

생활 코딩 깃허브 수업

저장소 생성

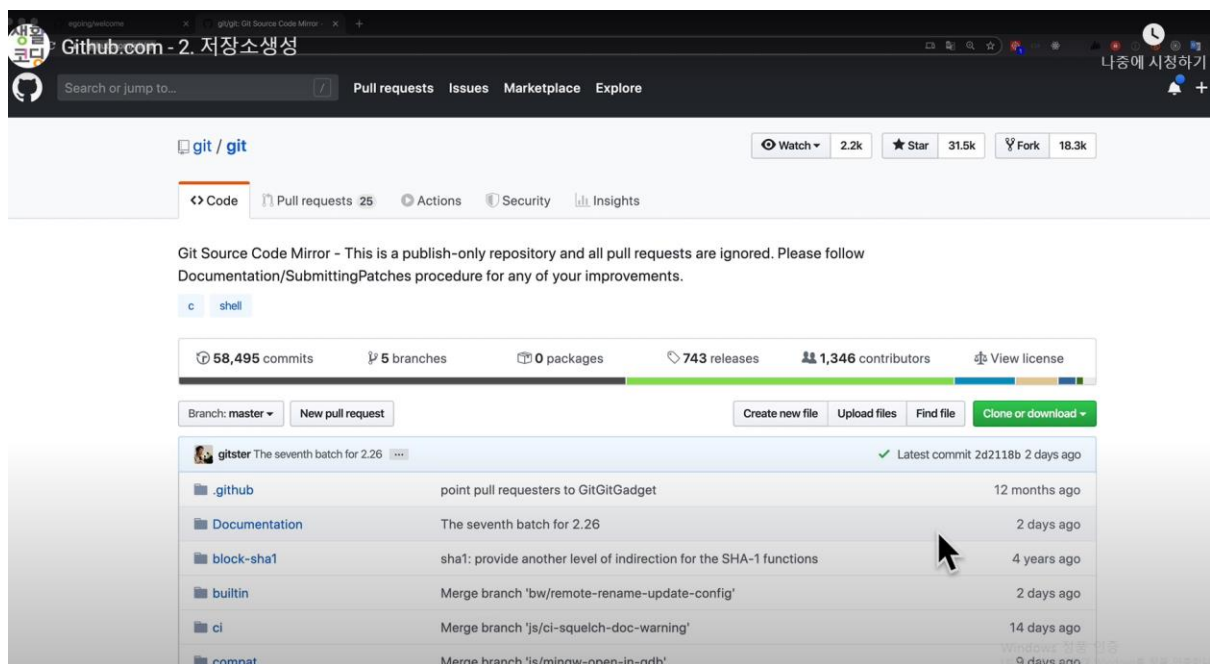
깃허브 서비스 안에서 프로젝트를 저장할 저장소를 만들어야함

각각의 프로젝트를 위한 소스코드 저장소 5개

깃허브 계정 상징성 있음.

깃은 깃허브와 다르다.

깃도 깃호브에서 관리됨.

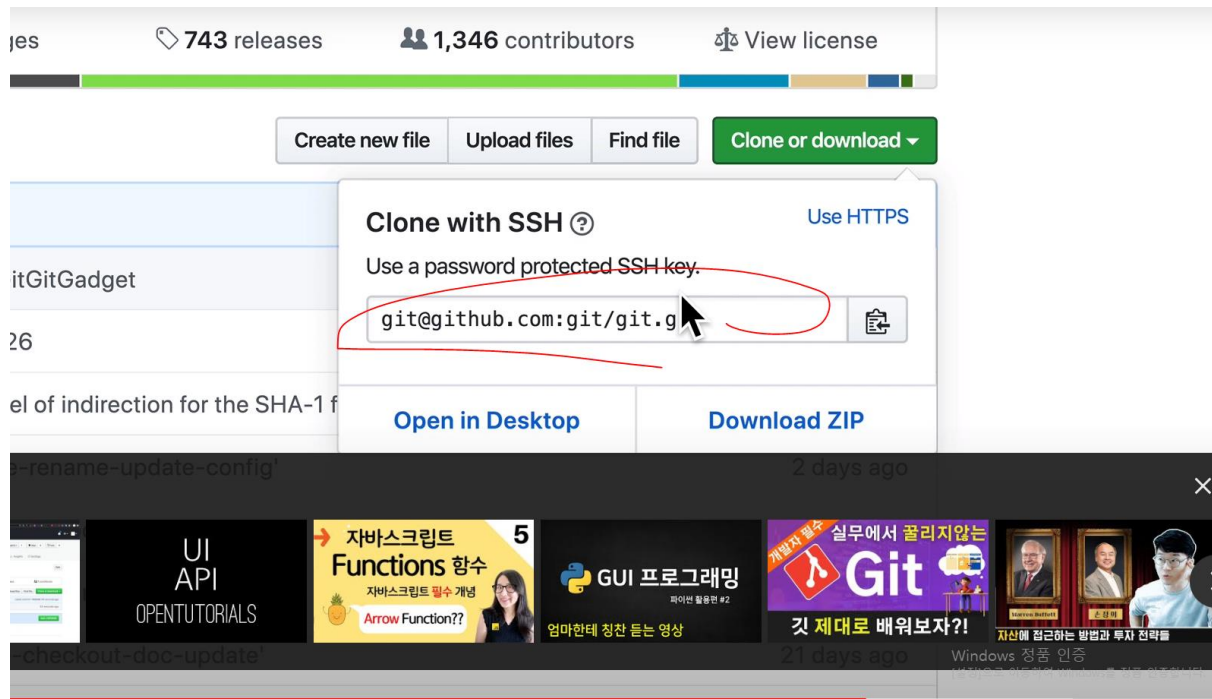


commits수를 보면 어떤 프로젝트인지 알 수 있음.

버전이 6만 개 가량

신뢰할 수 있는 프로젝트다.

