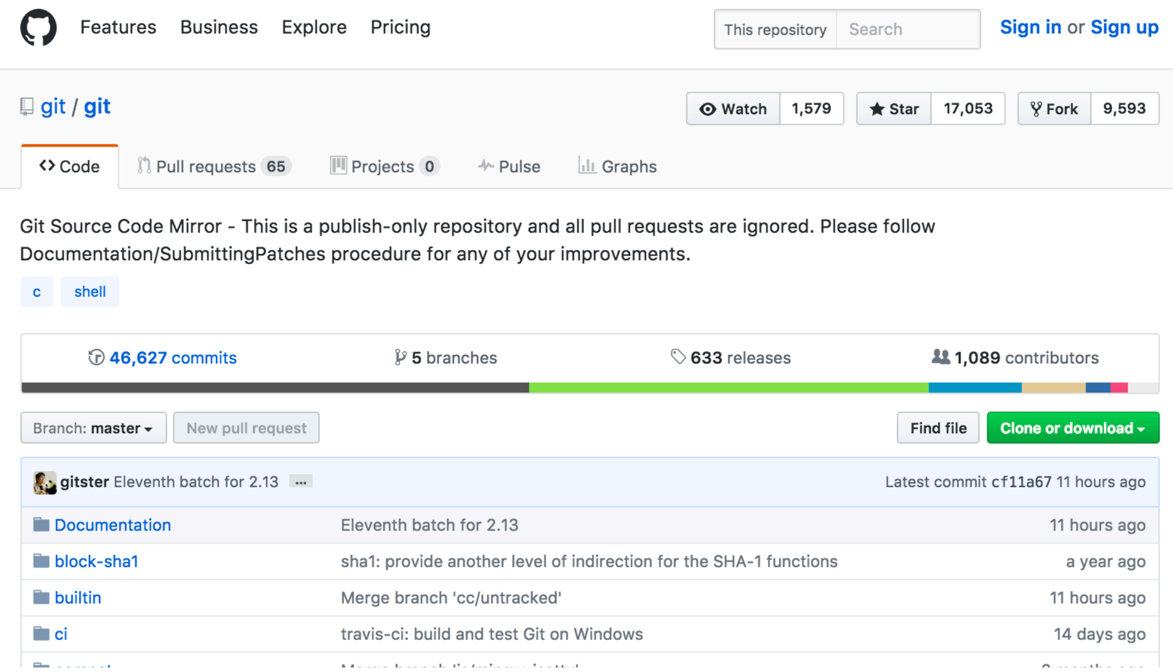
Github

1)원격 저장소를 지역 저장소로 복제(Github)

git의 원격저장소 링크

https://github.com/git/git



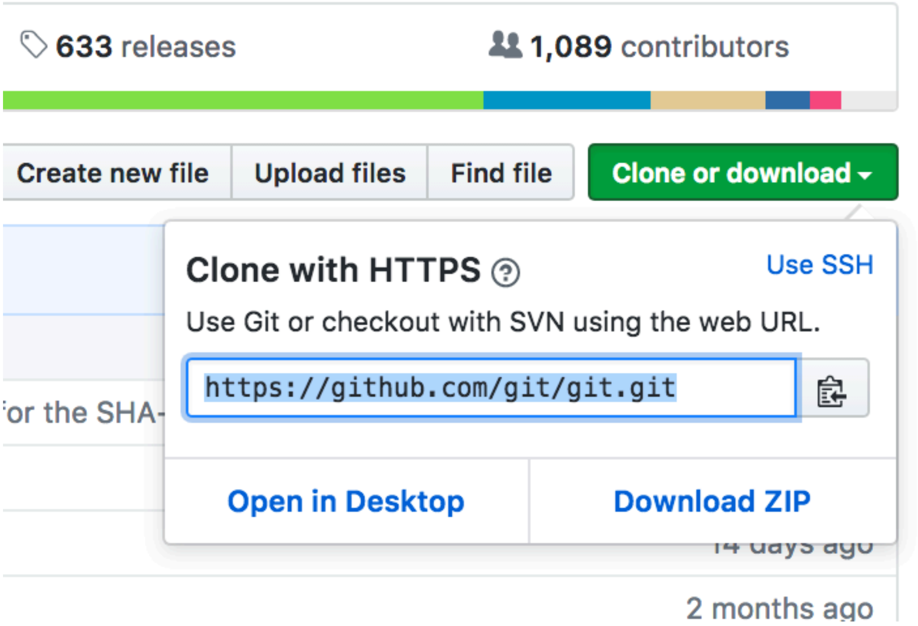
Git Source Code Mirror - This is a publish-only repository and all pull requests are ignored. Please follow Documentation/SubmittingPatches procedure for any of your improvements.

gitster Eleventh batch for 2.13 ... Latest commit cf11a67 11 hours ago

Documentation	Eleventh batch for 2.13	11 hours ago
block-sha1	sha1: provide another level of indirection for the SHA-1 functions	a year ago
builtin	Merge branch 'cc/untracked'	11 hours ago
ci	travis-ci: build and test Git on Windows	14 days ago

Fork를 하면 마음대로 소스코드를 수정할 수 있는 자유가 생긴다.

회원가입 로그인 을 하지 않아도....



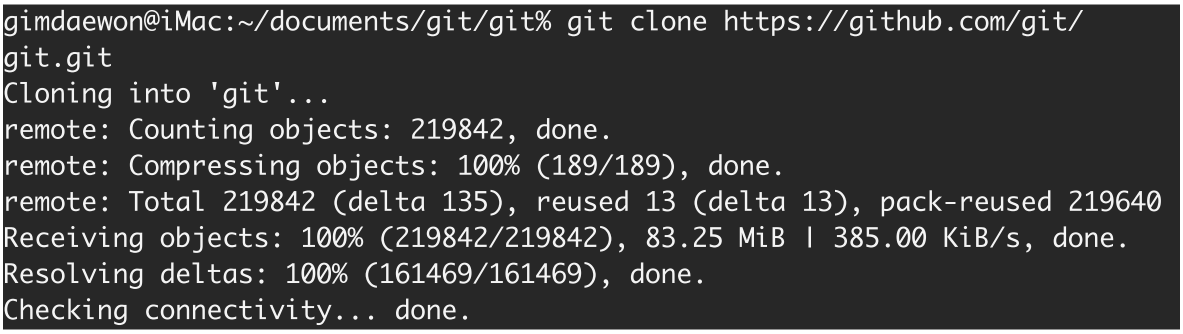
Clone with HTTPS

Use Git or checkout with SVN using the web URL.

```
https://github.com/git/git.git
```

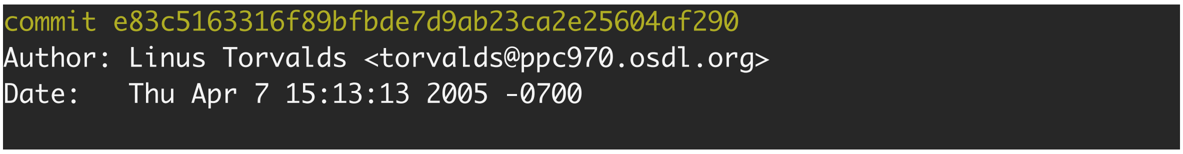
소스코드를 복사한다.

내 컴퓨터에 적당한 디렉토리를 생성해
>git clone https://github.com/git/git.git
//>git clone https://github.com/git/git.git gitsrc
//gitsrc라는 폴더에 저장

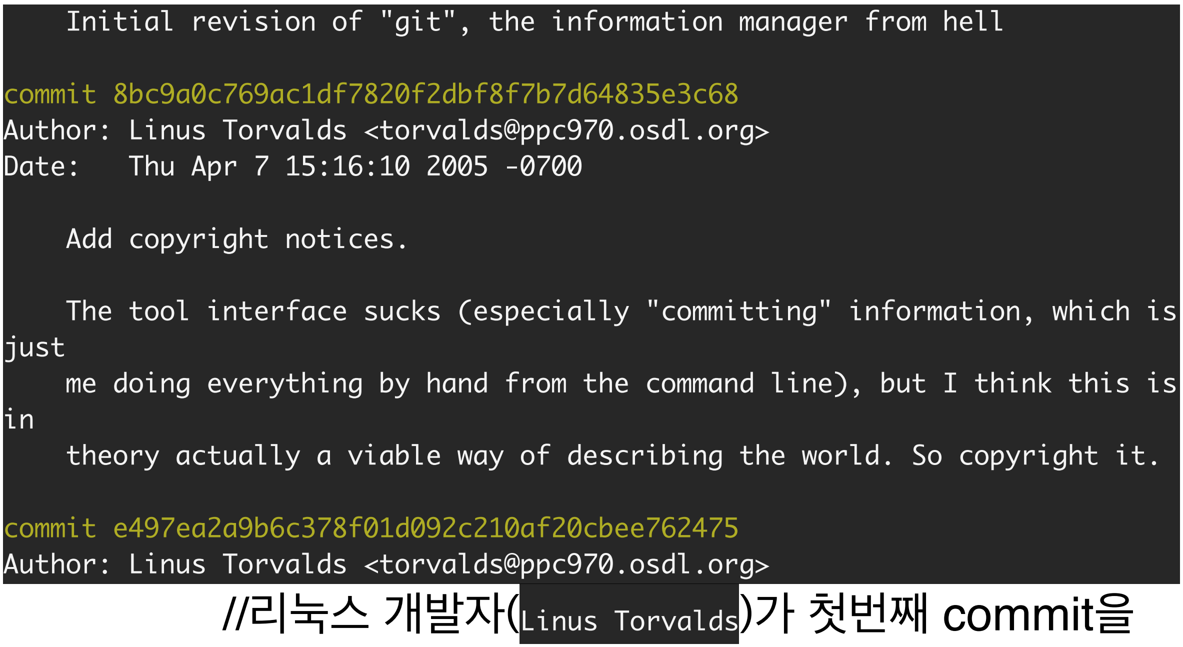


```
gimdaewon@iMac:~/documents/git/git% git clone https://github.com/git/
git.git
Cloning into 'git'...
remote: Counting objects: 219842, done.
remote: Compressing objects: 100% (189/189), done.
remote: Total 219842 (delta 135), reused 13 (delta 13), pack-reused 219640
Receiving objects: 100% (219842/219842), 83.25 MiB | 385.00 KiB/s, done.
Resolving deltas: 100% (161469/161469), done.
Checking connectivity... done.
```

>git log //엄청많은 로그들을 볼 수 있다.
>git log -reverse //로그를 거꾸로 출력해 첫번째 로그를 확인할 수 있다.



```
commit e83c5163316f89bfbde7d9ab23ca2e25604af290
Author: Linus Torvalds <torvalds@ppc970.osdl.org>
Date:   Thu Apr 7 15:13:13 2005 -0700
```

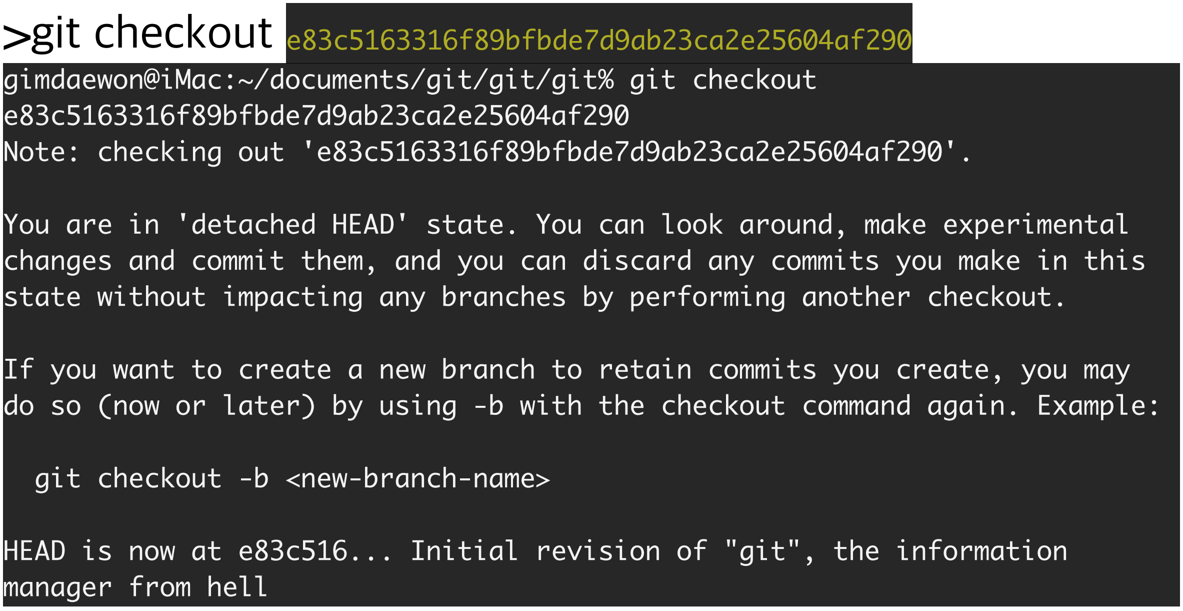


```
    Initial revision of "git", the information manager from hell

commit 8bc9a0c769ac1df7820f2dbf8f7b7d64835e3c68
Author: Linus Torvalds <torvalds@ppc970.osdl.org>
Date:   Thu Apr 7 15:16:10 2005 -0700

    Add copyright notices.

    The tool interface sucks (especially "committing" information, which is
just
    me doing everything by hand from the command line), but I think this is
in
    theory actually a viable way of describing the world. So copyright it.

commit e497ea2a9b6c378f01d092c210af20cbee762475
Author: Linus Torvalds <torvalds@ppc970.osdl.org>
```

//리눅스 개발자(`Linus Torvalds`)가 첫번째 commit을 했다.

