

프론트엔드 정규세션 3주차

깃허브

강의 : 구예원



LIKELION X DKU

Front-end 3th

- GITHUB 구조
- GITHUB 사용 FOR 개인
- GITHUB 사용 FOR 협업
- OG 태그
- 내 깃허브 꾸미기



GitHub란?



- ✓ Git : 컴퓨터 파일의 변경 사항을 추적하고 여러 명의 사용자들 간에 해당 파일들의 작업을 조율하기 위한 분산 버전 관리 시스템
- ✓ GitHub : 분산 버전 관리 툴인 깃(Git)을 사용하는 프로젝트를 지원하는 웹 호스팅 서비스

- Git과 GitHub



Git

내 원하는 시점으로 자유롭게 돌아갈 수 있는 소스 코드 버전 관리 시스템



GitHub

Git으로 관리하는 프로젝트를 올려둘 수 있는 Git 호스팅 사이트

● Git의 3가지 작업 상태

working directory
작업 디렉토리

Staging
스테이지

local repository
저장소

working directory : 현재 작업 중인 프로젝트가 위치한 디렉토리

● Git의 3가지 작업 상태

working directory
작업 디렉토리

Staging
스테이지

local repository
저장소

Staging : commit 할 파일의 예비 저장소

● Git의 3가지 작업 상태

working directory
작업 디렉토리

Staging
스테이지

local repository
저장소

local repository : 각 컴퓨터의 git이 관리하는 로컬 저장소

- Git의 3가지 작업 상태

working directory

작업 디렉토리

my file

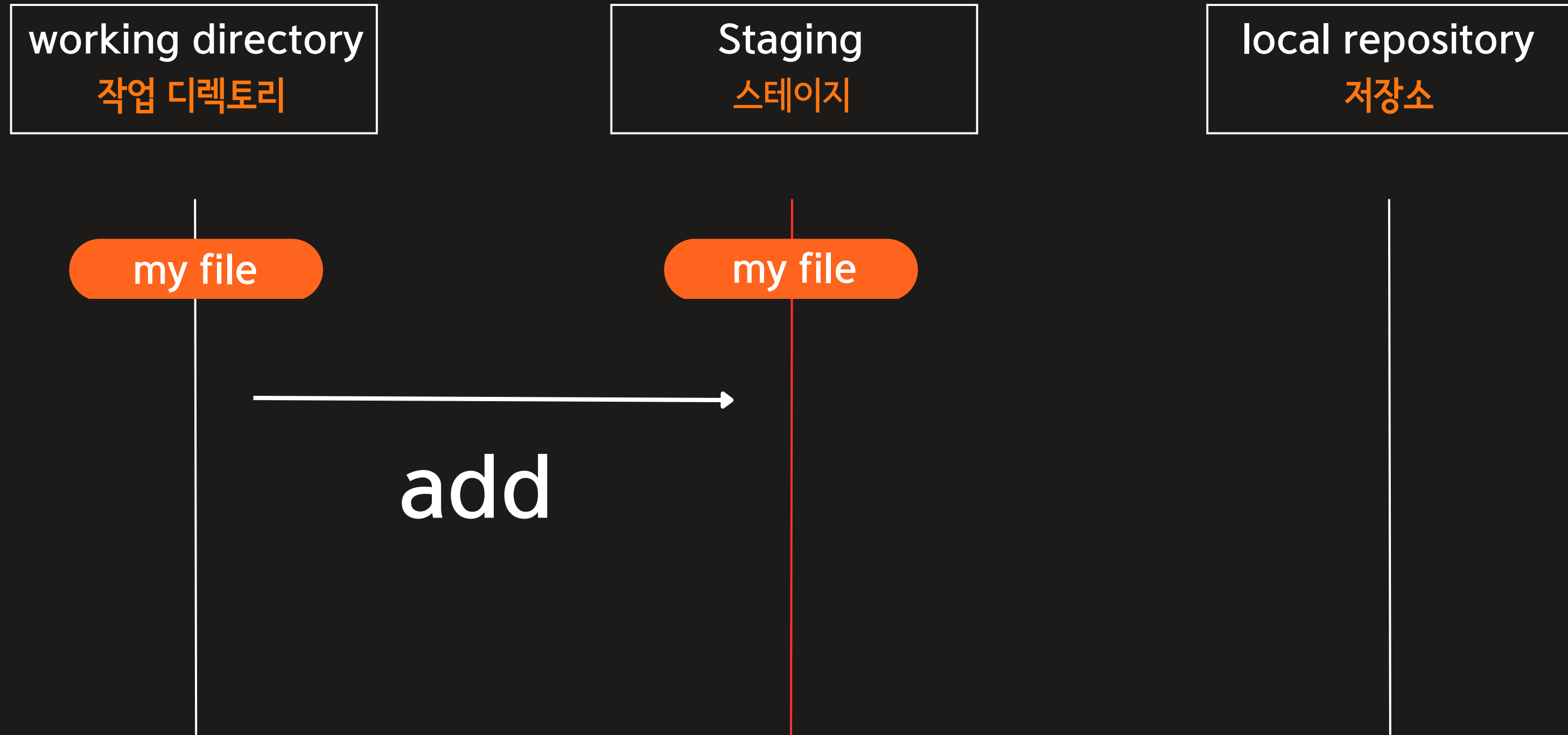
Staging

스테이지

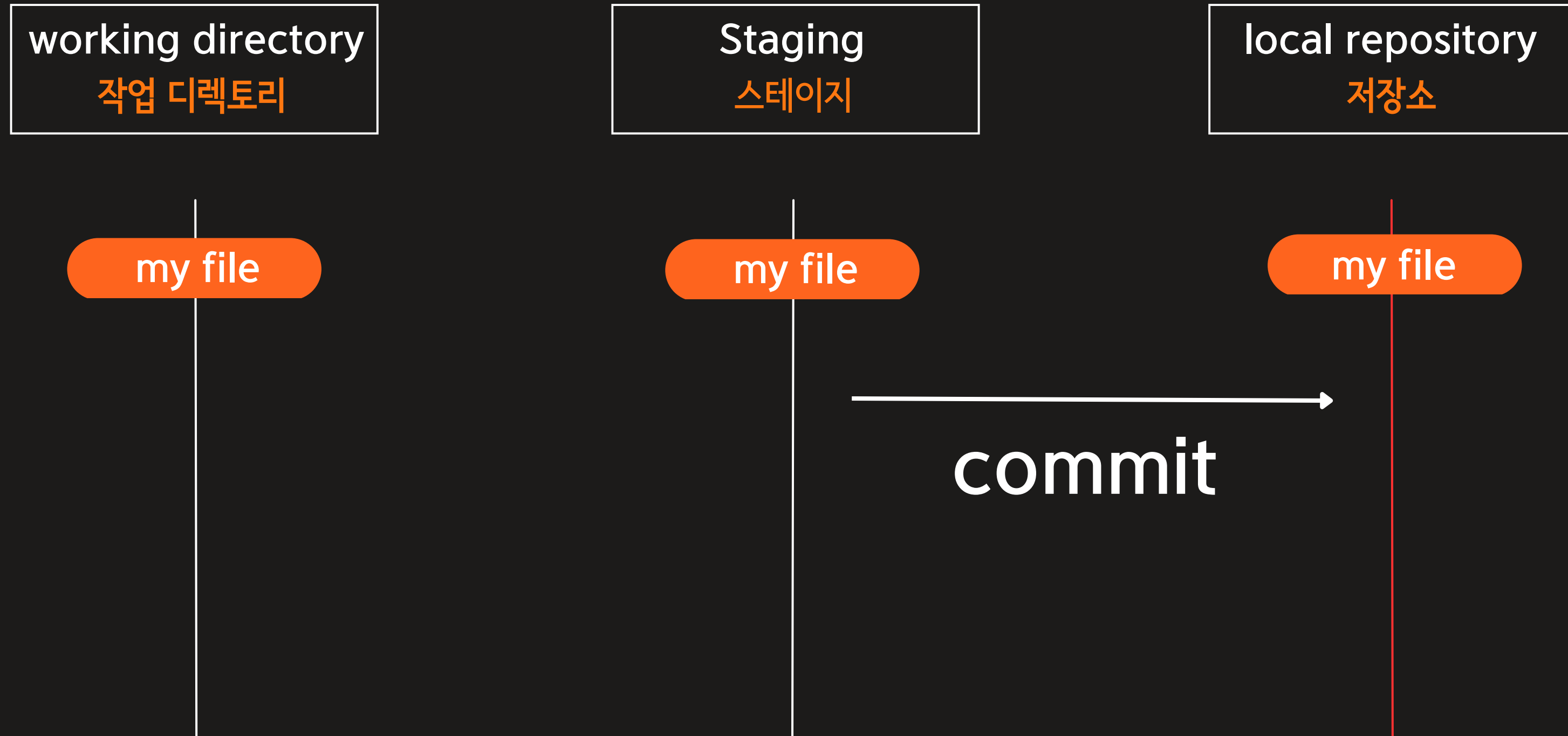