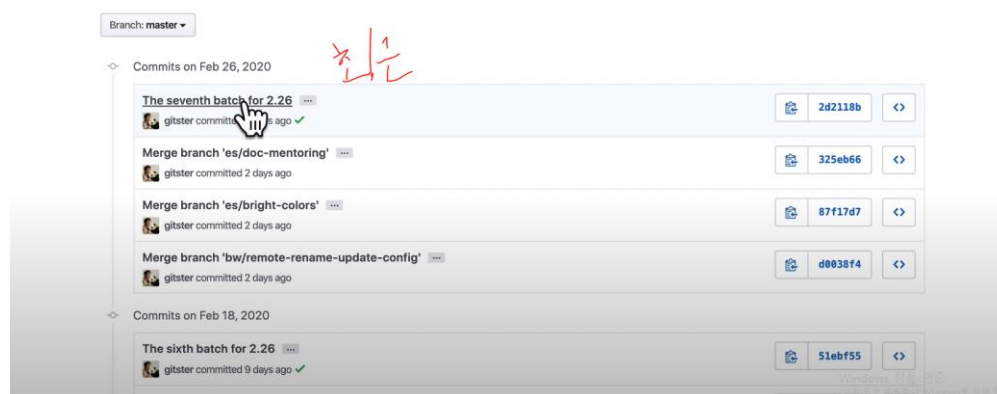
프로젝트를 사용하고 싶을때는 clone or download



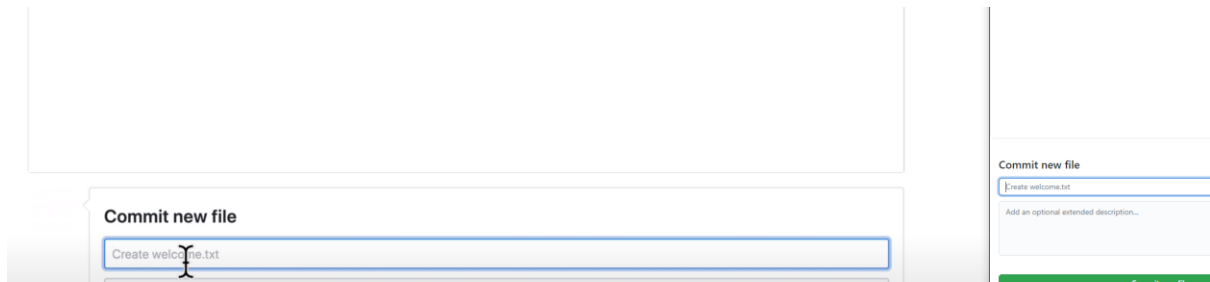
Git을 이용할 때 쓸 정보

굳이 쓰지 않는다면 다운로드 zip하면 그냥 받아짐.

파일들의 변화를 나타내는 것



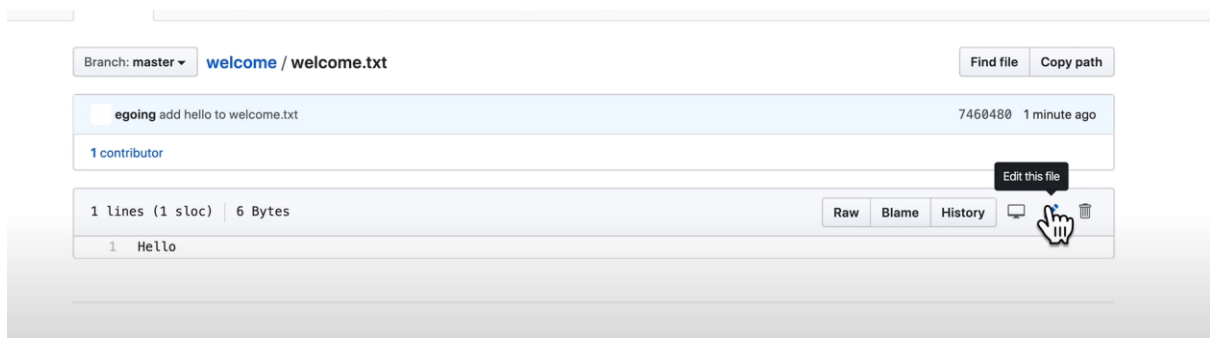




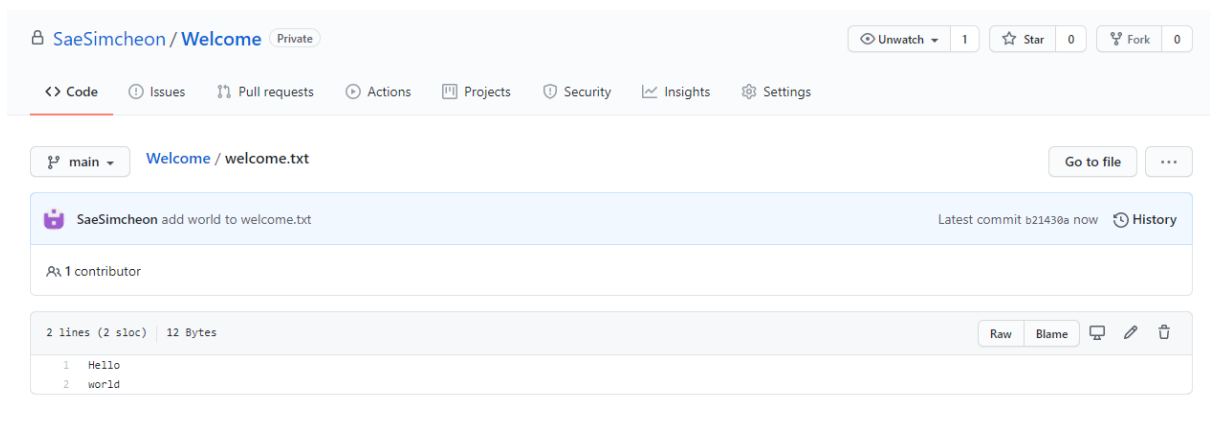
뭔가 업데이트 한다면 일지를 작성하게 됨.

일지 작성

Commit -> 제출하다. 파일에서 어떤 일을 했는지와 수정내용



만약 수정할 내용이 생겼다면



Issues Pull requests Actions Projects Security Insights Settings

main 1 branch 0 tags Go to file Add file Code

aeSimcheon add world to welcome.txt b21430a 1 minute ago 2 commits

welcome.txt add world to welcome.txt 1 minute ago

README with an overview of your project. Add a README

About No description provided.