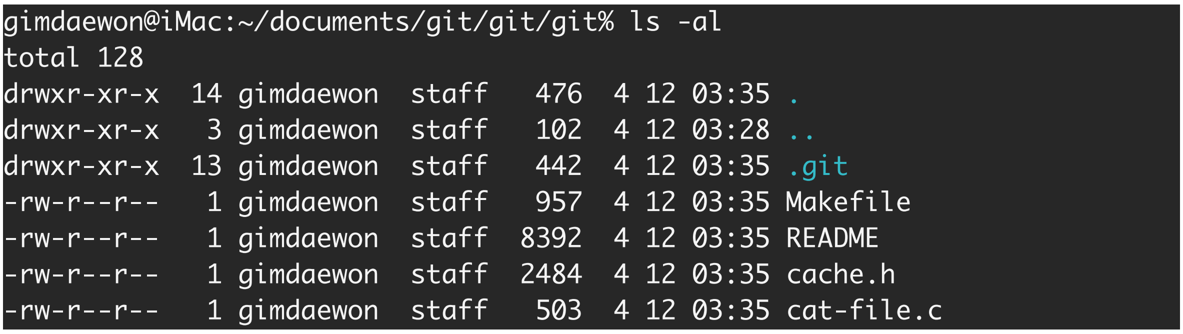
git의 첫번째 커밋으로 체크아웃 하기

>git checkout `e83c5163316f89bfbde7d9ab23ca2e25604af290`

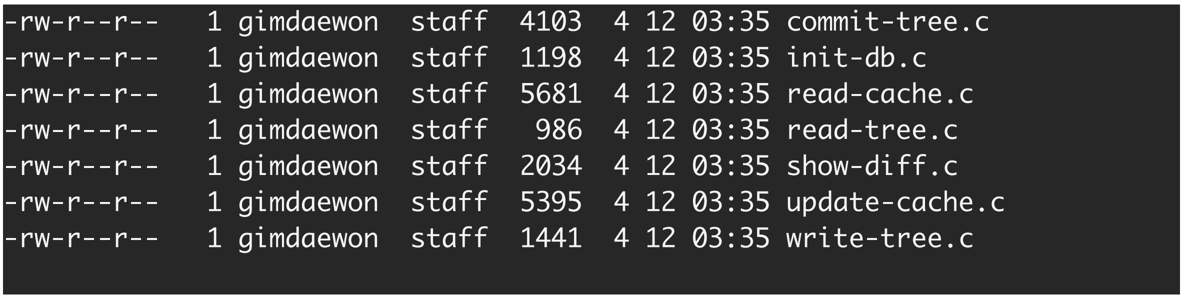


```
gimdaewon@iMac:~/documents/git/git/git% git checkout
e83c5163316f89bfbde7d9ab23ca2e25604af290
Note: checking out 'e83c5163316f89bfbde7d9ab23ca2e25604af290'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by performing another checkout.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -b with the checkout command again. Example:

  git checkout -b <new-branch-name>

HEAD is now at e83c516... Initial revision of "git", the information
manager from hell
```

>ls -al



```
gimdaewon@iMac:~/documents/git/git/git% ls -al
total 128
drwxr-xr-x  14 gimdaewon  staff   476  4 12 03:35 .
drwxr-xr-x   3 gimdaewon  staff   102  4 12 03:28 ..
drwxr-xr-x  13 gimdaewon  staff   442  4 12 03:35 .git
-rw-r--r--   1 gimdaewon  staff   957  4 12 03:35 Makefile
-rw-r--r--   1 gimdaewon  staff  8392  4 12 03:35 README
-rw-r--r--   1 gimdaewon  staff  2484  4 12 03:35 cache.h
-rw-r--r--   1 gimdaewon  staff   503  4 12 03:35 cat-file.c
```

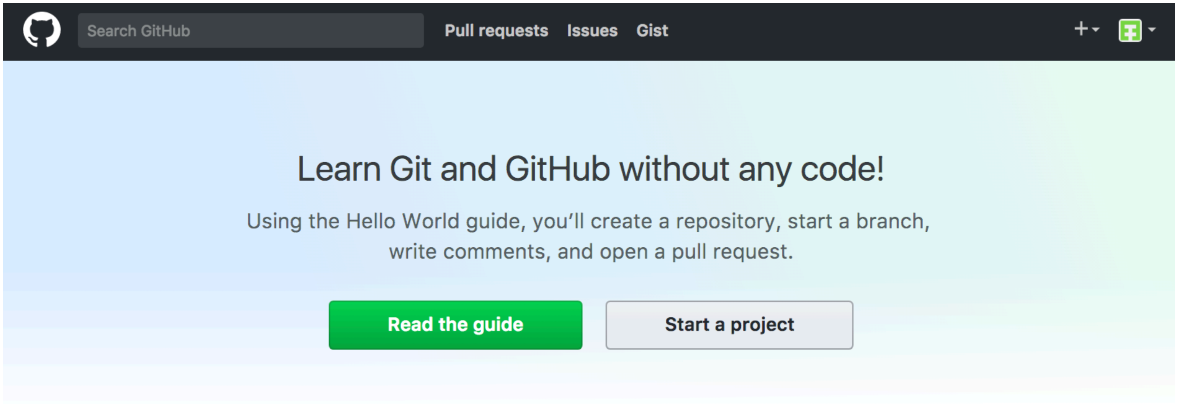


```
-rw-r--r--    1 gimdaewon  staff  4103  4 12 03:35 commit-tree.c
-rw-r--r--    1 gimdaewon  staff  1198  4 12 03:35 init-db.c
-rw-r--r--    1 gimdaewon  staff  5681  4 12 03:35 read-cache.c
-rw-r--r--    1 gimdaewon  staff   986  4 12 03:35 read-tree.c
-rw-r--r--    1 gimdaewon  staff  2034  4 12 03:35 show-diff.c
-rw-r--r--    1 gimdaewon  staff  5395  4 12 03:35 update-cache.c
-rw-r--r--    1 gimdaewon  staff  1441  4 12 03:35 write-tree.c
```

얼마 되지않은 파일들로 구성되게 되었고 분석할만한 양들이다…

2)원격 저장소 만들기(GitHub)

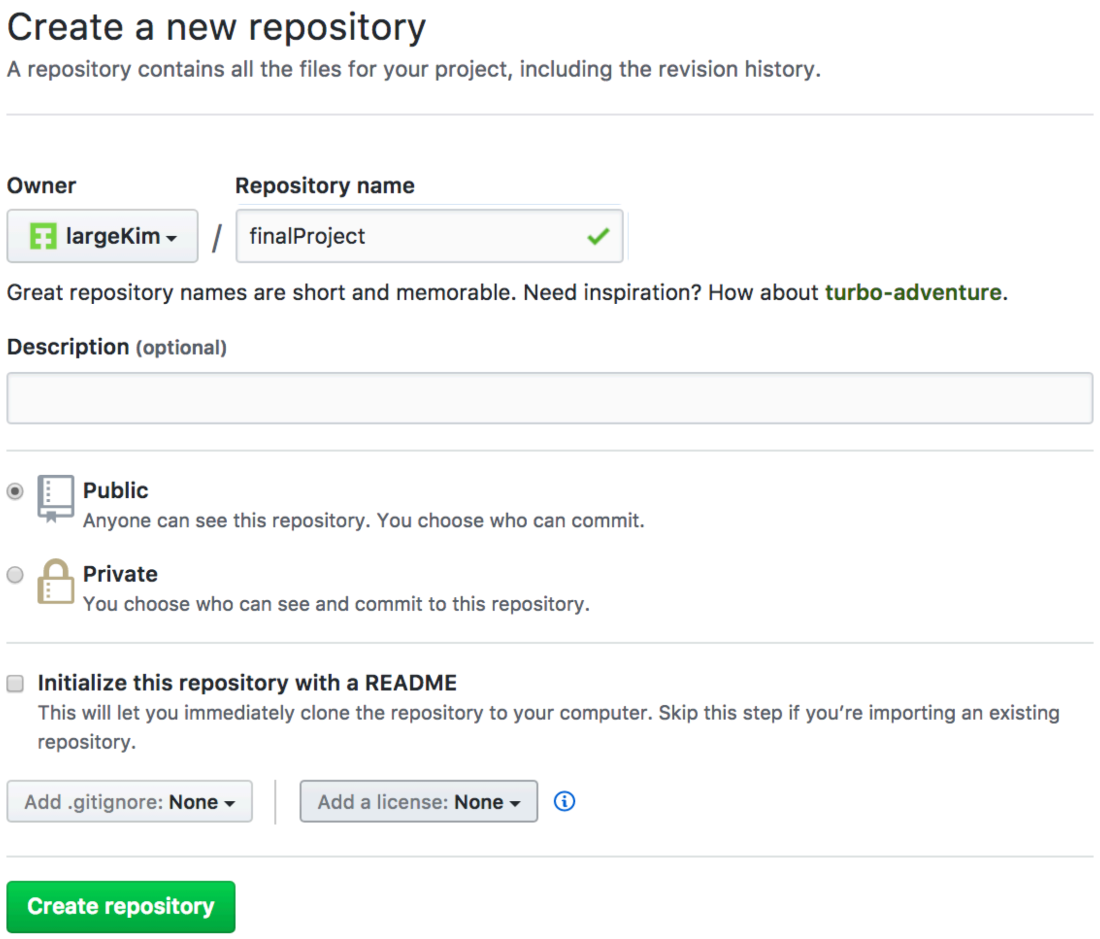
(1)Start a project 클릭



Learn Git and GitHub without any code!

Using the Hello World guide, you'll create a repository, start a branch, write comments, and open a pull request.

(2)Repository name을 입력후 public(무료)를 선택하고 Create repository 버튼 클릭!



Create a new repository

A repository contains all the files for your project, including the revision history.

Owner: largeKim / Repository name: finalProject ✓

Great repository names are short and memorable. Need inspiration? How about **turbo-adventure**.

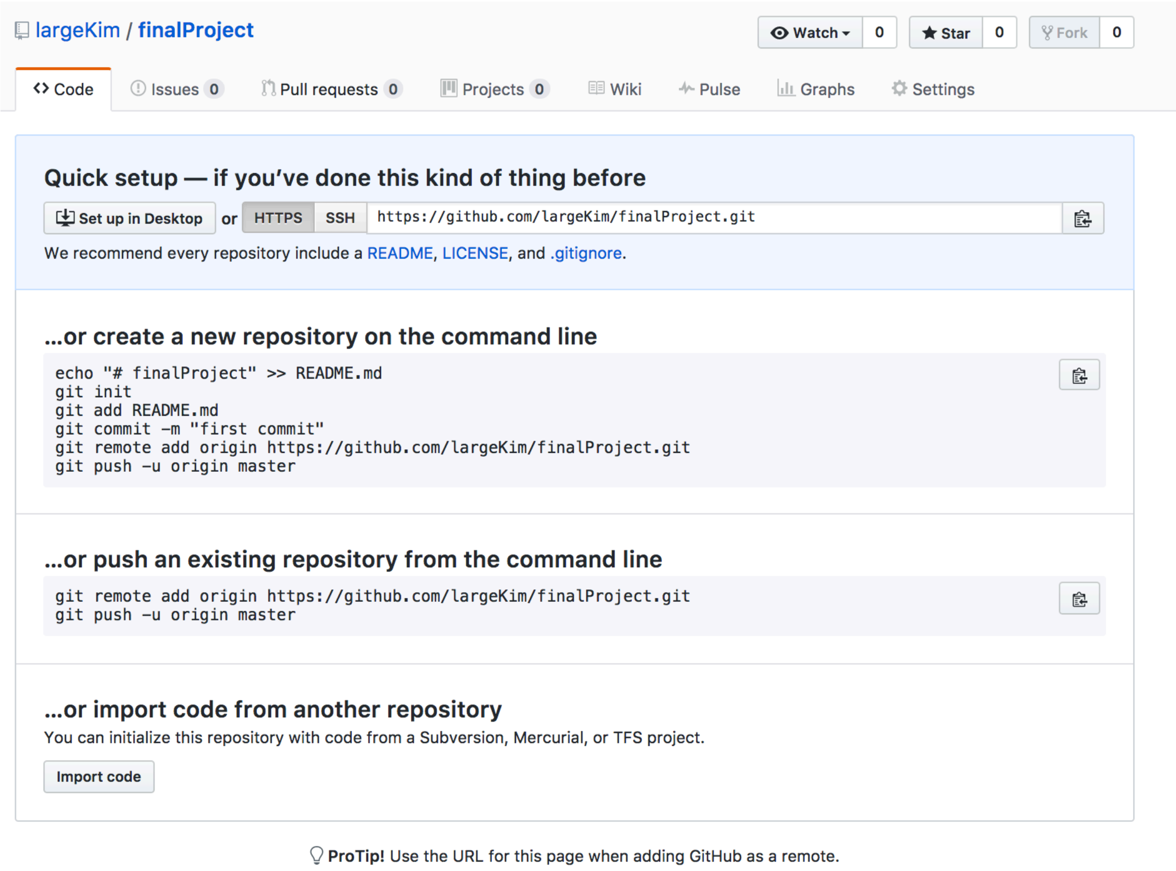
Description (optional)

● Public
Anyone can see this repository. You choose who can commit.