local repository

저장소

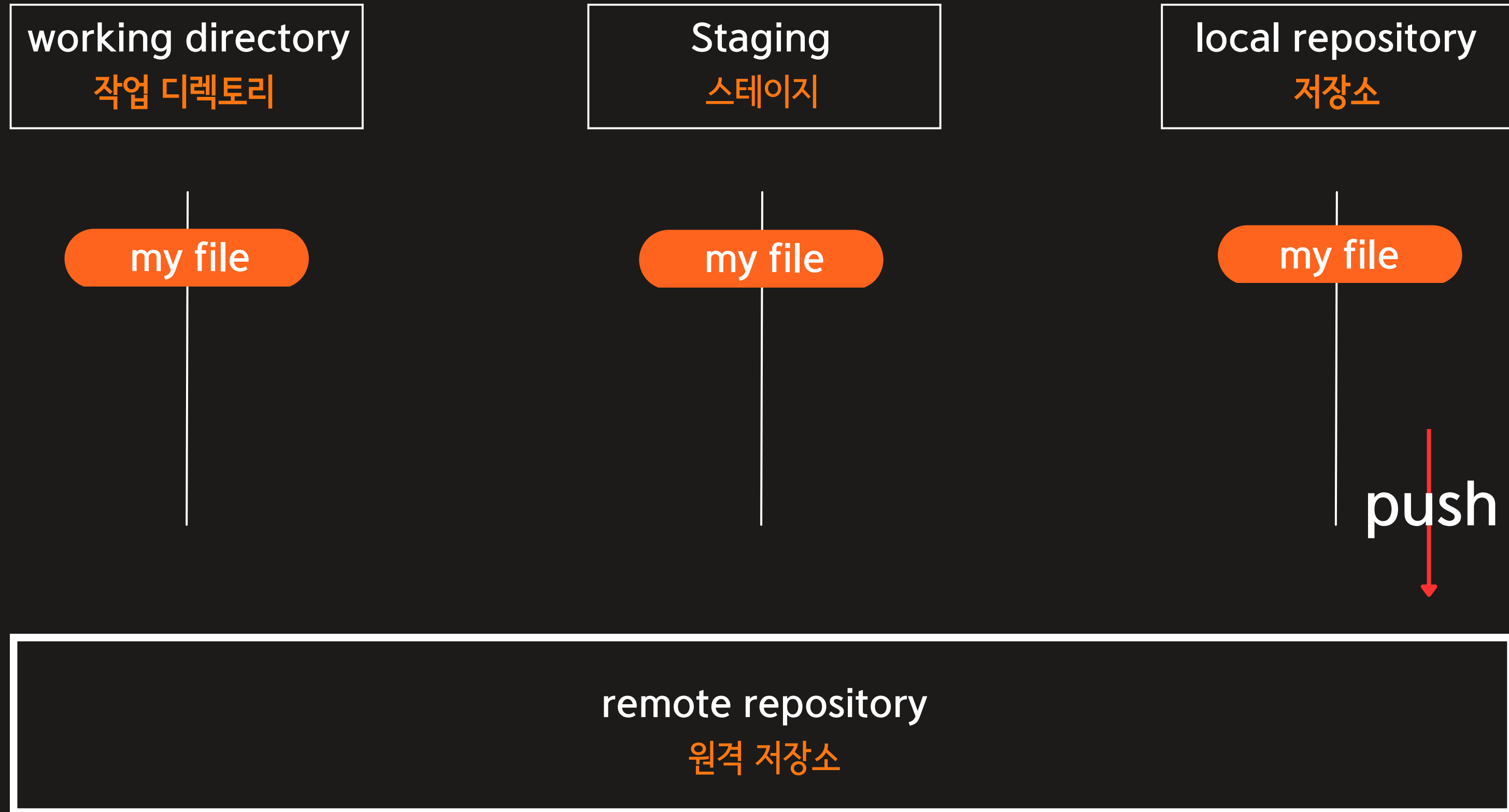
- Git의 3가지 작업 상태



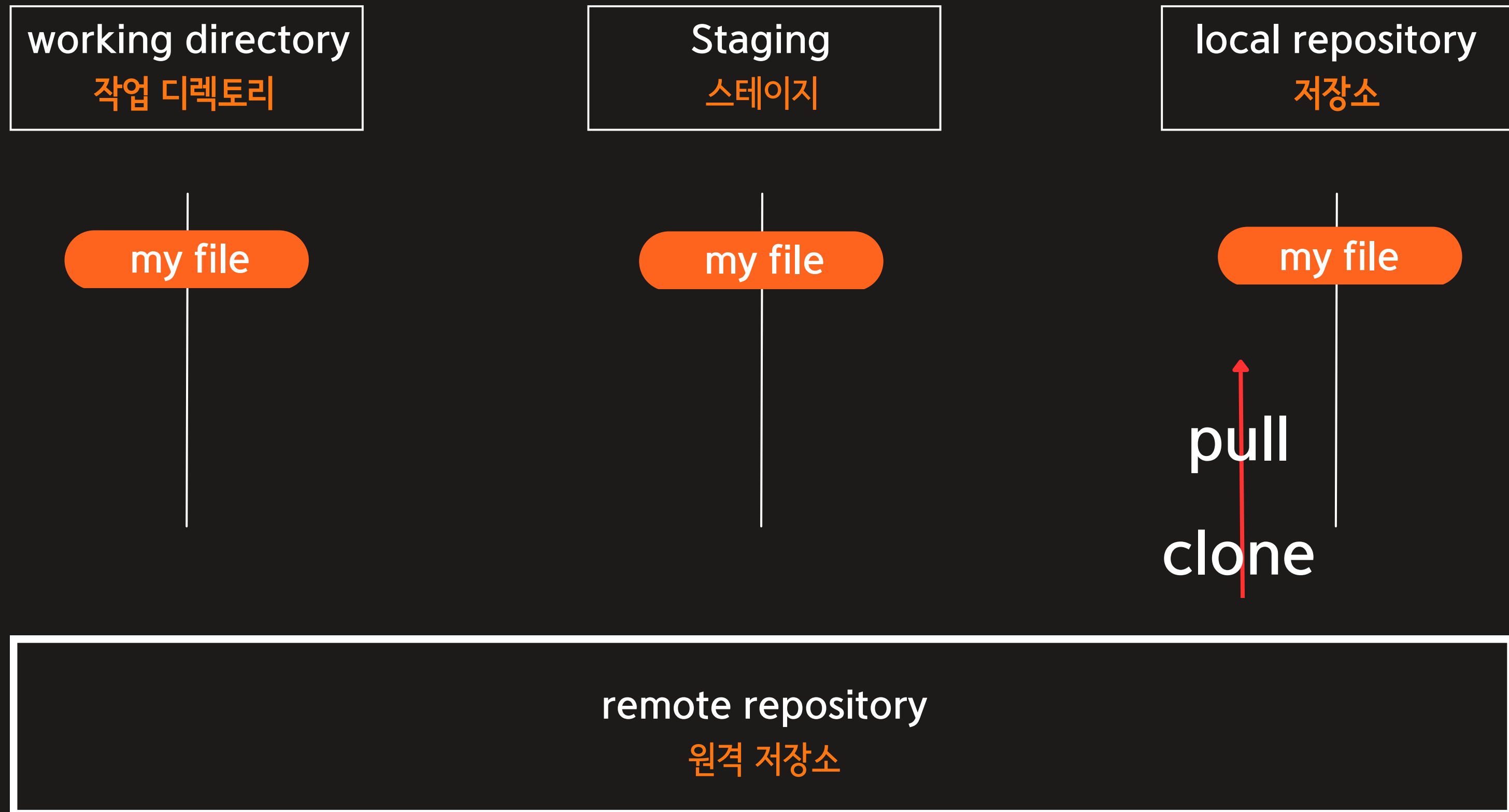
- Git의 3가지 작업 상태



- Git의 3가지 작업 상태



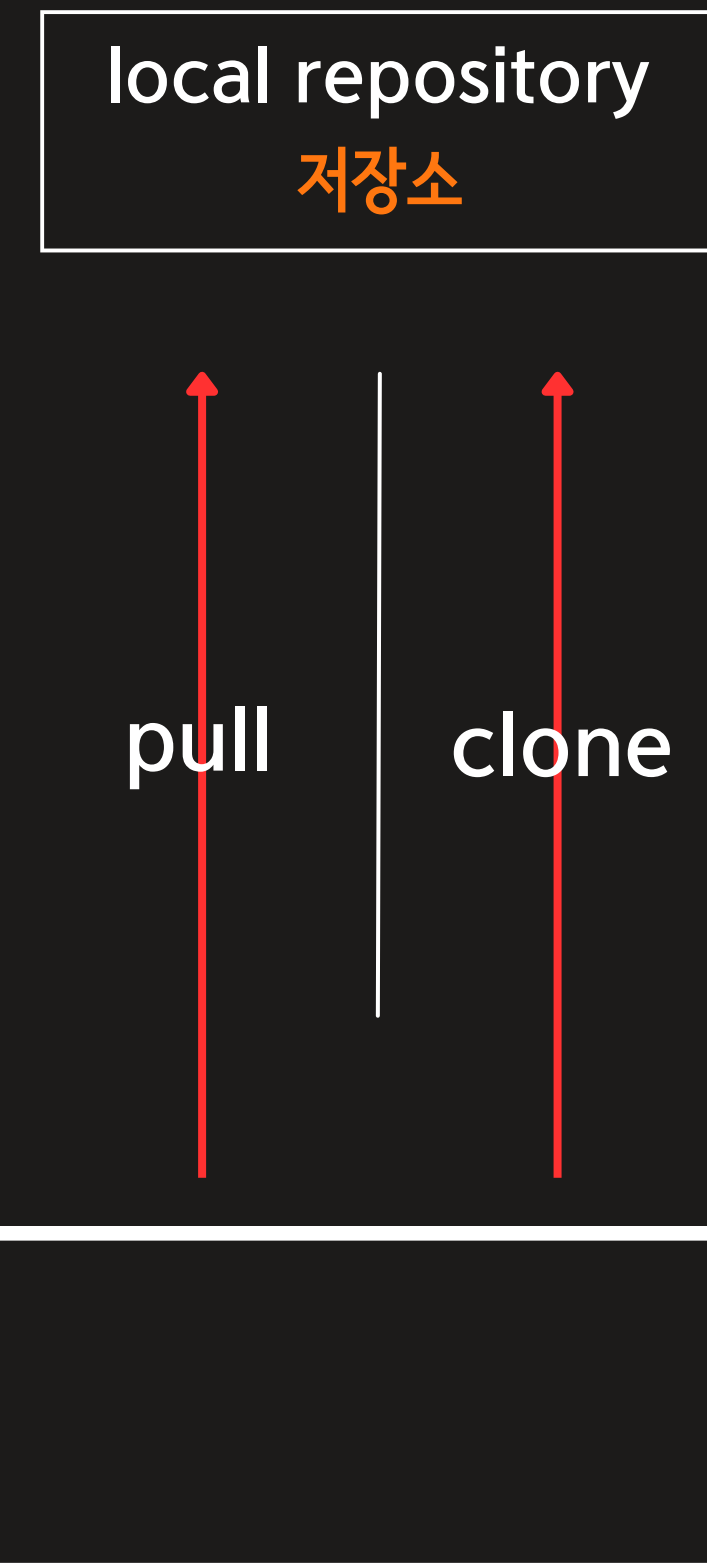
- Git의 3가지 작업 상태



● clone과 pull의 차이?

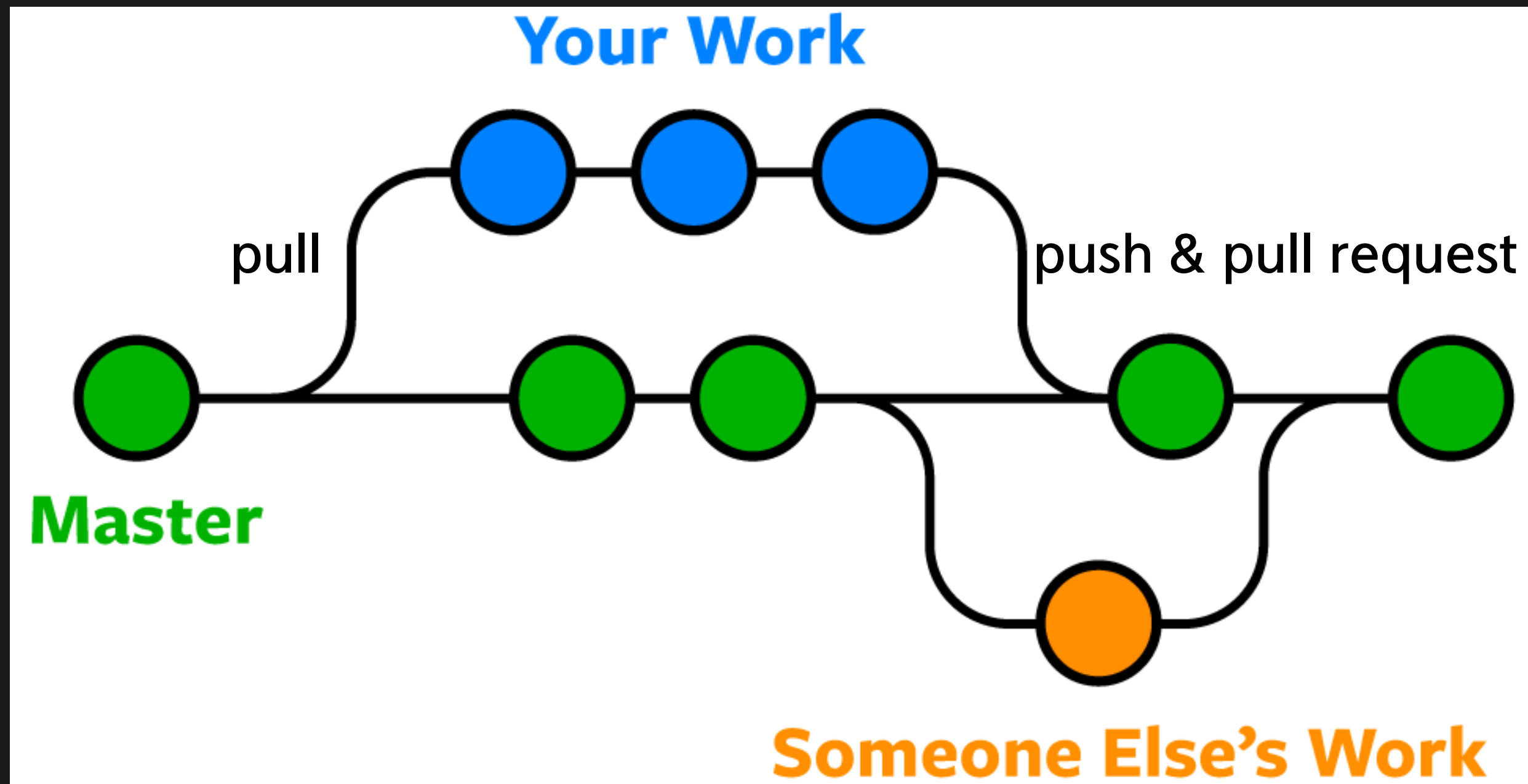
git clone: 해당 repository를 통째로 다운 받는 개념으로
내 컴퓨터에 해당 프로젝트의 폴더를 새롭게 생성한다.

git pull : 내가 참여하고 있는 프로젝트에서 변경 사항이 생긴 경우
내가 로컬로 작업하고 있는 프로젝트와의 병합을 위해서 사용한다.