Releases No releases published. Create a new release

Packages

확인 해보면

main

Commits on Feb 9, 2021

add world to welcome.txt SaeSimcheon committed 1 minute ago Verified b21430a

add hello to welcome.txt SaeSimcheon committed 4 minutes ago Verified b632c4b

Newer Older

작업 내용 잘 정리 되어 있음.

add world to welcome.txt Browse files

main

SaeSimcheon committed 2 minutes ago Verified 1 parent b632c4b commit b21430a94224aa230b1fe5309f2d39ea3c5b3c8b

Showing 1 changed file with 1 addition and 0 deletions. Unified Split

1 welcome.txt

```

@@ -1 +1,2 @@
1 Hello
2 + world

```

Comments on commit b21430a Lock conversation

Write Preview H B I  `< > @ ↩ ↪`

Leave a comment

Welcome /



Drag additional files here to add them to your repository

Or [choose your files](#)

file1.txt



file2.txt



### Commit changes

version3

Add an optional extended description...

☒ Commit directly to the `main` branch.

☐ Create a new branch for this commit and start a pull request. [Learn more about pull requests.](#)

Commit changes

Cancel

업로드로 버전 변화 확인

<> Code ⓘ Issues 🔗 Pull requests ▶ Actions 📁 Projects 🛡 Security 📊 Insights ⚙ Settings

version3

[Browse files](#)

main

SaeSimcheon committed 20 seconds ago

Verified

1 parent b21438a commit 742c1bca013490c5194482722c77d189f265f5d8

Showing 2 changed files with 2 additions and 0 deletions.

Unified Split

1 file1.txt

@@ -0,0 +1 @@  
+ version3

1 file2.txt

@@ -0,0 +1 @@  
+ version3

0 comments on commit 742c1bc

[Lock conversation](#)

버전 3에 속하는 변화들

버전 4

The screenshot shows a GitHub commit interface. At the top, there's a navigation bar with links for Code, Issues, Pull requests, Actions, Projects, Security, Insights, and Settings. Below this, a section titled 'Add files via upload' shows a commit by SaeSimcheon 16 seconds ago, verified, with parent 742c1bc and commit hash c63198d3fe854f2a7b0d6a031ed49e6af725ce09. The commit message is 'Showing 2 changed files with 3 additions and 1 deletion.' Below this, two file change sections are shown. The first section for 'file2.txt' shows a deletion of 'version3' and additions of 'version3' and 'version4'. The second section for 'file3.txt' shows an addition of 'version4'. At the bottom, there's a comment section with '0 comments on commit c63198d' and a 'Write' button.

## Git 구경하기

개발자들이 실제로 git을 사용하는 모습 -> 구경

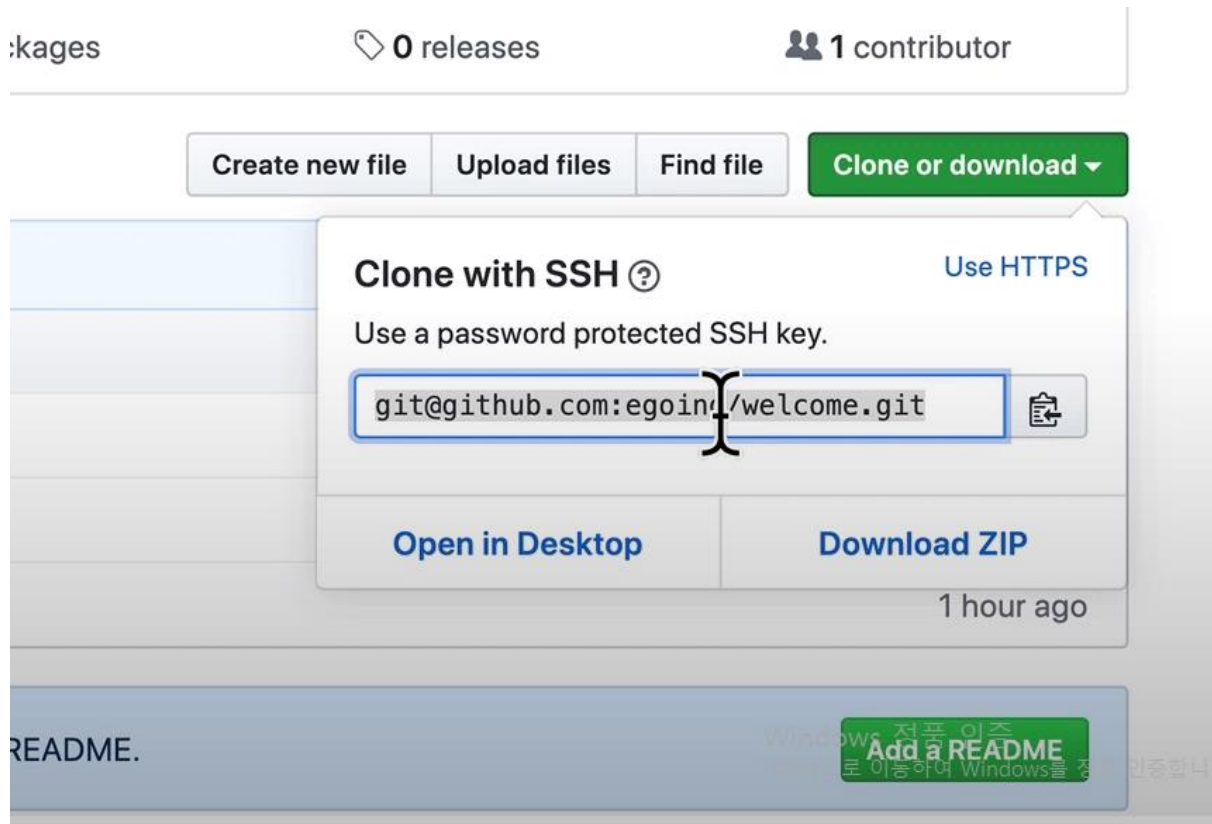
아 ~ 저렇게 하면 편하겠다.

내 컴퓨터에서 작업한 것을 쉽게 쉽게 보내려면?

Git이 내 컴퓨터에 깔아야함.

내 컴퓨터와 깃 허브와 연결 시켜서 서로가 주고 받도록 해야함.