○ Private
You choose who can see and commit to this repository.

□ Initialize this repository with a README
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

(3)



Quick setup — if you've done this kind of thing before

https://github.com/largeKim/finalProject.git

We recommend every repository include a README, LICENSE, and .gitignore.

…or create a new repository on the command line

```
echo "# finalProject" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/largeKim/finalProject.git
git push -u origin master
```

…or push an existing repository from the command line

```
git remote add origin https://github.com/largeKim/finalProject.git
git push -u origin master
```

…or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

ProTip! Use the URL for this page when adding GitHub as a remote.

첫번째

…or create a new repository on the command line

-> 원격저장소를 만들고 그것을 복제해서, 로컬저장소를 만든 다음에 로컬저장소에서 작업

두번째

…or push an existing repository from the command line

-> 이미 로컬저장소에서 작업을 해왔고, 원격저장소로 작업 했던 내용을 올리는

지금은 두번째로 한다.

적당한 폴더를 생성한후

>git init

>vim f1.txt //생성

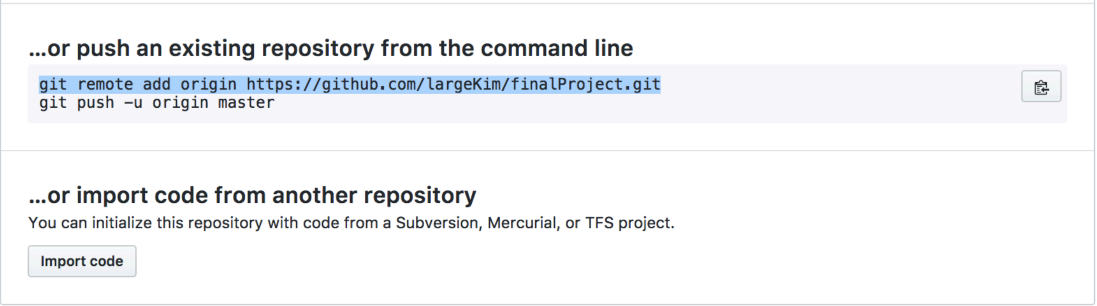
```
a
~
```

```
~
```

>git add f1.txt
>git commit -m '1'

github로 돌아가서

첫번째 블록지정된 부분을 copy



...or push an existing repository from the command line

```
git remote add origin https://github.com/largeKim/finalProject.git
git push -u origin master
```

...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

>git remote add origin https://github.com/largeKim/finalProject.git

//현재 원격저장소 remote repository를 연결(add)시키며, origin이라는 별명으로

// 그뒤에 경로로 연결할 것이다.

>git remote

```
gimdaewon@iMac:~/documents/git/gitfth10% git remote
origin
```

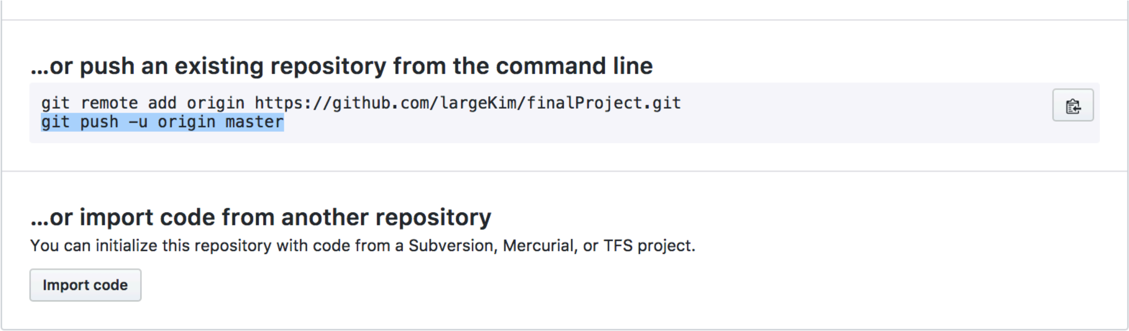
>git remote -v

```
gimdaewon@iMac:~/documents/git/gitfth10% git remote -v
origin	https://github.com/largeKim/finalProject.git (fetch)
origin	https://github.com/largeKim/finalProject.git (push)
```

// >git remote add friend https://github.com/largeKim/finalProject.git

// 동일한 주소를 friend 라는 다른 이름의 별명으로 저장도 가능하다.

//지우고 싶을때는 >git remote remove '별명' 으로 삭제가 가능하다.



...or push an existing repository from the command line

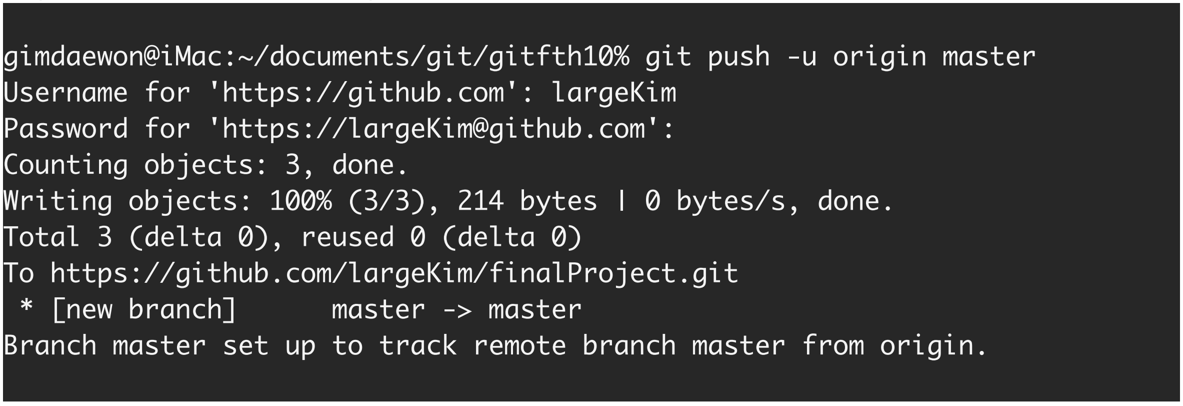
```
git remote add origin https://github.com/largeKim/finalProject.git
git push -u origin master
```

...or import code from another repository

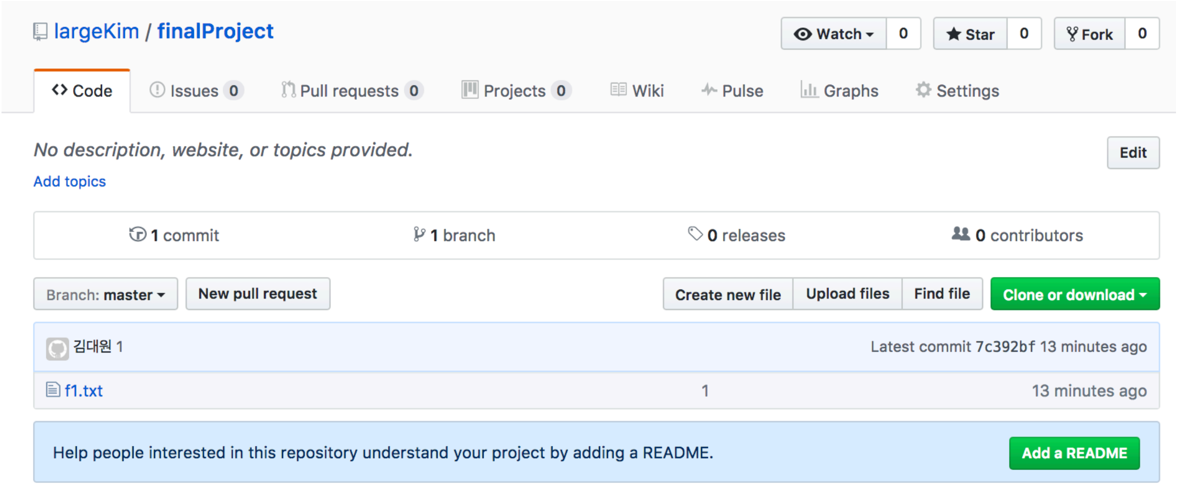
You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

Git에서는 로컬저장소를 기준으로 원격저장소로 나의 작업을 보낼 때, 로컬저장소는 원격저장소로 push한다고 한다. >`git push -u origin master`
이 명령어를 사용한후 >`git push` 명령어 만으로도 자동으로 원격저장소의 origin의 master branch로 push된다.

>`git push -u origin master`



```
gimdaewon@iMac:~/documents/git/gitfth10% git push -u origin master
Username for 'https://github.com': largeKim
Password for 'https://largeKim@github.com':
Counting objects: 3, done.
Writing objects: 100% (3/3), 214 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/largeKim/finalProject.git
 * [new branch]      master -> master
Branch master set up to track remote branch master from origin.
```

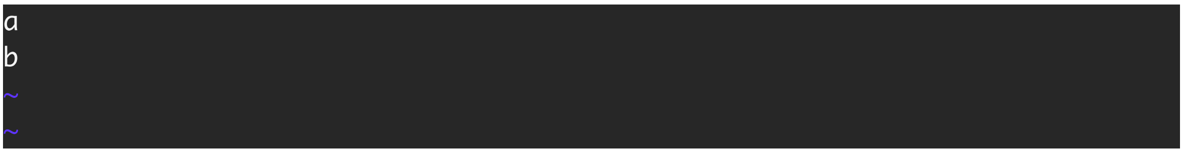


No description, website, or topics provided.

김대원 1		Latest commit 7c392bf 13 minutes ago
f1.txt	1	13 minutes ago

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

>vim f1.txt   //수정

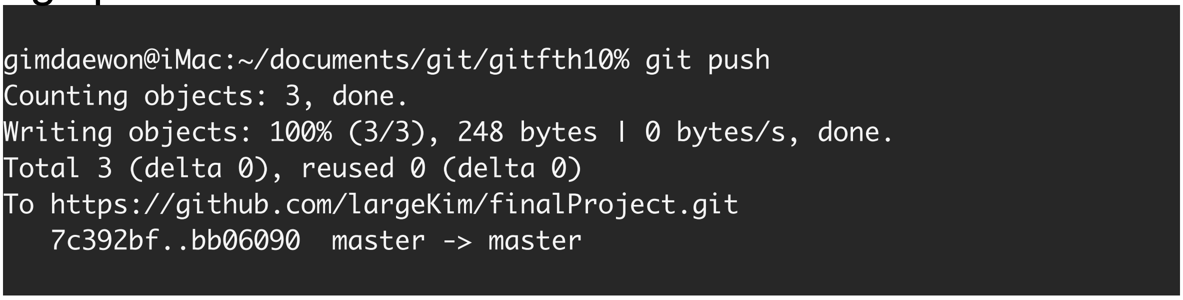


```
a
b
~
~
```

[esc] + :wq + [enter]

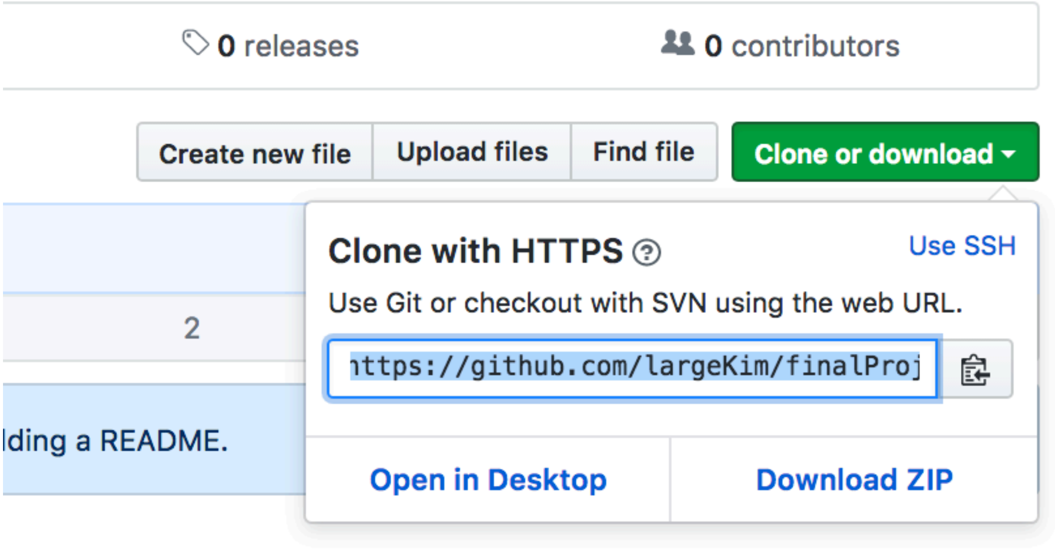
>git commit -am '2'

>git push



```
gimdaewon@iMac:~/documents/git/gitfth10% git push
Counting objects: 3, done.
Writing objects: 100% (3/3), 248 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/largeKim/finalProject.git
   7c392bf..bb06090  master -> master
```

(4)원격저장소로부터 가져오기(처음 git 가져오는 부분과 중복)



