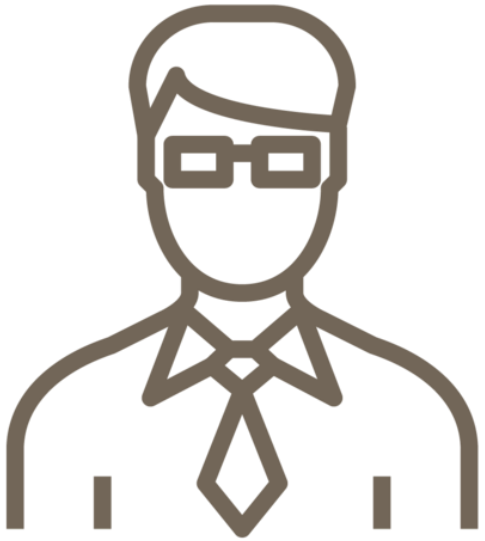
상식적으로 배우는 git/github

윤성국

Modulabs Meani.mo

버전관리

처음 걸로 해주세요



```
#include "polynomial.h"
#include <iostream>

using std::cout;
using std::endl;


//using std::endl;

/* df;hkajasdlhjioeltheklhy
using std::endl;dklklashj
dhklas
aklhjioaenklayhseldhgld */


Polynomial::Polynomial(int termCount): nTerm(termCount)



      {polyno = new double[nTerm];for(int i = 0; i < nTerm; i++)
        polyno[i] = 0;}

Polynomial::Polynomial(const double arr[], int nArr): nTerm(nArr)
{polyno = new double[nTerm];
for(int i = 0; i < nTerm; i++)
polyno[i] = arr[i];}




Polynomial::Polynomial(const Polynomial &poly): nTerm(poly.nTerm)
{polyno = new double[nTerm];
for     (int i = 0; i < nTerm; i++)


        polyno[i] = poly.polyno[i];}

Polynomial::~Polynomial(void)
   {delete [] polyno;}

void Polynomial::reset(int termCount)
{nTerm = termCount; delete [] polyno;



                          polyno = new double[nTerm];

for(int i = 0; i < nTerm; i++)
polyno[i] = 0;}

void Polynomial::show(void)
{bool plus = false;for(int i = nTerm - 1; i >= 0; i--)
```

어떻게 되돌리지?

?

?

버전관리

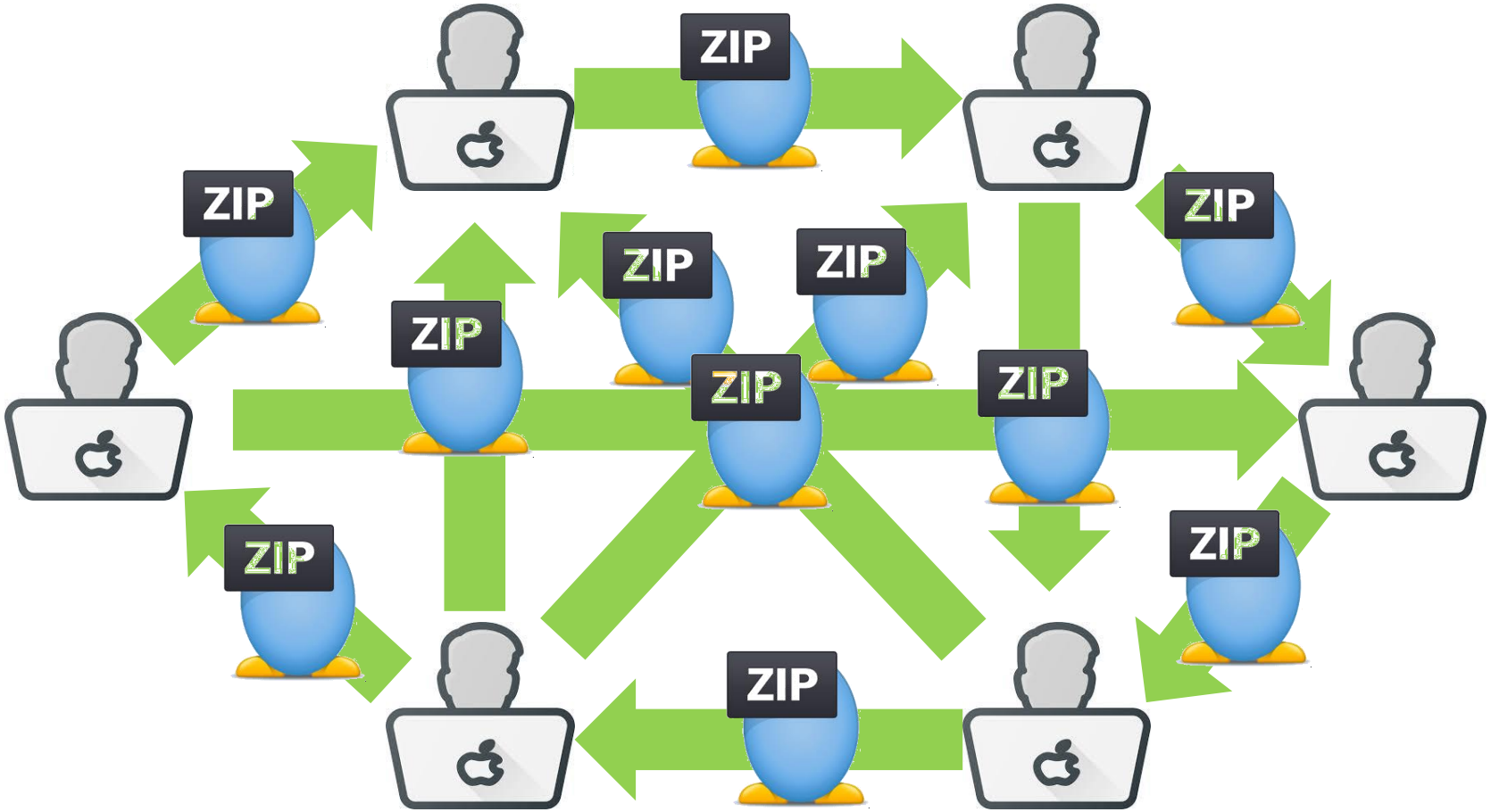
버전관리

최종_시안.png
최종_시안_(1).png
진짜최종.png
Final.png
이게_마지막_시아ㄴ.png
이게_진짜_끝.png
끝.png
최종(2).png
진짜진짜_진짜_최종.png
위에거_다아니고_이게_최종.png

어떤 것이 진짜 최종인 것인가?