일단 깃허브 주소를 알아야함 -> 인터넷 말고 저장소의 주소



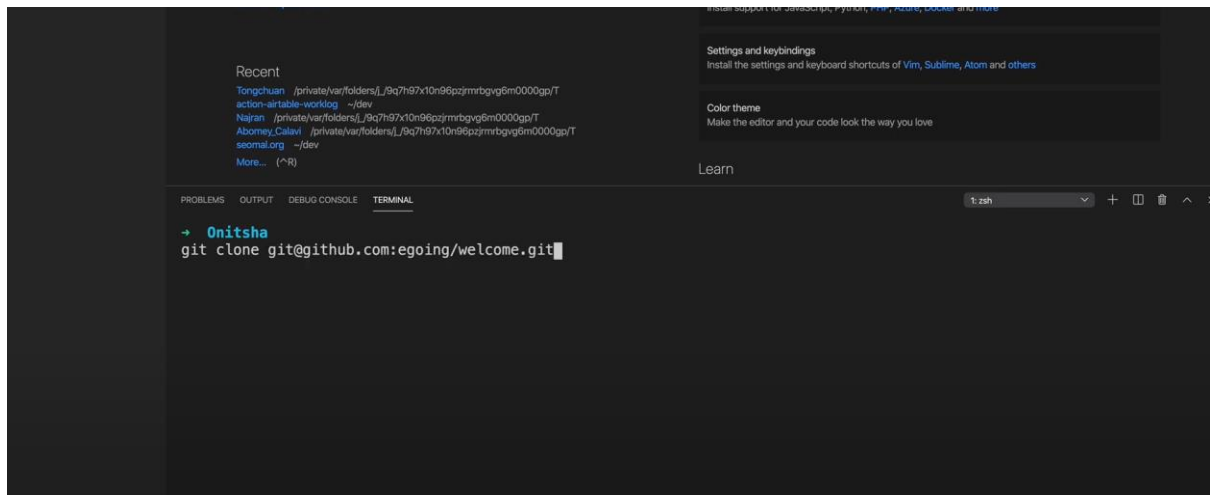
깃허브 상의 저장소를

그리고 복제할 것임. -> clone

컴퓨터한테 말로 명령을 시켜볼 것임.