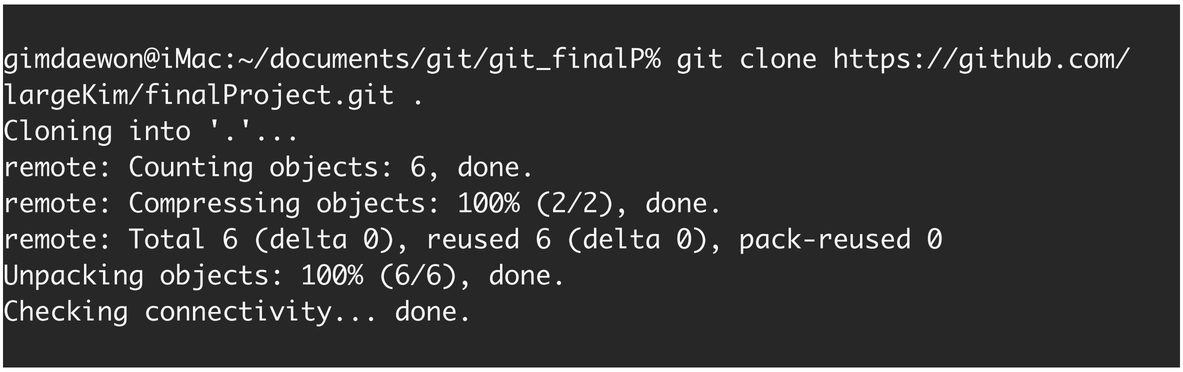
Clone with HTTPS

Use Git or checkout with SVN using the web URL.

```
https://github.com/largeKim/finalProj
```

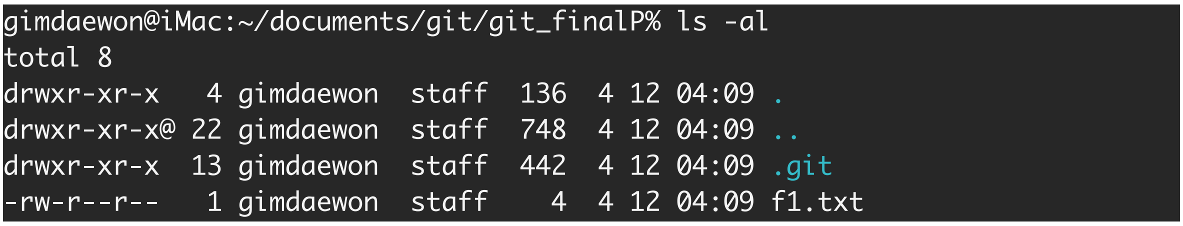
저장할 적당한 디렉토리 생성

>cd ..
>mkdir git_finalP
>git clone https://github.com/largeKim/finalProject.git .
//마지막 .은 현재디렉토리라는 뜻이다.



```
gimdaewon@iMac:~/documents/git/git_finalP% git clone https://github.com/
largeKim/finalProject.git .
Cloning into '.'...
remote: Counting objects: 6, done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 6 (delta 0), reused 6 (delta 0), pack-reused 0
Unpacking objects: 100% (6/6), done.
Checking connectivity... done.
```

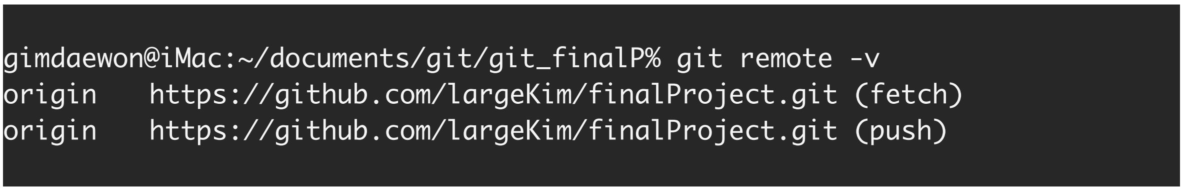
>ls -al



```
gimdaewon@iMac:~/documents/git/git_finalP% ls -al
total 8
drwxr-xr-x   4 gimdaewon  staff  136  4 12 04:09 .
drwxr-xr-x@ 22 gimdaewon  staff  748  4 12 04:09 ..
drwxr-xr-x  13 gimdaewon  staff  442  4 12 04:09 .git
-rw-r--r--   1 gimdaewon  staff    4  4 12 04:09 f1.txt
```

파일이 잘 clone 된것을 알 수 있다.

>git remote -v



```
gimdaewon@iMac:~/documents/git/git_finalP% git remote -v
origin   https://github.com/largeKim/finalProject.git (fetch)
origin   https://github.com/largeKim/finalProject.git (push)
```

따로 하지 않았던 원격저장소와의 연결이 이미 완료되어있다.

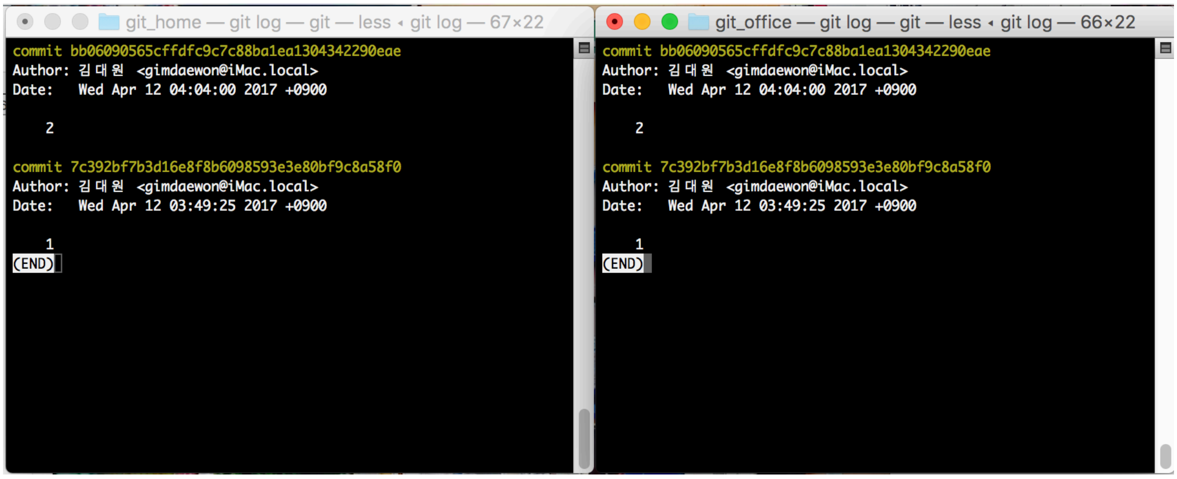
2개의 다른이름의 같은링크 저장소 생성
>git clone https://github.com/largeKim/finalProject.git git_home
>git clone https://github.com/largeKim/finalProject.git git_office

git_home

git_office

>git log

>git log

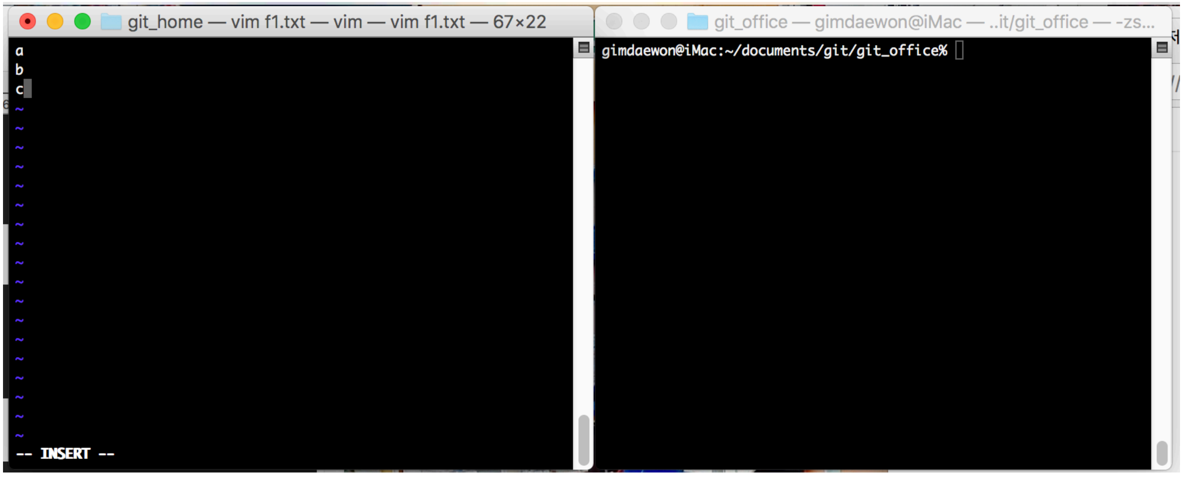

git_home — git log — git — less ◂ git log — 67×22
git_office — git log — git — less ◂ git log — 66×22

```
commit bb06090565cffdfc9c7c88ba1ea1304342290eae
Author: 김대원 <gimdaewon@iMac.local>
Date:   Wed Apr 12 04:04:00 2017 +0900

    2

commit 7c392bf7b3d16e8f8b6098593e3e80bf9c8a58f0
Author: 김대원 <gimdaewon@iMac.local>
Date:   Wed Apr 12 03:49:25 2017 +0900

    1
(END)
```

```
commit bb06090565cffdfc9c7c88ba1ea1304342290eae
Author: 김대원 <gimdaewon@iMac.local>
Date:   Wed Apr 12 04:04:00 2017 +0900

    2

commit 7c392bf7b3d16e8f8b6098593e3e80bf9c8a58f0
Author: 김대원 <gimdaewon@iMac.local>
Date:   Wed Apr 12 03:49:25 2017 +0900

    1
(END)
```

집으로 와서

>vim f1.txt　　//수정


git_home — vim f1.txt — vim — vim f1.txt — 67×22
git_office — gimdaewon@iMac — ..it/git_office — -zs...

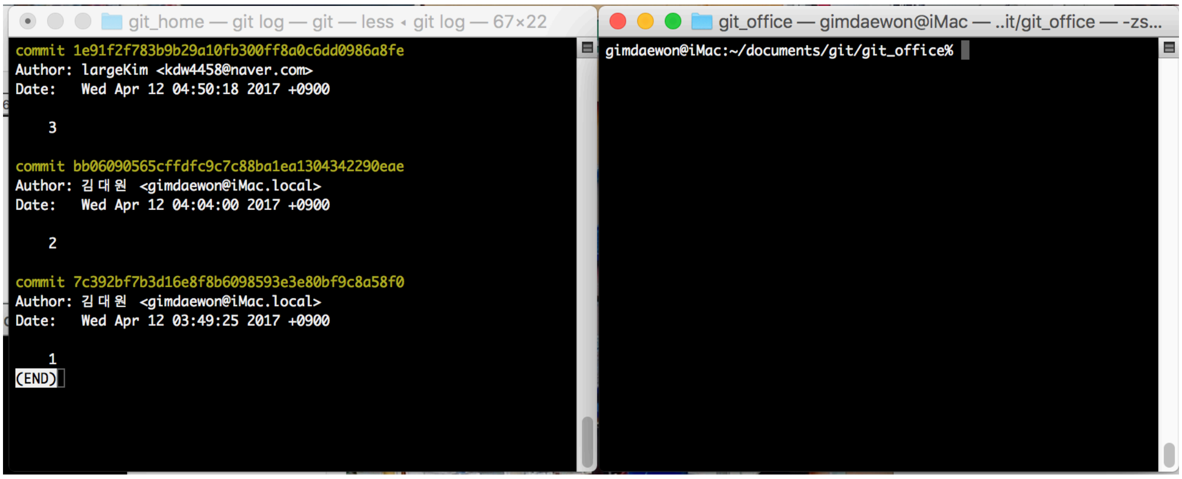
```
a
b
c
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
-- INSERT --
```

```
gimdaewon@iMac:~/documents/git/git_office%
```

>git commit -am '3'

//git commit --amend

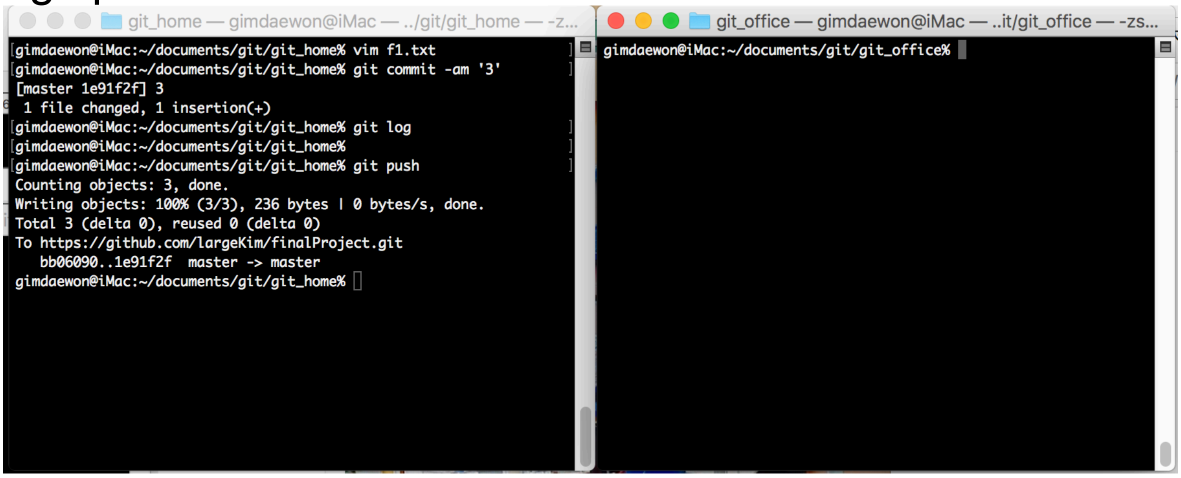
//　commit명('3')을 바꿀수있다.