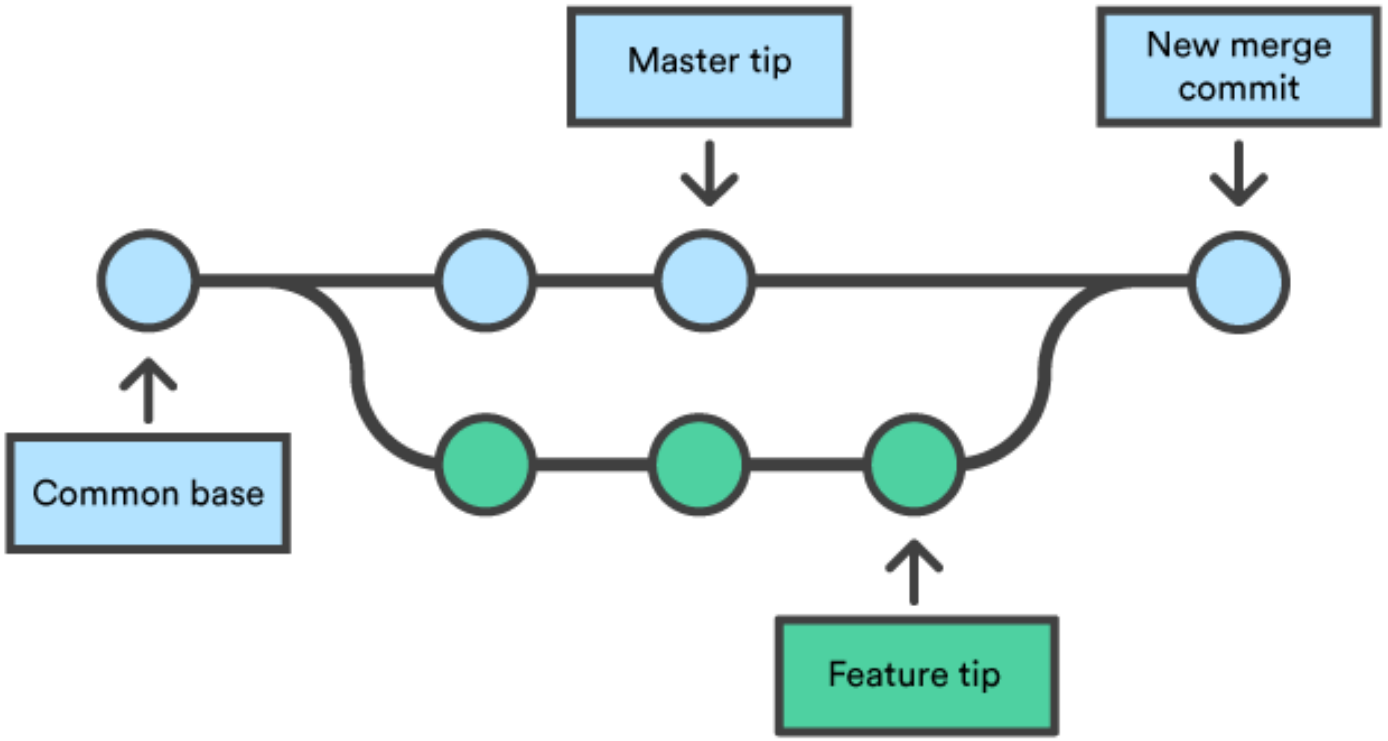
~~이름 제일 긴 게 제일 마지막이더라~~

소스코드 협업

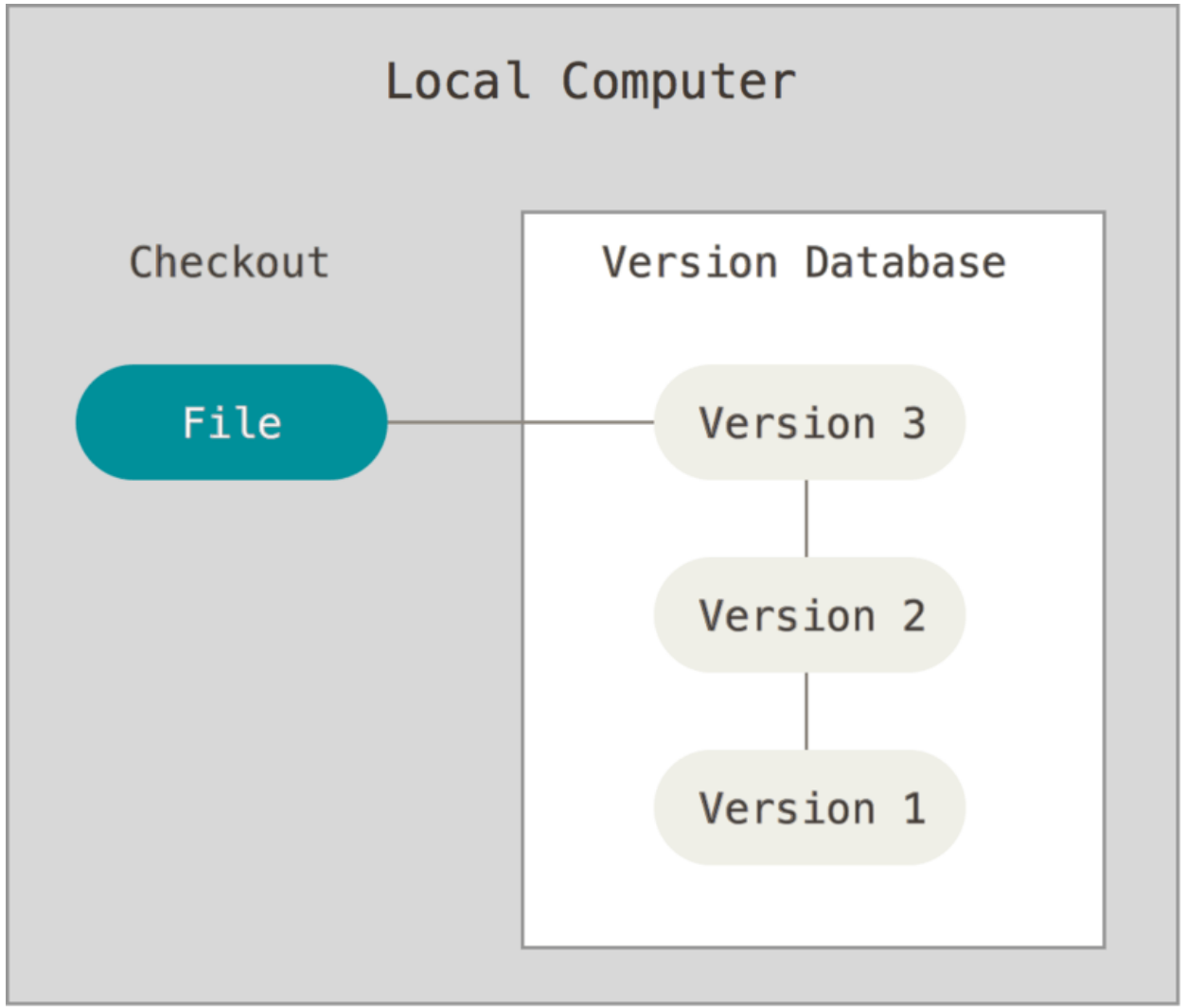


관리가 되지 않음 – 작업했던 게 없어지기도 함

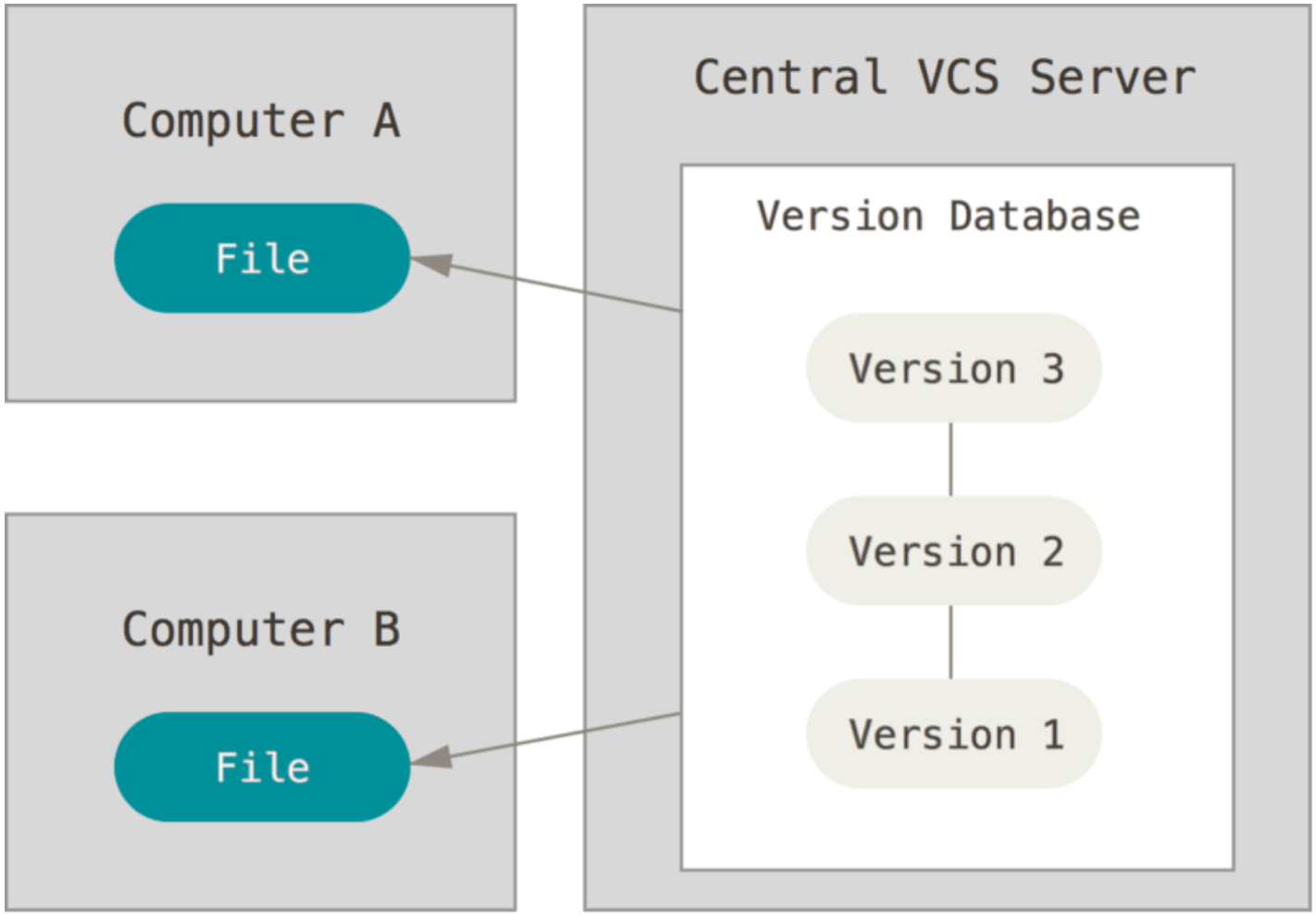
VCS : 버전관리 시스템



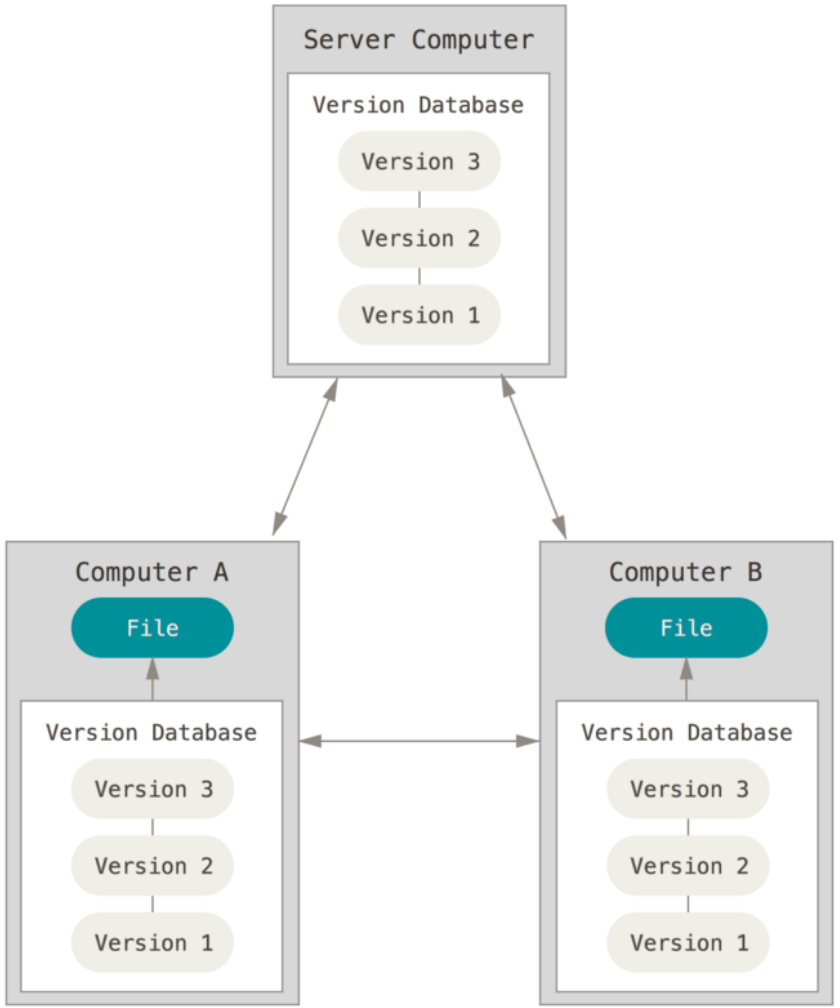
버전관리의 역사 – LVCS



버전관리의 역사 – CVCS



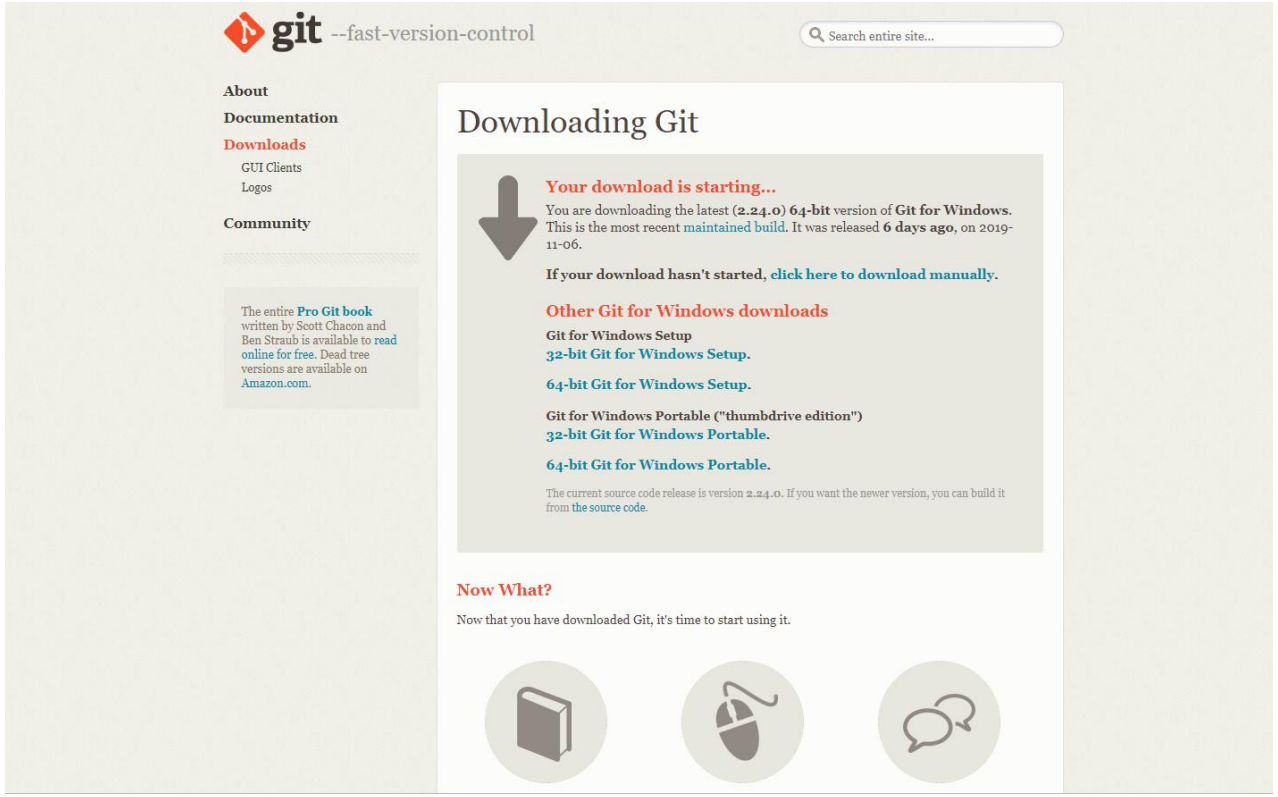
버전관리의 역사 - DVCS



git 설치

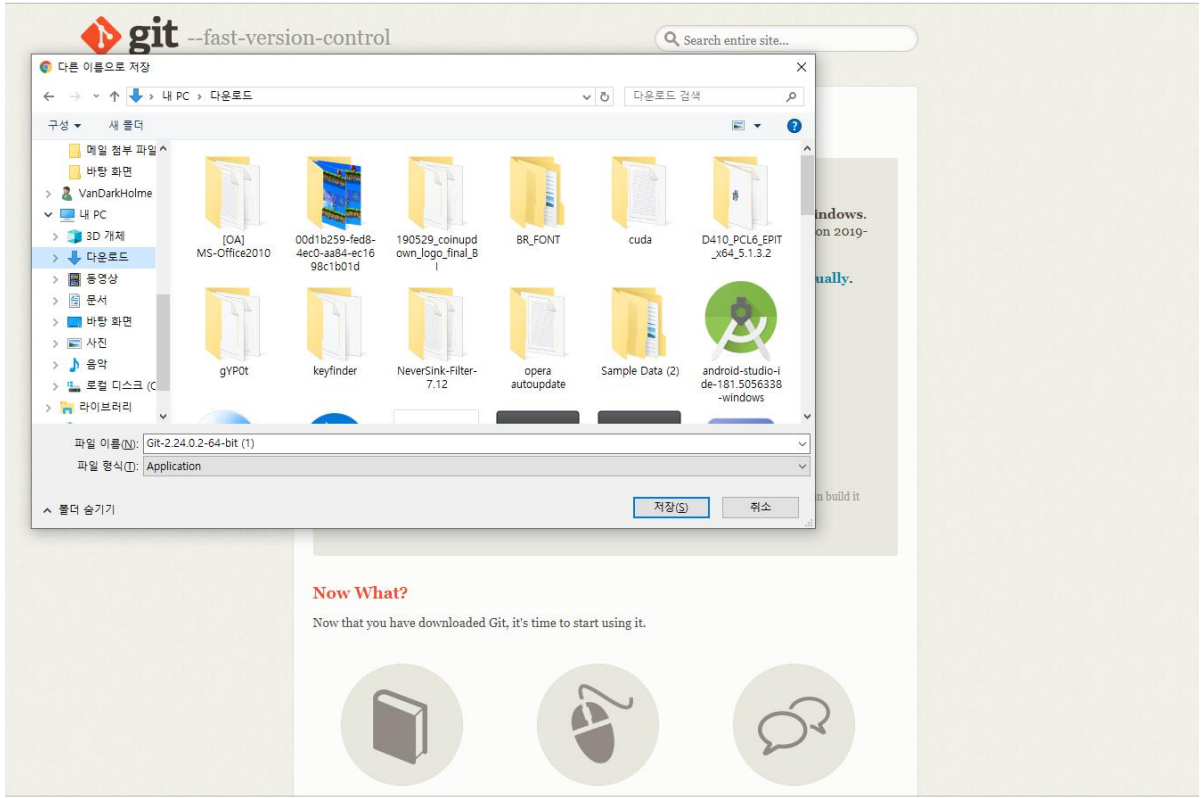
- windows

https://git-scm.com/download/win



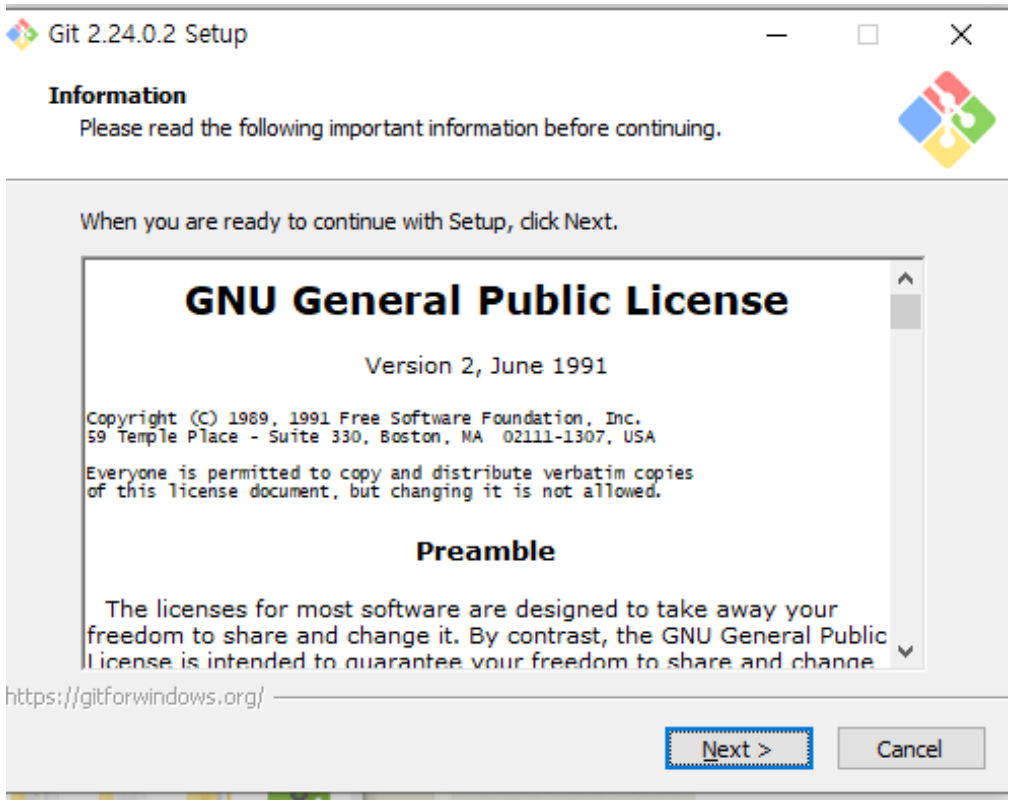
git 설치

- windows



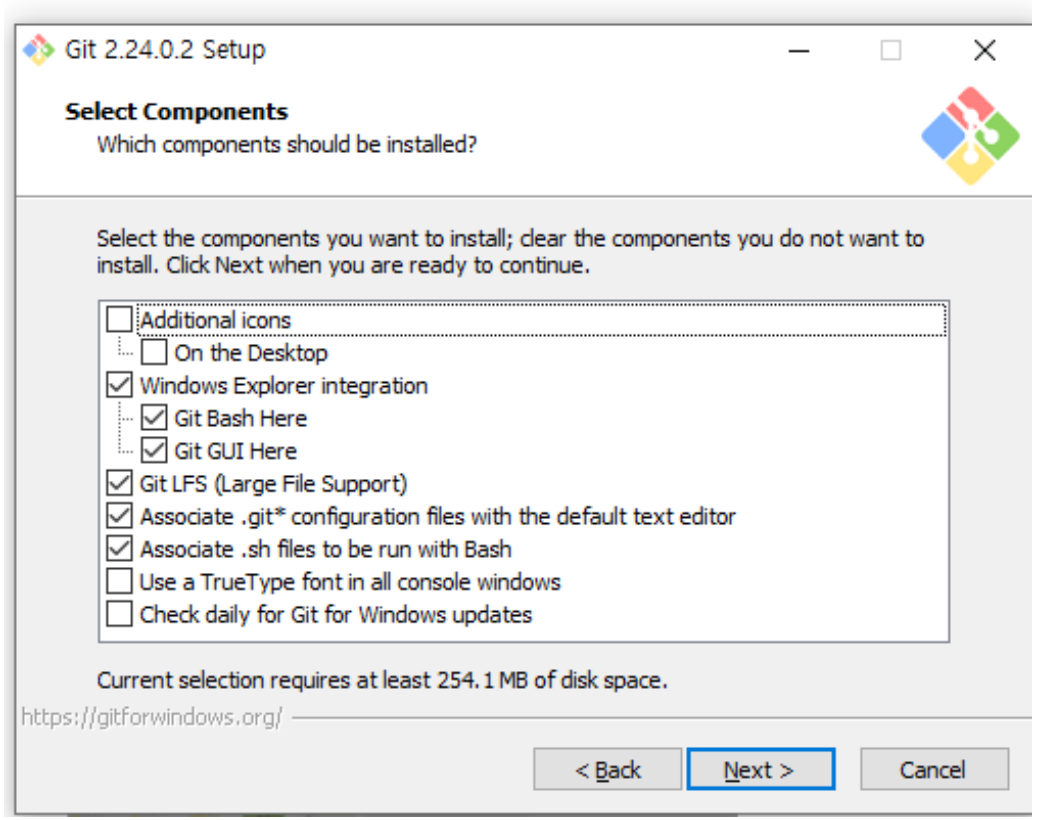
git 설치

- windows



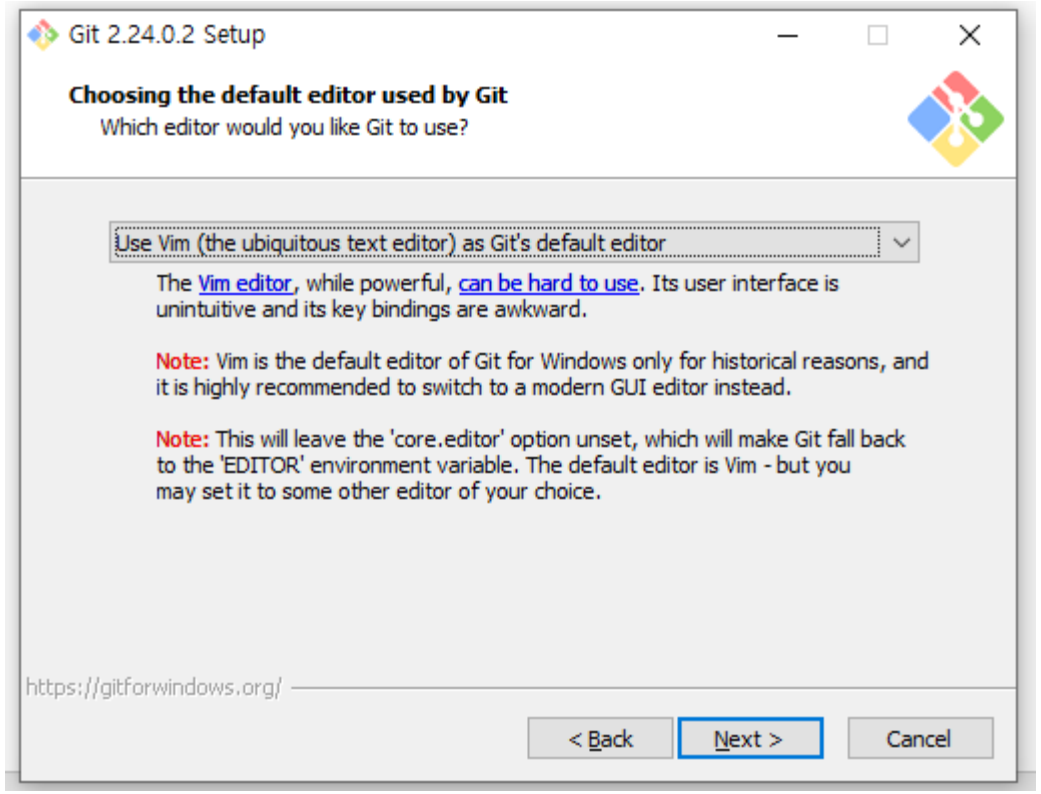
git 설치

- windows



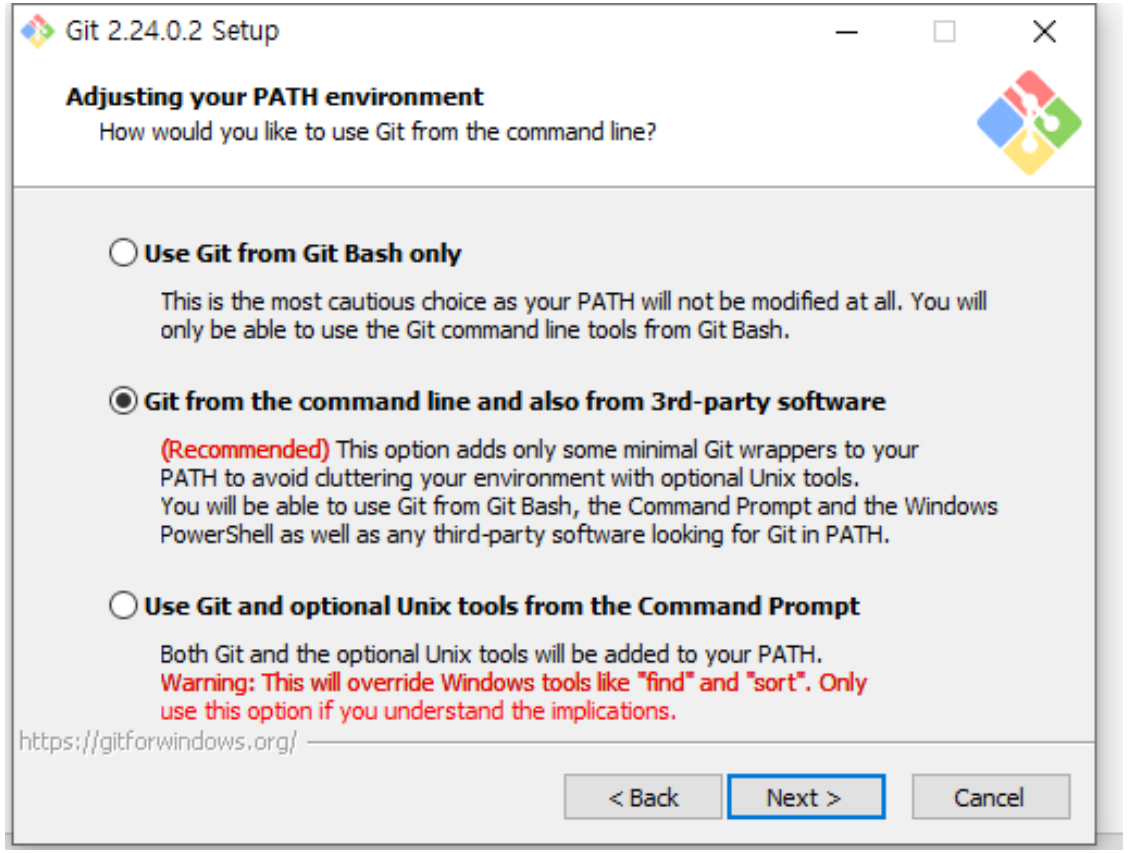
git 설치

- windows



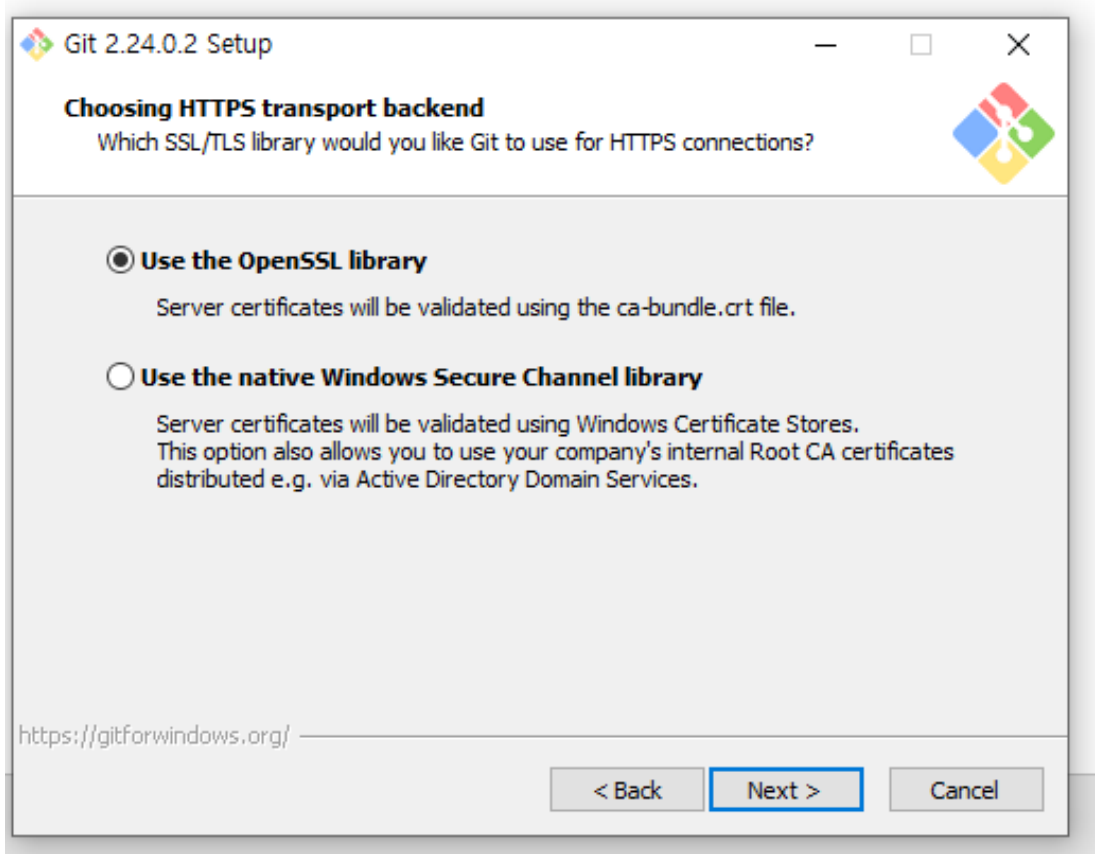
git 설치

- windows



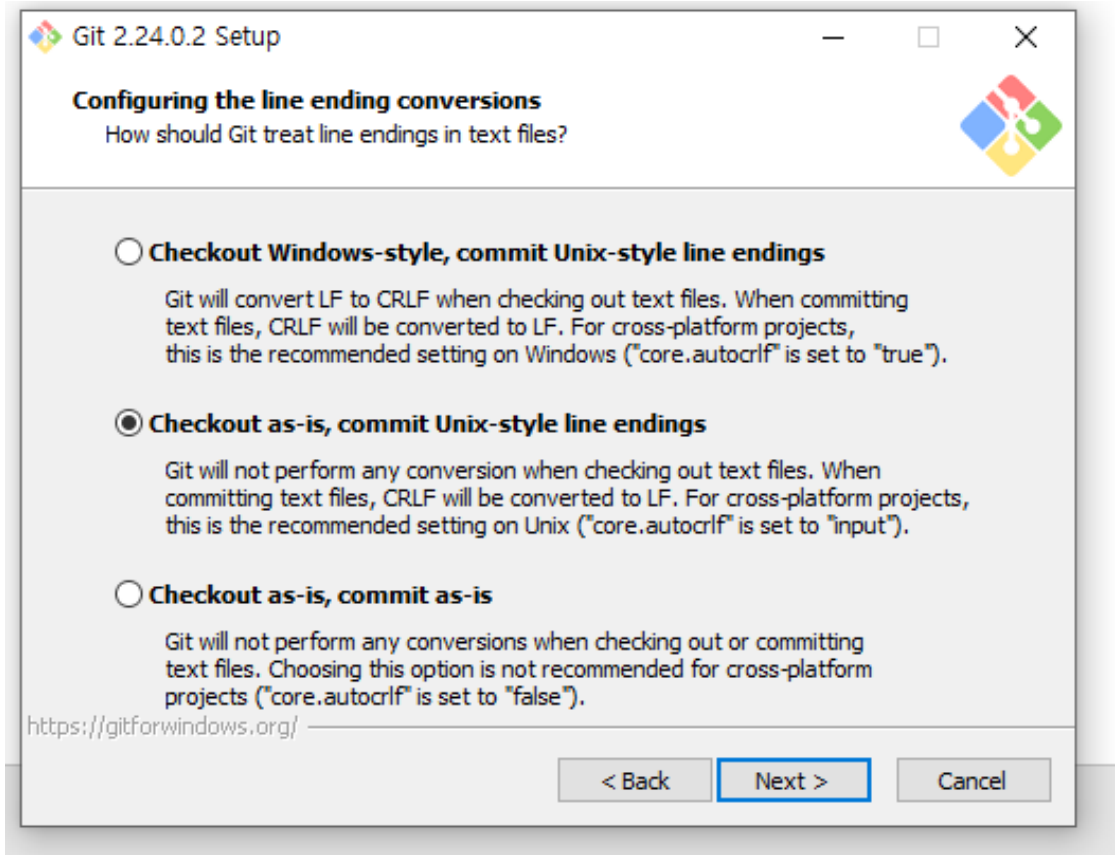
git 설치

- windows



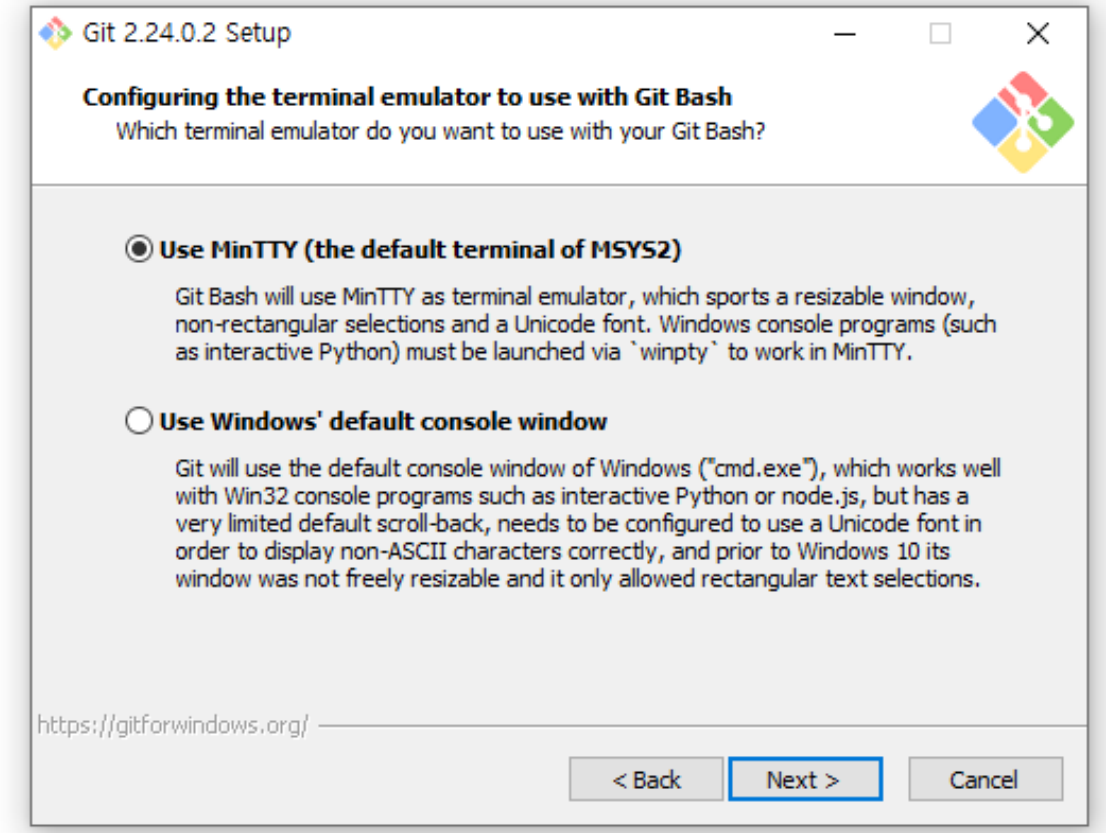
git 설치

- windows



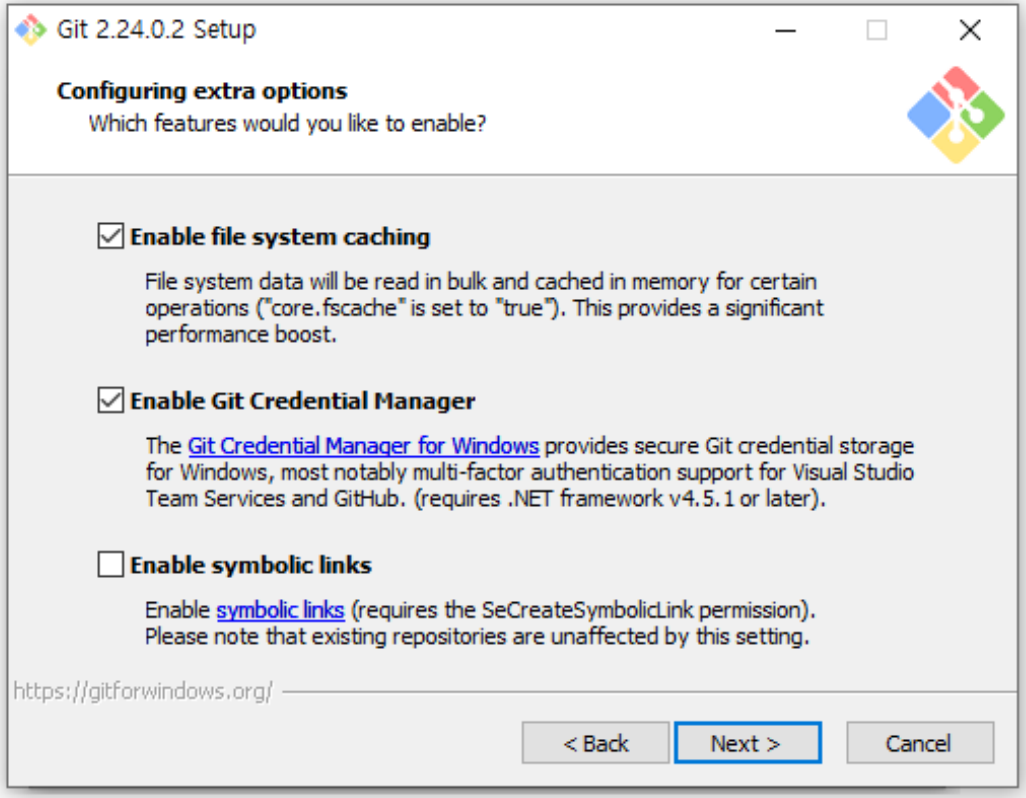
git 설치

- windows



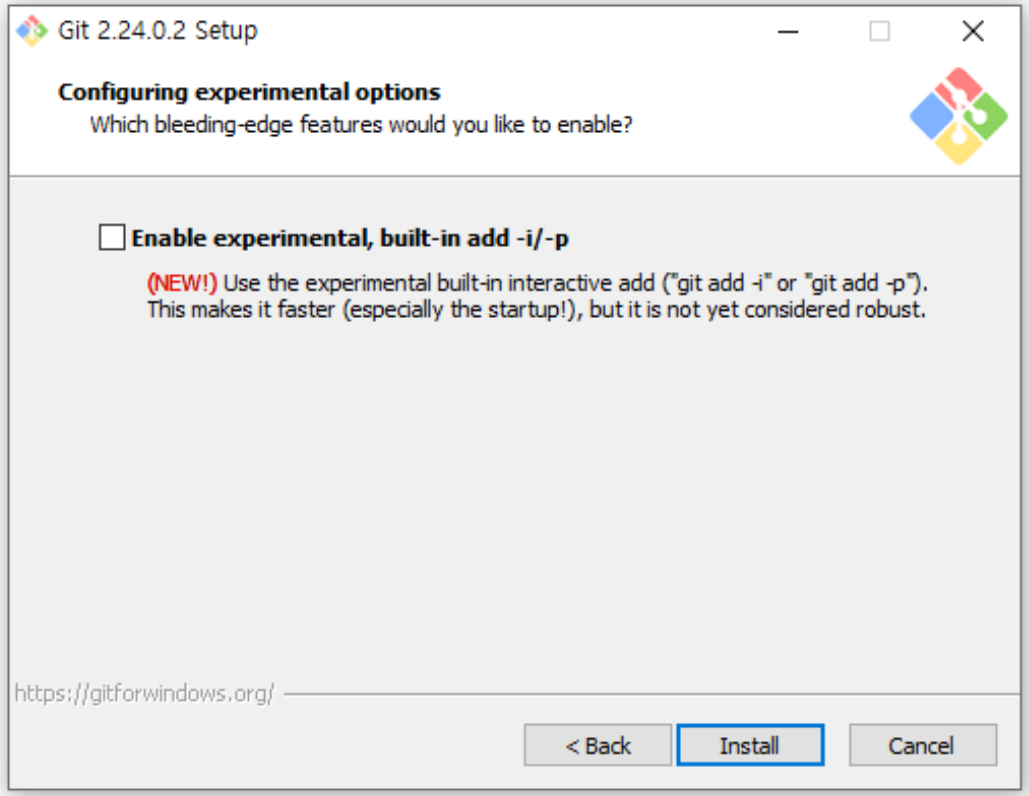
git 설치

- windows



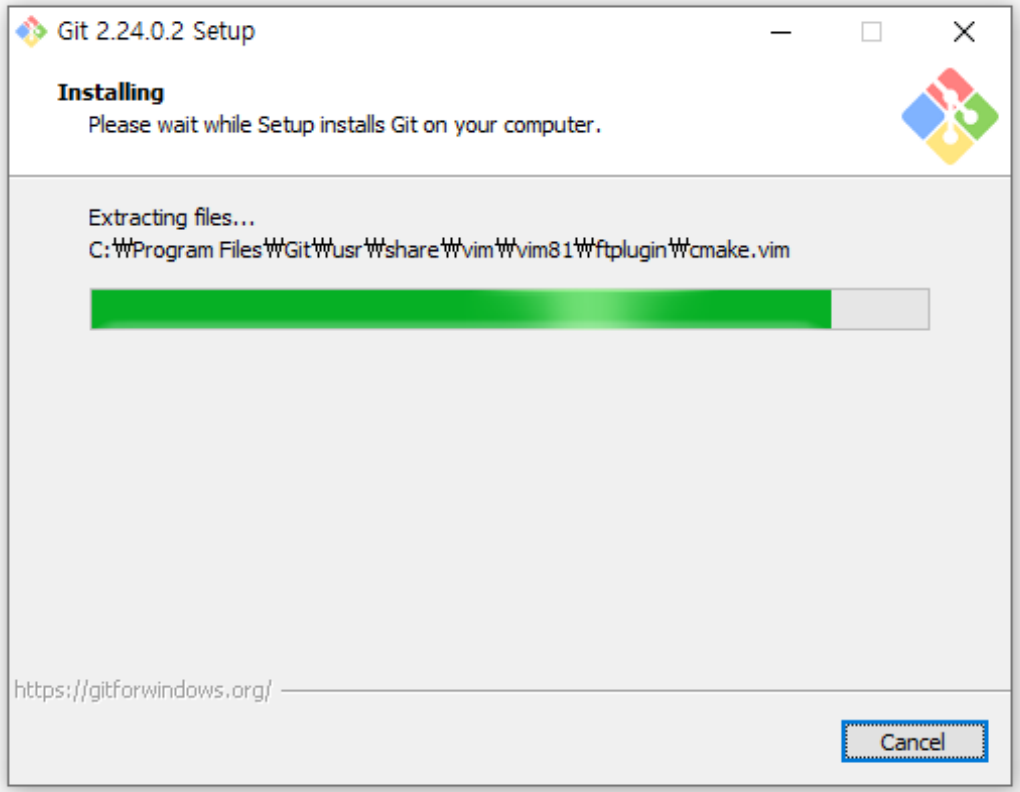
git 설치

- windows



git 설치

- windows



git 설치

- Mac OS