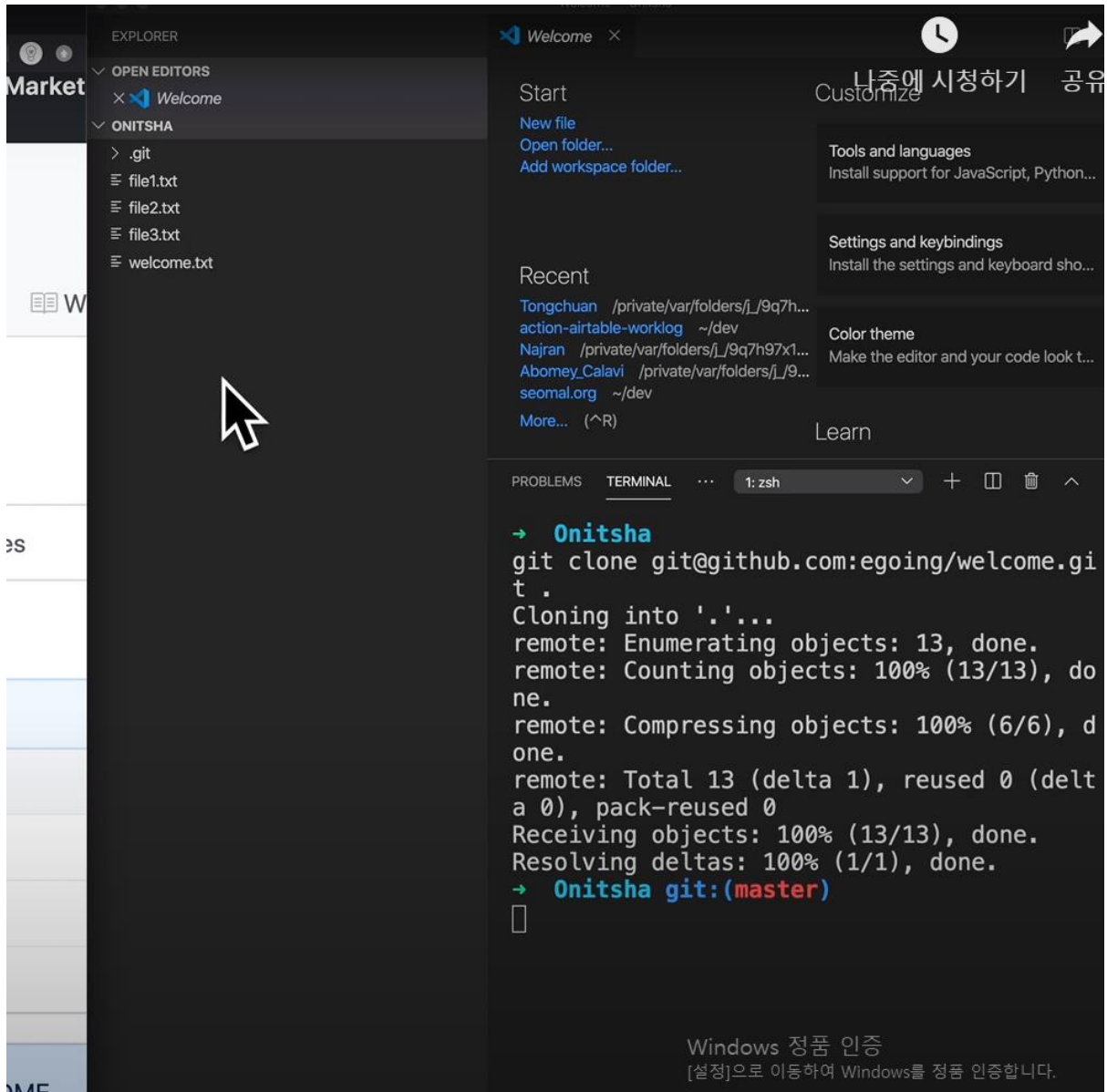
컴퓨터 한테





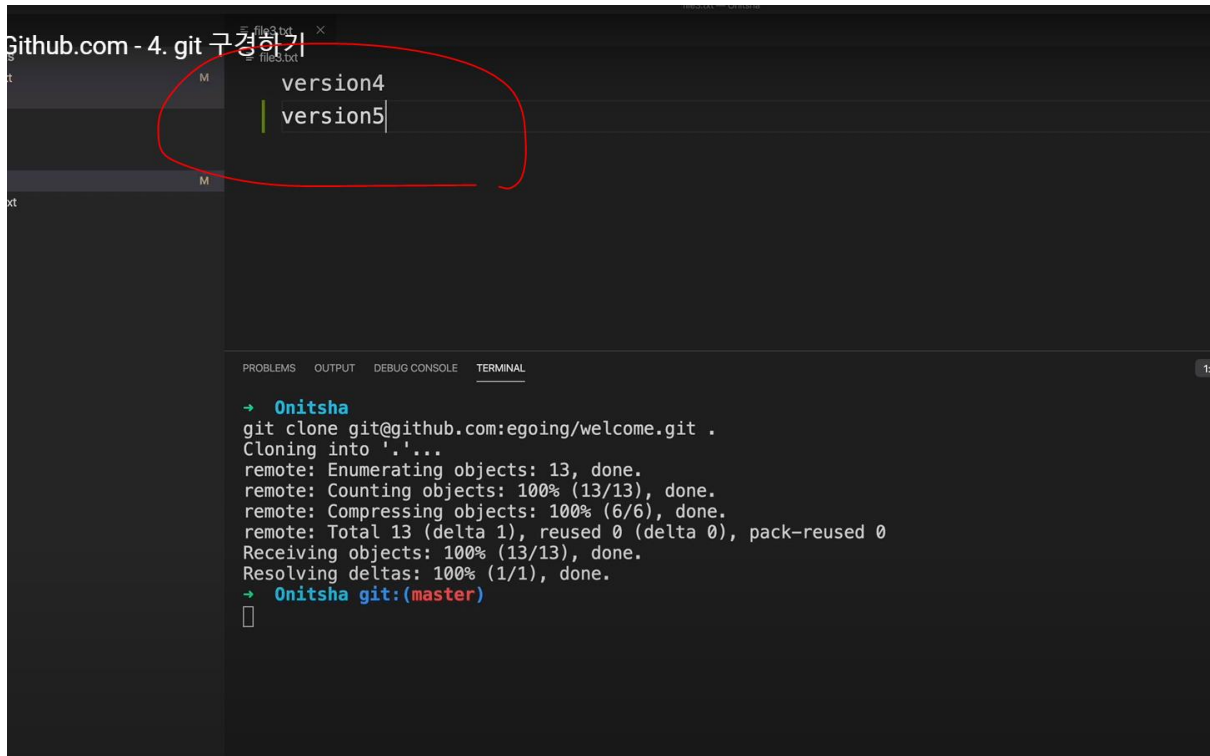
. -> 현재 디렉터리로

Git이



똑같은 상태가 됨.

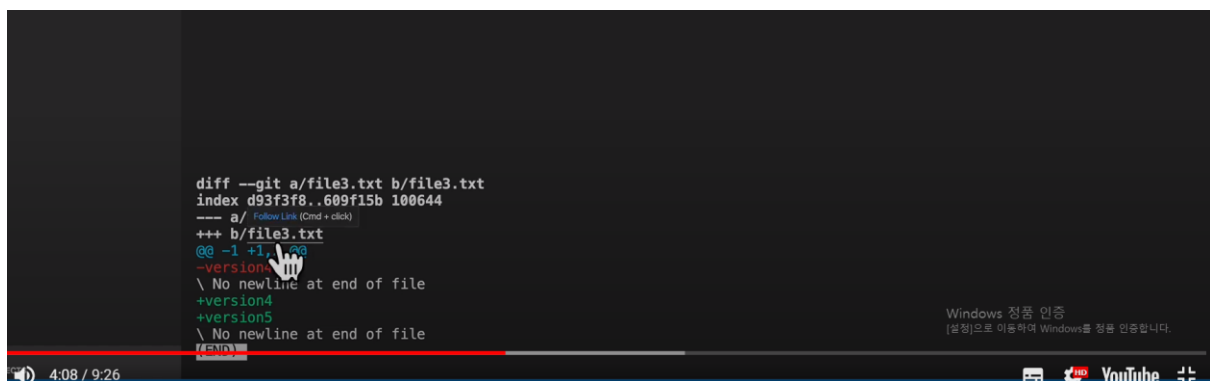
이상태에서 파일을 수정한다면?



수정했음.

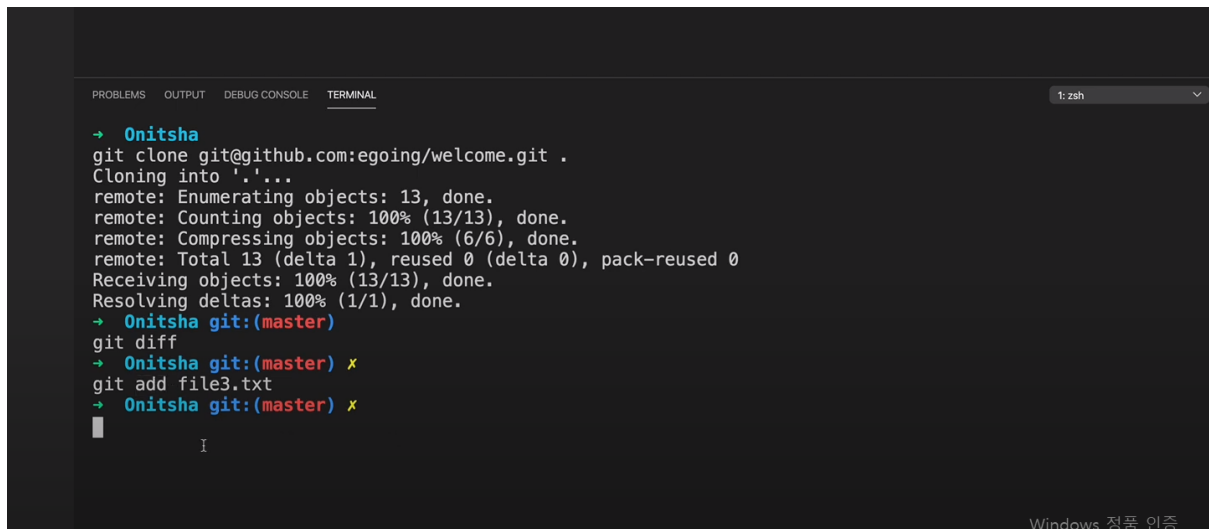
수정했다면 컴퓨터에 설치된 git에서 마지막 버전 이후에 내가 수정한 것이 뭔지 보여줘.