git_home — git log — git — less ◂ git log — 67×22
git_office — gimdaewon@iMac — ..it/git_office — -zs...

```
commit 1e91f2f783b9b29a10fb300ff8a0c6dd0986a8fe
Author: largeKim <kdw4458@naver.com>
Date:   Wed Apr 12 04:50:18 2017 +0900

    3

commit bb06090565cffdfc9c7c88ba1ea1304342290eae
Author: 김대원 <gimdaewon@iMac.local>
Date:   Wed Apr 12 04:04:00 2017 +0900

    2

commit 7c392bf7b3d16e8f8b6098593e3e80bf9c8a58f0
Author: 김대원 <gimdaewon@iMac.local>
Date:   Wed Apr 12 03:49:25 2017 +0900

    1
(END)
```

```
gimdaewon@iMac:~/documents/git/git_office%
```

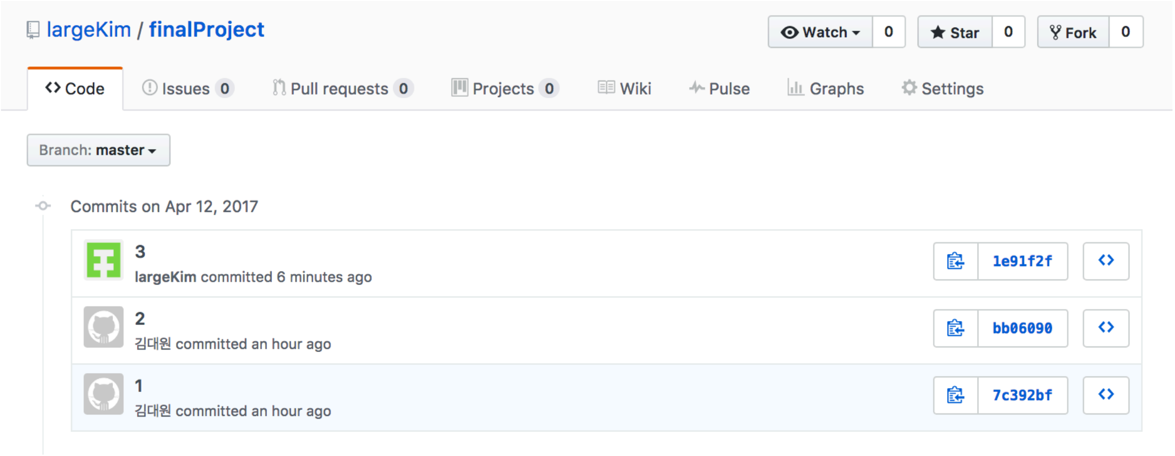
>git push


git_home — gimdaewon@iMac — ../git/git_home — -z...
git_office — gimdaewon@iMac — ..it/git_office — -zs...

```
gimdaewon@iMac:~/documents/git/git_home% vim f1.txt
gimdaewon@iMac:~/documents/git/git_home% git commit -am '3'
[master 1e91f2f] 3
 1 file changed, 1 insertion(+)
gimdaewon@iMac:~/documents/git/git_home% git log
gimdaewon@iMac:~/documents/git/git_home%
gimdaewon@iMac:~/documents/git/git_home% git push
Counting objects: 3, done.
Writing objects: 100% (3/3), 236 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/largeKim/finalProject.git
   bb06090..1e91f2f  master -> master
gimdaewon@iMac:~/documents/git/git_home%
```

```
gimdaewon@iMac:~/documents/git/git_office%
```

github를 확인해보면 commit이 3개가 되어있다.



Commits on Apr 12, 2017

3
largeKim committed 6 minutes ago
1e91f2f

2
김대원 committed an hour ago
bb06090

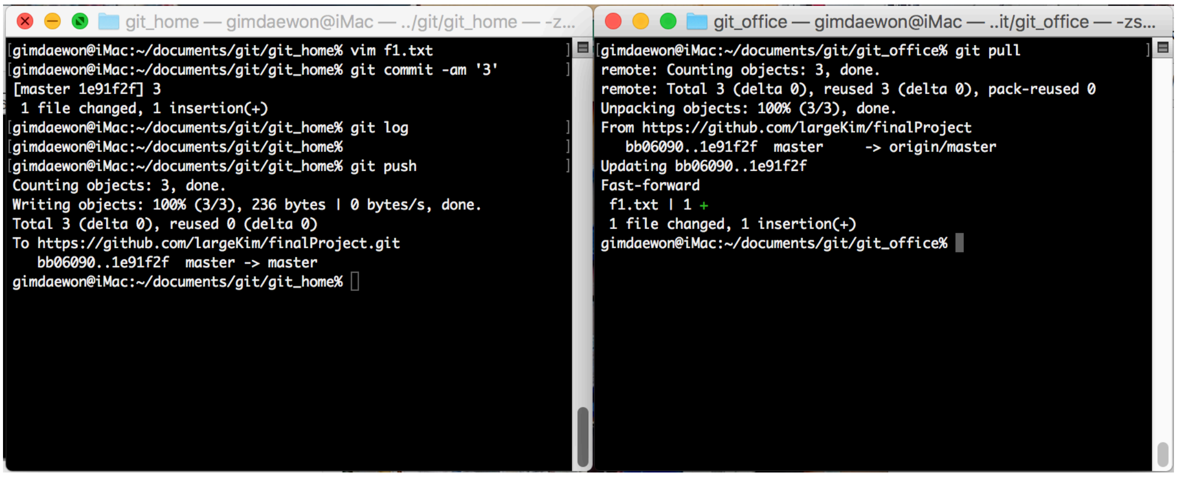
1
김대원 committed an hour ago
7c392bf

회

사로 가서

>git pull　　//원격저장소의 내용을 로컬로..

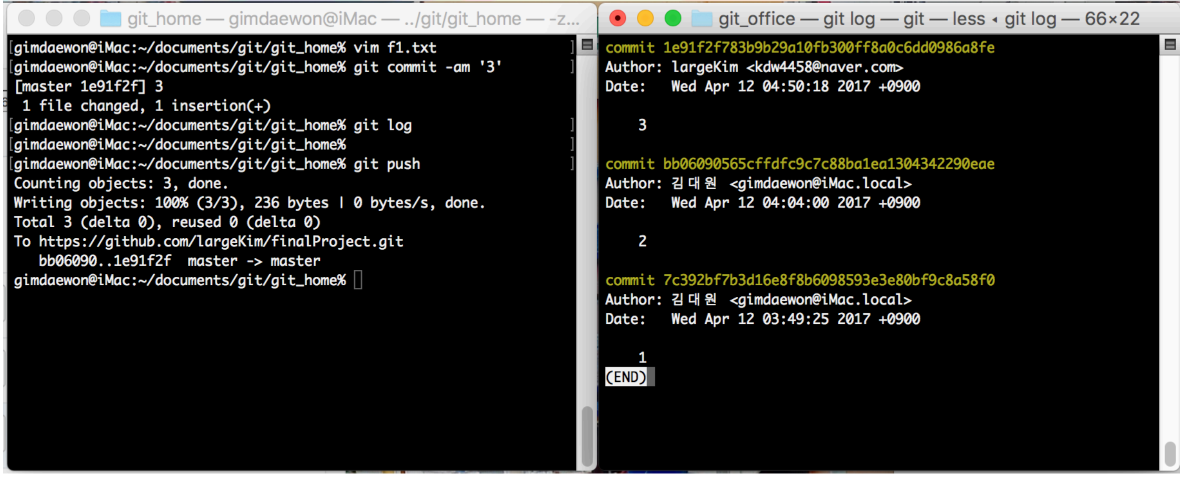
//공개저장소라 id/pwd안물어봄


git_home — gimdaewon@iMac — ../git/git_home — -z...
git_office — gimdaewon@iMac — ..it/git_office — -zs...

```
gimdaewon@iMac:~/documents/git/git_home% vim f1.txt
gimdaewon@iMac:~/documents/git/git_home% git commit -am '3'
[master 1e91f2f] 3
 1 file changed, 1 insertion(+)
gimdaewon@iMac:~/documents/git/git_home% git log
gimdaewon@iMac:~/documents/git/git_home%
gimdaewon@iMac:~/documents/git/git_home% git push
Counting objects: 3, done.
Writing objects: 100% (3/3), 236 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/largeKim/finalProject.git
   bb06090..1e91f2f  master -> master
gimdaewon@iMac:~/documents/git/git_home%
```

```
gimdaewon@iMac:~/documents/git/git_office% git pull
remote: Counting objects: 3, done.
remote: Total 3 (delta 0), reused 3 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
From https://github.com/largeKim/finalProject
   bb06090..1e91f2f  master     -> origin/master
Updating bb06090..1e91f2f
Fast-forward
 f1.txt | 1 +
 1 file changed, 1 insertion(+)
gimdaewon@iMac:~/documents/git/git_office%
```

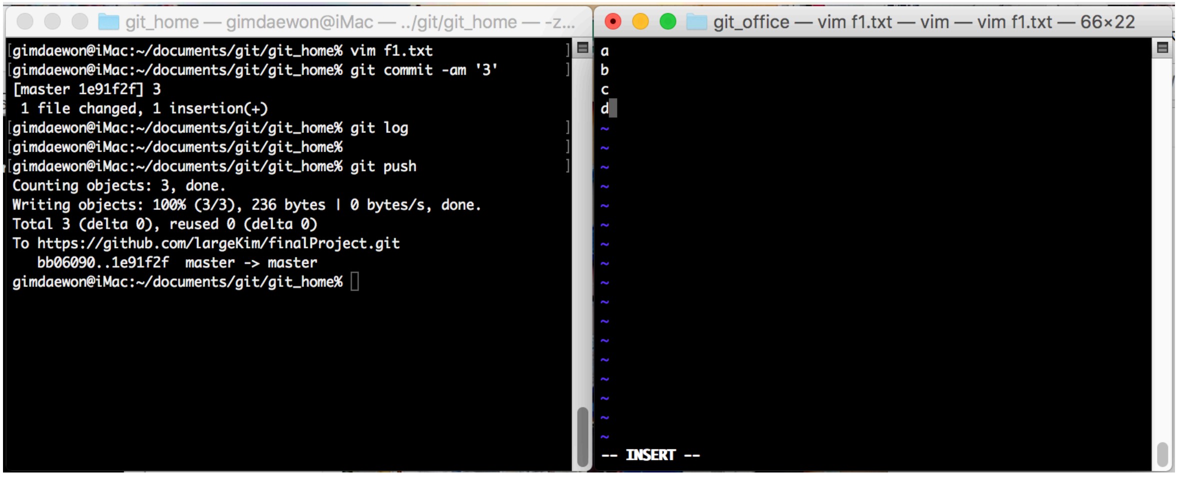
>git log


git_home — gimdaewon@iMac — ../git/git_home — -z...
git_office — git log — git — less ◂ git log — 66×22

```
gimdaewon@iMac:~/documents/git/git_home% vim f1.txt
gimdaewon@iMac:~/documents/git/git_home% git commit -am '3'
[master 1e91f2f] 3
 1 file changed, 1 insertion(+)
gimdaewon@iMac:~/documents/git/git_home% git log
gimdaewon@iMac:~/documents/git/git_home%
gimdaewon@iMac:~/documents/git/git_home% git push
Counting objects: 3, done.
Writing objects: 100% (3/3), 236 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/largeKim/finalProject.git
   bb06090..1e91f2f  master -> master
gimdaewon@iMac:~/documents/git/git_home%
```

```
commit 1e91f2f783b9b29a10fb300ff8a0c6dd0986a8fe
Author: largeKim <kdw4458@naver.com>
Date:   Wed Apr 12 04:50:18 2017 +0900

    3

commit bb06090565cffdfc9c7c88ba1ea1304342290eae
Author: 김대원 <gimdaewon@iMac.local>
Date:   Wed Apr 12 04:04:00 2017 +0900

    2

commit 7c392bf7b3d16e8f8b6098593e3e80bf9c8a58f0
Author: 김대원 <gimdaewon@iMac.local>
Date:   Wed Apr 12 03:49:25 2017 +0900

    1
(END)
```

