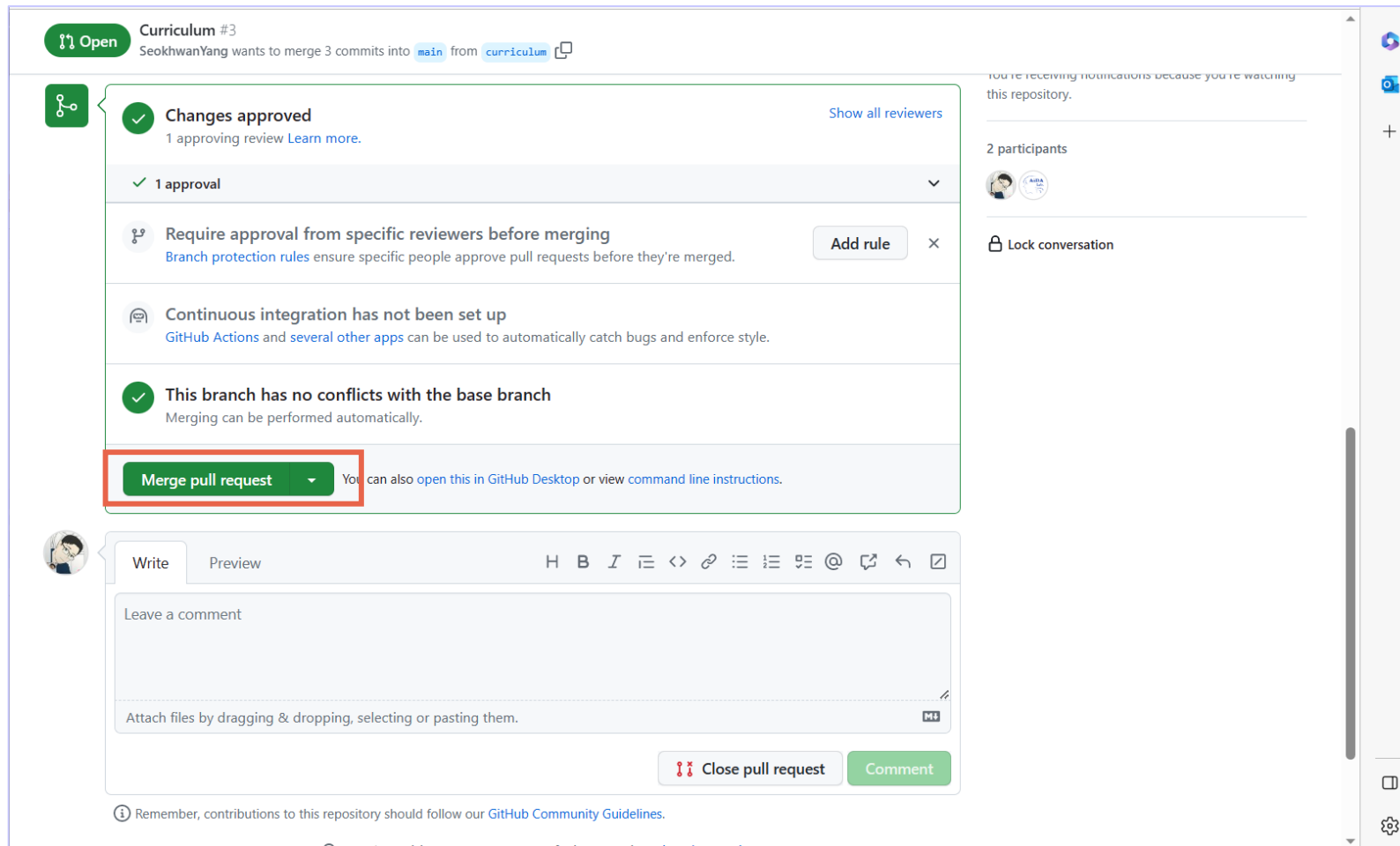
코멘트를 작성한 후 "Approve"를 체크,  
Submit review 버튼을 눌러서 승인