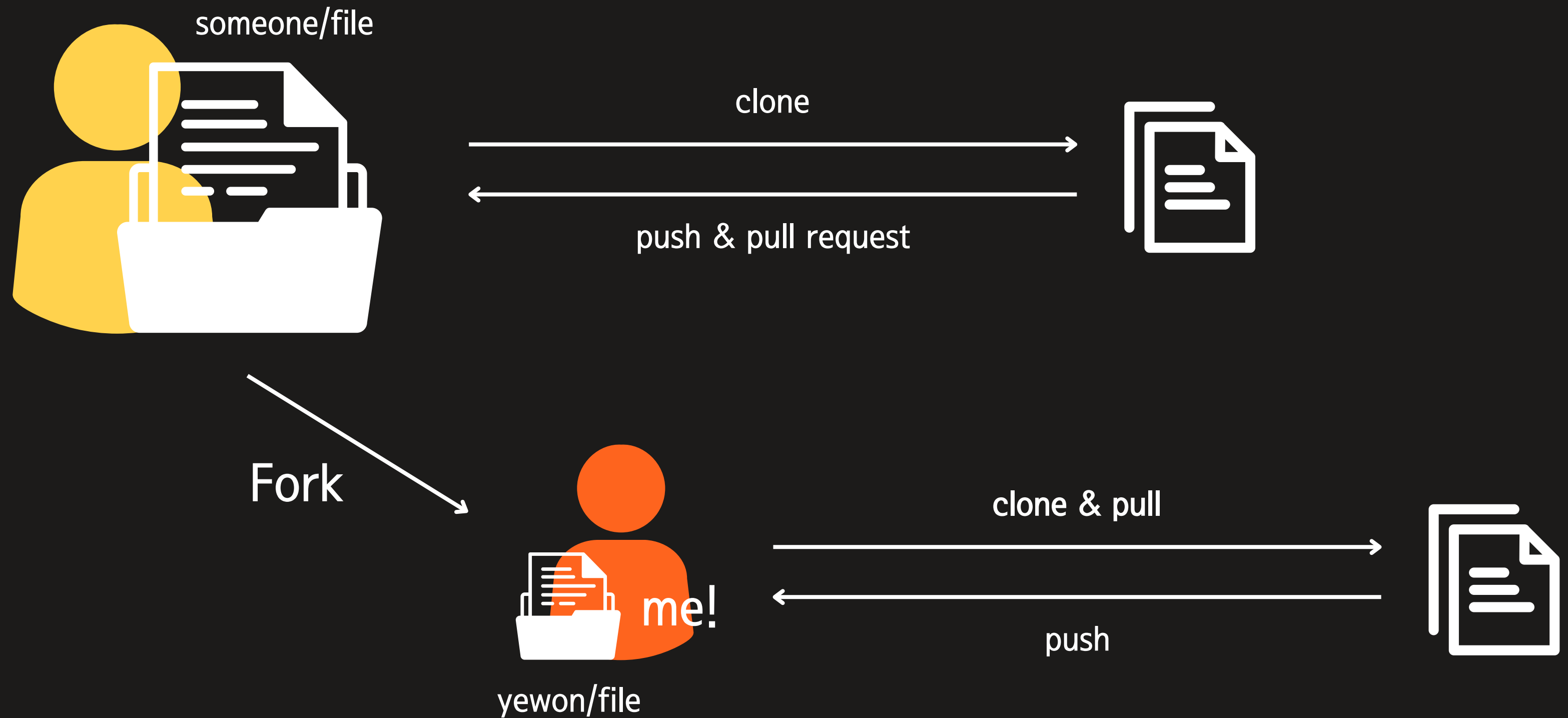


• Branch

여러 개발자들이 함께 개발할 때 충돌이 일어나지 않고 다양한 작업을 동시에 할 수 있게 만들어주는 기능



• Fork



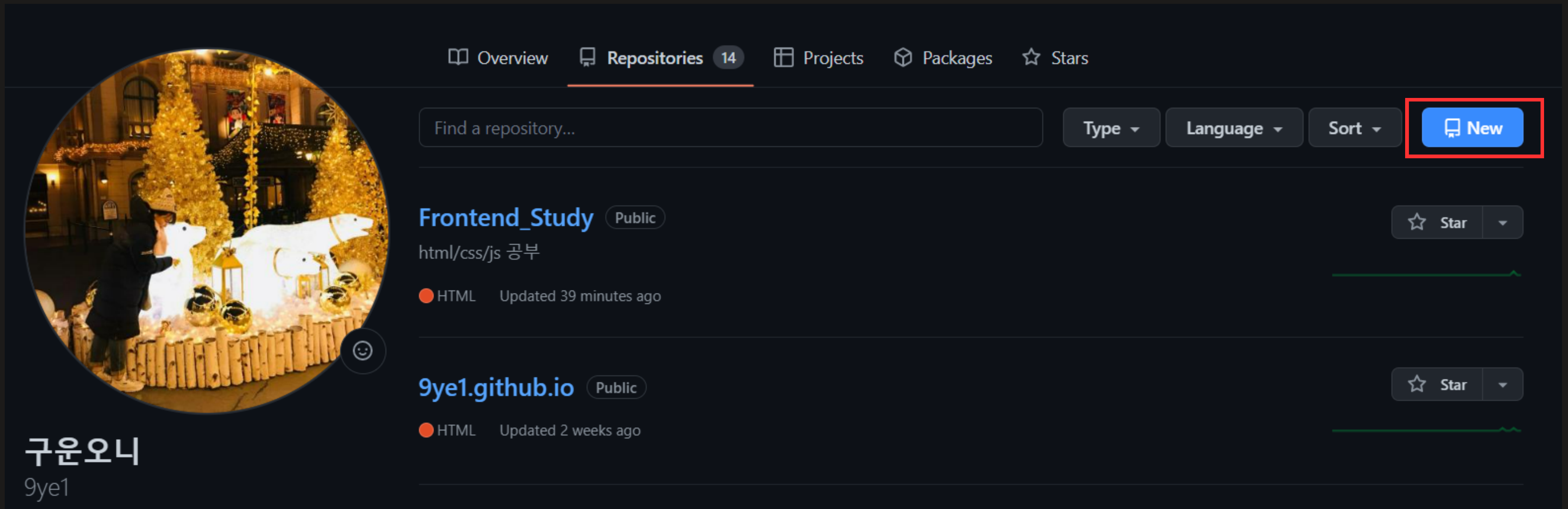


GitHub 사용하기

- ✓ 내 프로젝트 관리하기
- ✓ 협업하기

내 개인 프로젝트 관리하기

● 원격 저장소 생성하기



Overview Repositories 14 Projects Packages Stars

Find a repository... Type Language Sort **New**

Frontend_Study Public
html/css/js 공부
HTML Updated 39 minutes ago

9ye1.github.io Public
HTML Updated 2 weeks ago

구운오니
9ye1


원격 저장소 생성하기

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner *

Repository name *

 9ye1 ▾ /

Great repository names are short and memorable. Need inspiration? How about [supreme-meme?](#)

Description (optional)



Public

Anyone on the internet can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.

Initialize this repository with:

Skip this step if you're importing an existing repository.



Add a README file

This is where you can write a long description for your project. [Learn more.](#)

Add .gitignore

Choose which files not to track from a list of templates. [Learn more.](#)

.gitignore template: None ▾

Choose a license

A license tells others what they can and can't do with your code. [Learn more.](#)