>vim f1.txt　　//수정

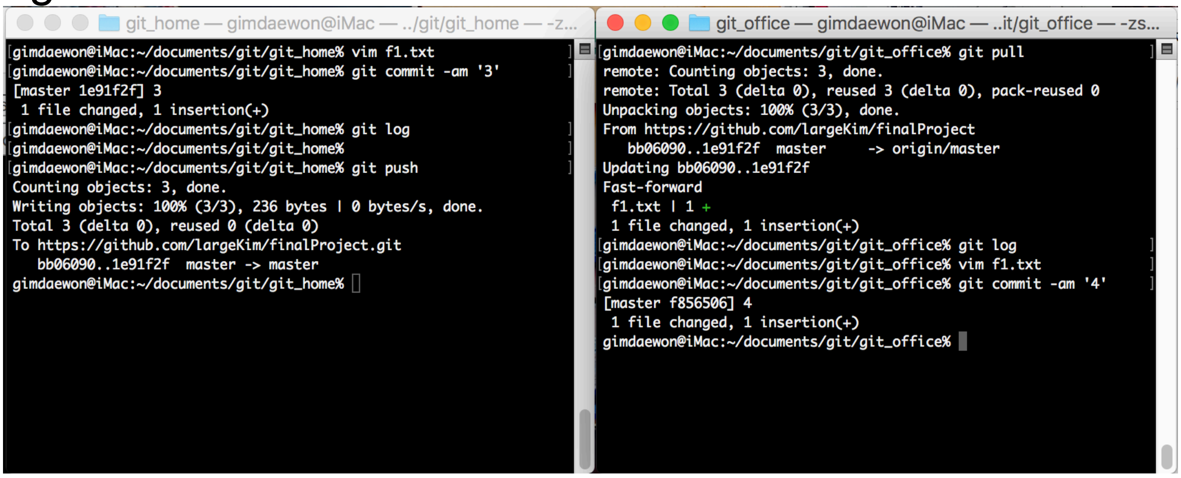


git_home — gimdaewon@iMac — ../git/git_home — -z...
git_office — vim f1.txt — vim — vim f1.txt — 66×22

```
[gimdaewon@iMac:~/documents/git/git_home% vim f1.txt
[gimdaewon@iMac:~/documents/git/git_home% git commit -am '3'
[master 1e91f2f] 3
 1 file changed, 1 insertion(+)
[gimdaewon@iMac:~/documents/git/git_home% git log
[gimdaewon@iMac:~/documents/git/git_home%
[gimdaewon@iMac:~/documents/git/git_home% git push
Counting objects: 3, done.
Writing objects: 100% (3/3), 236 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/largeKim/finalProject.git
   bb06090..1e91f2f  master -> master
gimdaewon@iMac:~/documents/git/git_home%
```

```
a
b
c
d
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
-- INSERT --
```

>git commit -am '4'

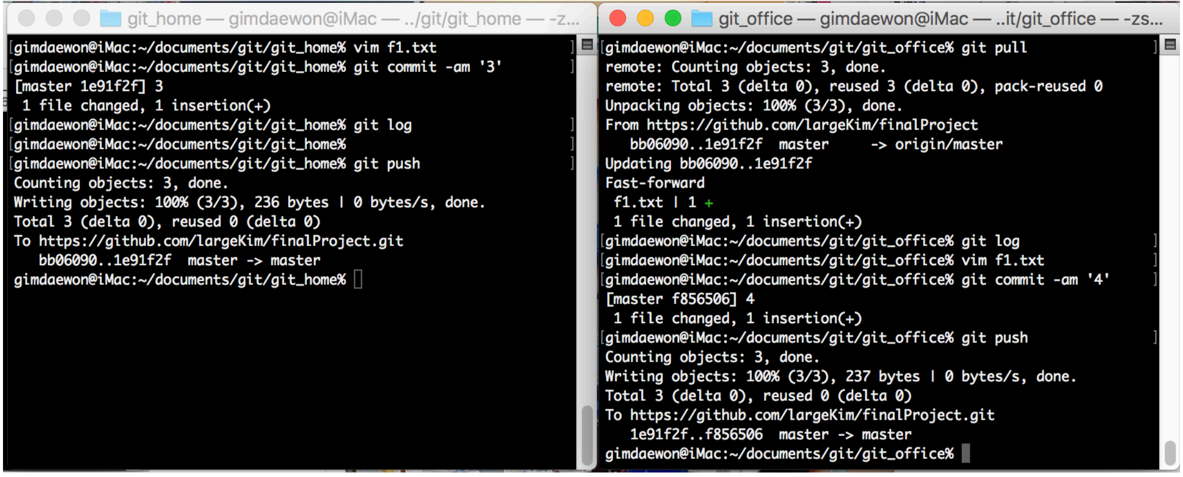


git_home — gimdaewon@iMac — ../git/git_home — -z...
git_office — gimdaewon@iMac — ..it/git_office — -zs...

```
[gimdaewon@iMac:~/documents/git/git_home% vim f1.txt
[gimdaewon@iMac:~/documents/git/git_home% git commit -am '3'
[master 1e91f2f] 3
 1 file changed, 1 insertion(+)
[gimdaewon@iMac:~/documents/git/git_home% git log
[gimdaewon@iMac:~/documents/git/git_home%
[gimdaewon@iMac:~/documents/git/git_home% git push
Counting objects: 3, done.
Writing objects: 100% (3/3), 236 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/largeKim/finalProject.git
   bb06090..1e91f2f  master -> master
gimdaewon@iMac:~/documents/git/git_home%
```

```
[gimdaewon@iMac:~/documents/git/git_office% git pull
remote: Counting objects: 3, done.
remote: Total 3 (delta 0), reused 3 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
From https://github.com/largeKim/finalProject
   bb06090..1e91f2f  master     -> origin/master
Updating bb06090..1e91f2f
Fast-forward
 f1.txt | 1 +
 1 file changed, 1 insertion(+)
[gimdaewon@iMac:~/documents/git/git_office% git log
[gimdaewon@iMac:~/documents/git/git_office% vim f1.txt
[gimdaewon@iMac:~/documents/git/git_office% git commit -am '4'
[master f856506] 4
 1 file changed, 1 insertion(+)
gimdaewon@iMac:~/documents/git/git_office%
```

>git push



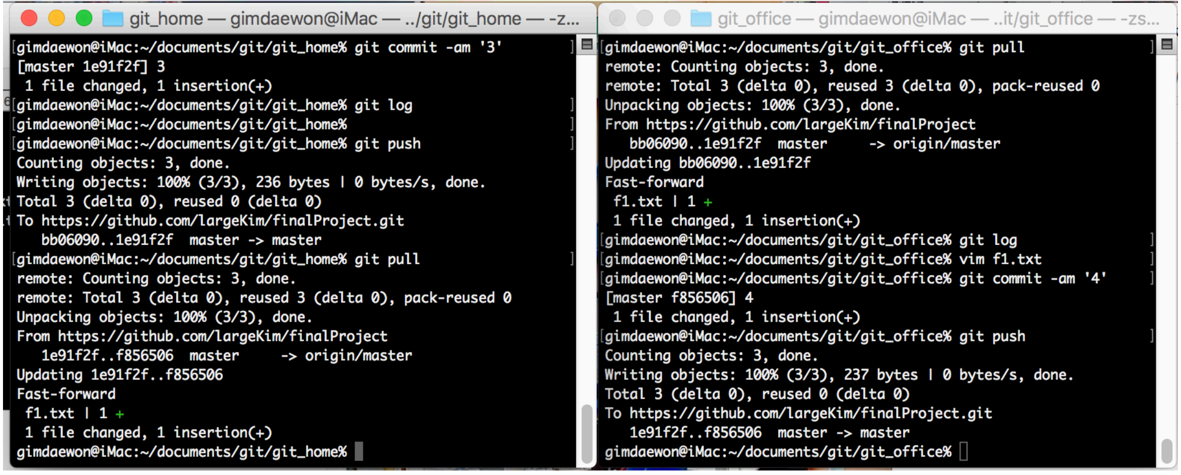
git_home — gimdaewon@iMac — ../git/git_home — -z...
git_office — gimdaewon@iMac — ..it/git_office — -zs...

```
[gimdaewon@iMac:~/documents/git/git_home% vim f1.txt
[gimdaewon@iMac:~/documents/git/git_home% git commit -am '3'
[master 1e91f2f] 3
 1 file changed, 1 insertion(+)
[gimdaewon@iMac:~/documents/git/git_home% git log
[gimdaewon@iMac:~/documents/git/git_home%
[gimdaewon@iMac:~/documents/git/git_home% git push
Counting objects: 3, done.
Writing objects: 100% (3/3), 236 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/largeKim/finalProject.git
   bb06090..1e91f2f  master -> master
gimdaewon@iMac:~/documents/git/git_home%
```

```
[gimdaewon@iMac:~/documents/git/git_office% git pull
remote: Counting objects: 3, done.
remote: Total 3 (delta 0), reused 3 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
From https://github.com/largeKim/finalProject
   bb06090..1e91f2f  master     -> origin/master
Updating bb06090..1e91f2f
Fast-forward
 f1.txt | 1 +
 1 file changed, 1 insertion(+)
[gimdaewon@iMac:~/documents/git/git_office% git log
[gimdaewon@iMac:~/documents/git/git_office% vim f1.txt
[gimdaewon@iMac:~/documents/git/git_office% git commit -am '4'
[master f856506] 4
 1 file changed, 1 insertion(+)
[gimdaewon@iMac:~/documents/git/git_office% git push
Counting objects: 3, done.
Writing objects: 100% (3/3), 237 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/largeKim/finalProject.git
   1e91f2f..f856506  master -> master
gimdaewon@iMac:~/documents/git/git_office%
```

집으로와서!

>git pull


git_home — gimdaewon@iMac — ../git/git_home — -z...
git_office — gimdaewon@iMac — ..it/git_office — -zs...

```
gimdaewon@iMac:~/documents/git/git_home% git commit -am '3'
[master 1e91f2f] 3
 1 file changed, 1 insertion(+)
gimdaewon@iMac:~/documents/git/git_home% git log
gimdaewon@iMac:~/documents/git/git_home%
gimdaewon@iMac:~/documents/git/git_home% git push
Counting objects: 3, done.
Writing objects: 100% (3/3), 236 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/largeKim/finalProject.git
   bb06090..1e91f2f  master -> master
gimdaewon@iMac:~/documents/git/git_home% git pull
remote: Counting objects: 3, done.
remote: Total 3 (delta 0), reused 3 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
From https://github.com/largeKim/finalProject
   1e91f2f..f856506  master     -> origin/master
Updating 1e91f2f..f856506
Fast-forward
 f1.txt | 1 +
 1 file changed, 1 insertion(+)
gimdaewon@iMac:~/documents/git/git_home%
```

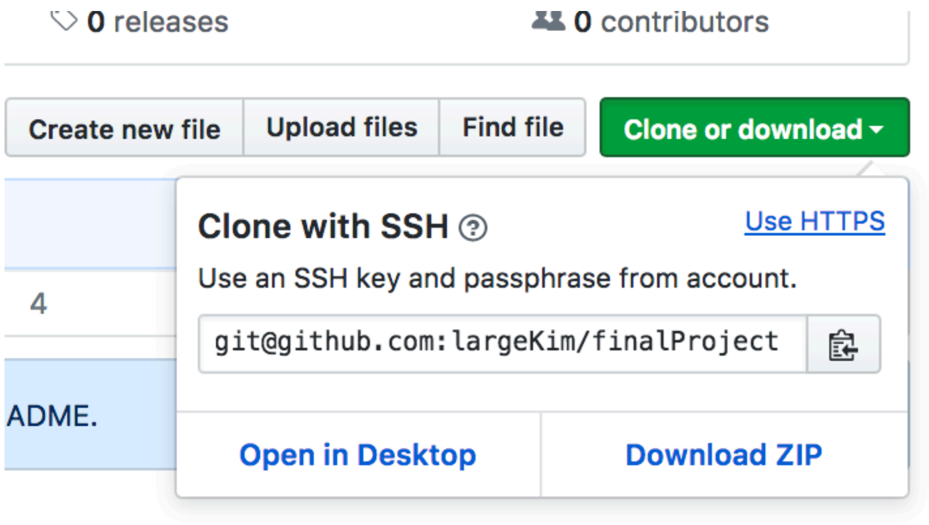
```
gimdaewon@iMac:~/documents/git/git_office% git pull
remote: Counting objects: 3, done.
remote: Total 3 (delta 0), reused 3 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
From https://github.com/largeKim/finalProject
   bb06090..1e91f2f  master     -> origin/master
Updating bb06090..1e91f2f
Fast-forward
 f1.txt | 1 +
 1 file changed, 1 insertion(+)
gimdaewon@iMac:~/documents/git/git_office% git log
gimdaewon@iMac:~/documents/git/git_office% vim f1.txt
gimdaewon@iMac:~/documents/git/git_office% git commit -am '4'
[master f856506] 4
 1 file changed, 1 insertion(+)
gimdaewon@iMac:~/documents/git/git_office% git push
Counting objects: 3, done.
Writing objects: 100% (3/3), 237 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/largeKim/finalProject.git
   1e91f2f..f856506  master -> master
gimdaewon@iMac:~/documents/git/git_office%
```

이러한것들로 pull(땡기고) push(보내고)를 반복해서 백업하는 것이 가능하다.

집 github 회사 이렇게 구축이 된다면
지구적인 위기가 아닌이상 소스코드를 잃어버리는 일이 없다.

4)로그인 없이 원격 저장소 이용하기(Github)

ssh 통신방식을 이용해서 자동으로 로그인 해준다.



Use an SSH key and passphrase from account.

```
git@github.com:largeKim/finalProject
```

>ssh-keygen

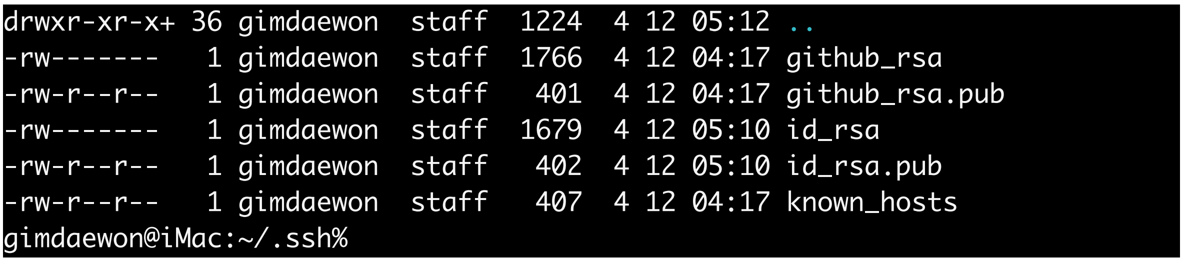
```
gimdaewon@iMac:~% ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/Users/gimdaewon/.ssh/id_rsa):
```

경로(`/Users/gimdaewon/.ssh/id_rsa`)를 잘 체크해두기 그리고 [enter]!