# Pull Requests 요청 및 처리

The screenshot shows a GitHub Pull Request interface. At the top, the repository is identified as 'SeokhwanYang / FOSS'. The pull request title is 'Curriculum #3'. A comment from 'SeokhwanYang' states: '웹페이지의 첫 진입점이 될 04\_index.html을 생성하여 추가함'. Below this, a commit history shows three commits: 'Create 04\_index.html', 'Change Footer', and '연락처를 5555-6666으로 변경'. A review from 'aidalabs' is shown at the bottom, with the text '수정 내용 확인 완료했습니다.' (Checked and approved the changes). On the right side, the 'Reviewers' section lists 'aidalabs' with a green checkmark, indicating approval. A red box highlights the 'aidalabs' reviewer and the approval checkmark. Another red box highlights the approval comment and the reviewer's name in the 'Reviewers' list. The 'Reviewers' list also includes options like 'Still in progress? Convert to draft', 'Assignees', 'Labels', 'Projects', 'Milestone', and 'Development'.

메인 페이지에서 리뷰완료 여부를 확인할 수 있음

- Pull Requests가 완료되면 기준 브랜치에 병합



- 최종 승인

The screenshot shows a GitHub Pull Request interface for 'Curriculum #3'. At the top, it indicates 'SeokhwanYang wants to merge 3 commits into main from curriculum'. Below this, a user has initiated a merge confirmation. The dialog box contains the following elements:

- A title: 'Merge pull request #3 from SeokhwanYang/curriculum'
- A text input field containing the Korean text 'Curriculum 추가 메시지 작성' (Write additional message for curriculum).
- A dropdown menu for email selection, currently showing 'yang.seokhwan@gmail.com'.
- Two buttons: 'Confirm merge' (highlighted with a red box) and 'Cancel'.

Below the merge confirmation, there is a comment section with a rich text editor toolbar (including bold, italic, link, and list icons) and a text area with the placeholder 'Leave a comment'. At the bottom of the comment section are 'Close pull request' and 'Comment' buttons.

On the right side of the interface, there is a sidebar with the following information:

- A notification: 'You're receiving notifications because you're watching this repository.'
- A section for '2 participants' showing two user avatars.
- A 'Lock conversation' button.

At the bottom of the page, there is a 'ProTip!' icon and text: 'ProTip! Add comments to specific lines under Files changed.'

- 완료

The screenshot displays a GitHub notification for a pull request. At the top, a commit by SeokhwanYang is shown as merged into the 'main' branch, with a 'Revert' button. Below this, a purple-bordered box contains the message: 'Pull request successfully merged and closed. You're all set—the curriculum branch can be safely deleted.' A 'Delete branch' button is located to the right of this message. At the bottom, there is a text input area with 'Write' and 'Preview' tabs, a rich text editor toolbar (including icons for bold, italic, code, link, list, quote, mention, share, undo, and redo), and a 'Leave a comment' placeholder.

Notifications

You're receiving notificat  
this repository.