https://git-scm.com/download/mac

git 설치

- Linux

RedHat 계열

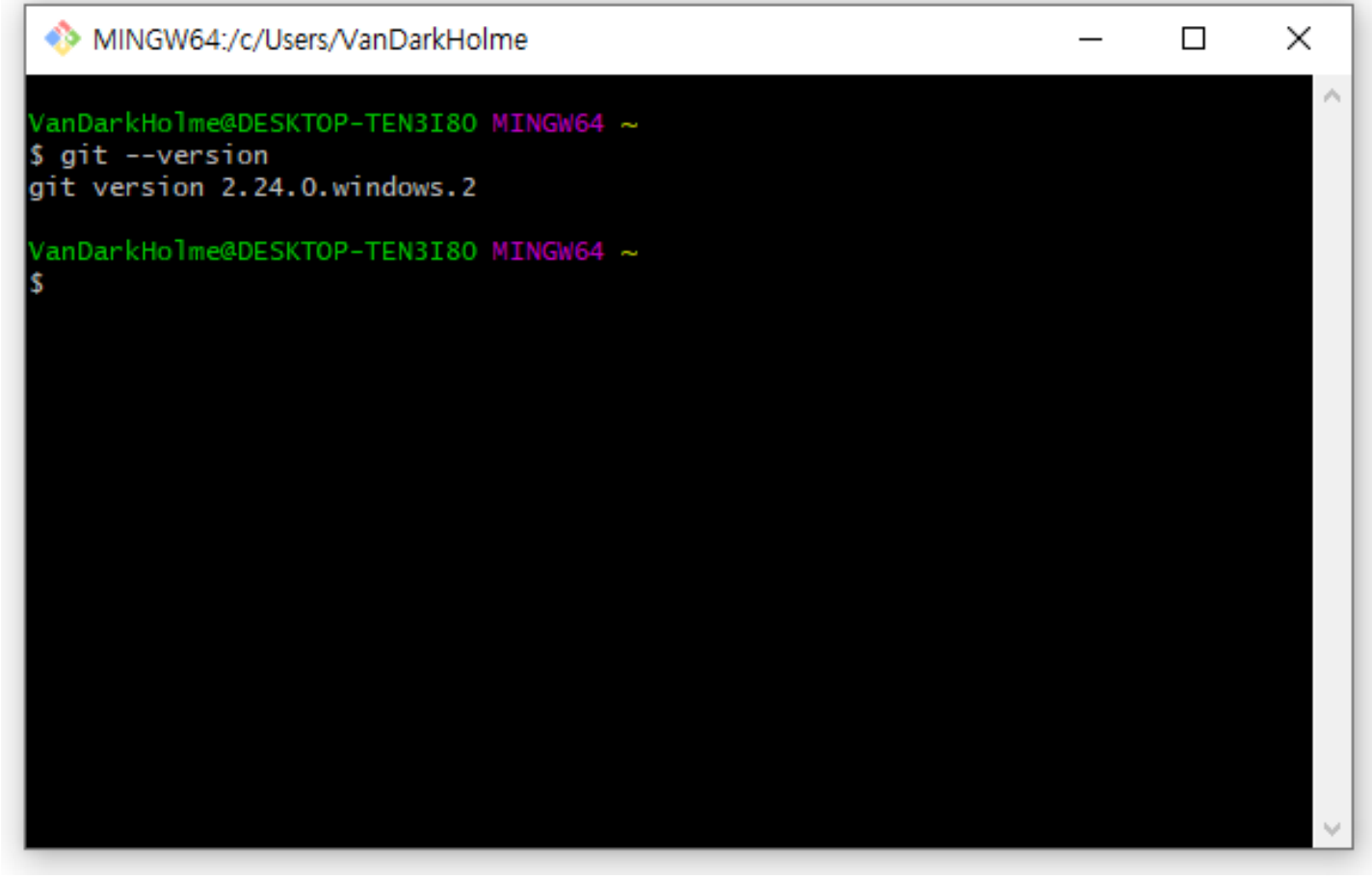
```
$ sudo dnf install git-all
```

Debian 계열

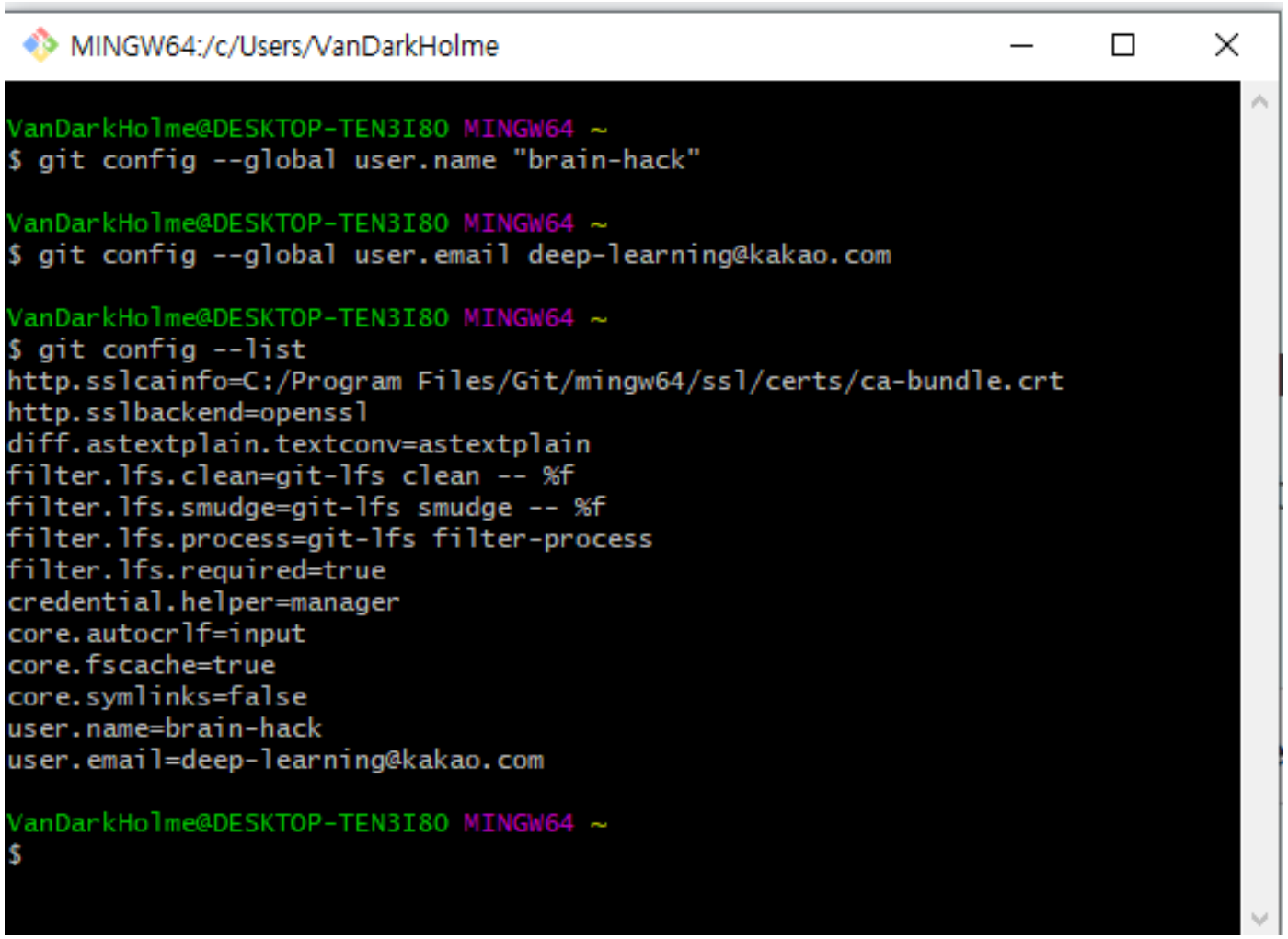
```
$ sudo apt install git-all
```

git 설치

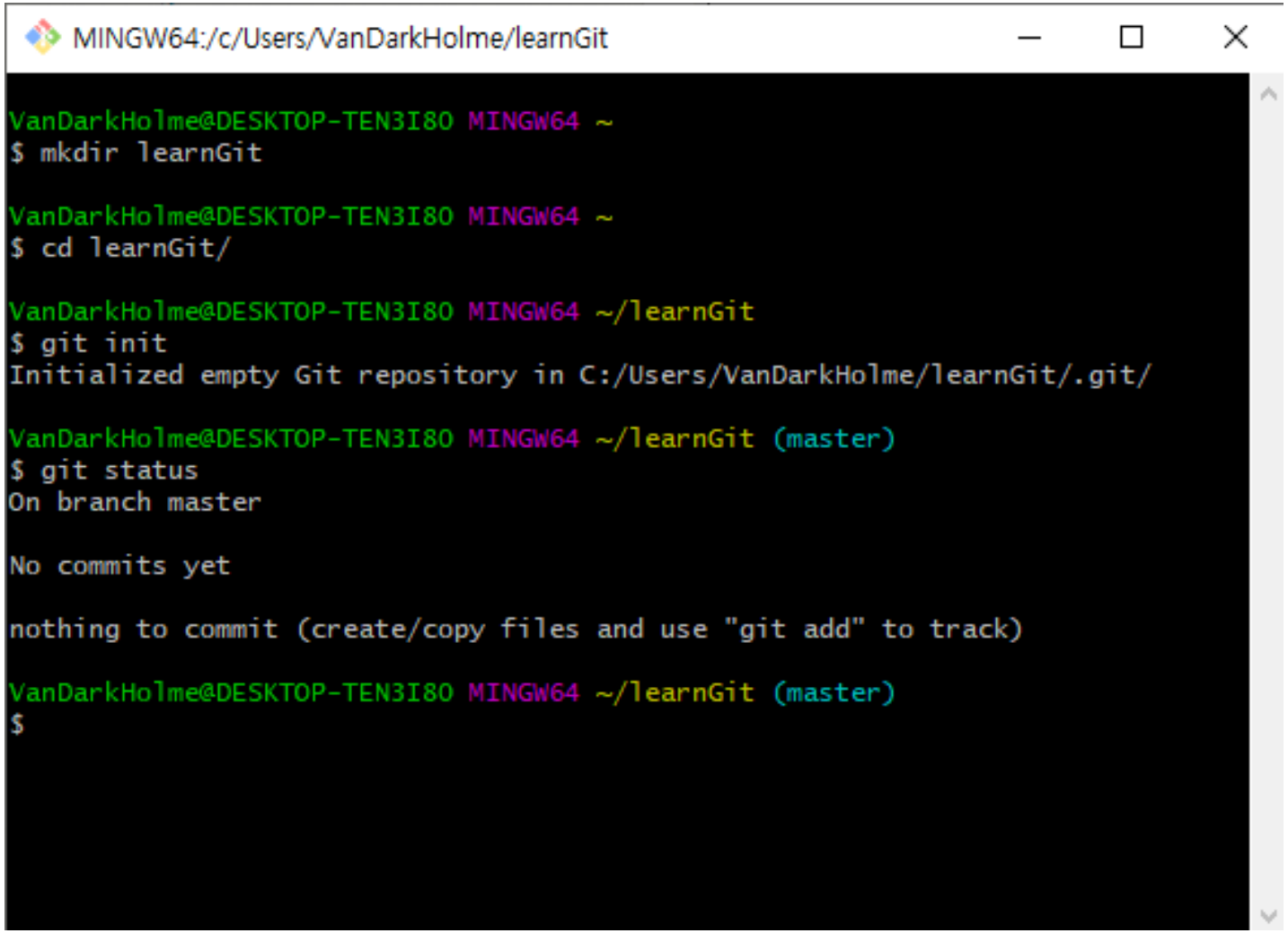
- 설치확인


MINGW64:/c/Users/VanDarkHolme
VanDarkHolme@DESKTOP-TEN3I8O MINGW64 ~
$ git --version
git version 2.24.0.windows.2
VanDarkHolme@DESKTOP-TEN3I8O MINGW64 ~
$

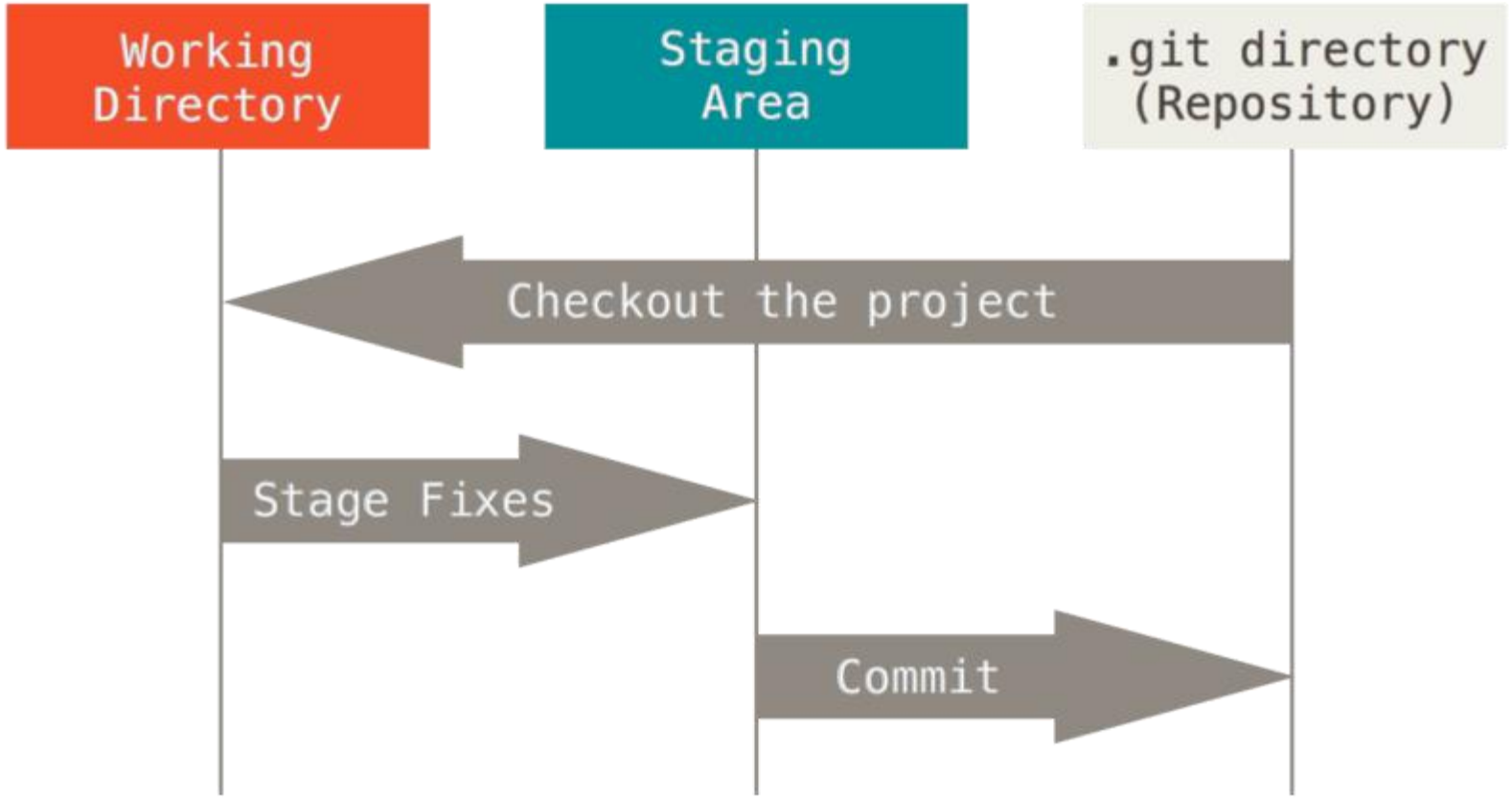
사용자 정보 지정


MINGW64:/c/Users/VanDarkHolme
VanDarkHolme@DESKTOP-TEN3I8O MINGW64 ~
$ git config --global user.name "brain-hack"
VanDarkHolme@DESKTOP-TEN3I8O MINGW64 ~
$ git config --global user.email deep-learning@kakao.com
VanDarkHolme@DESKTOP-TEN3I8O MINGW64 ~
$ git config --list
http.sslcainfo=C:/Program Files/Git/mingw64/ssl/certs/ca-bundle.crt
http.sslbackend=openssl
diff.astextplain.textconv=astextplain
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
credential.helper=manager
core.autocrlf=input
core.fscache=true
core.symlinks=false
user.name=brain-hack
user.email=deep-learning@kakao.com
VanDarkHolme@DESKTOP-TEN3I8O MINGW64 ~
$