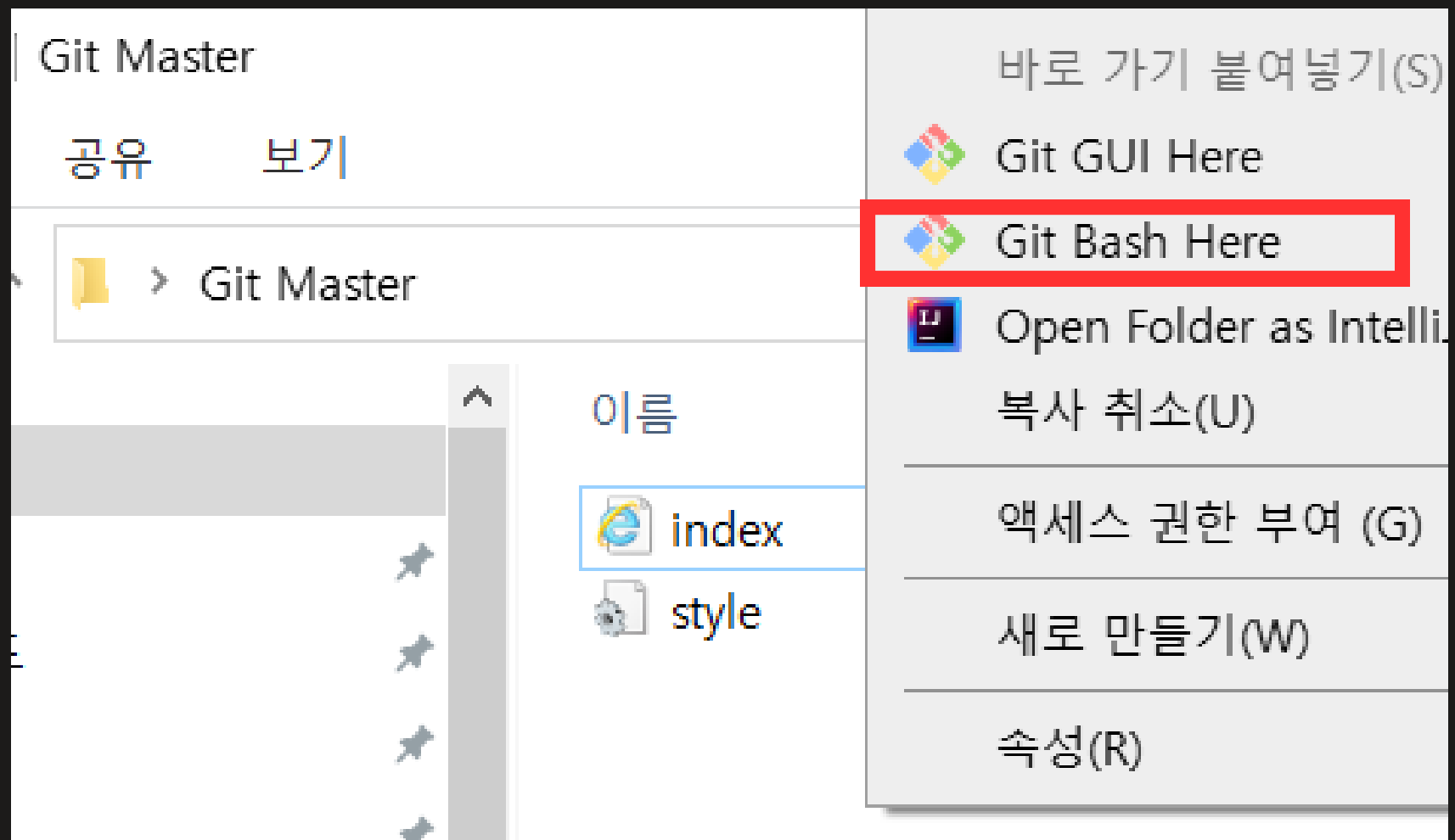
License: None ▾



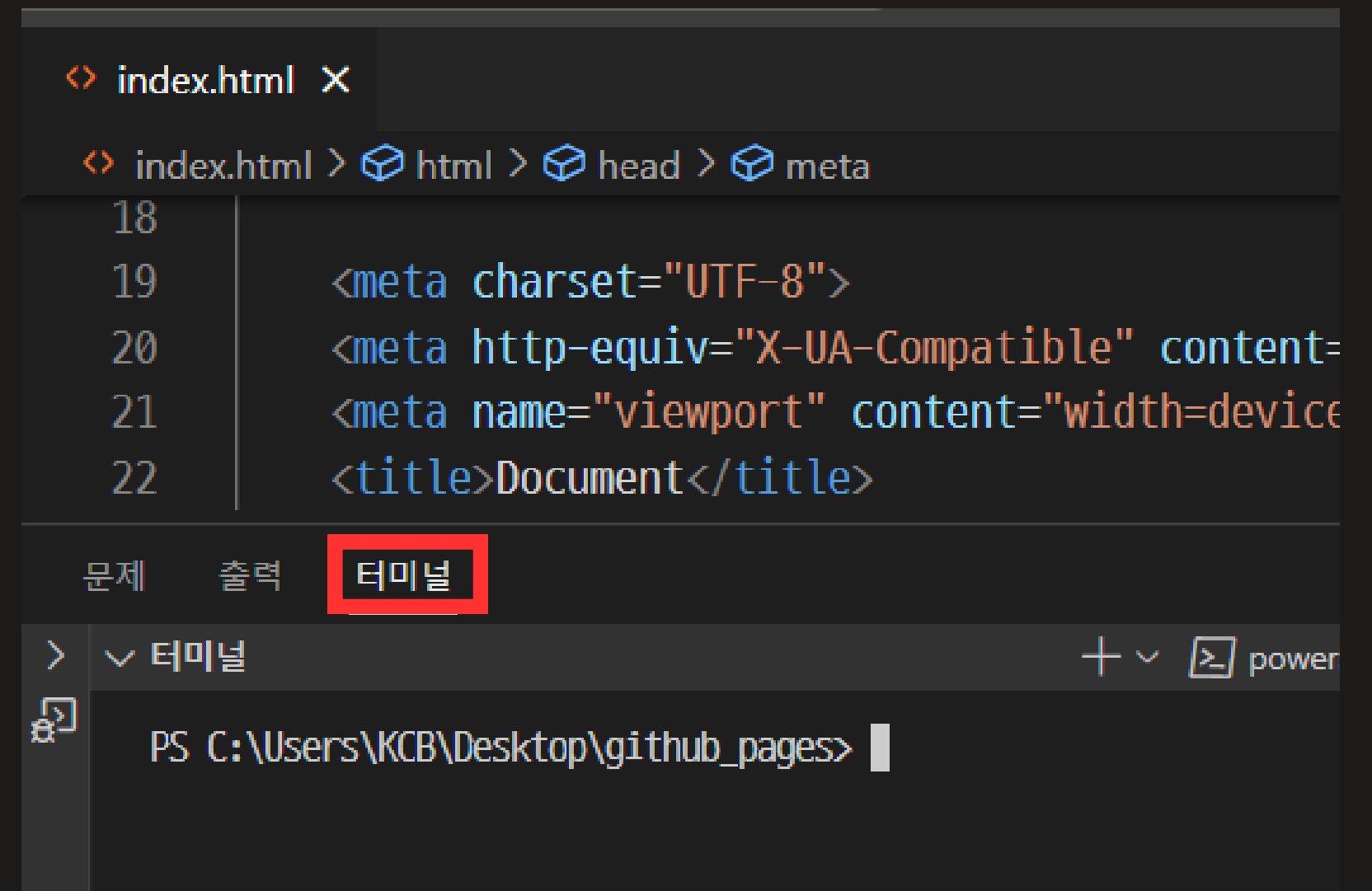
You are creating a public repository in your personal account.

Create repository

폴더 내 우클릭



VScode의 터미널



● 초기 작업

사용자 저장

- git config --global user.name [깃허브 닉네임]
 - git config --global user.email [깃허브 아이디(이메일)]
-
- git config user.name
 - git config user.email

git 저장소 생성

- git init

● 명령어 입력

`[$ git status]`

`[$ git add . ('.'은 모든 파일을 의미)]`

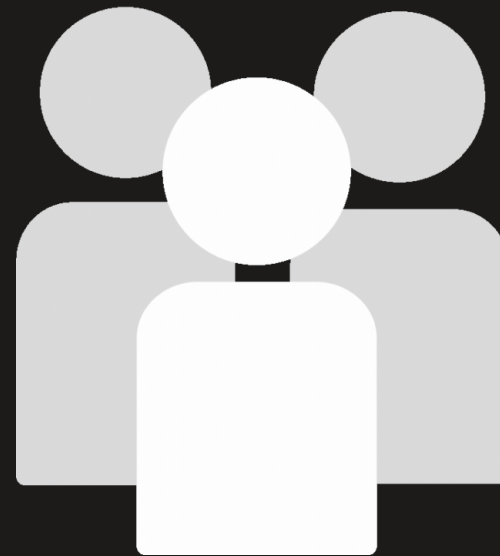
`[$ git commit -m "commit 메시지"]`

`[$ git remote add origin [Repository주소]]`

`[$ git push origin [Branch이름]]`

- 새로운침 후 변경사항 확인해보기

프로젝트 협업하기 (member 입장)



● 프로젝트 clone하기

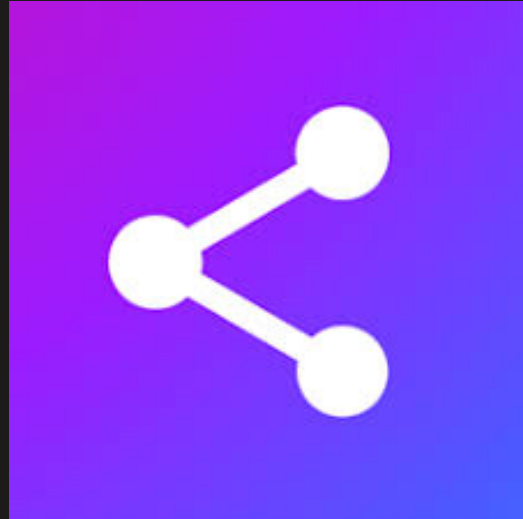
```
git clone [repository 주소] [폴더 명]
```

주소 <https://github.com/likelion-DKU-frontend/homework>



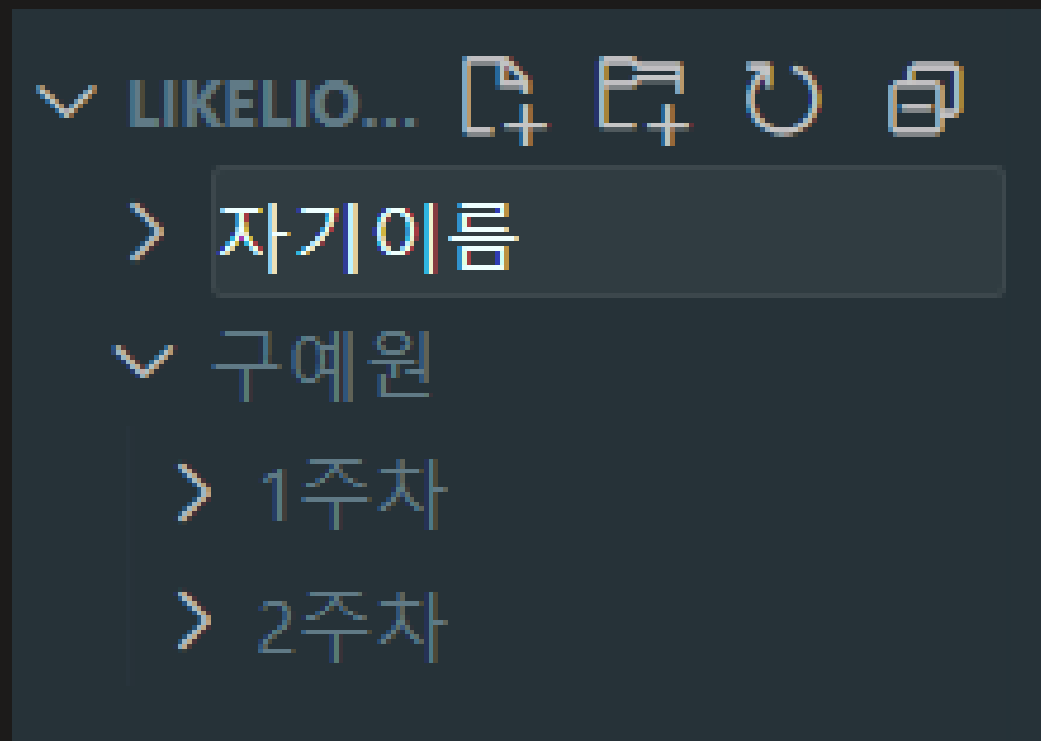
폴더 명 : 자신이 원하는 폴더 명 설정 가능 or 비워두면 repository 이름 그대로 자동 설정

● 각자 브랜치 생성하기



1. [`$ git branch`] branch list & 내가 위치한 branch 확인
2. [`$ git branch 브랜치 명`] 내 branch 생성
3. [`$ git checkout 브랜치 명`] 내 branch 로 이동
4. [`$ git branch`] 내가 위치한 branch 확인

● 프로젝트 각자 폴더 생성 & 파일 업로드



1. 자기 이름으로 된 폴더 추가
2. 자신의 폴더 안에 1주차 폴더 생성
3. 1주차 폴더에 1주차 과제(인스타그램 클론코딩) 파일 추가
(index.html / style.css / images)