➔ 이거 진짜 중요한 것임



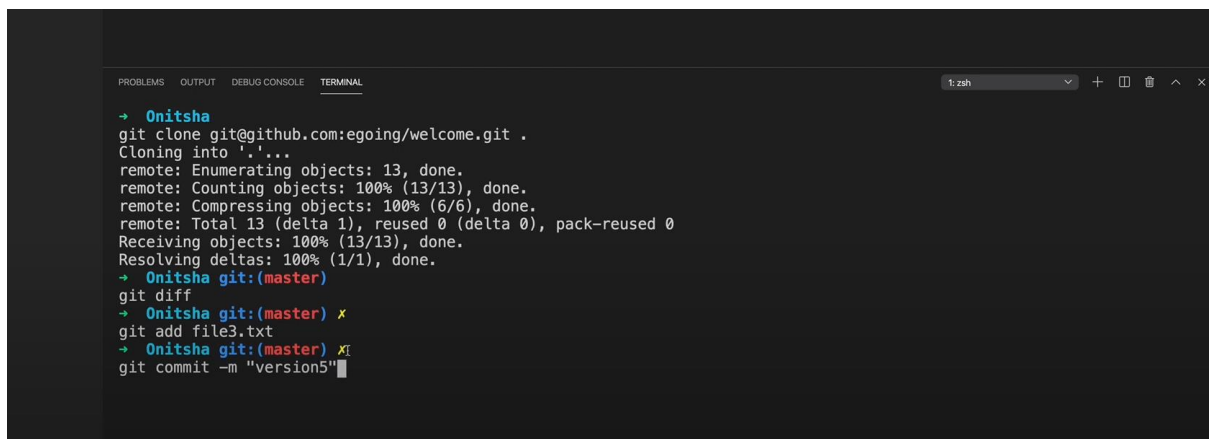
이거 추가했어요. 수정한 사항 쉽게 볼 수 있음.

## 커밋하는 방법



```
→ Onitsha
git clone git@github.com:egoing/welcome.git .
Cloning into '.'...
remote: Enumerating objects: 13, done.
remote: Counting objects: 100% (13/13), done.
remote: Compressing objects: 100% (6/6), done.
remote: Total 13 (delta 1), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (13/13), done.
Resolving deltas: 100% (1/1), done.
→ Onitsha git:(master)
git diff
→ Onitsha git:(master) x
git add file3.txt
→ Onitsha git:(master) x
```

Windows 정품 인증



```
→ Onitsha
git clone git@github.com:egoing/welcome.git .
Cloning into '.'...
remote: Enumerating objects: 13, done.
remote: Counting objects: 100% (13/13), done.
remote: Compressing objects: 100% (6/6), done.
remote: Total 13 (delta 1), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (13/13), done.
Resolving deltas: 100% (1/1), done.
→ Onitsha git:(master)
git diff
→ Onitsha git:(master) x
git add file3.txt
→ Onitsha git:(master) x
git commit -m "version5"
```

지금 수정한 내용을 version5로 만들어라

git에서 hisotry를 보여줘

➔ Git log

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: zsh + [ ] [x] [^] [v]
→ Onitsha
git clone git@github.com:egoing/welcome.git .
Cloning into '.'...
remote: Enumerating objects: 13, done.
remote: Counting objects: 100% (13/13), done.
remote: Compressing objects: 100% (6/6), done.
remote: Total 13 (delta 1), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (13/13), done.
Resolving deltas: 100% (1/1), done.
→ Onitsha git:(master)
git diff
→ Onitsha git:(master) x
git add file3.txt
→ Onitsha git:(master) x
git commit -m "version5"
[master 60742ac] version5
1 file changed, 2 insertions(+), 1 deletion(-)
→ Onitsha git:(master)
git log
```

Windows 정품 인증  
[설정]으로 이동하여 Windows를 정품 인증합니다.

9:26 [ ] [x] [^] [v] YouTube [ ]

컴퓨터에서 만든 버전을 동기화 시켜야함 깃허브로

지역저장소를 원격저장소로 밀어넣어

Push해

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: zsh + [ ] [x] [^] [v] x
→ Onitsha git:(master)
git diff
→ Onitsha git:(master) x
git add file3.txt
→ Onitsha git:(master) x
git commit -m "version5"
[master 60742ac] version5
1 file changed, 2 insertions(+), 1 deletion(-)
→ Onitsha git:(master)
git log
→ Onitsha git:(master)
git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 259 bytes | 259.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:egoing/welcome.git
dfd07df..60742ac master -> master
→ Onitsha git:(master)
```

Windows 정품 인증  
[설정]으로 이동하여 Windows를 정품 인증합니다.

[ ] [x] [^] [v] YouTube [ ]