2 participants

Lock conversation

- 원격 저장소 최신 내역 → 지역 저장소 반영
  - `git pull {원격 저장소 식별자} {원격 저장소 브랜치}`

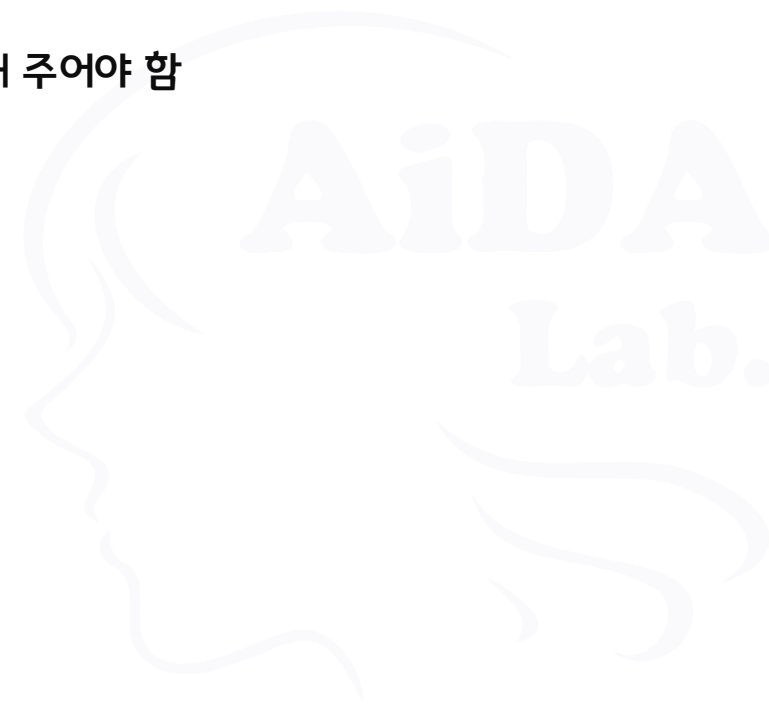
```
Windows PowerShell
PS C:\AiDALab\Workspace\FOSS> git checkout main
Already on 'main'
PS C:\AiDALab\Workspace\FOSS> git pull origin main
remote: Enumerating objects: 1, done.
remote: Counting objects: 100% (1/1), done.
remote: Total 1 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (1/1), 662 bytes | 220.00 KiB/s, done.
From https://github.com/SeokhwanYang/FOSS
 * branch          main          -> FETCH_HEAD
   f0daa46..53f4872 main          -> origin/main
Merge made by the 'ort' strategy.
PS C:\AiDALab\Workspace\FOSS> |
```

## • 최종 확인

최신 커밋을 통해 승인된 Pull Request가 반영된 것을 확인할 수 있음

```
Windows PowerShell
PS C:\AiDALab\Workspace\FOSS> git log --pretty=oneline --graph
* 7d67f3c301994c38028e5ebf211f8f9e42b7c47f (HEAD -> main) Merge branch 'main' of https://github.com/SeokhwanYang/FOSS
|
| * 53f48728af2042f2e598fd9fcd9eebe094230932 (origin/main) Merge pull request #3 from SeokhwanYang/curriculum
| |
| | * f0daa4678688e543c0151cacef5cf48c7f731c30 Merge pull request #2 from SeokhwanYang/curriculum
| | |
| | | * ef5bf60b615b72493fb6bca2b430bb964895cbc0 충돌 해결
| | | |
| | | | * 820e5d78034141e442662c0190aa685d04ab5f0b (origin/curriculum, curriculum) 연락처를 5555-6666으로 변경
| | | | * 1908cb487deee385bed2b2b624639dfb408bf4e2 Change Footer
| | | | * 710cf83d90f66b7bbd2a94049139d5675bb7f606 연락처를 3333-4444로 변경
| | | | * 6f89bf05864b19e078e243a2a5467999dfb9a886 Change Body
| | | |
| | | | * baa40098ca885a1e8d8bef8c43c63e6e16c4c588 Create 04_index.html
| | | |
| | | | * ef89f475385901efd59cf1191b5ae6aa501cab77 Curriculum File Created.
| | | | * 52e86f63cb84d4172ce1d0240eb5d6aa71cb8bb3 First File Created.
| | | |
| | | | * 43a46a6a7699d0b67ccc5cceb6c71783ef8ddd9f (origin/test/remote_branch) 두 번째 파일 생성: 내용 수정
| | | | * 1a7c90ca5ba22bafef4fe7902aab59eb20b2eee88 ReadMe.md 파일 생성
| | | |
PS C:\AiDALab\Workspace\FOSS>
```

- 원격 저장소 최신 내역을 지역 저장소로 반영하는 또 다른 방법
  - git fetch 후 직접 git merge를 수행
    - git fetch는 git pull과 달리 원격 저장소의 변경 내역을 가져올 뿐,
    - 지역 저장소의 작업 브랜치에 병합 작업은 수행하지 않음
    - 따라서 git fetch를 사용했을 경우에는 직접 git merge로 병합 작업을 해 주어야 함



**THANK  
YOU**