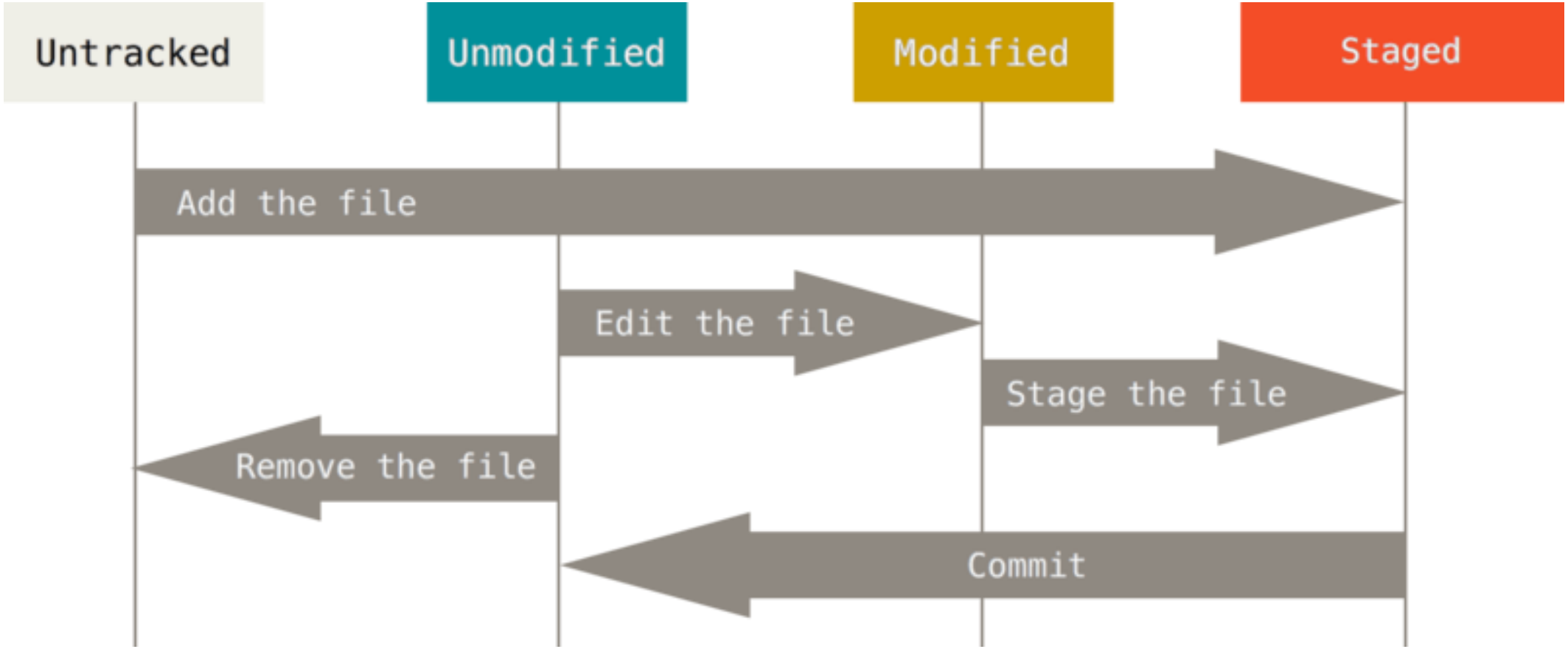
git repository 만들기



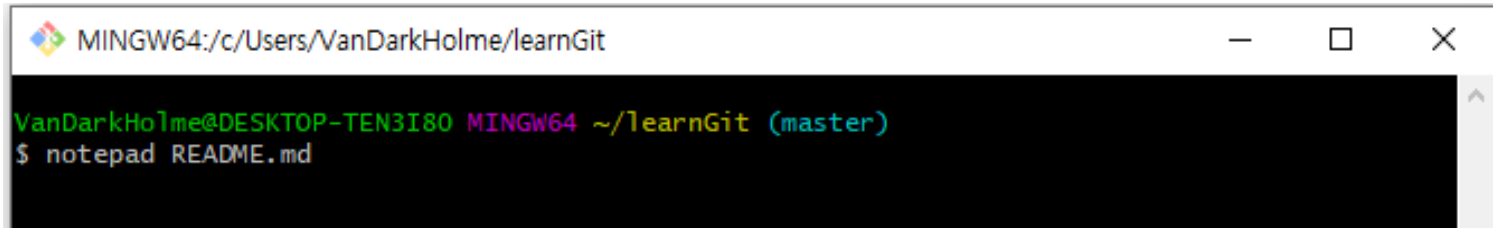
git project의 세가지 단계

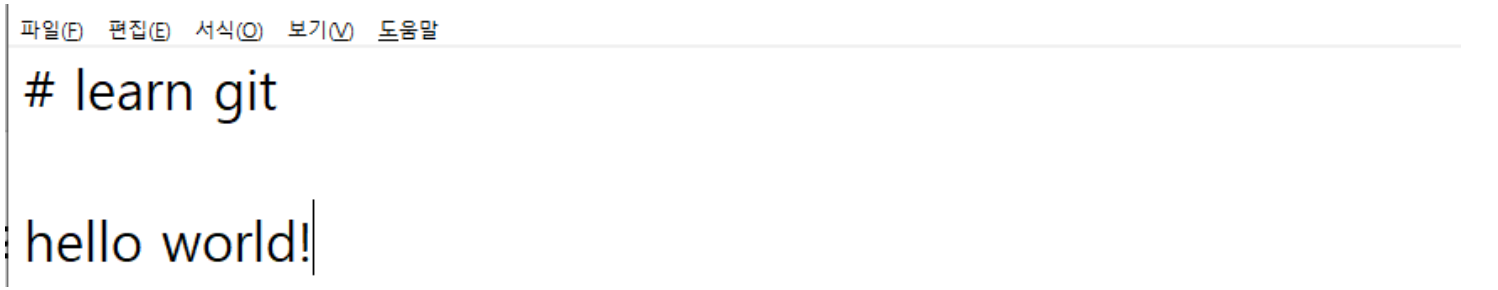


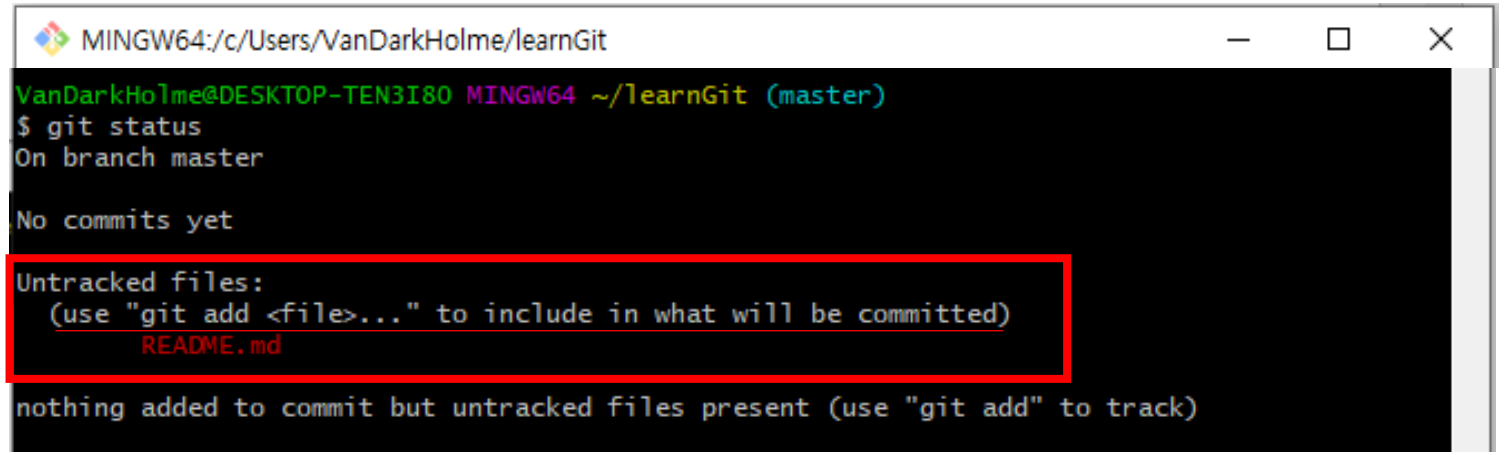
git repo에서 file의 life cycle



git repo에 파일 추가해보기

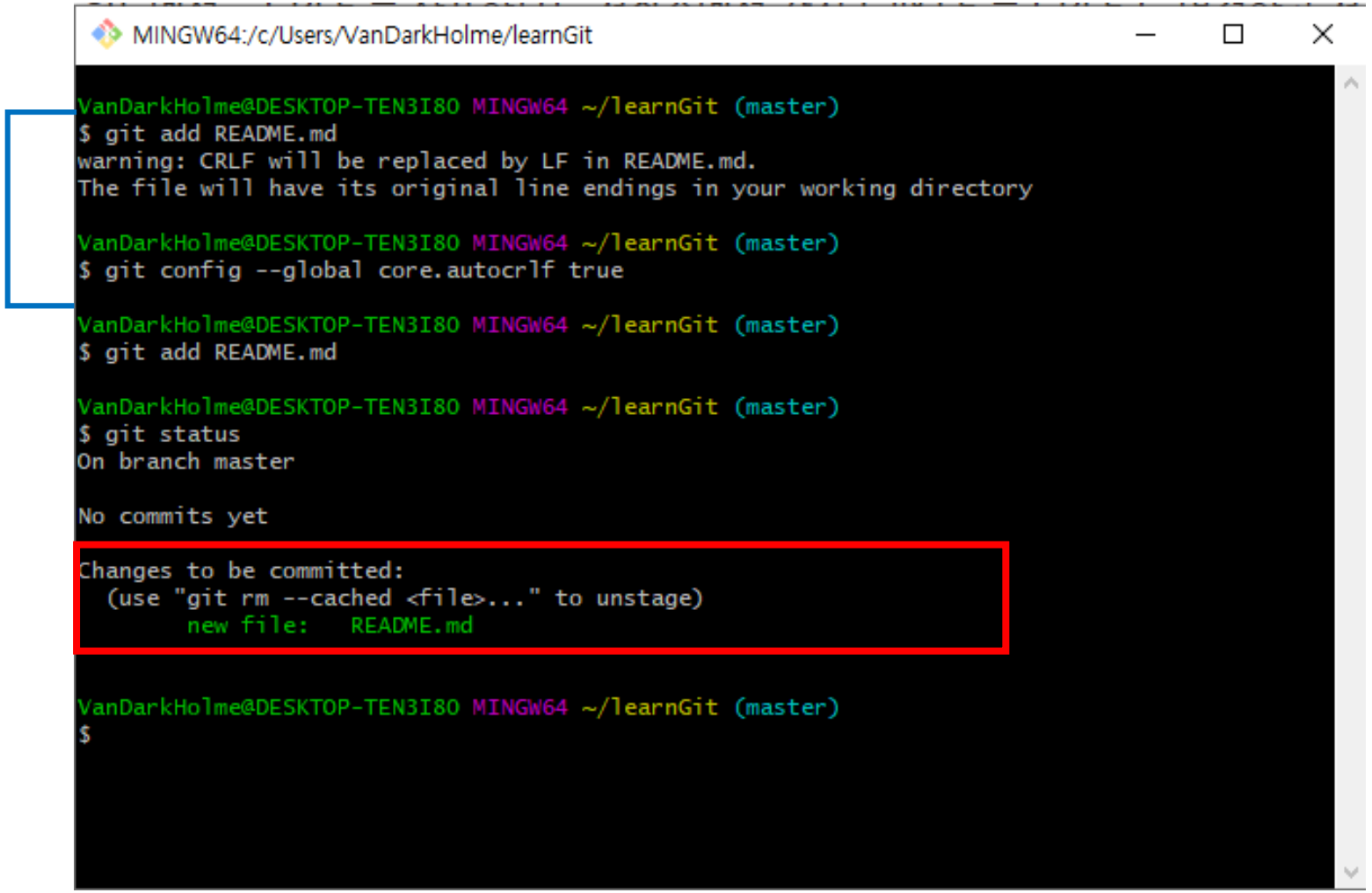




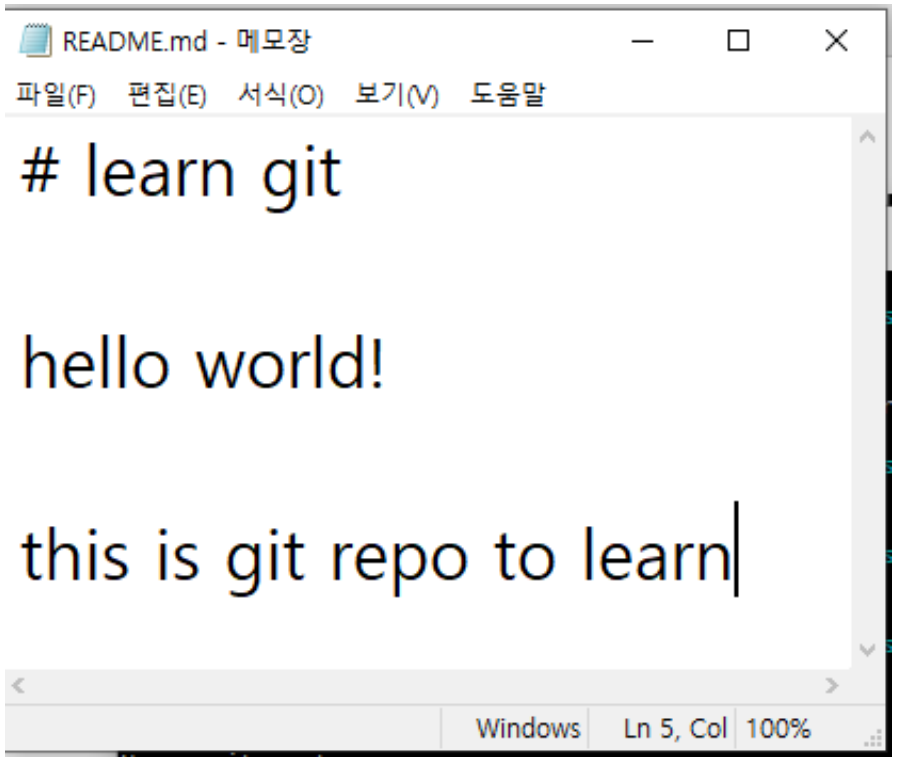


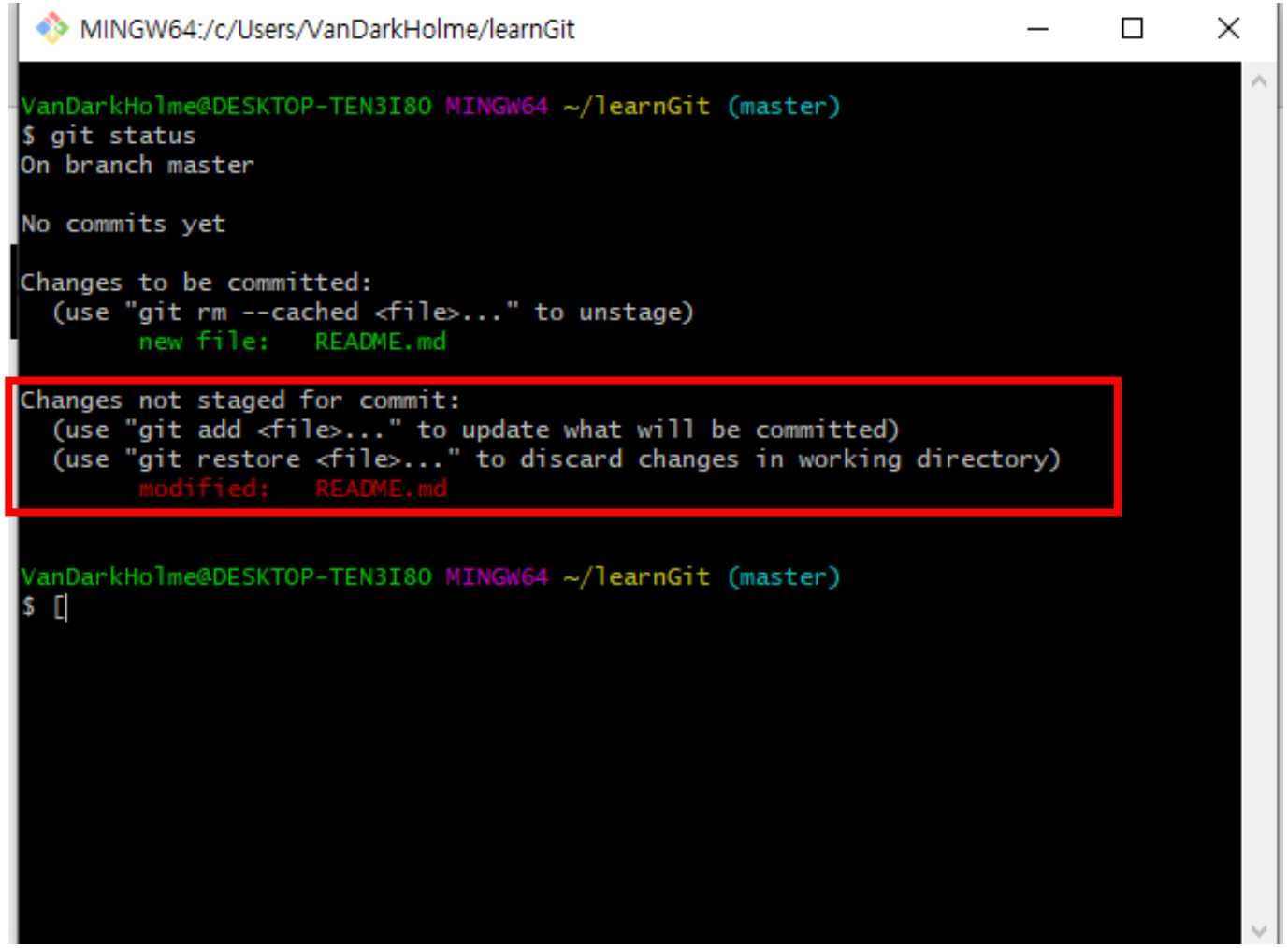
file을 stage에 올리기

windows의 경우

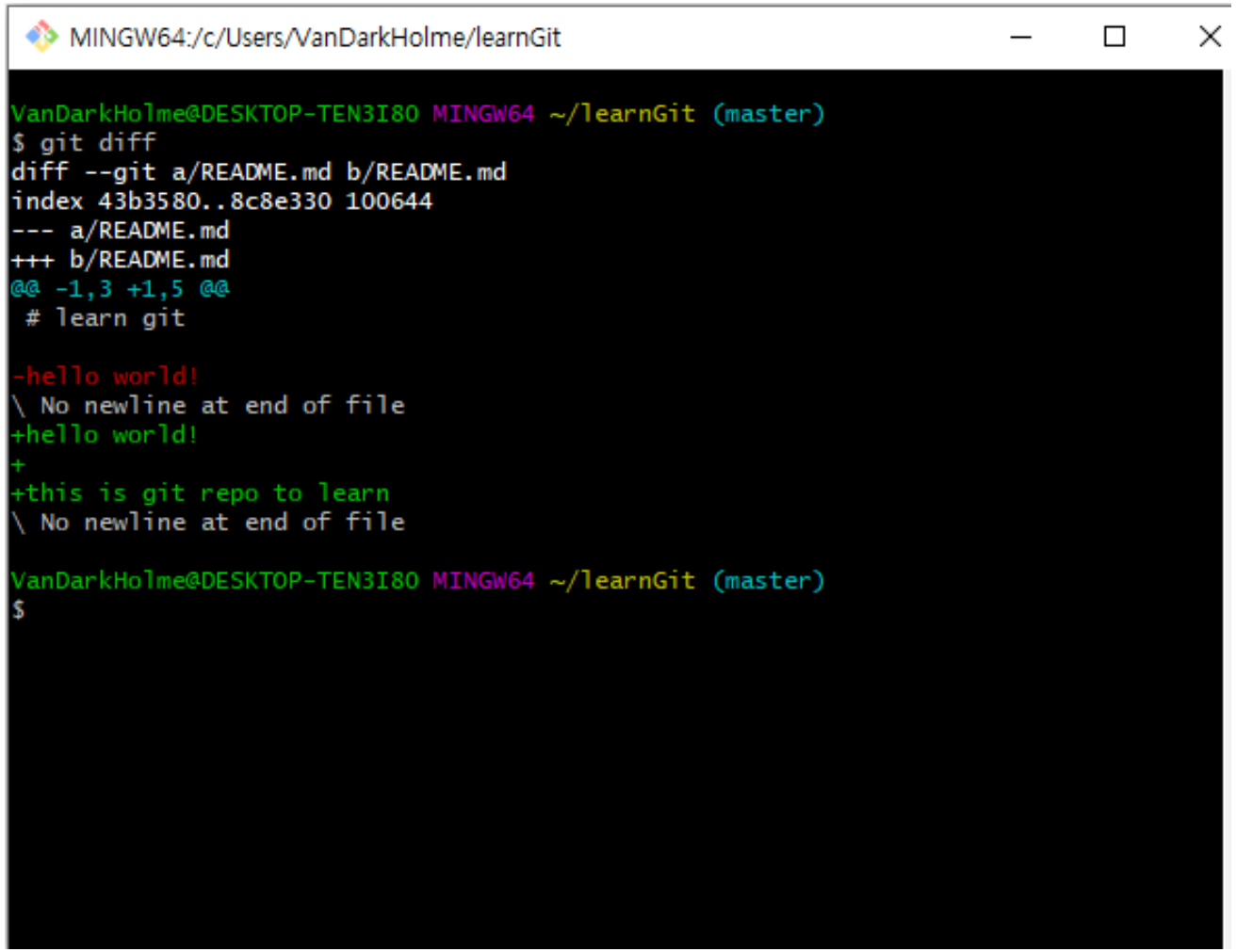


file 내용을 추가해보기

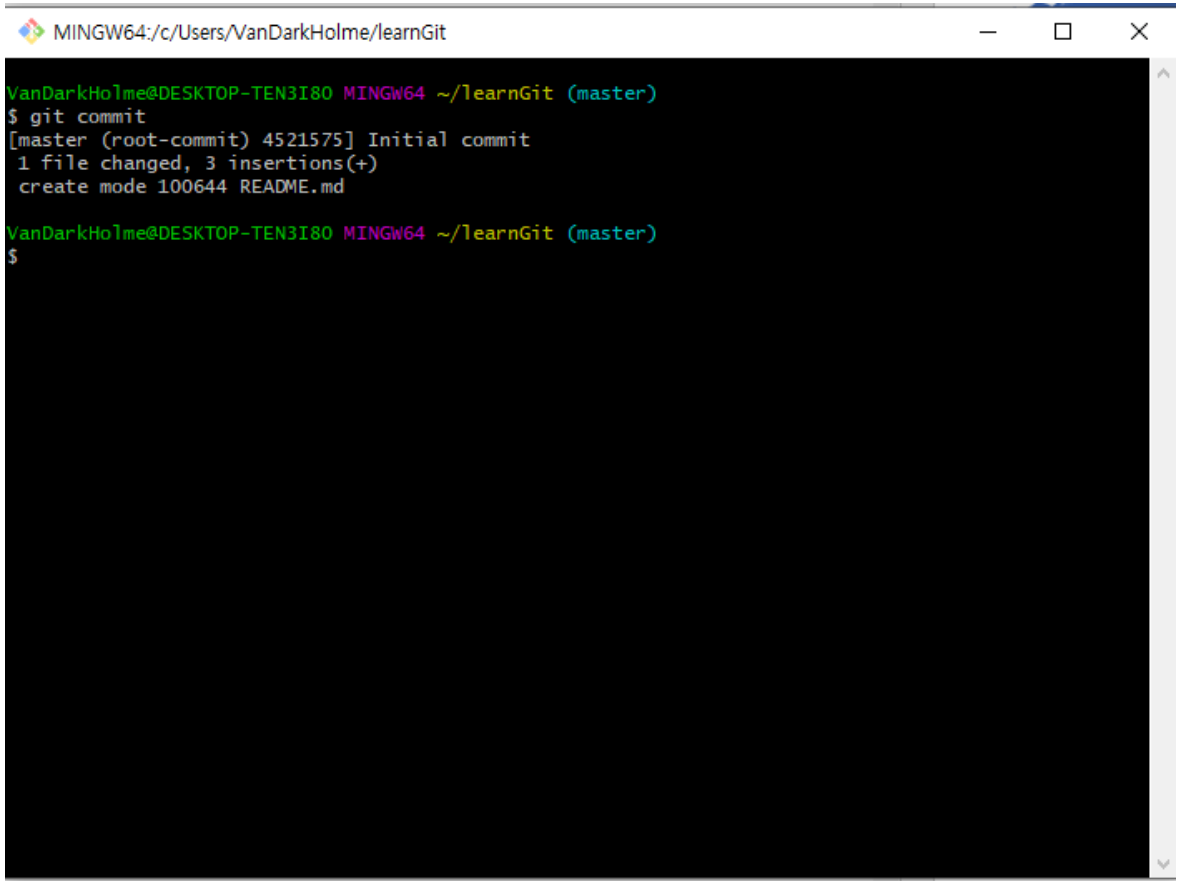


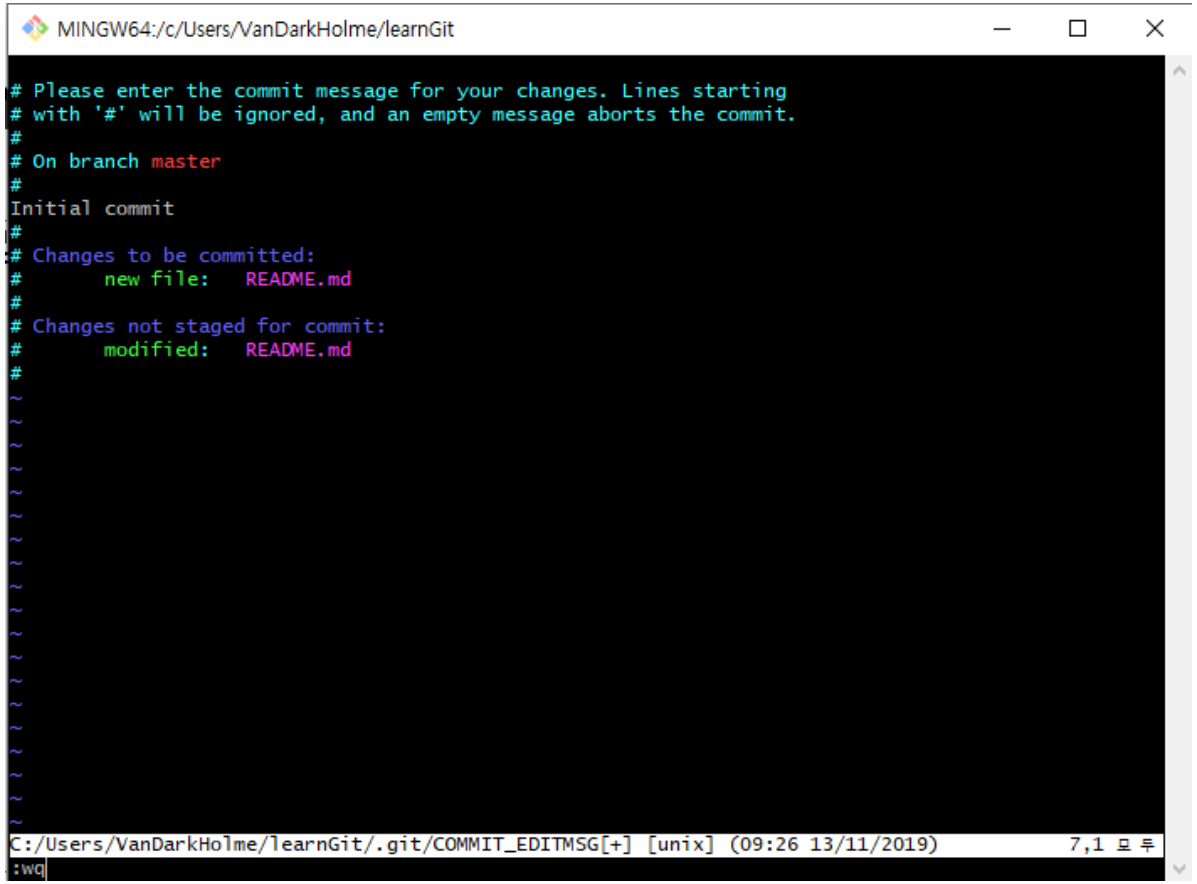


변경된 file 내용을 확인해보기

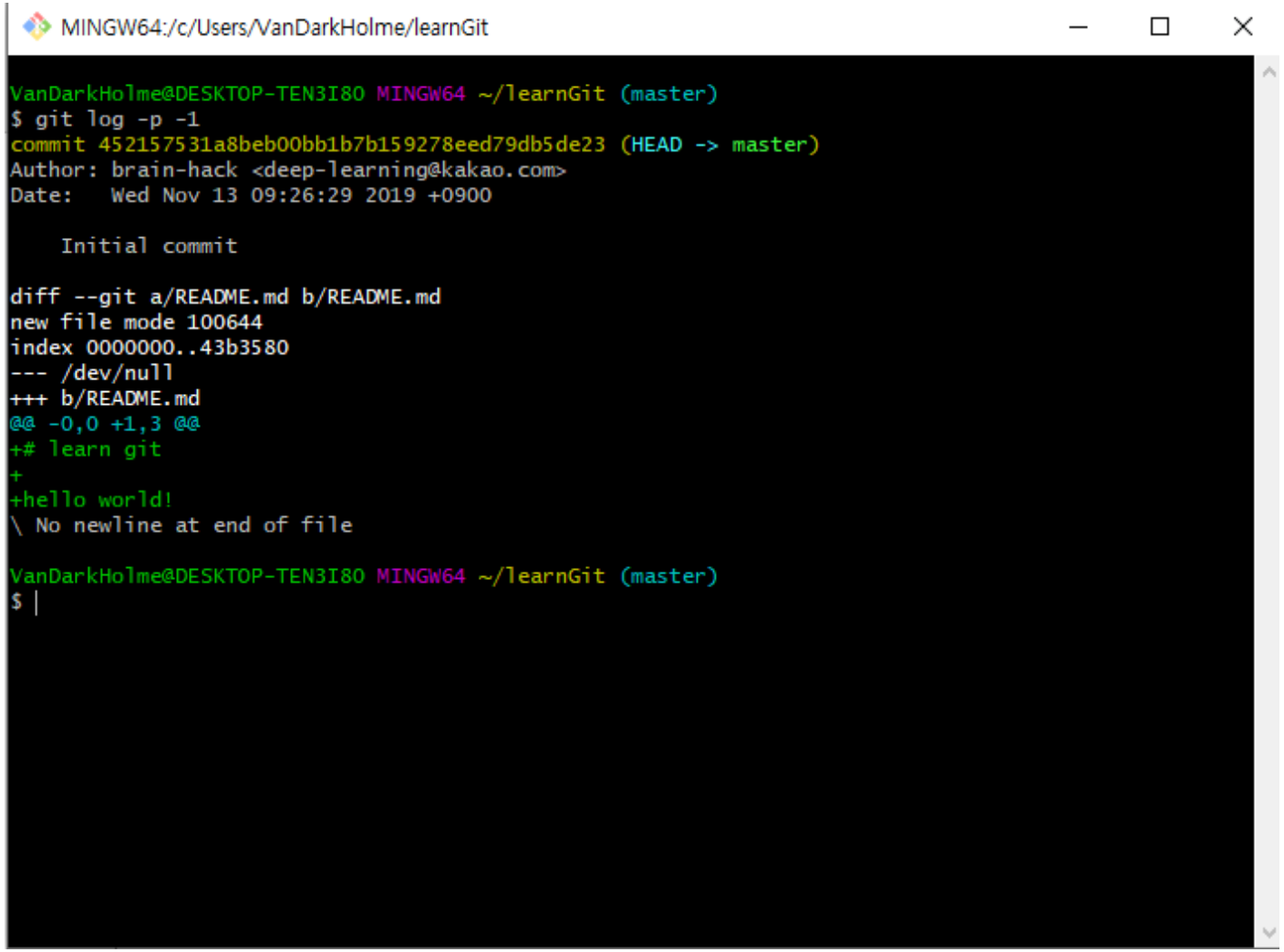


버전 commit 하기


MINGW64:/c/Users/VanDarkHolme/learnGit
VanDarkHolme@DESKTOP-TEN3I8O MINGW64 ~/learnGit (master)
$ git commit
[master (root-commit) 4521575] Initial commit
 1 file changed, 3 insertions(+)
 create mode 100644 README.md
VanDarkHolme@DESKTOP-TEN3I8O MINGW64 ~/learnGit (master)
$


MINGW64:/c/Users/VanDarkHolme/learnGit
Please enter the commit message for your changes. Lines starting
with '#' will be ignored, and an empty message aborts the commit.
#
On branch master
#
Initial commit
#
Changes to be committed:
new file: README.md
#
Changes not staged for commit:
modified: README.md
#
C:/Users/VanDarkHolme/learnGit/.git/COMMIT_EDITMSG[+] [unix] (09:26 13/11/2019) 7,1 모 두
:wq

commit 내역 확인하기



좀 이상하지 않나요?

MINGW64:/c/Users/VanDarkHolme/learnGit

```
VanDarkHolme@DESKTOP-TEN3I8O MINGW64 ~/learnGit (master)
$ git log -p -1
commit 452157531a8beb00bb1b7b159278eed79db5de23 (HEAD -> master)
Author: brain-hack <deep-learning@kakao.com>
Date:   Wed Nov 13 09:26:29 2019 +0900

    Initial commit

diff --git a/README.md b/README.md
new file mode 100644
index 0000000..43b3580
--- /dev/null
+++ b/README.md
@@ -0,0 +1,3 @@
+# learn git
+
+hello world!
\ No newline at end of file

VanDarkHolme@DESKTOP-TEN3I8O MINGW64 ~/learnGit (master)
$
```

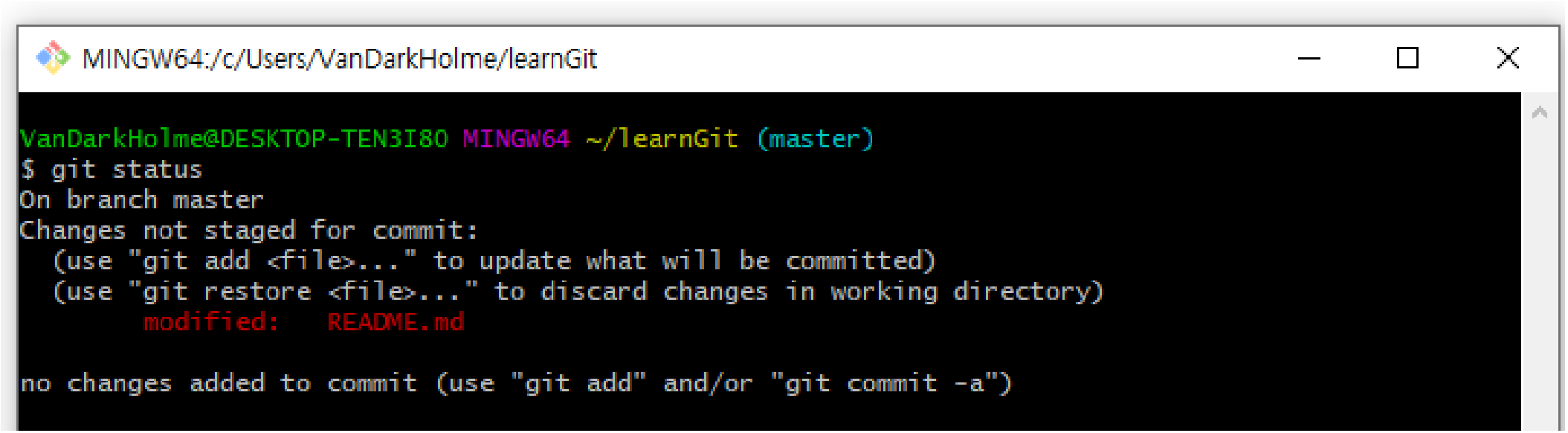
README.md - 메모장

파일(F) 편집(E) 서식(O) 보기(V) 도움말

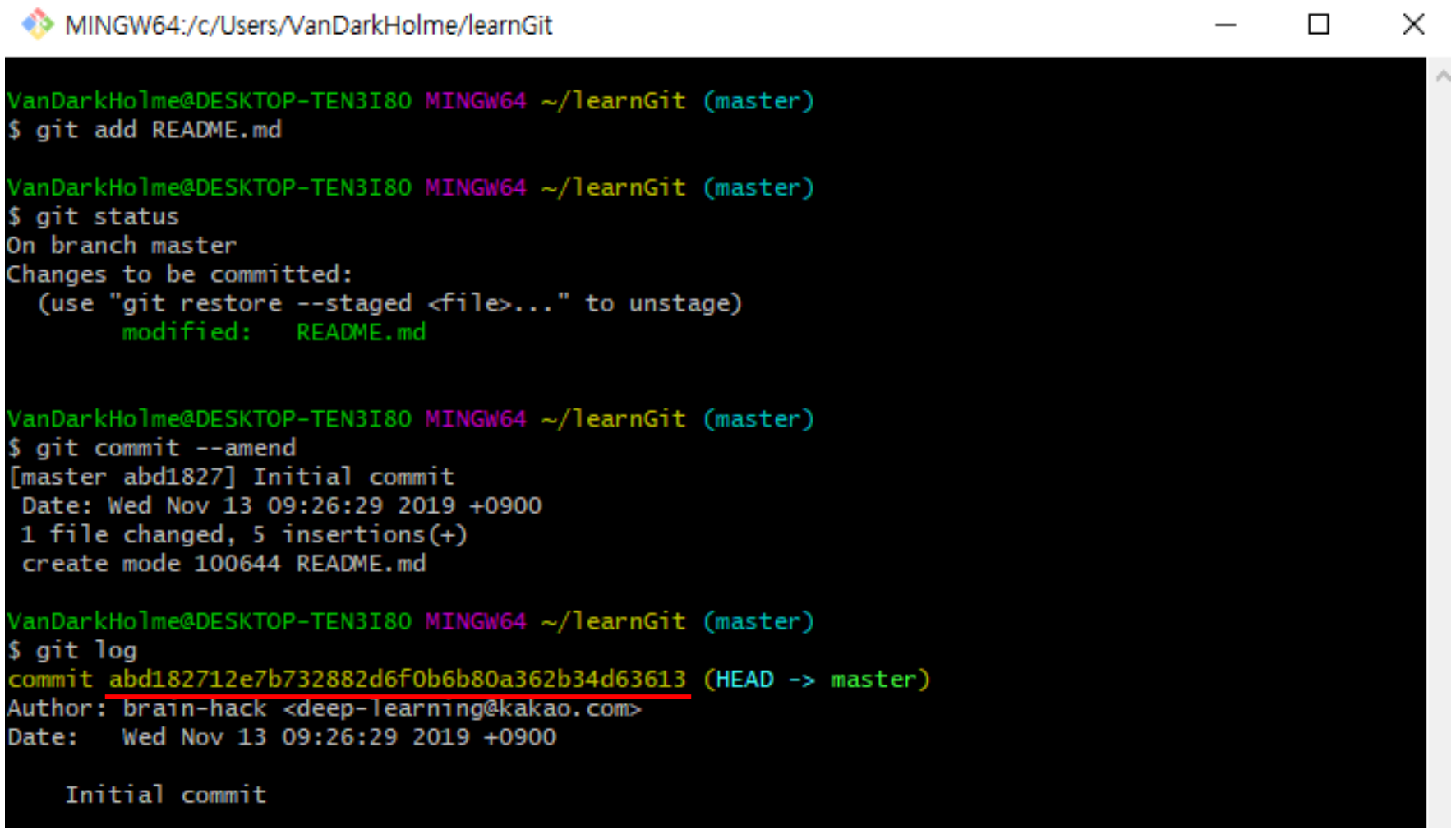
```
# learn git

hello world!

this is git repo to learn
```