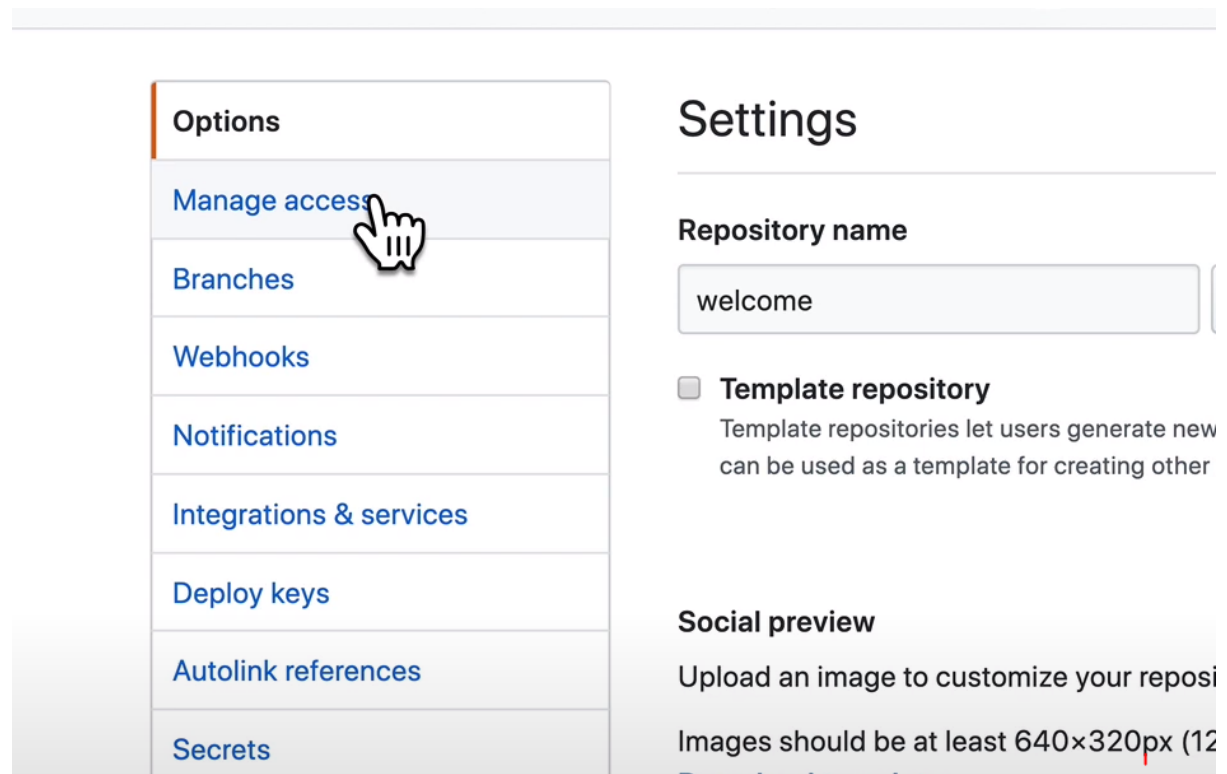
파일을 하나하나 업로드할 필요 없이

Git 이라는 이름의 어떤 프로그램이 git임.

업로드를 했을 때 그 버전을 보관하고 관리해주는 여러 솔루션 중 하나.

Git1 등 수업 들을 것.

협업



## Manage access



### You haven't invited any collaborators yet

If you're using GitHub Free, you can add unlimited collaborators on public repositories, and up to three collaborators on private repositories owned by your personal account. [Learn more](#)

Invite a collaborator



Windows 창용 인증  
[설정]으로 이동하여 Windows를 창용 인증합니다.

PUBLIC REPOSITORY

DIRECT ACCESS

This repository is public and

0 collaborators have access to



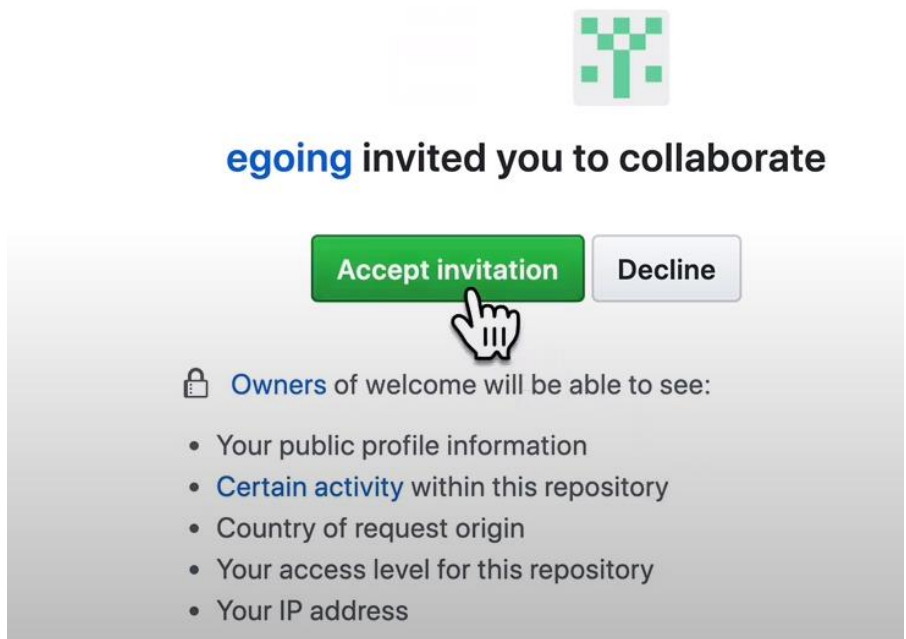
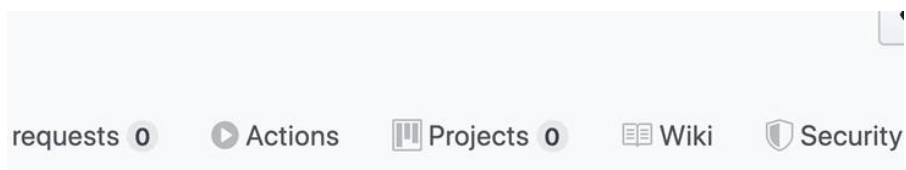
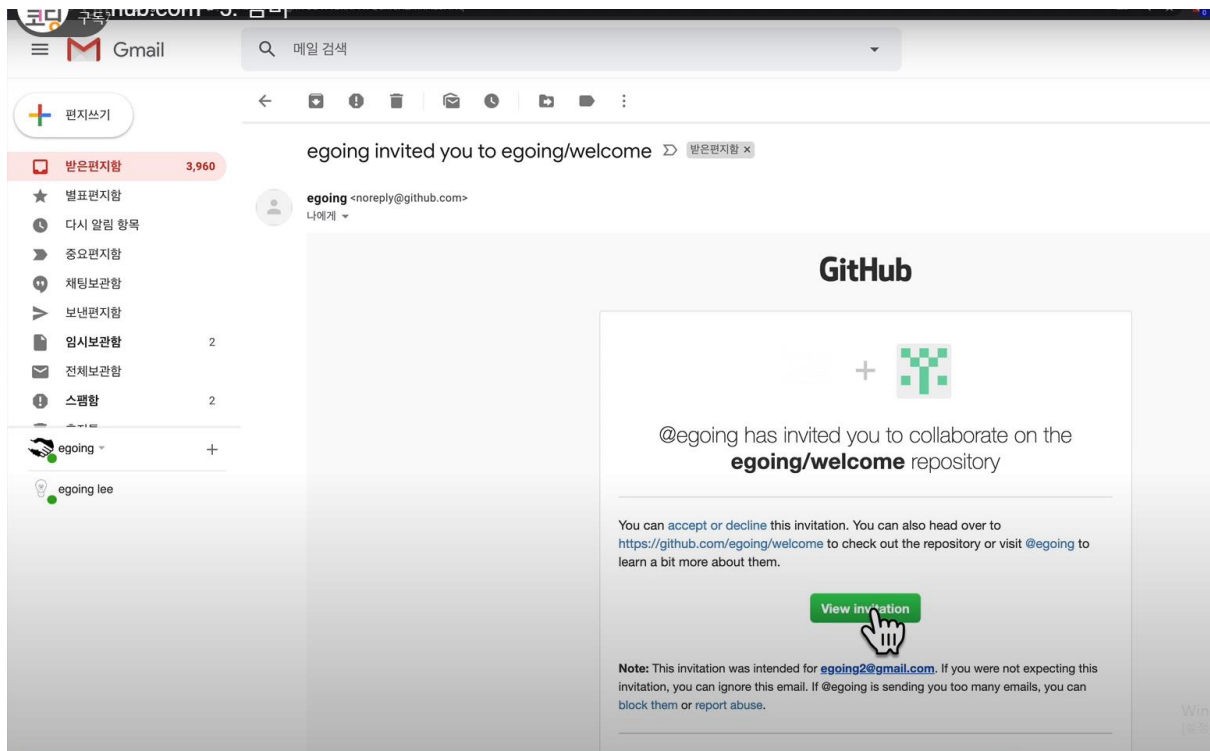
### Invite a collaborator to welcome

Search by username, full name, or email address. Please fill out this field.