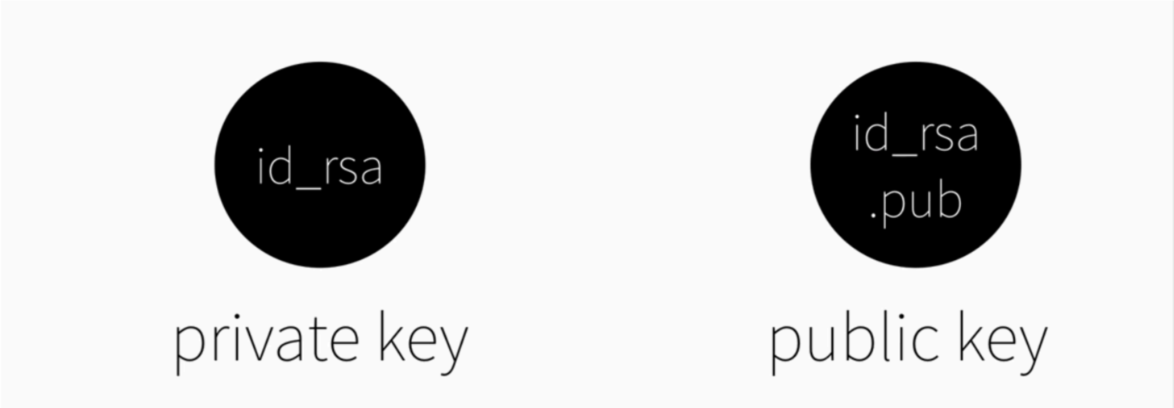
```
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /Users/gimdaewon/.ssh/id_rsa.
Your public key has been saved in /Users/gimdaewon/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:spwwxp4wF6KDqn74GEaZRGd1D3xqfBUFPz45AQykTQA gimdaewon@iMac.local
The key's randomart image is:
+---[RSA 2048]----+
|  . o.Eo+o+o++.  |
|. o    ..*. oo   |
| .. . ..oo.  +   |
|o.oo . + .  . +  |
|++o * o S    =   |
|o. * = +      o  |
|.o. o +          |
|o.o.             |
|ooo.             |
+----[SHA256]-----+
gimdaewon@iMac:~%
```

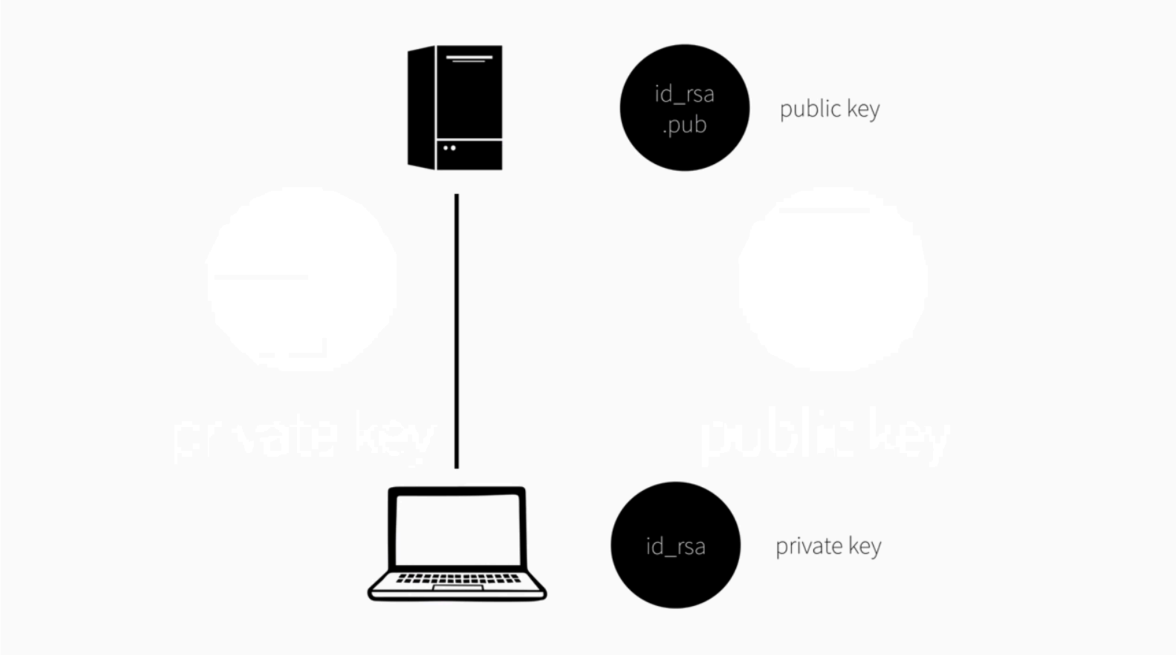
>cd /Users/gimdaewon/.ssh/

```
gimdaewon@iMac:~% cd /Users/gimdaewon/.ssh/
gimdaewon@iMac:~/.ssh% ls -al
total 40
drwx------   7 gimdaewon  staff   238  4 12 05:10 .
```



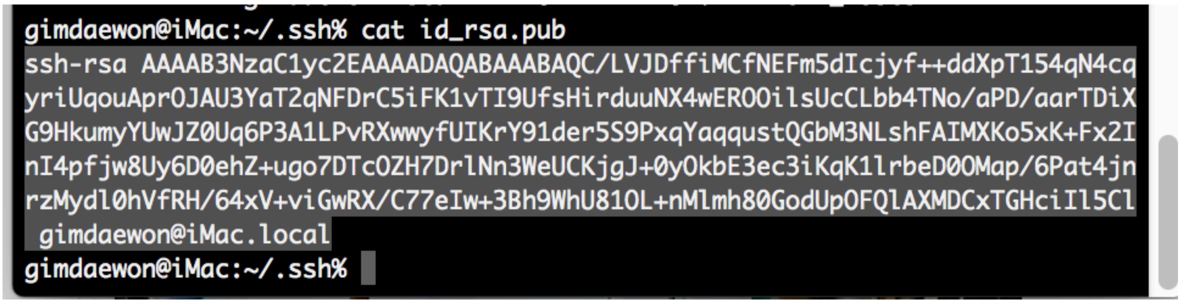
```
drwxr-xr-x+ 36 gimdaewon  staff  1224  4 12 05:12 ..
-rw-------   1 gimdaewon  staff  1766  4 12 04:17 github_rsa
-rw-r--r--   1 gimdaewon  staff   401  4 12 04:17 github_rsa.pub
-rw-------   1 gimdaewon  staff  1679  4 12 05:10 id_rsa
-rw-r--r--   1 gimdaewon  staff   402  4 12 05:10 id_rsa.pub
-rw-r--r--   1 gimdaewon  staff   407  4 12 04:17 known_hosts
gimdaewon@iMac:~/.ssh%
```





원격저장소가있는 곳에 public key(id_rsa.pub)파일을 저장해둔다.

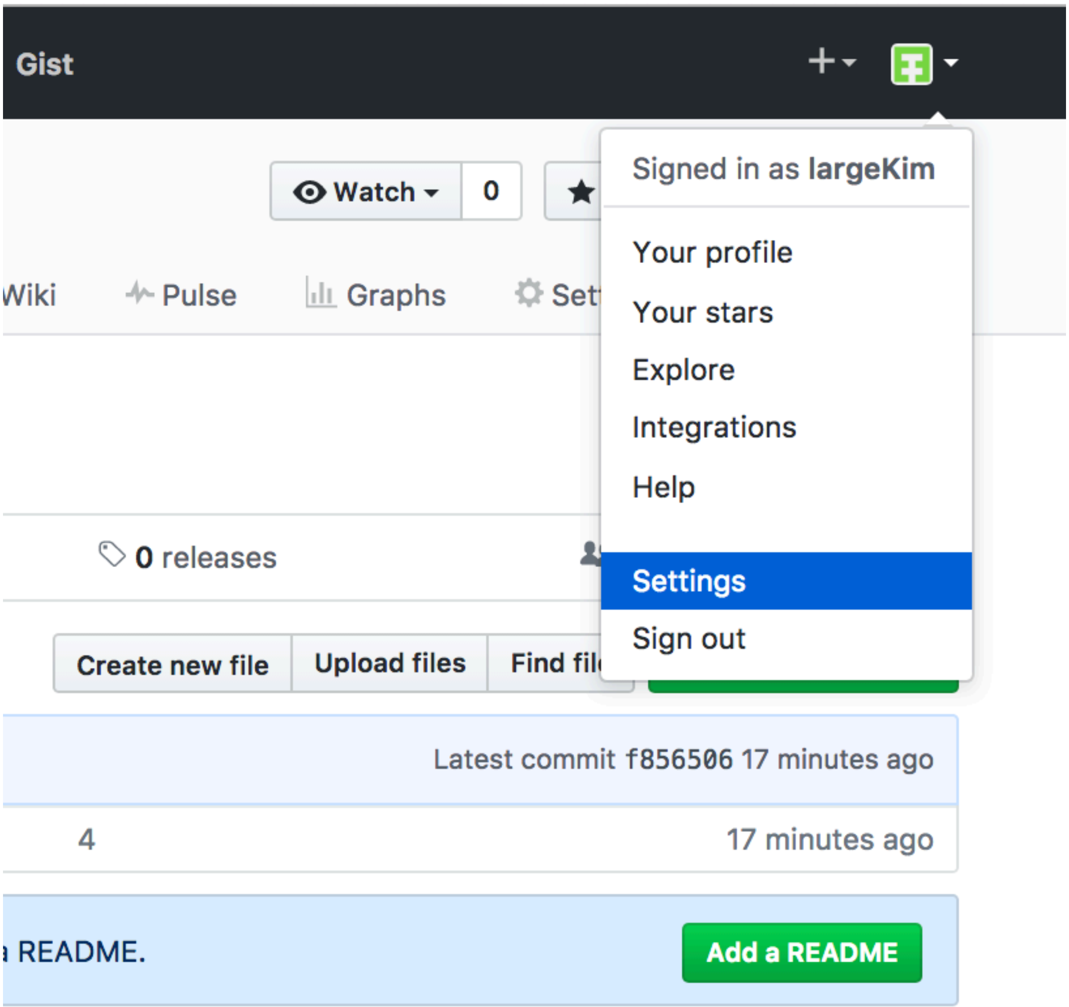
>cat id_rsa.pub



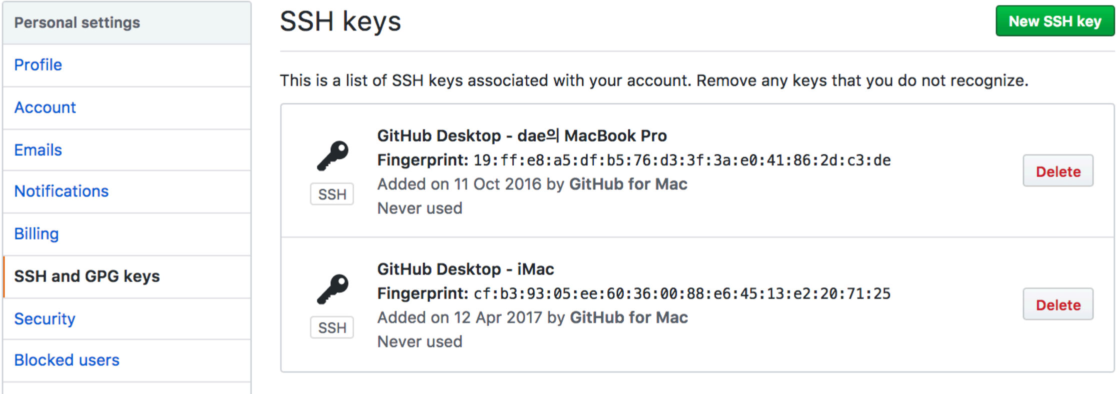
```
gimdaewon@iMac:~/.ssh% cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQC/LVJDffiMCfNEFm5dIcjyf++ddXpT154qN4cq
yriUqouAprOJAU3YaT2qNFDrC5iFK1vTI9UfsHirduuNX4wEROOilsUcCLbb4TNo/aPD/aarTDiX
G9HkumyYUwJZ0Uq6P3A1LPvRXwwyfUIKrY91der5S9PxqYaqqustQGbM3NLshFAIMXKo5xK+Fx2I
nI4pfjw8Uy6D0ehZ+ugo7DTcOZH7DrlNn3WeUCKjgJ+0yOkbE3ec3iKqK1lrbeD0OMap/6Pat4jn
rzMydl0hVfRH/64xV+viGwRX/C77eIw+3Bh9WhU81OL+nMlmh80GodUpOFQlAXMDCxTGHciIl5Cl
 gimdaewon@iMac.local
gimdaewon@iMac:~/.ssh%
```

//정교하게 복사해둔다.