Windows Ln 5, Col 100%

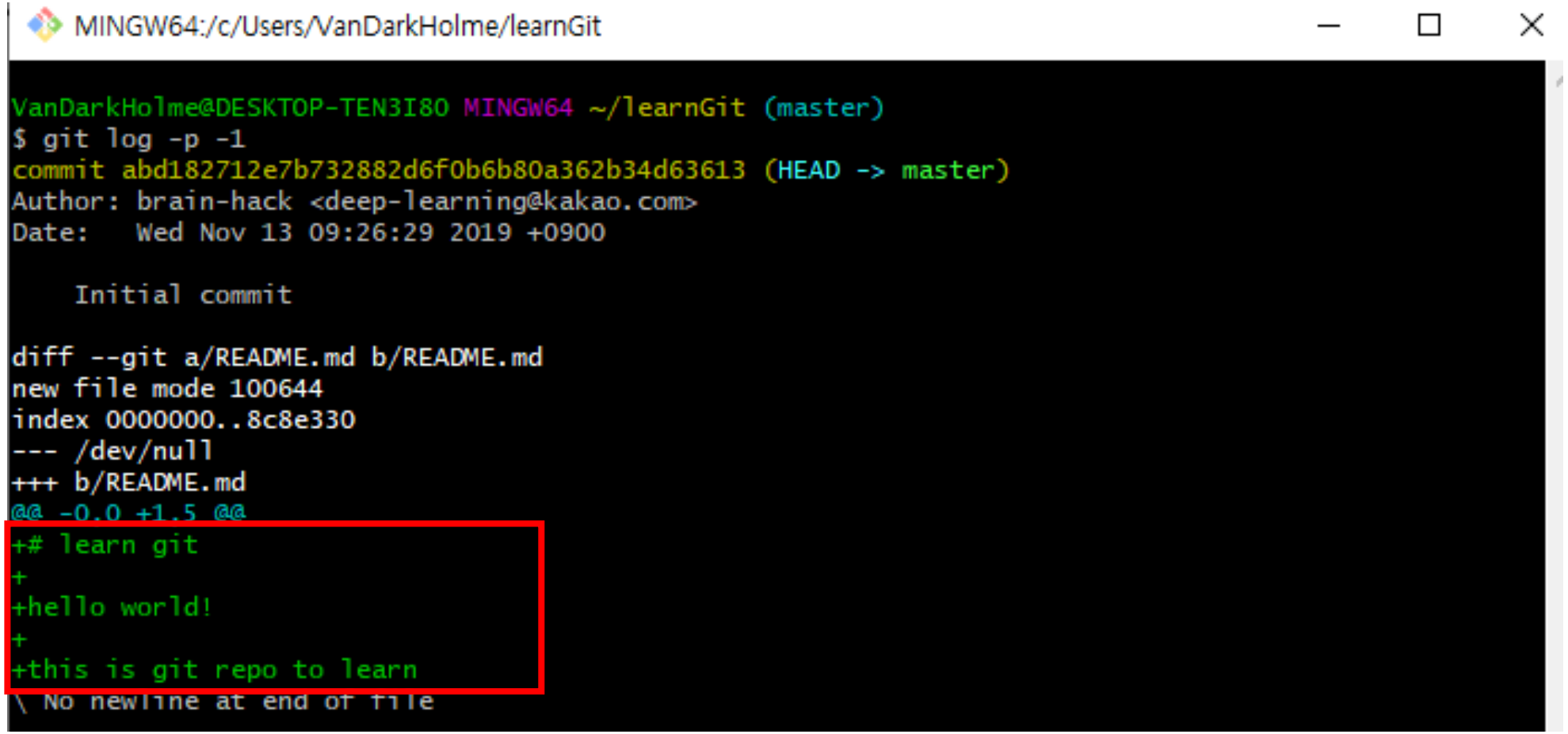
현재 git 상태를 확인해보죠



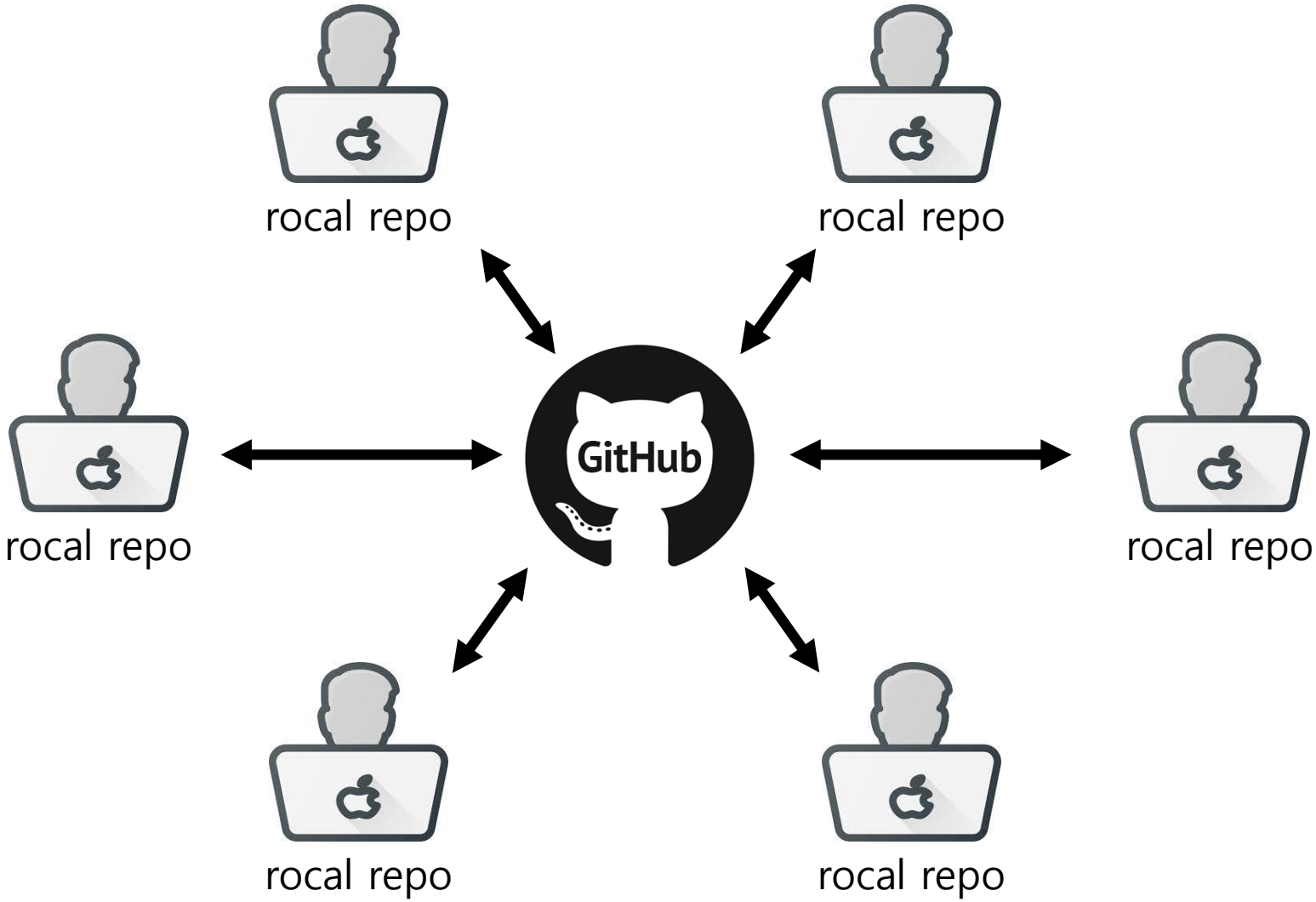
commit을 수정해봅시다



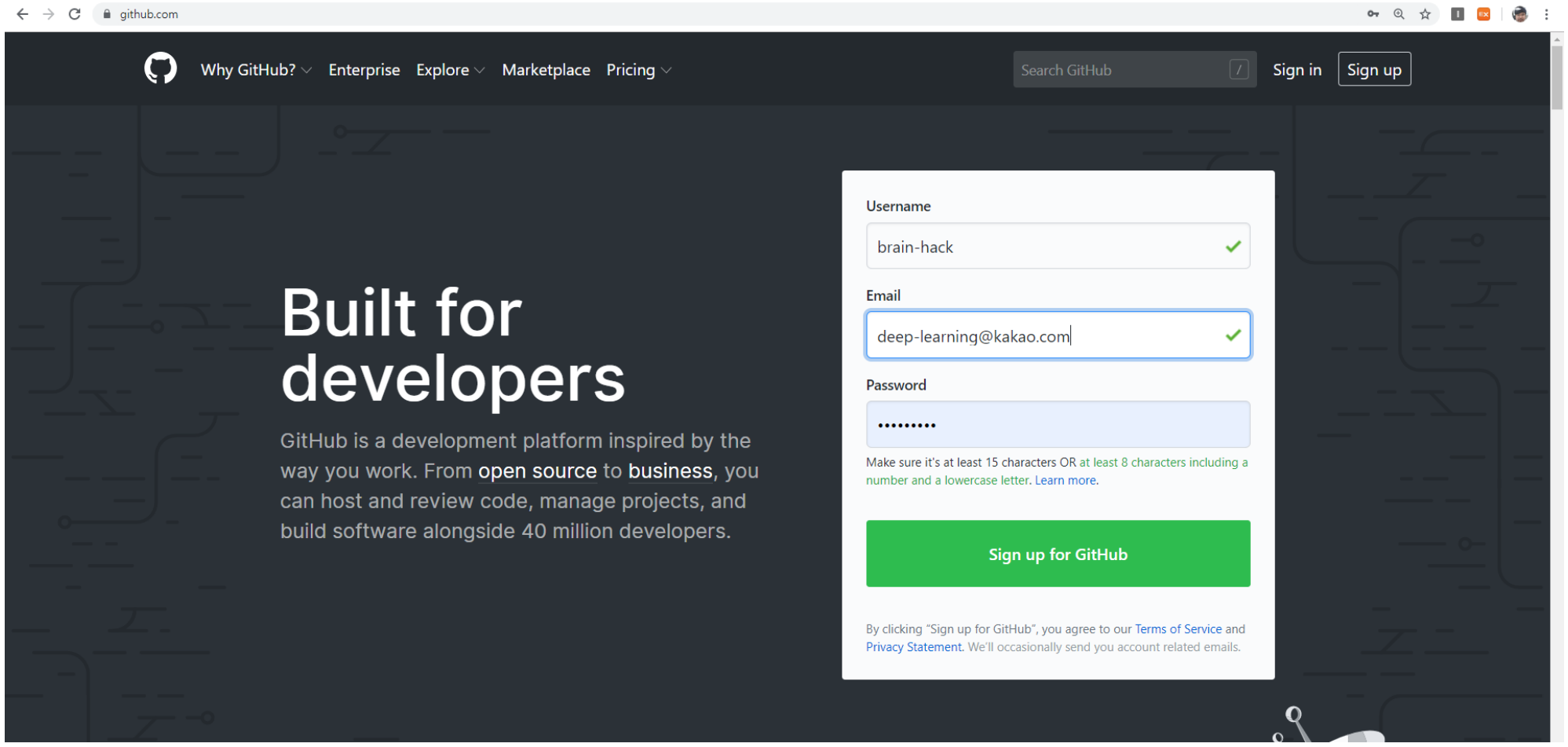
수정된 commit 내용을 볼까요



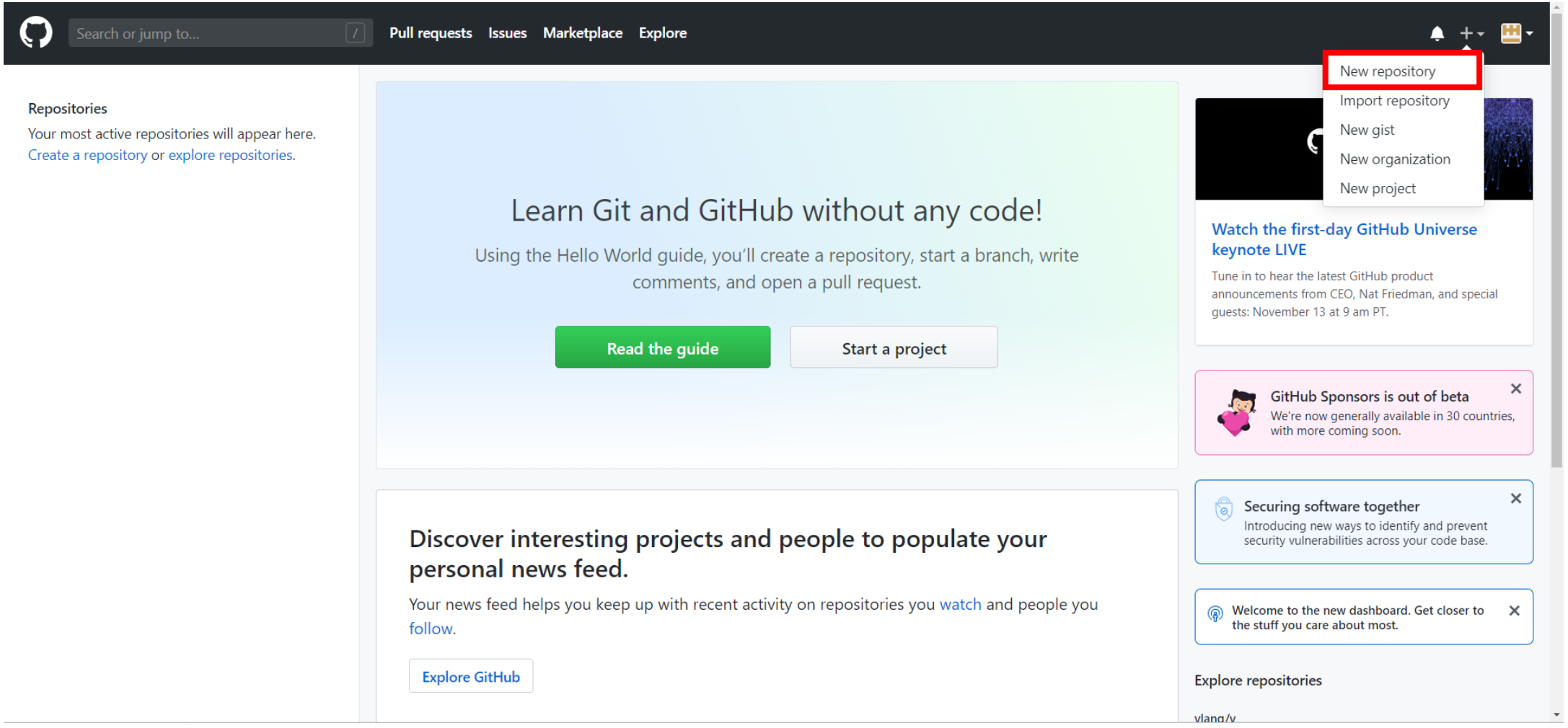
remote repository : github



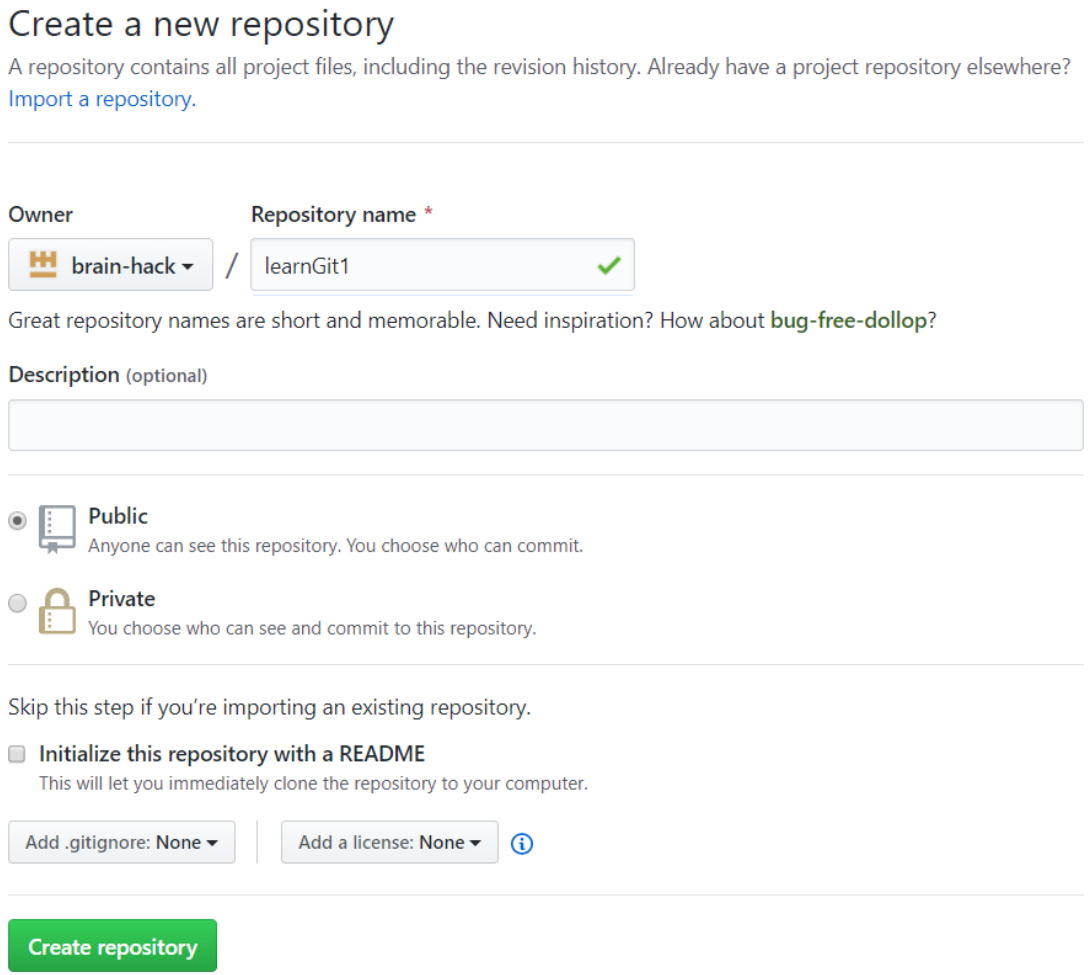
github 가입


github.com
Why GitHub?
Enterprise
Explore
Marketplace
Pricing
Search GitHub
Sign in
Sign up
Built for developers
GitHub is a development platform inspired by the way you work. From open source to business, you can host and review code, manage projects, and build software alongside 40 million developers.
Username
brain-hack
Email
deep-learning@kakao.com
Password
Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. Learn more.
Sign up for GitHub
By clicking "Sign up for GitHub", you agree to our Terms of Service and Privacy Statement. We'll occasionally send you account related emails.

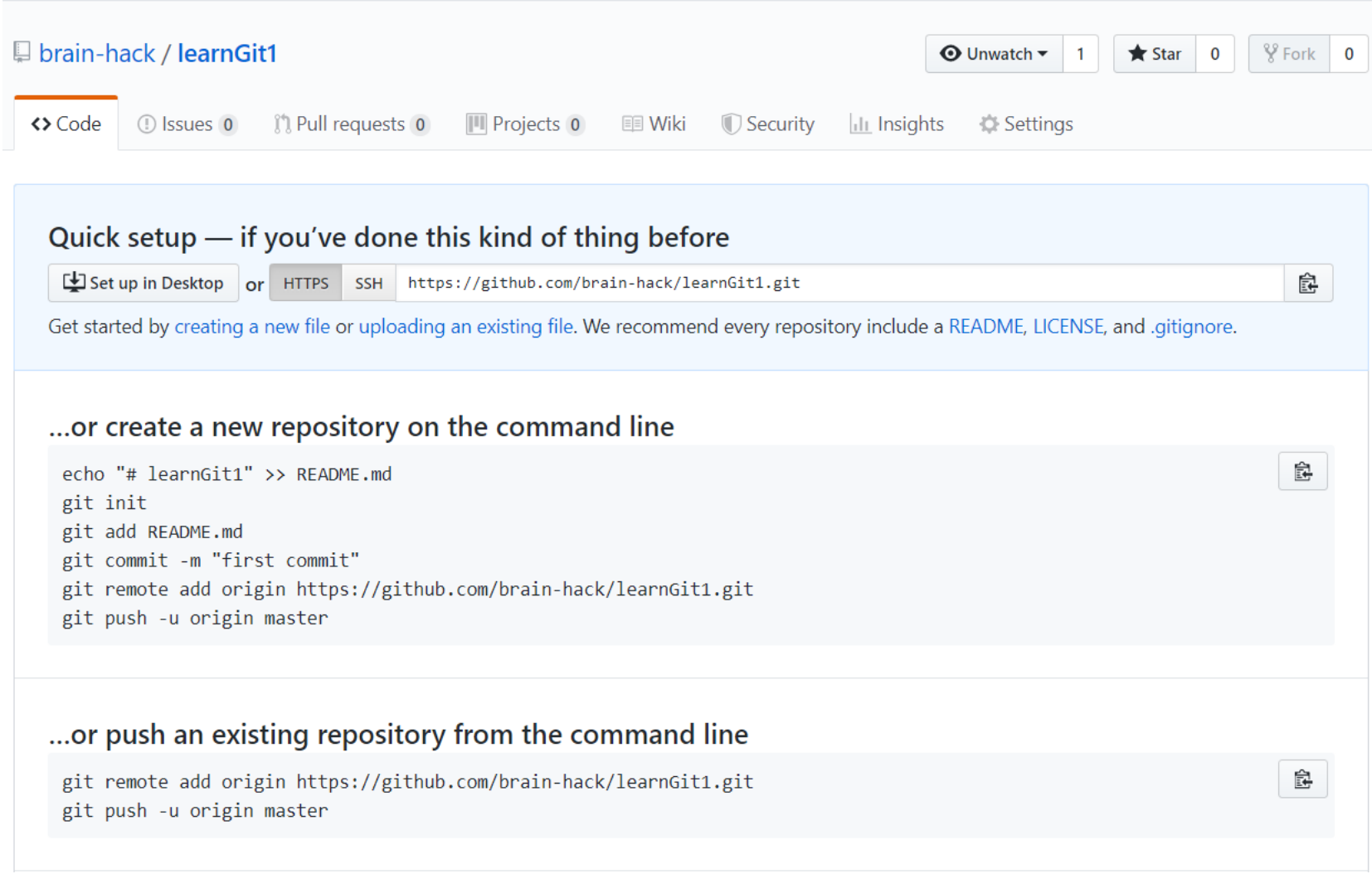
새 github repo 만들기



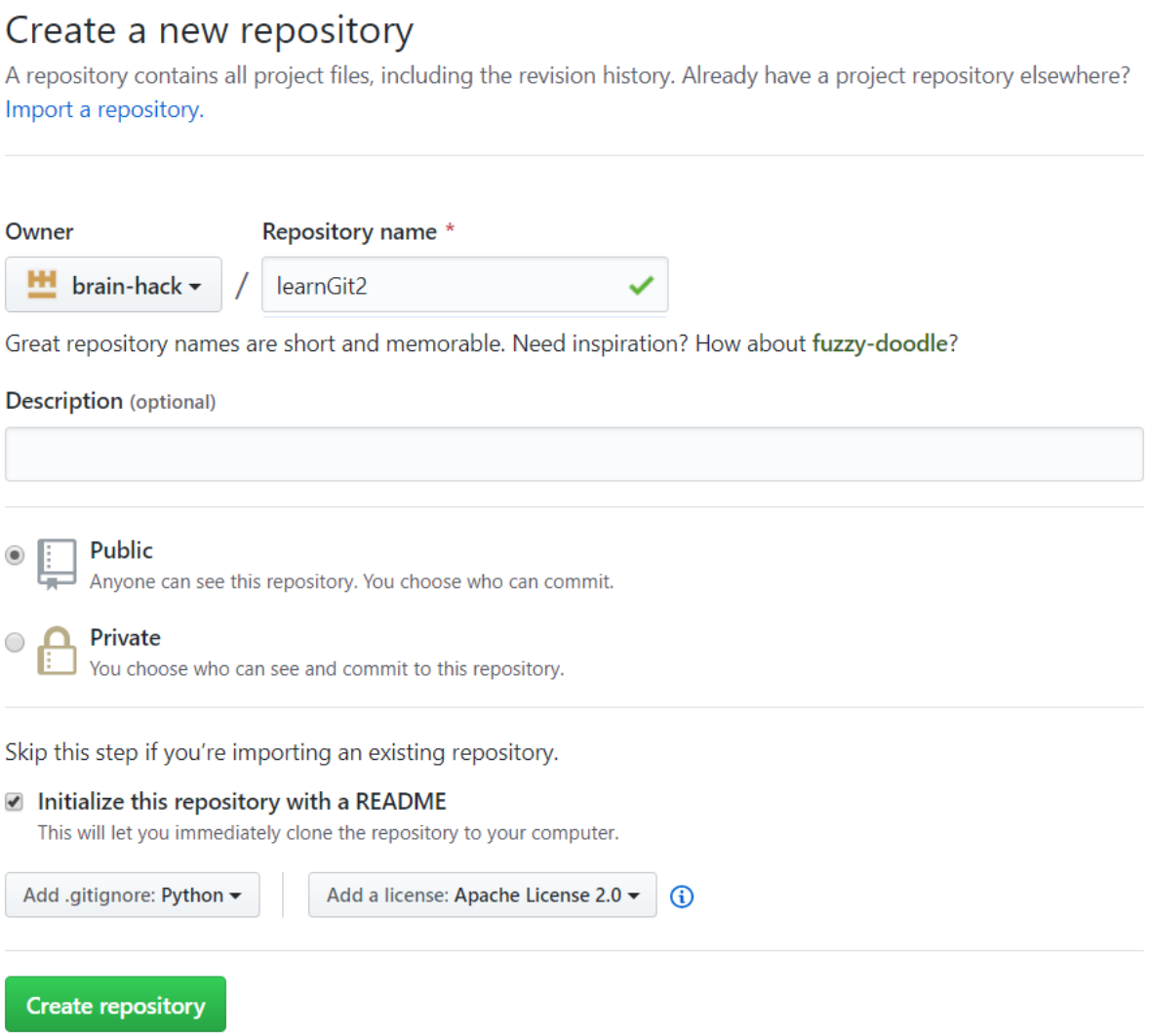
새 github repo 만들기



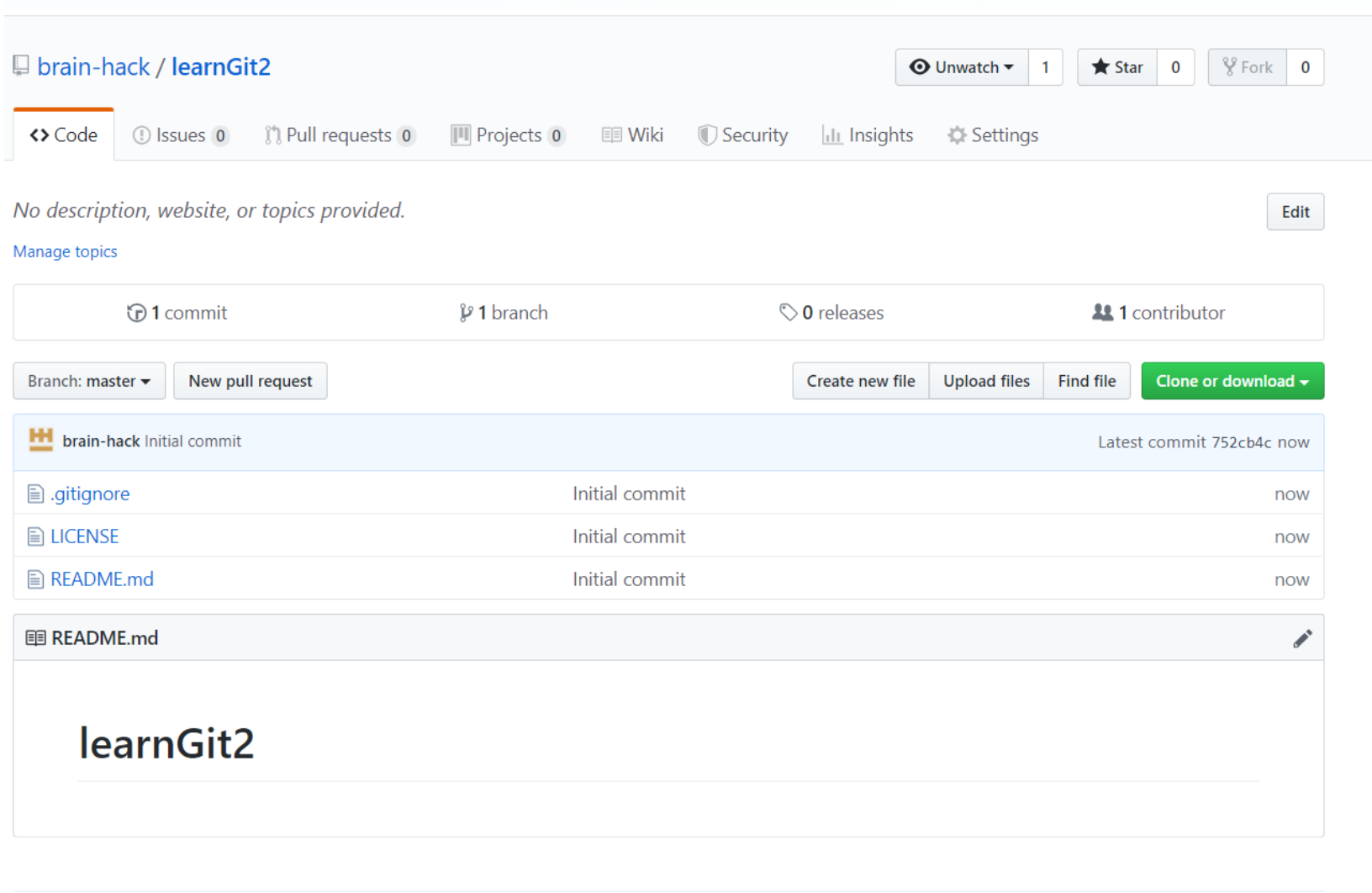
새 github repo 만들기



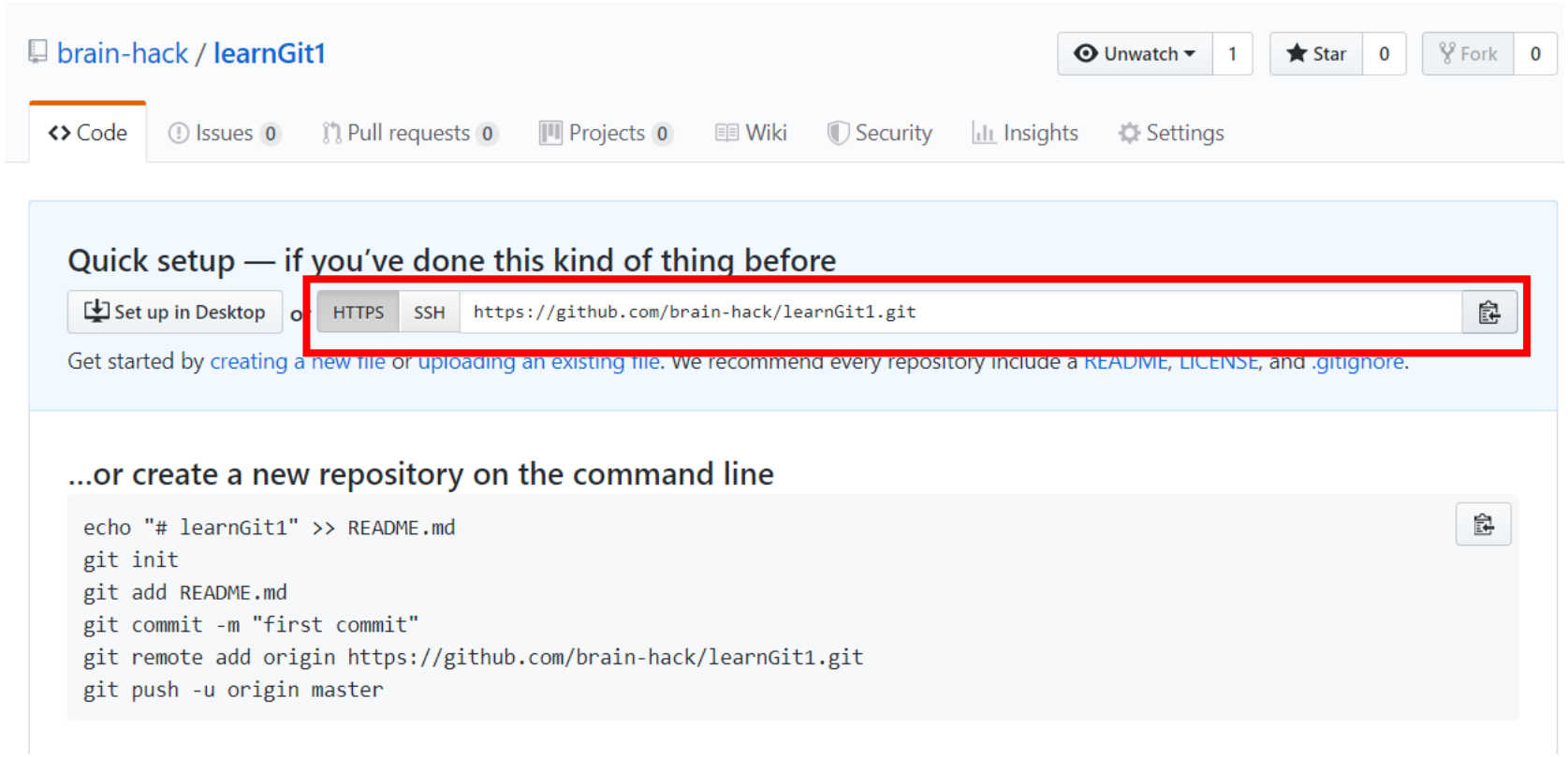
새 github repo 만들기(2)



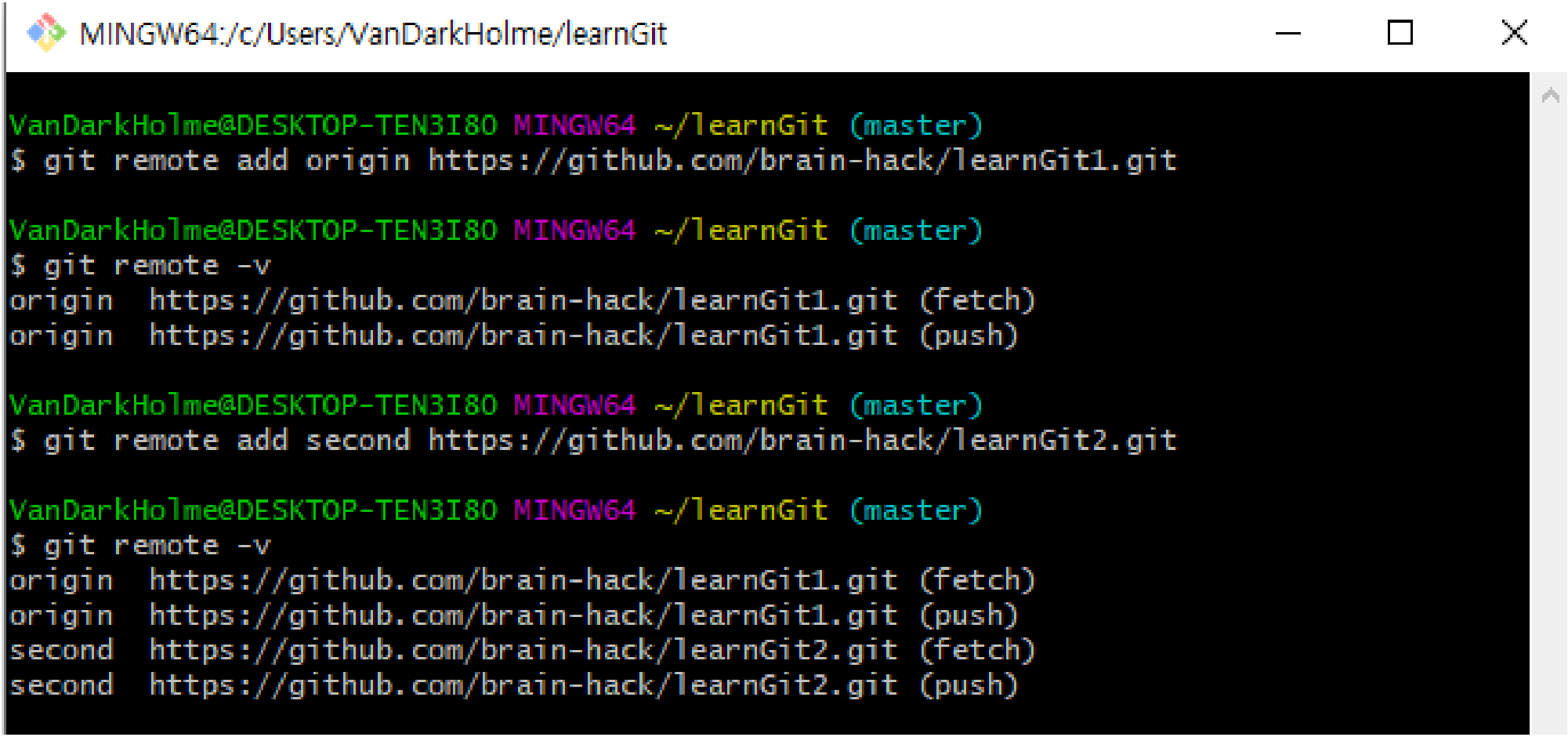
새 github repo 만들기(2)