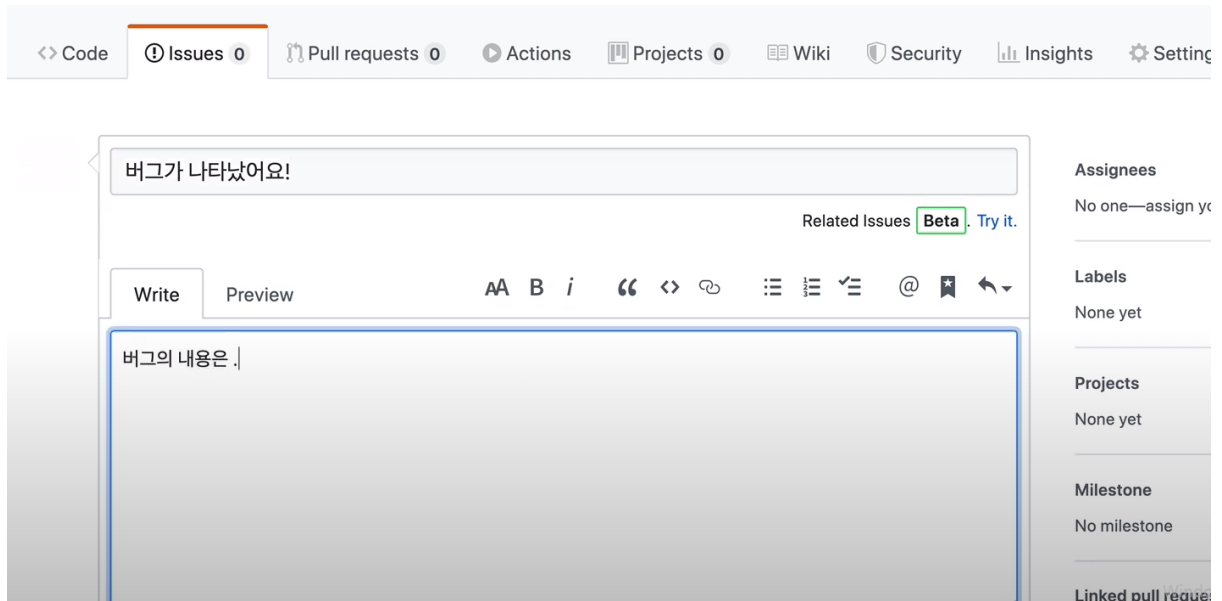
Select a collaborator above

### You haven't invited any collaborators yet

If you're using GitHub Free, you can add unlimited collaborators on public repositories, and up to three collaborators on private repositories owned by your personal account. [Learn more](#)







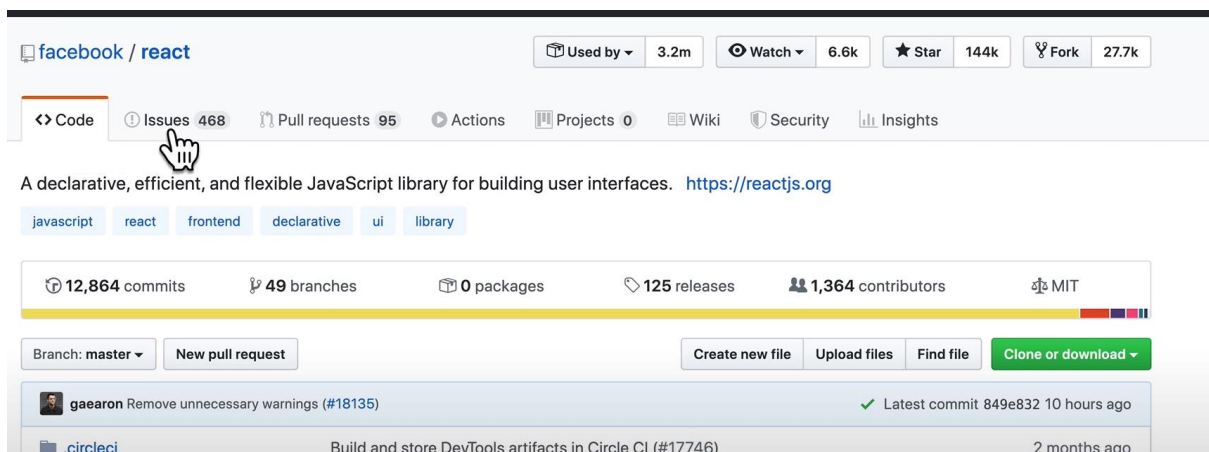
이슈 관리

담당자 지정 가능 -> 알림 감

Label -> 그 리스트 중에서 bug, documentation..

Invalid 필요 없음

수업을 마치며



위 순서대로 중요함.

다른 것들은 후속수업에서 다룰 것.

기능들만 언급.

Wiki란? 프로젝트를 진행 하다보면, 지식을 정리정돈할 필요성이 있음.

프로젝트와 관련된 지식들

위키피디아처럼

여러 지표들을 확인 할 수 있는 방법 -> insights

Fork -> 내 것이 아니지만, 내 것인 것처럼 저장소에 복제할 수 있음.

내 마음대로 수정할 수 있음. 포크의 숫자가 높다는 것은 복제가 많고 인기있는 프로젝트.

Pull requests -> 복제하면 마음대로 수정할 수 있음. 내가 작업한 것을 가져가도 좋을 것 같아요. 가치있어요.

Actions -> 일종의 예제들이 있는 것 -> push 저장소로 작업자들이 push를 하면 밑의 작업들을 시행하도록 git hub 한테 setting할 수 있음.

누군가가 우리 저장소로 push 한다면