settings 클릭



SSH and GPG keys 클릭후 New SSH Key 클릭



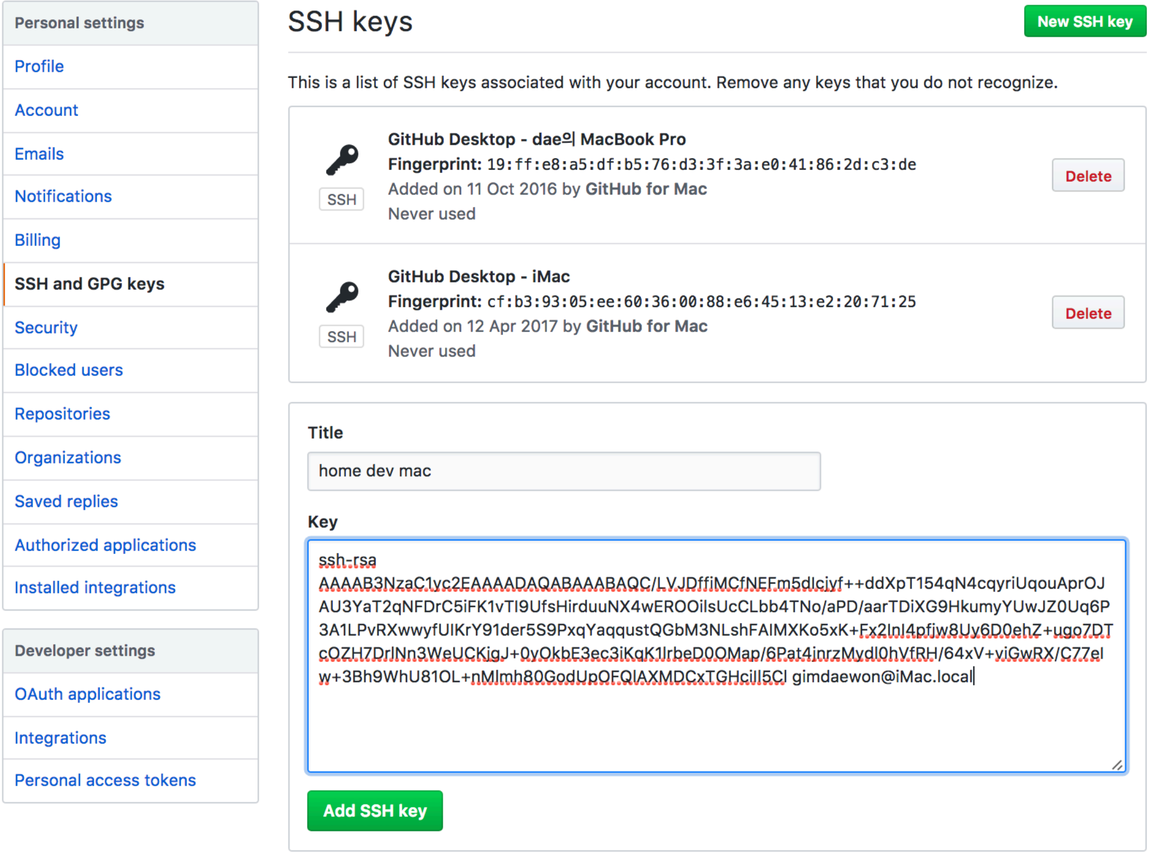
SSH keys

This is a list of SSH keys associated with your account. Remove any keys that you do not recognize.

GitHub Desktop - dae의 MacBook Pro
Fingerprint: `19:ff:e8:a5:df:b5:76:d3:3f:3a:e0:41:86:2d:c3:de`
Added on 11 Oct 2016 by GitHub for Mac
Never used

GitHub Desktop - iMac
Fingerprint: `cf:b3:93:05:ee:60:36:00:88:e6:45:13:e2:20:71:25`
Added on 12 Apr 2017 by GitHub for Mac
Never used

title과 복사해둔 것을 Key에 붙여넣은후 Add SSH Key 클릭



SSH keys

This is a list of SSH keys associated with your account. Remove any keys that you do not recognize.

GitHub Desktop - dae의 MacBook Pro
Fingerprint: `19:ff:e8:a5:df:b5:76:d3:3f:3a:e0:41:86:2d:c3:de`
Added on 11 Oct 2016 by GitHub for Mac
Never used

GitHub Desktop - iMac
Fingerprint: `cf:b3:93:05:ee:60:36:00:88:e6:45:13:e2:20:71:25`
Added on 12 Apr 2017 by GitHub for Mac
Never used

Title

home dev mac

Key

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQC/LVJDffiMCfNEFm5dlcjyf++ddXpT154qN4cqyriUqouAprOJ
AU3YaT2qNFDrC5iFK1vTI9UfsHirduuNX4wEROOilsUcCLbb4TNo/aPD/aarTDiXG9HkumyYUwJZ0Uq6P
3A1LPvRXwwyfUlKrY91der5S9PxqYaqqustQGbM3NLshFAIMXKo5xK+Fx2Inl4pfjw8Uy6D0ehZ+ugo7DT
cOZH7DrlNn3WeUCKjgJ+0yOkbE3ec3iKqK1lrbeD0OMap/6Pat4jnrzMydl0hVfRH/64xV+yiGwRX/C77el
w+3Bh9WhU81OL+nMlmh80GodUpOFQlAXMDCxTGHcilI5Cl gimdaewon@iMac.local
```

간단하게 확인을 위해

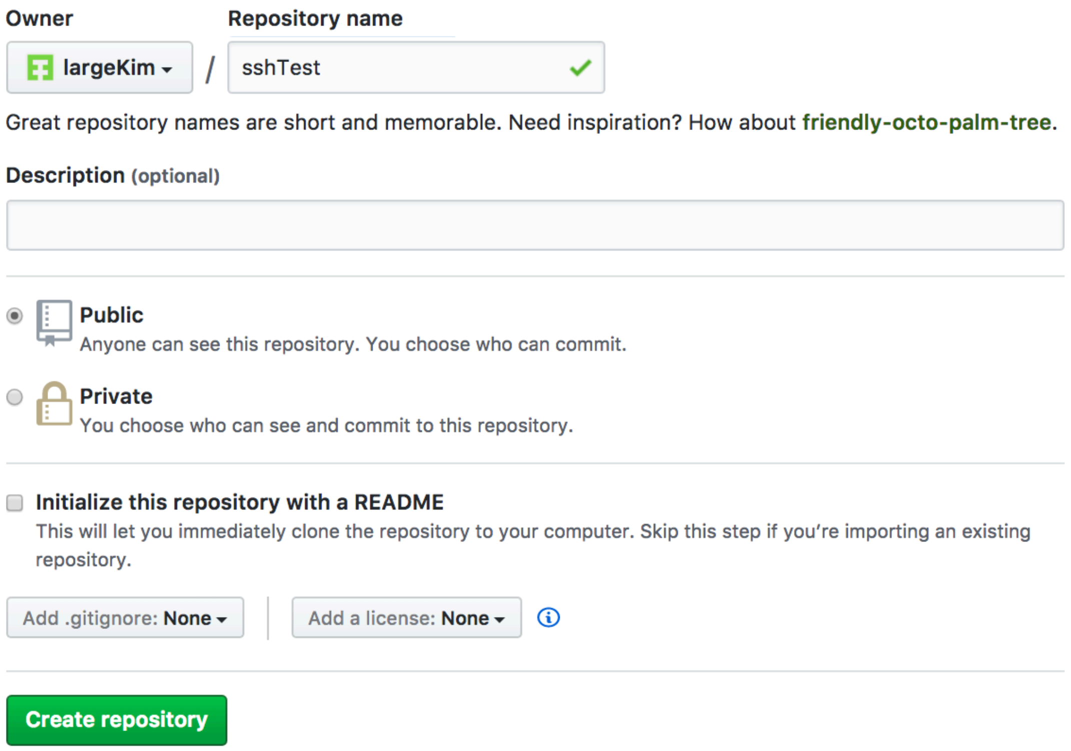
새로운 저장소를 만들어보자

Start a project 클릭

Learn Git and GitHub without any code!

Using the Hello World guide, you'll create a repository, start a branch, write comments, and open a pull request.

이름 입력후 Create repository 클릭



Owner: largeKim / Repository name: sshTest ✓

Great repository names are short and memorable. Need inspiration? How about **friendly-octo-palm-tree**.

Description (optional)

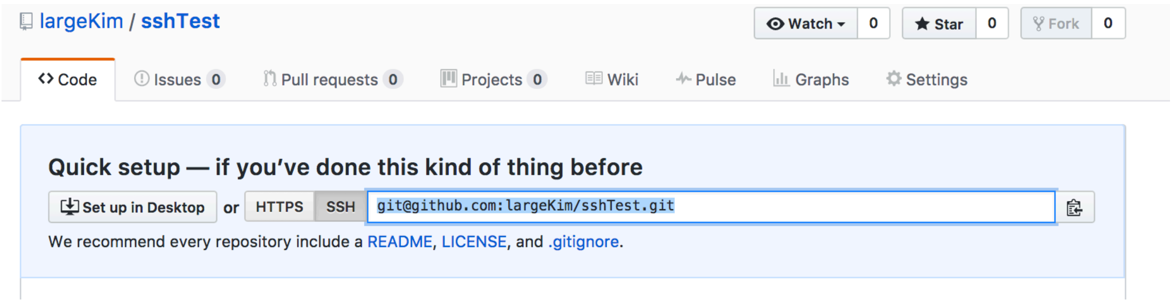
● **Public**
Anyone can see this repository. You choose who can commit.

○ **Private**
You choose who can see and commit to this repository.

□ **Initialize this repository with a README**
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: None | Add a license: None

SSH 클릭후 주소 copy 하기



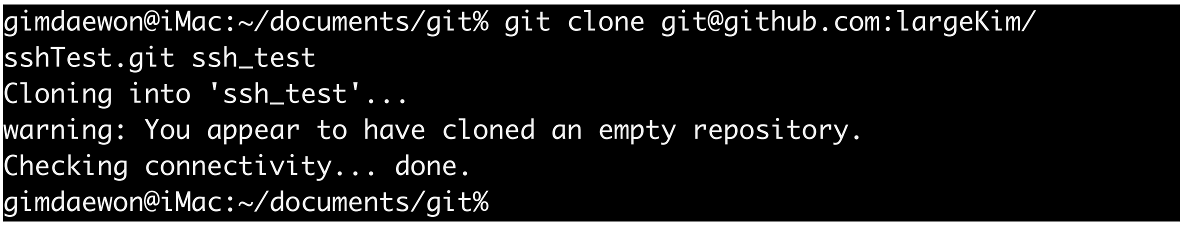
Quick setup — if you've done this kind of thing before

Set up in Desktop or HTTPS SSH `git@github.com:largeKim/sshTest.git`

We recommend every repository include a README, LICENSE, and .gitignore.

적당한 저장소를 찾아들어가서

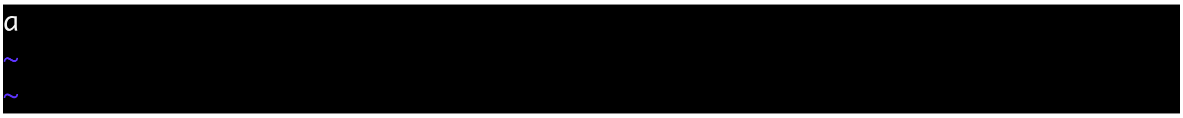
>git clone git@github.com:largeKim/sshTest.git ssh_test



```
gimdaewon@iMac:~/documents/git% git clone git@github.com:largeKim/
sshTest.git ssh_test
Cloning into 'ssh_test'...
warning: You appear to have cloned an empty repository.
Checking connectivity... done.
gimdaewon@iMac:~/documents/git%
```

>cd ssh_test

>vim f1.txt   //생성

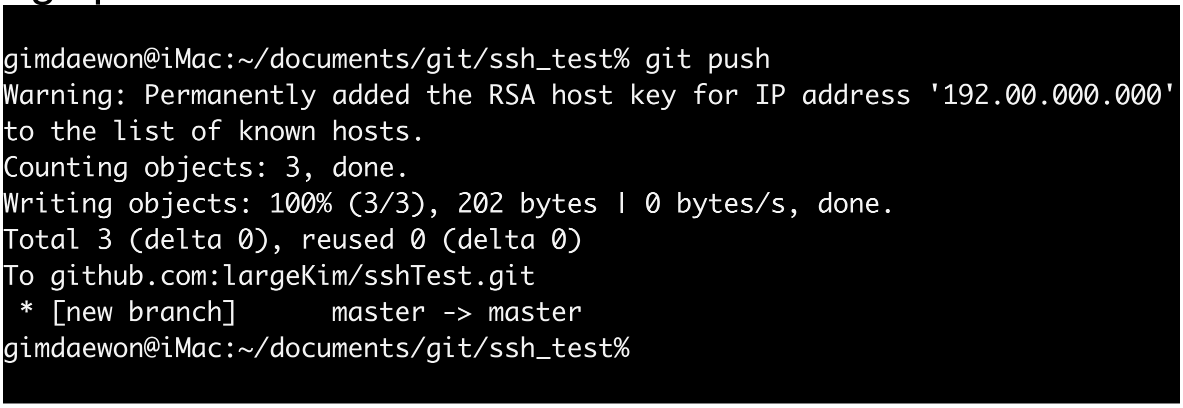


```
a
~
~
```

>git add f1.txt

>git commit -m '1'

>git push



```
gimdaewon@iMac:~/documents/git/ssh_test% git push
Warning: Permanently added the RSA host key for IP address '192.00.000.000'
to the list of known hosts.
Counting objects: 3, done.
Writing objects: 100% (3/3), 202 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To github.com:largeKim/sshTest.git
 * [new branch]      master -> master
gimdaewon@iMac:~/documents/git/ssh_test%
```

잘된다면 성공!!!!