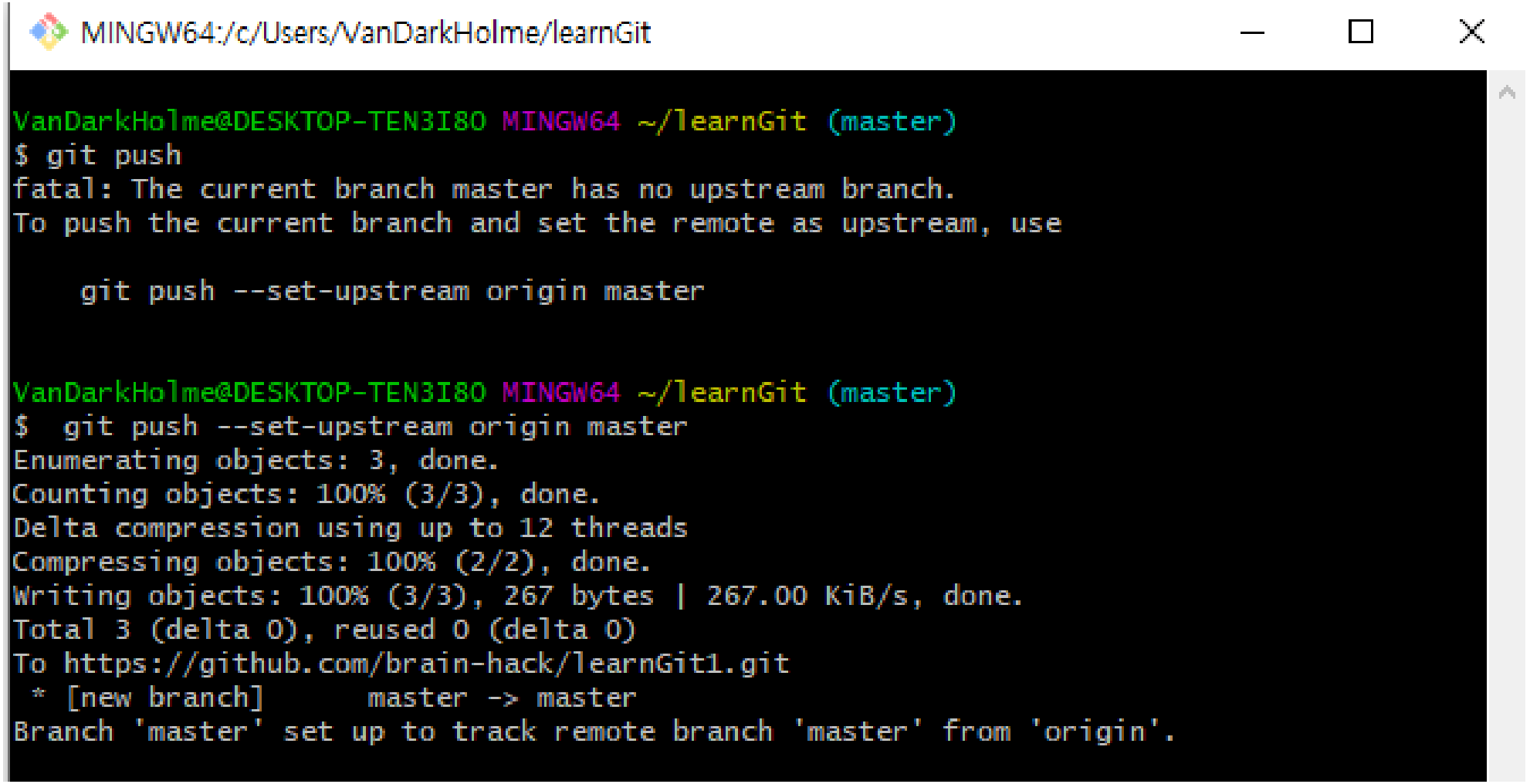
local과 github repo 연결



local과 github repo 연결

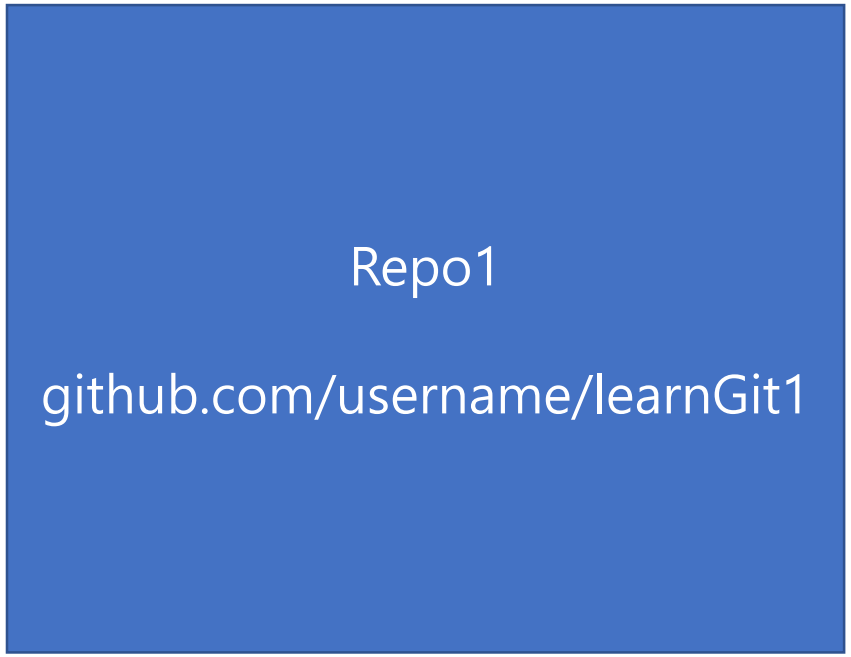


github에 local commits 업로드

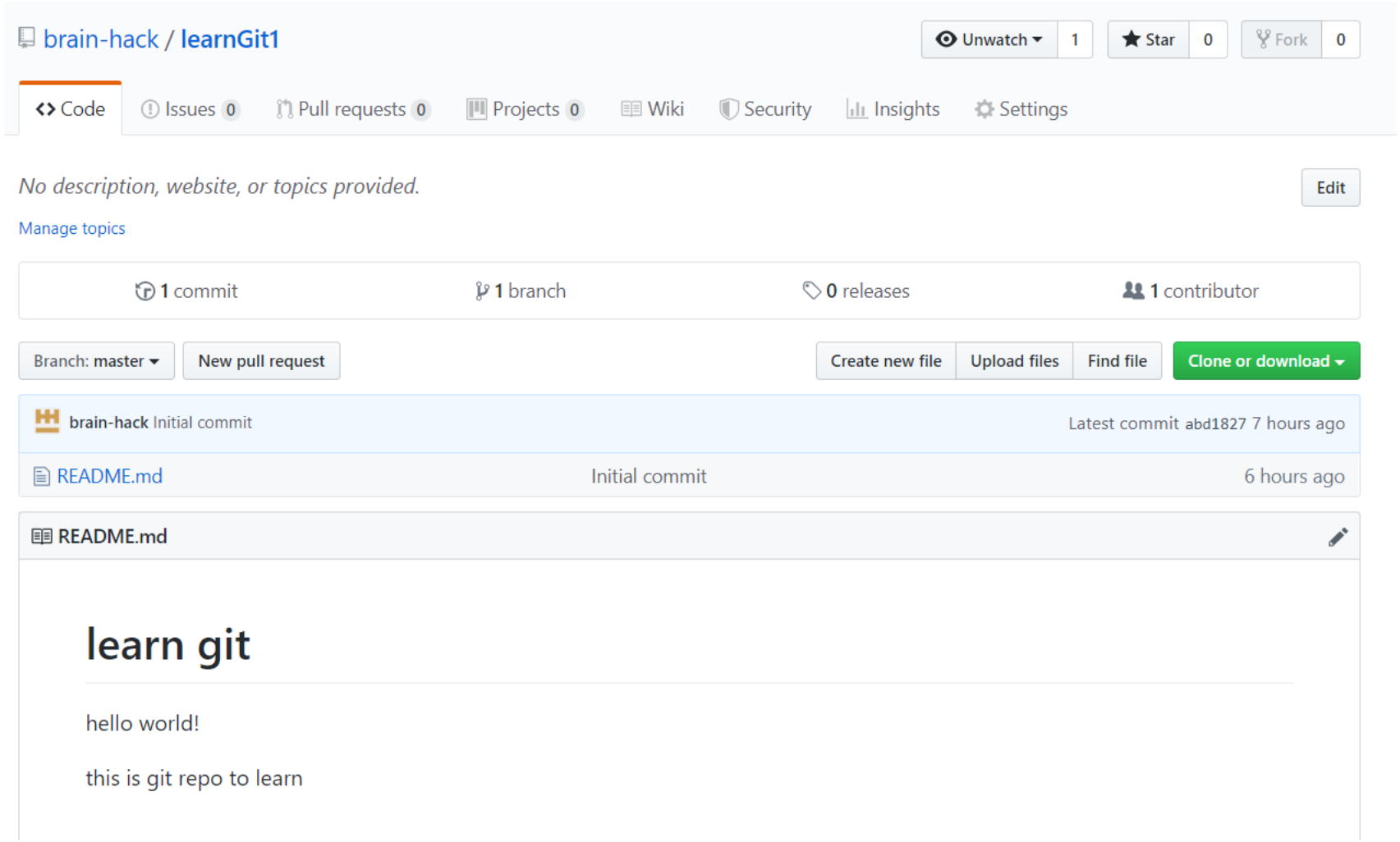


여기서 문제

- 과연 push한 local commit은 어디로 업로드가 되었을까요?