- add/ commit/ push/ pull reqeust

add

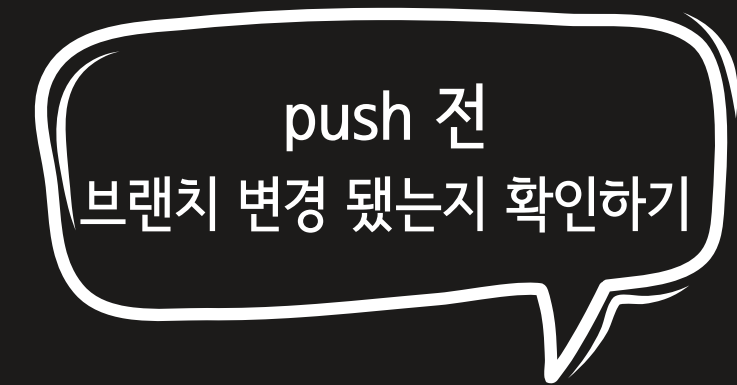
[\$ git add .]

commit


[\$ git commit -m "커밋 메세지"]

push

[\$ git push origin 자기 브랜치 명]



pull reqeust

 yewon had recent pushes 2 minutes ago

Compare & pull request

Repository 페이지를 새로고침 후 Compare&Pull Request 버튼

● Pull/Push 오류 1

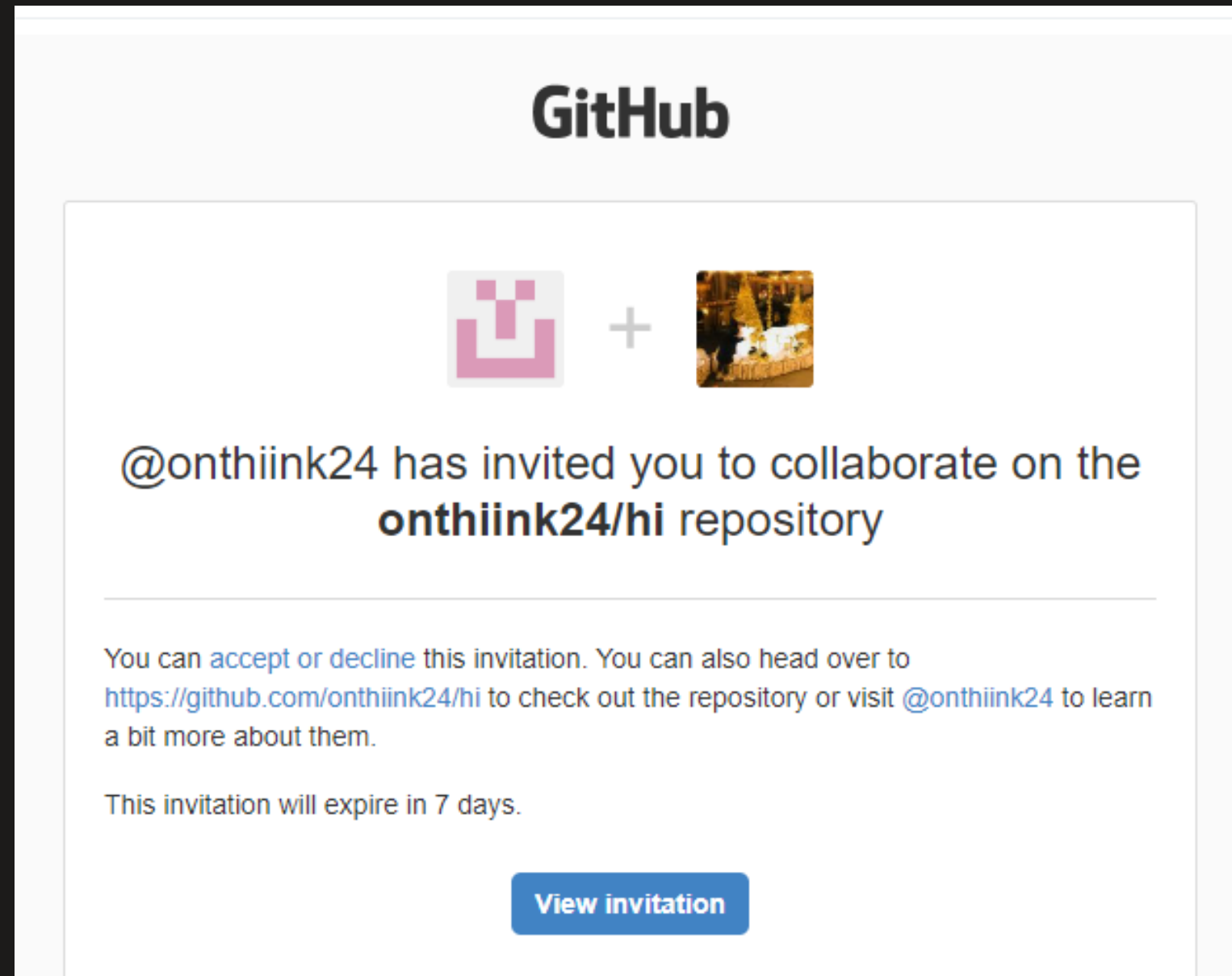


Permission to <주소> denied to <ID>

수정자에게 권한이 없어서 생기는 오류

- > repo를 만든 leader가 invite collaborators에 팀원 아이디 추가
- > 팀원은 메일을 통해 invite accept

● 프로젝트 권한 가지기



프로젝트 협업하기 (leader 입장)



- leader-member 2명씩 실습

leader

1. 새로운 repository 생성
2. 권한 부여

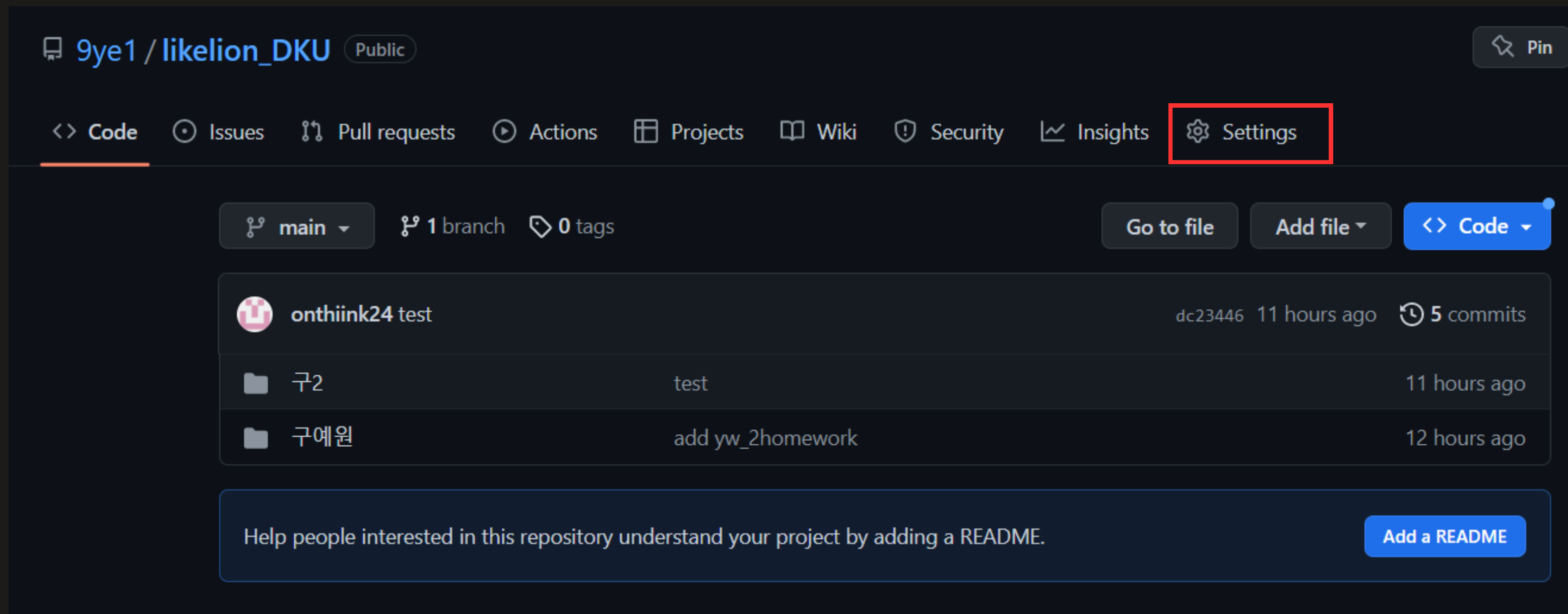
3. Merge

member

1. 이메일로 Invite Accept
2. clone
3. 새로운 브랜치 생성
4. 폴더&파일 추가 후 push
5. pull request 요청

• Collaborators (권한 부여하기) 1

Settings






The screenshot shows the GitHub repository page for `9ye1 / likelion_DKU`, which is public. The **Settings** tab is highlighted with a red box in the top navigation bar. Below the navigation bar, the repository details show the `main` branch with 1 branch and 0 tags. The commit history table lists two commits: one by `onthink24` titled `test` (11 hours ago) and another titled `add yw_2homework` (12 hours ago). At the bottom, there is a prompt to add a README.

9ye1 / likelion_DKU Public

Code Issues Pull requests Actions Projects Wiki Security Insights **Settings**

main 1 branch 0 tags

Go to file Add file <> Code

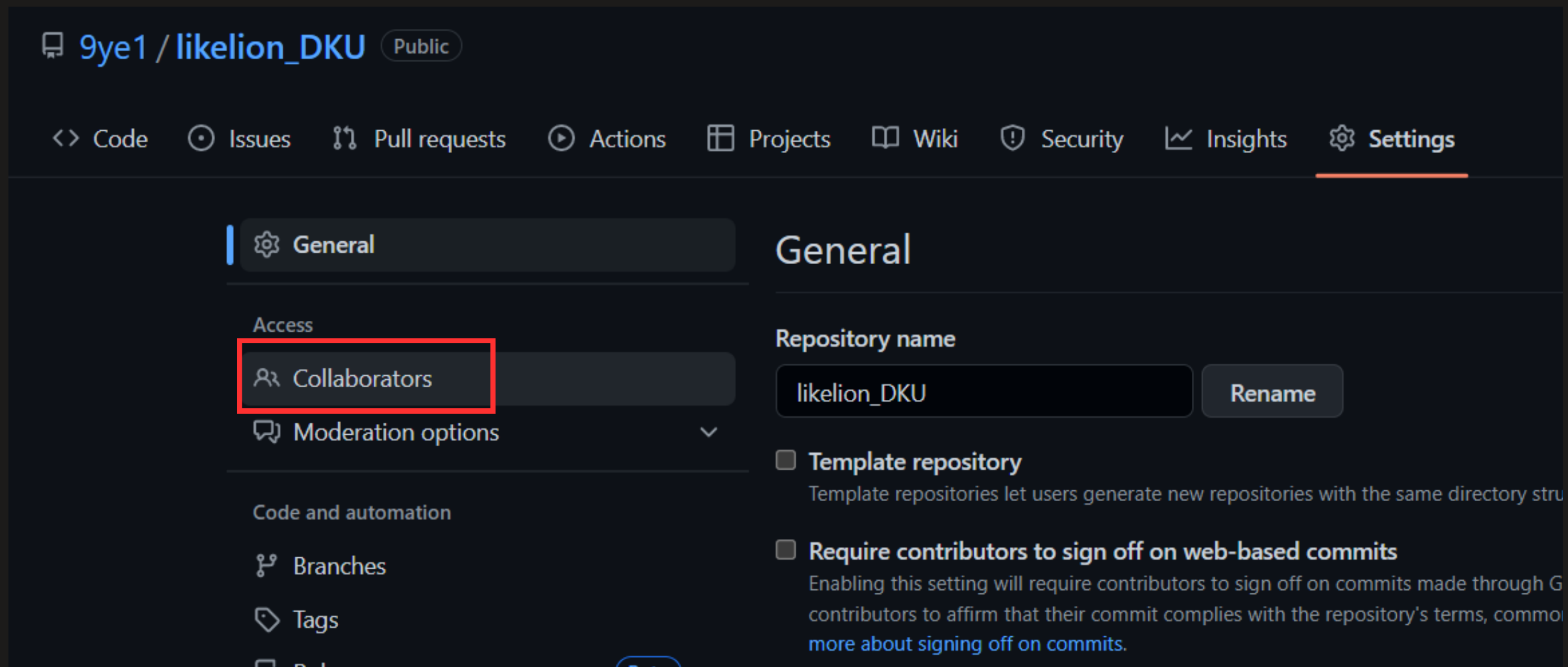
	onthink24 test	dc23446 11 hours ago	🕒 5 commits
	구2	test	11 hours ago
	구예원	add yw_2homework	12 hours ago

Help people interested in this repository understand your project by adding a README.

Add a README


● Collaborators (권한 부여하기) 2

Collaborators



• Collaborators (권한 부여하기) 3

비밀번호 입력



Confirm access

Password

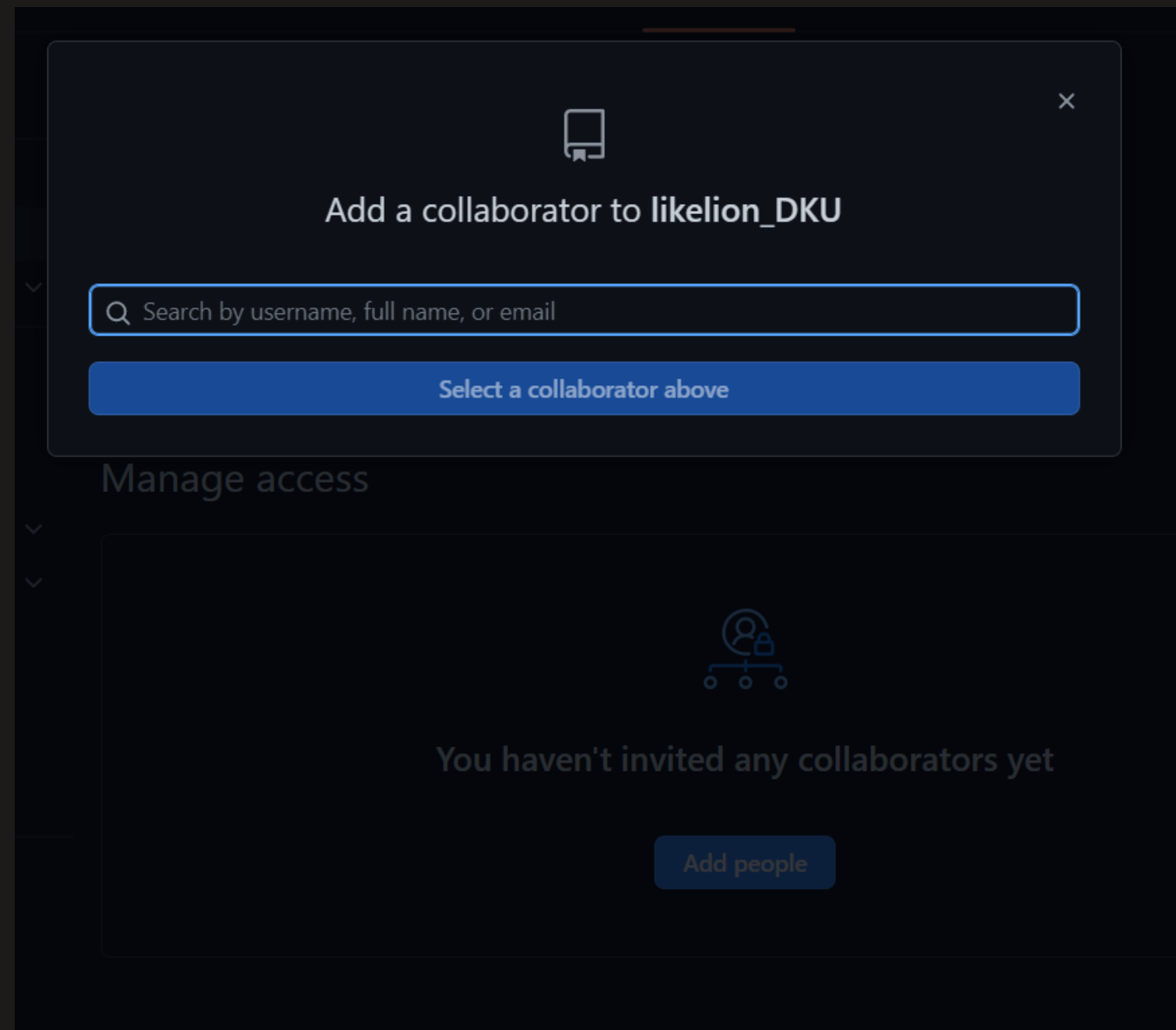
[Forgot password?](#)

Confirm

Tip: You are entering **sudo mode**. After you've performed a sudo-protected action, you'll only be asked to re-authenticate again after a few hours of inactivity.