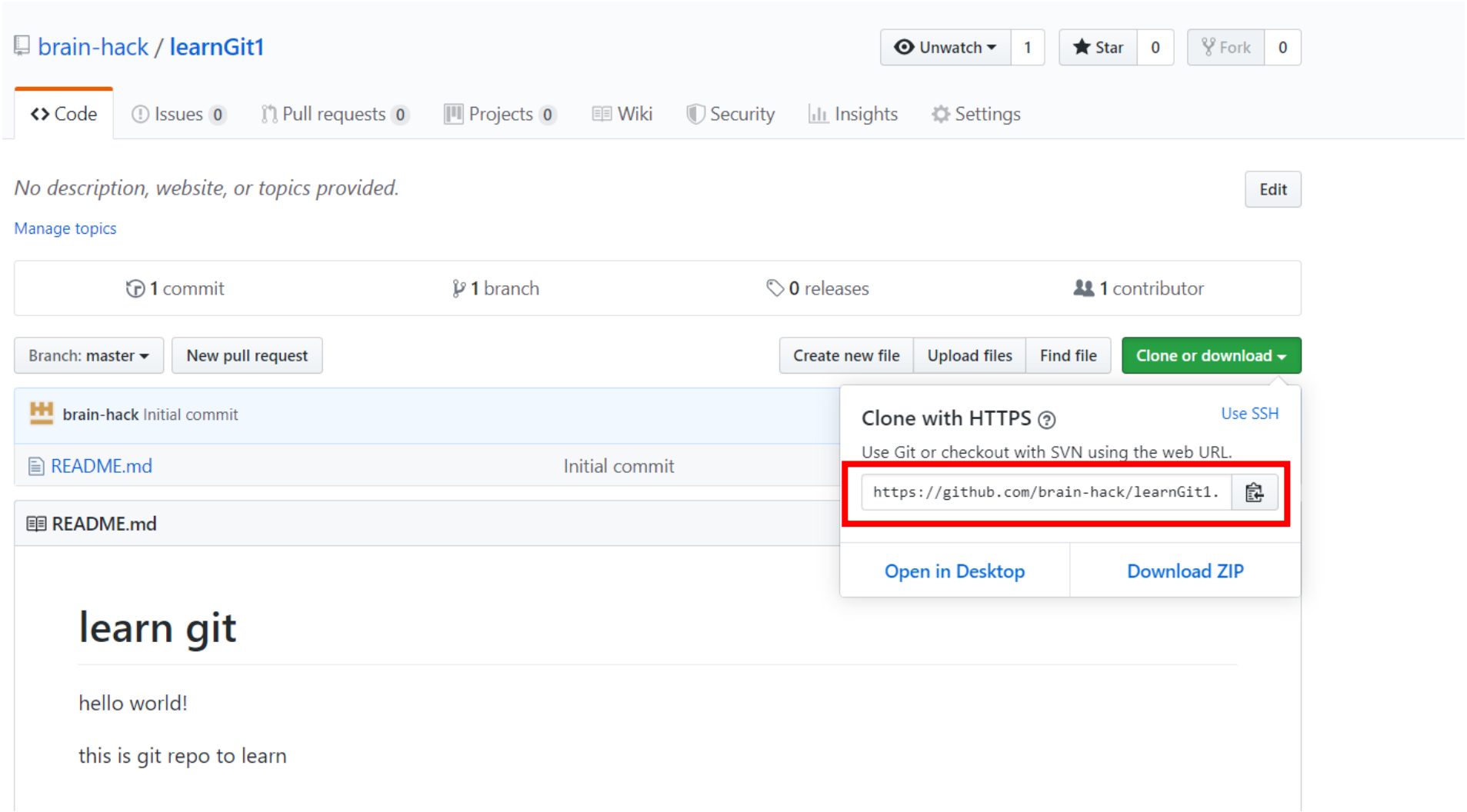


github repo 확인



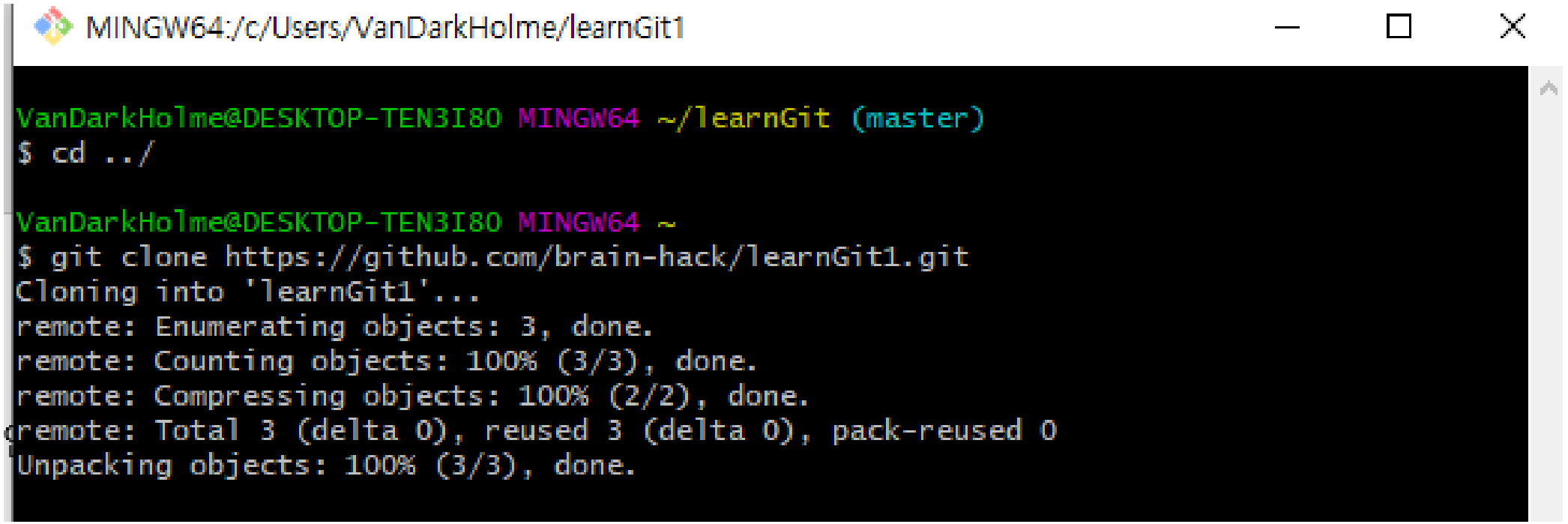
local에 새 저장소를 만들어보겠습니다

근데 이제 clone을 곁들인

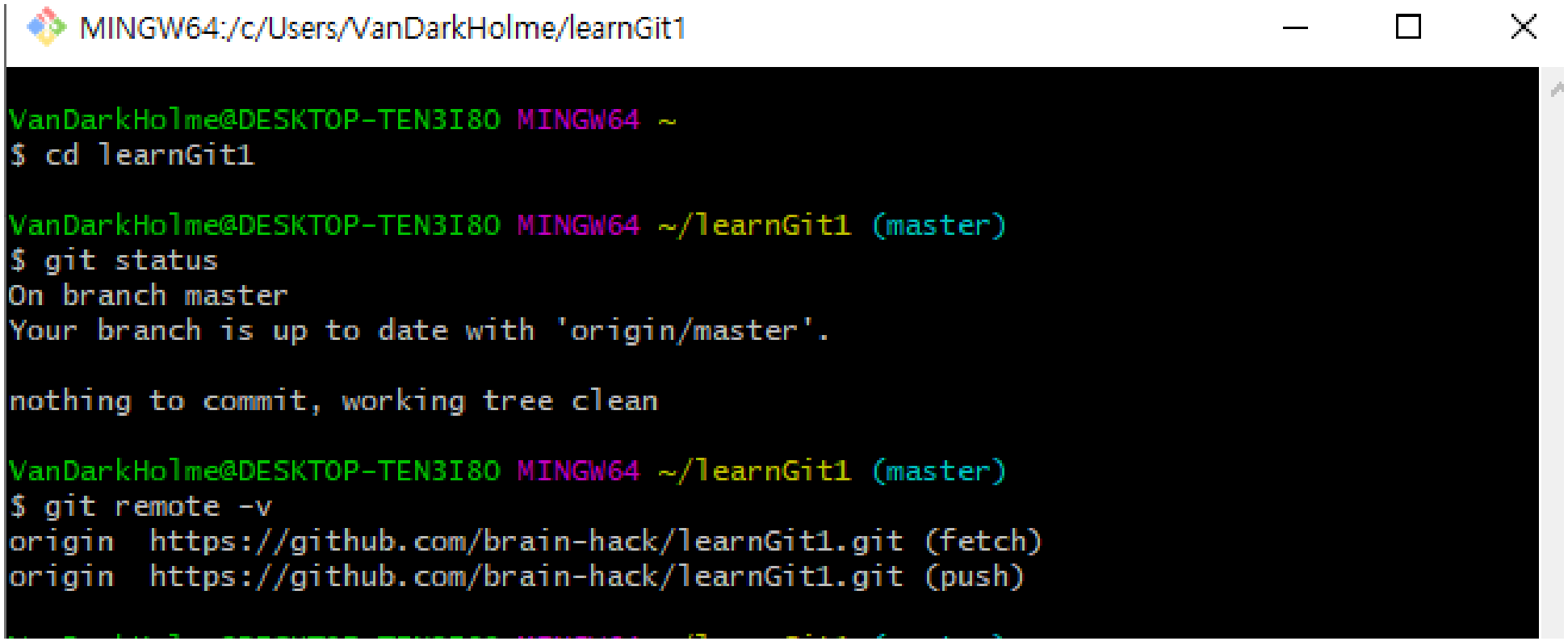


local에 새 저장소를 만들어보겠습니다

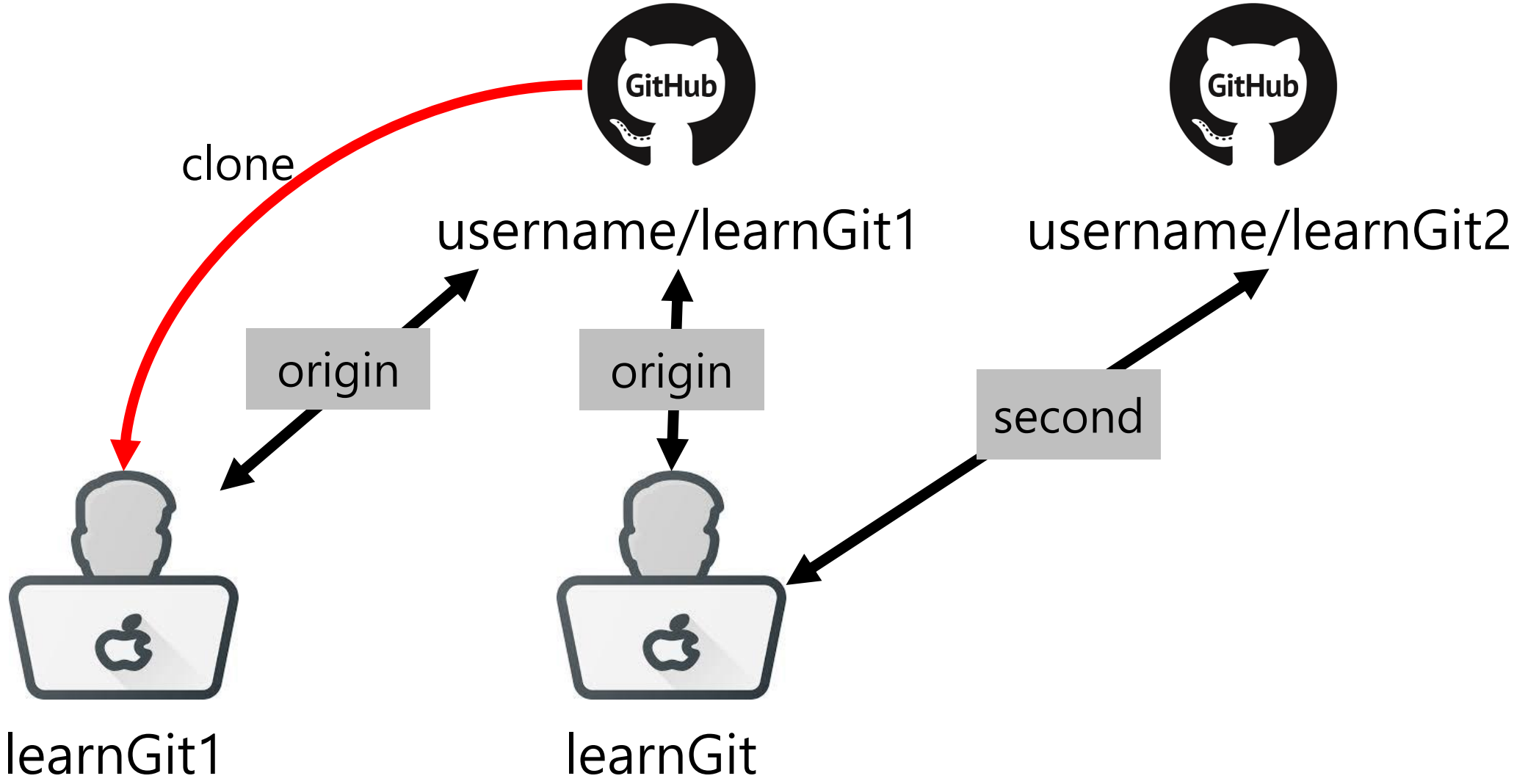
근데 이제 clone을 곁들인



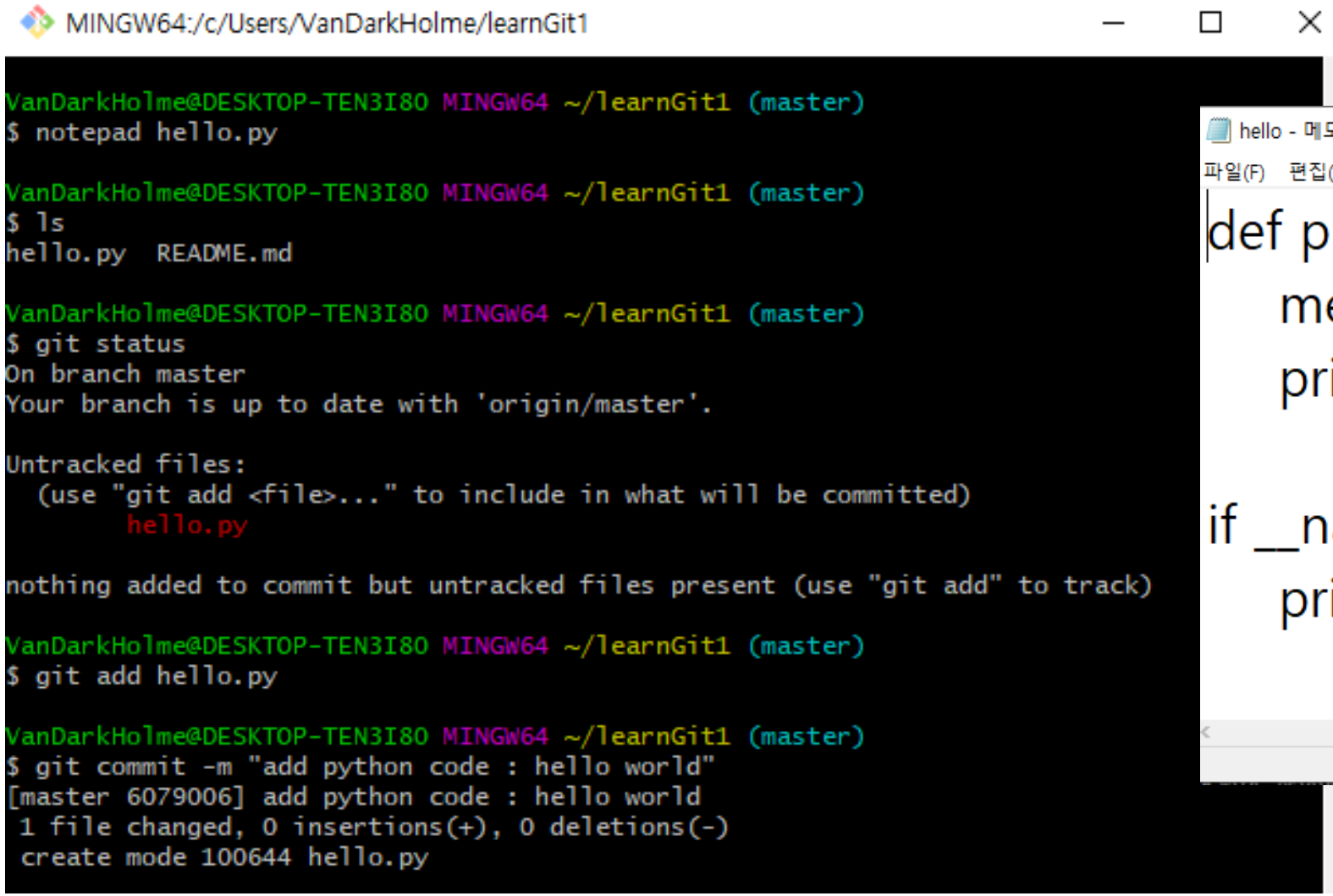
clone된 repo확인

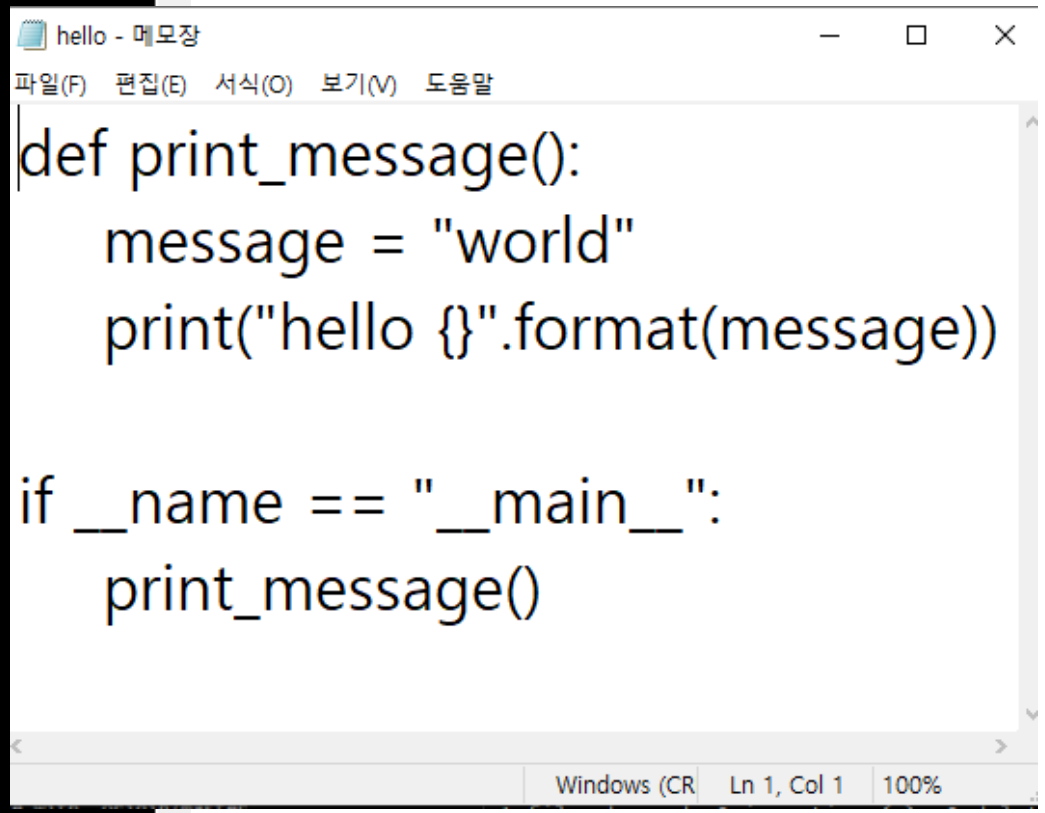


지금 현재 local과 remote의 관계

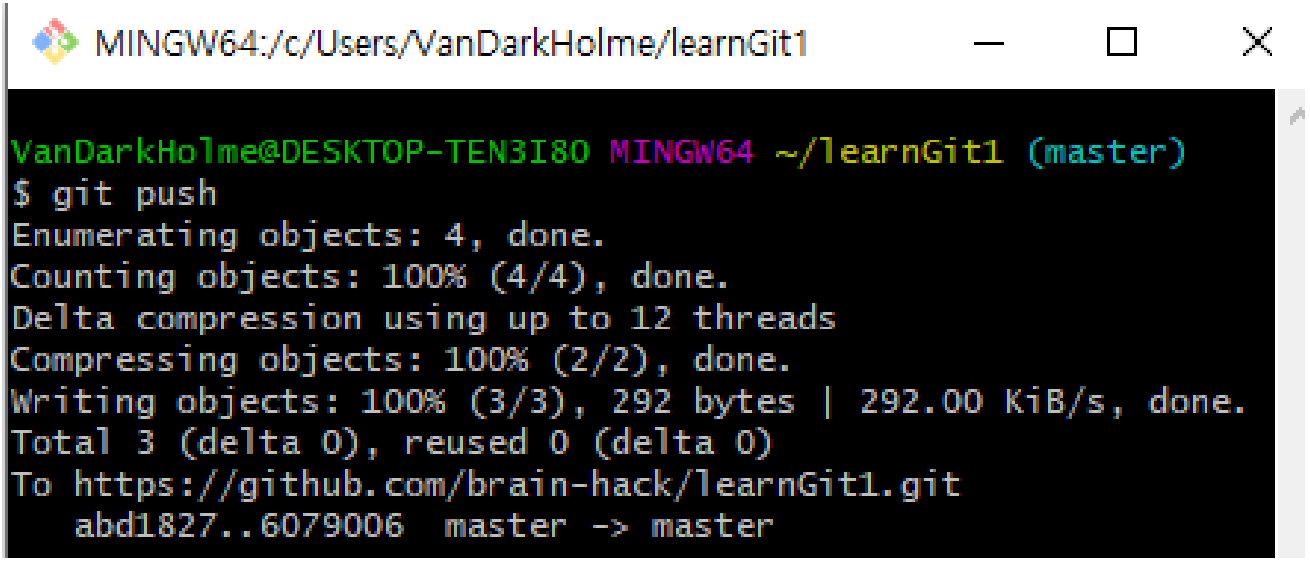


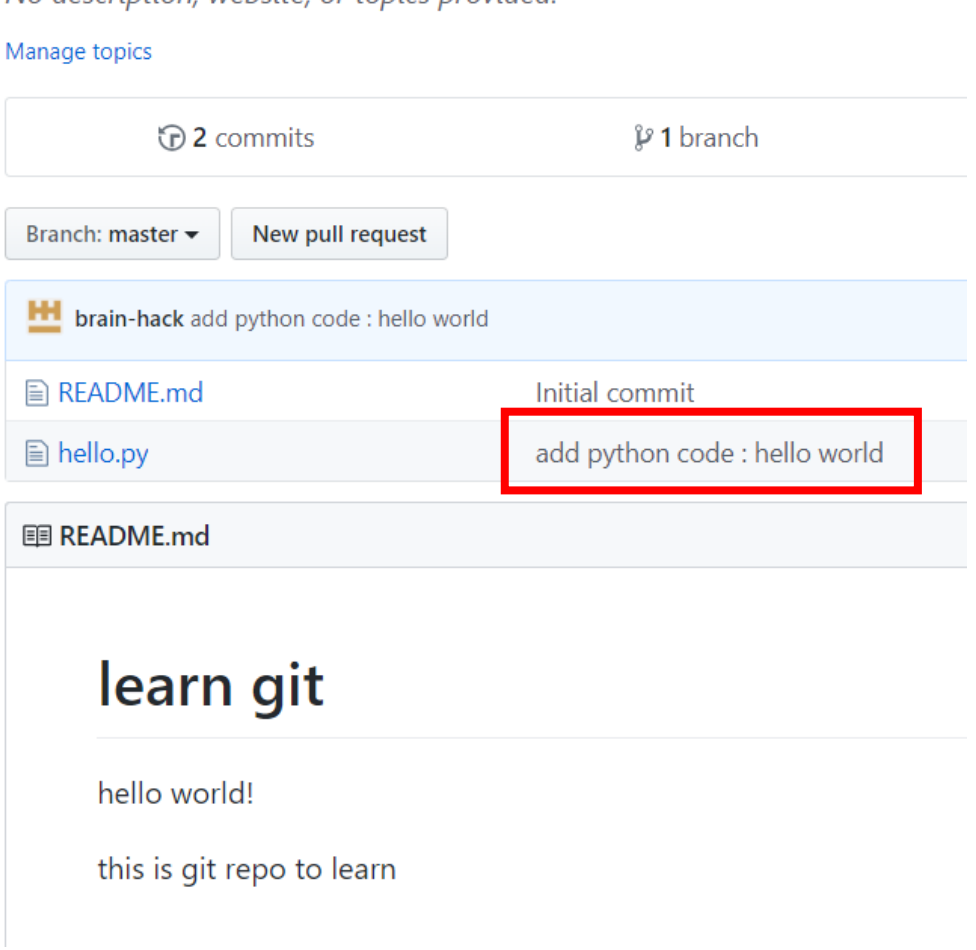
local repo : learnGit1에서 작업하기



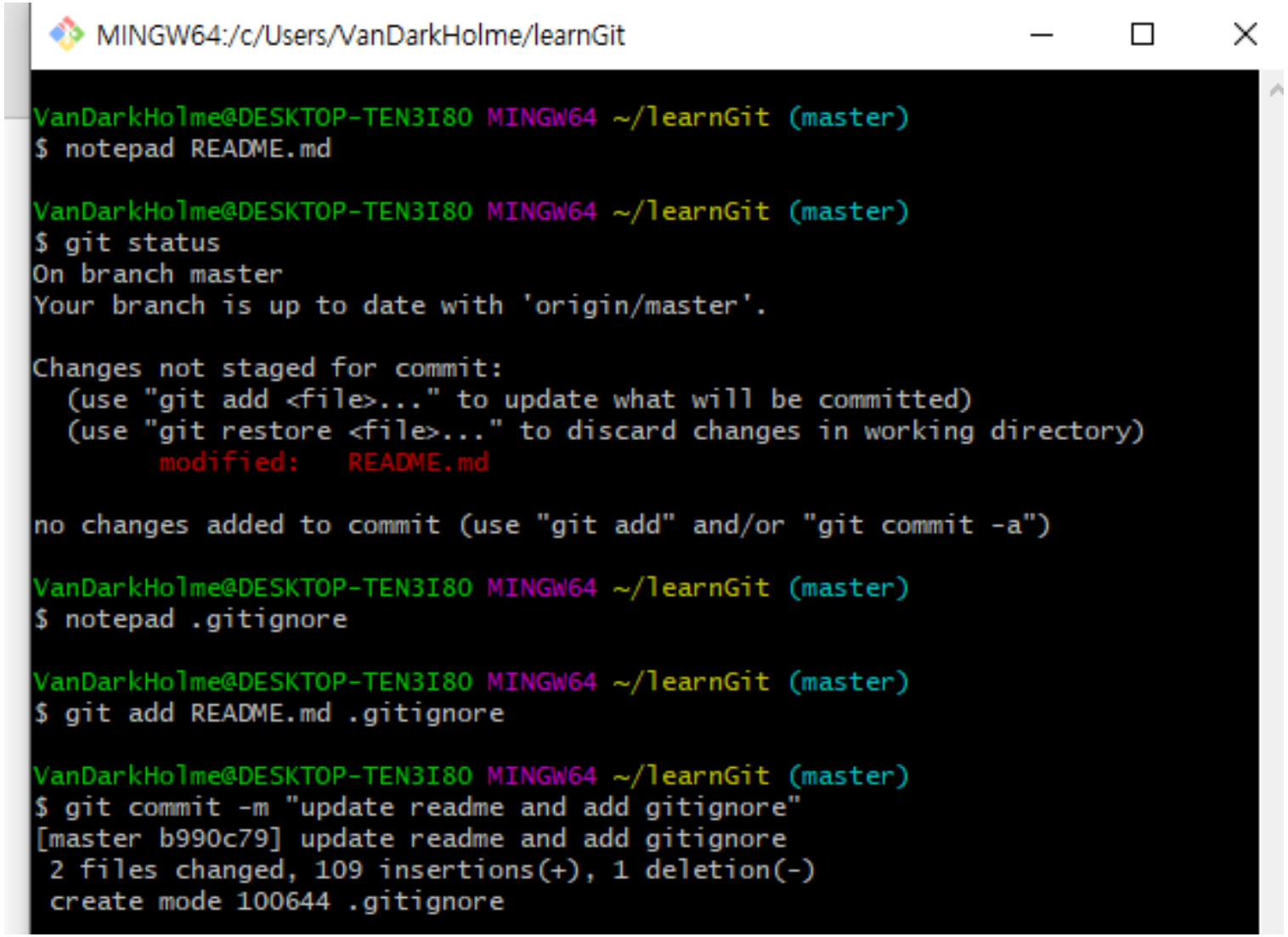


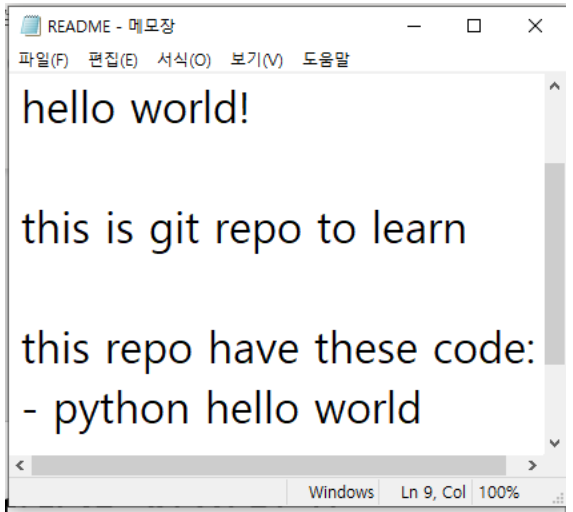
local repo : learnGit1에서 push

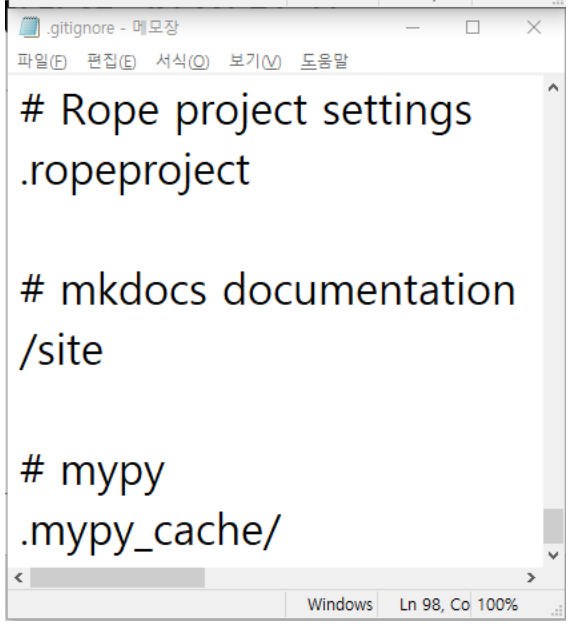




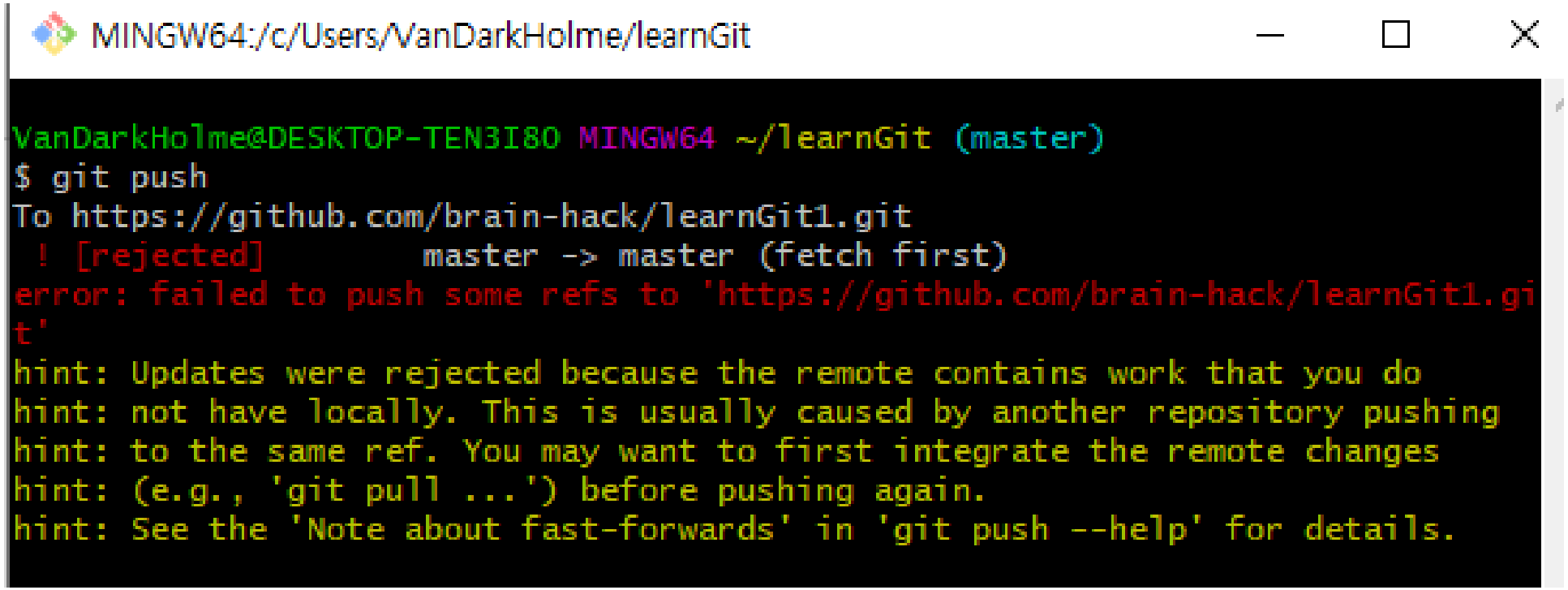