• Collaborators (권한 부여하기) 4

Add people -> 깃허브 name 검색

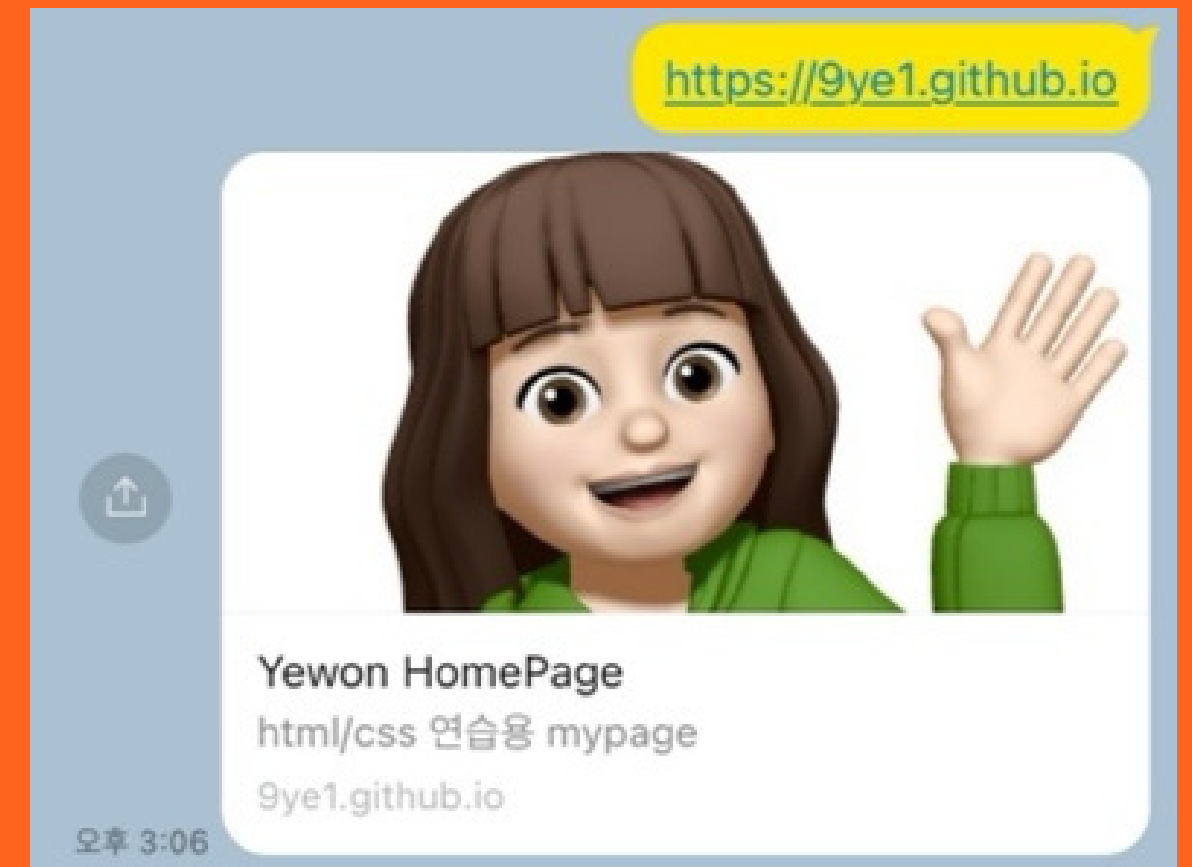


OG 태그

Open Graph

대부분의 콘텐츠는 URL로 공유된다.

이 때, OG태그는 미리보기 형식으로 콘텐츠의 요약 내용을 노출해준다.
어떤 HTML 문서의 메타정보를 쉽게 표시하기 위해서 메타정보에 해당하는 **제목, 설명, 문서의 타입, 대표 URL** 등 다양한 요소들에 대해서 사람들이 통일해서 쓸 수 있도록 정의해놓은 프로토콜이다.



● 1 - OG 태그 추가하기




```
<head>  
  <meta property="og:title" content="타이틀">  
  <meta property="og:description" content="간단한 설명">  
  <meta property="og:image" content="이미지 경로">  
</head>
```

html 파일 이름은 반드시 index.html 로!

- 2 - Github Pages 원격 저장소 만들기

Owner * Repository name *

 9ye1 ▾ / 9ye1.github.io ✓

Repository 이름 주의!

[깃허브ID].github.io

● 3 - Github 반영

아래의 명령어 입력

```
[$ git init ]
```

```
[$ git add . ]
```

```
[$ git commit -m "커밋 메세지" ]
```

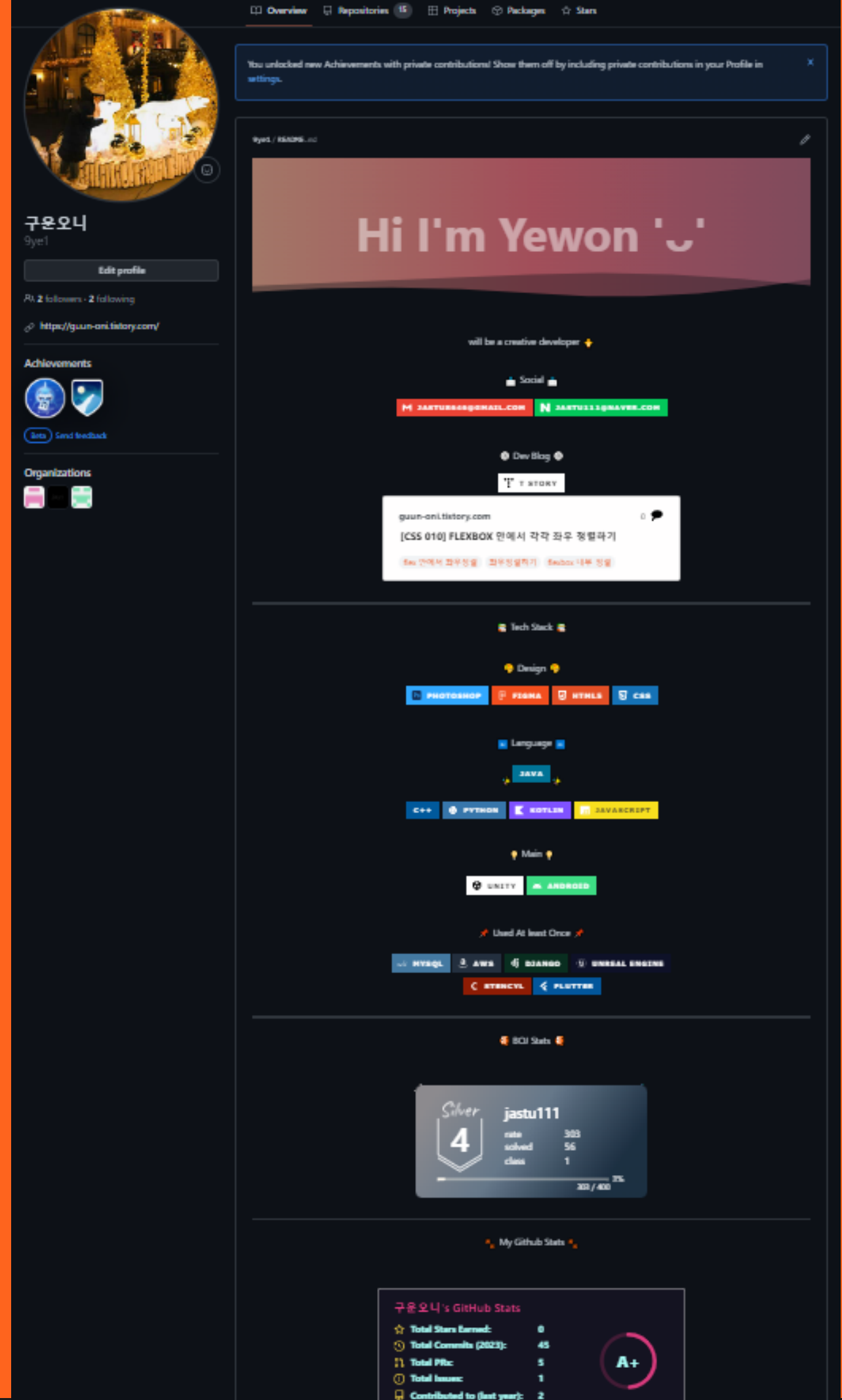
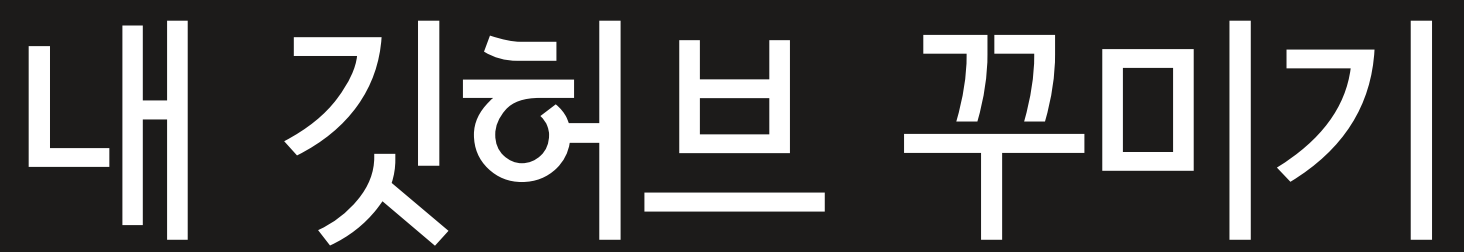
```
[$ git remote add origin [원격저장소주소] ]
```

```
[$ git push -u origin main ]
```


● 4 - 배포된 페이지 확인하기

내 페이지 주소 : [http://\[깃허브닉네임\].github.io](http://[깃허브닉네임].github.io)

카톡방에 공유해주세요~



● 3주차 정규세션 과제 두등

```
▼ 구예원 ●
  > 1주차
  ▼ 2주차 ●
    > images ●
    <> index.html U
    # style.css U
```

likelion-DKU-frontend/homework 깃허브에

- 자기 이름 폴더 안에 2주차 폴더 생성
- 2주차 폴더 안에 2주차 과제 업로드(자기소개 페이지)

POSSIBILITY  TO REALITY

 LIKELION UNIV.