local repo : learnGit에서 작업하기



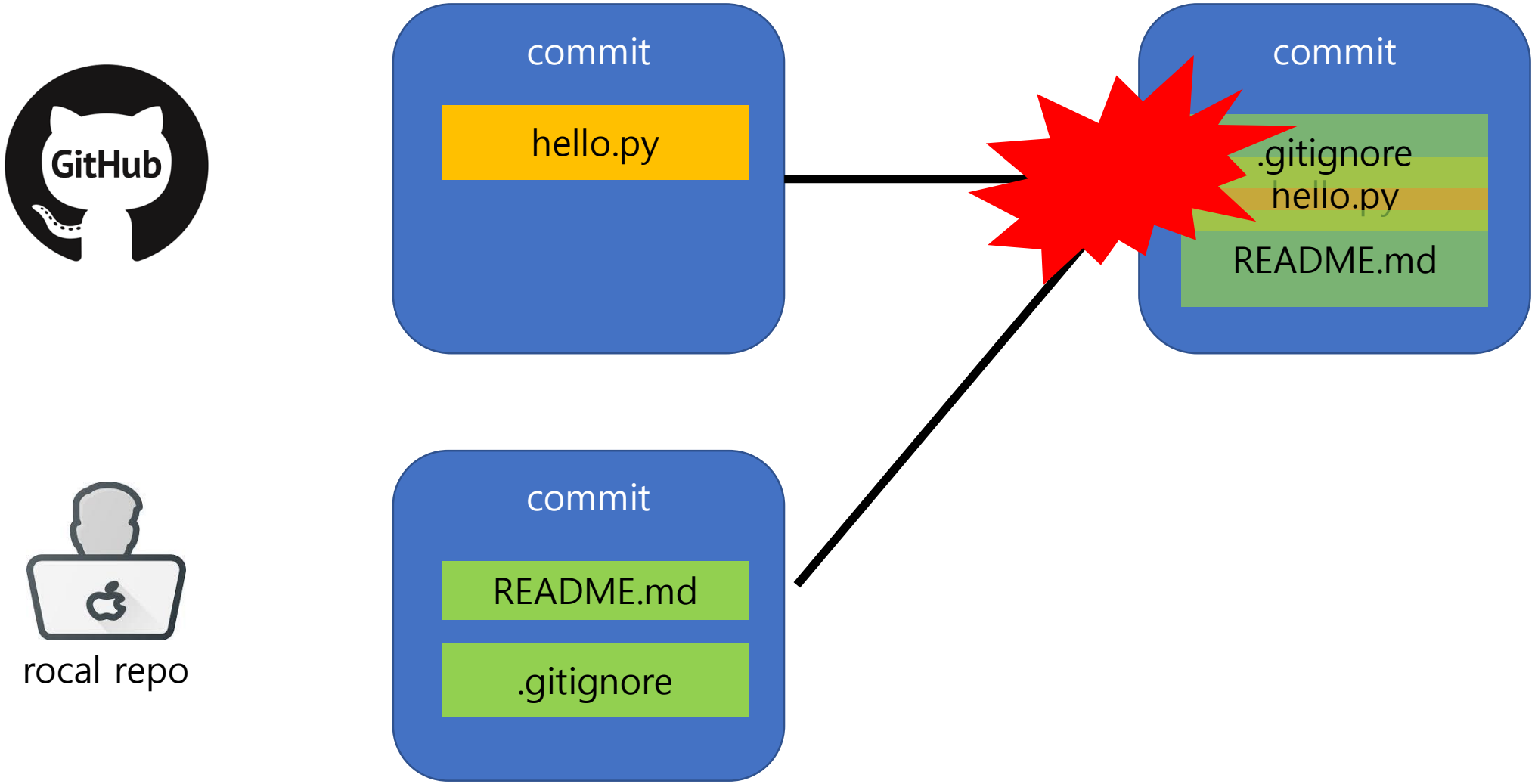




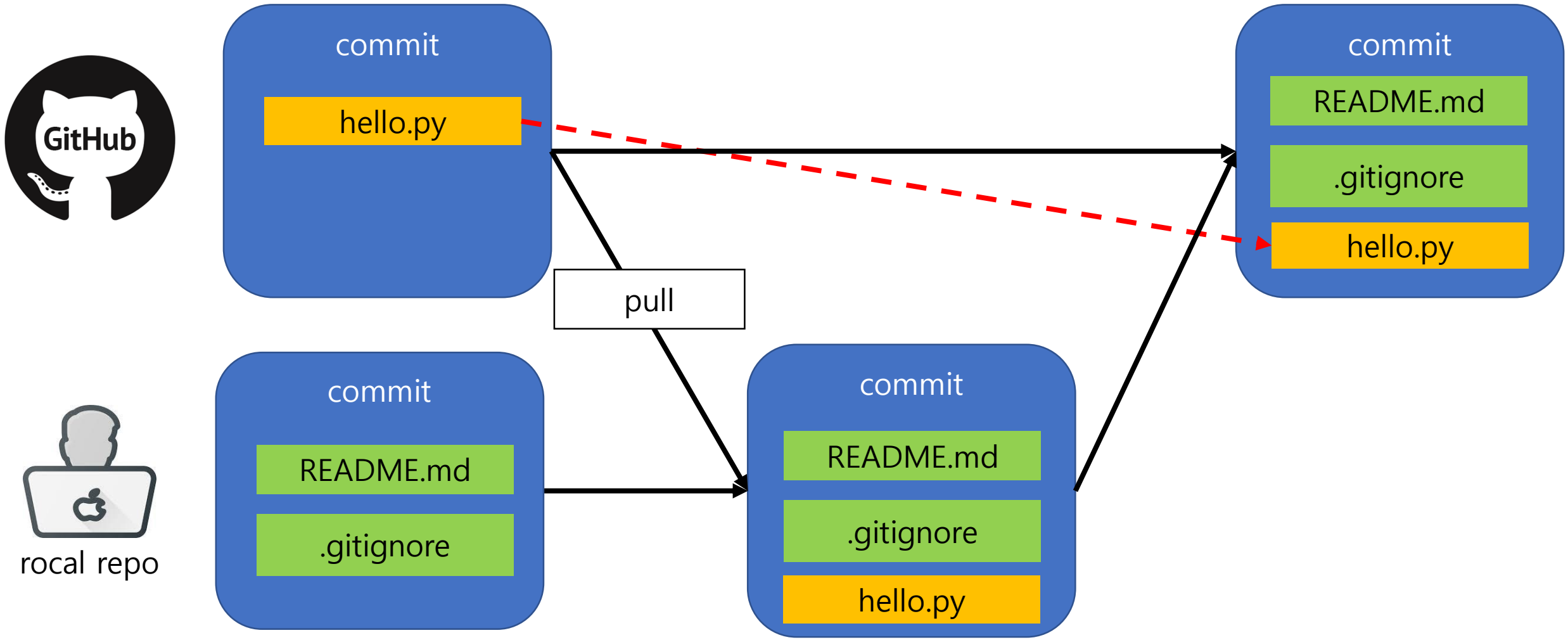
push 문제 발생


MINGW64:/c/Users/VanDarkHolme/learnGit
VanDarkHolme@DESKTOP-TEN3I8O MINGW64 ~/learnGit (master)
$ git push
To https://github.com/brain-hack/learnGit1.git
 ! [rejected]   master -> master (fetch first)
error: failed to push some refs to 'https://github.com/brain-hack/learnGit1.gi
t'
hint: Updates were rejected because the remote contains work that you do
hint: not have locally. This is usually caused by another repository pushing
hint: to the same ref. You may want to first integrate the remote changes
hint: (e.g., 'git pull ...') before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.

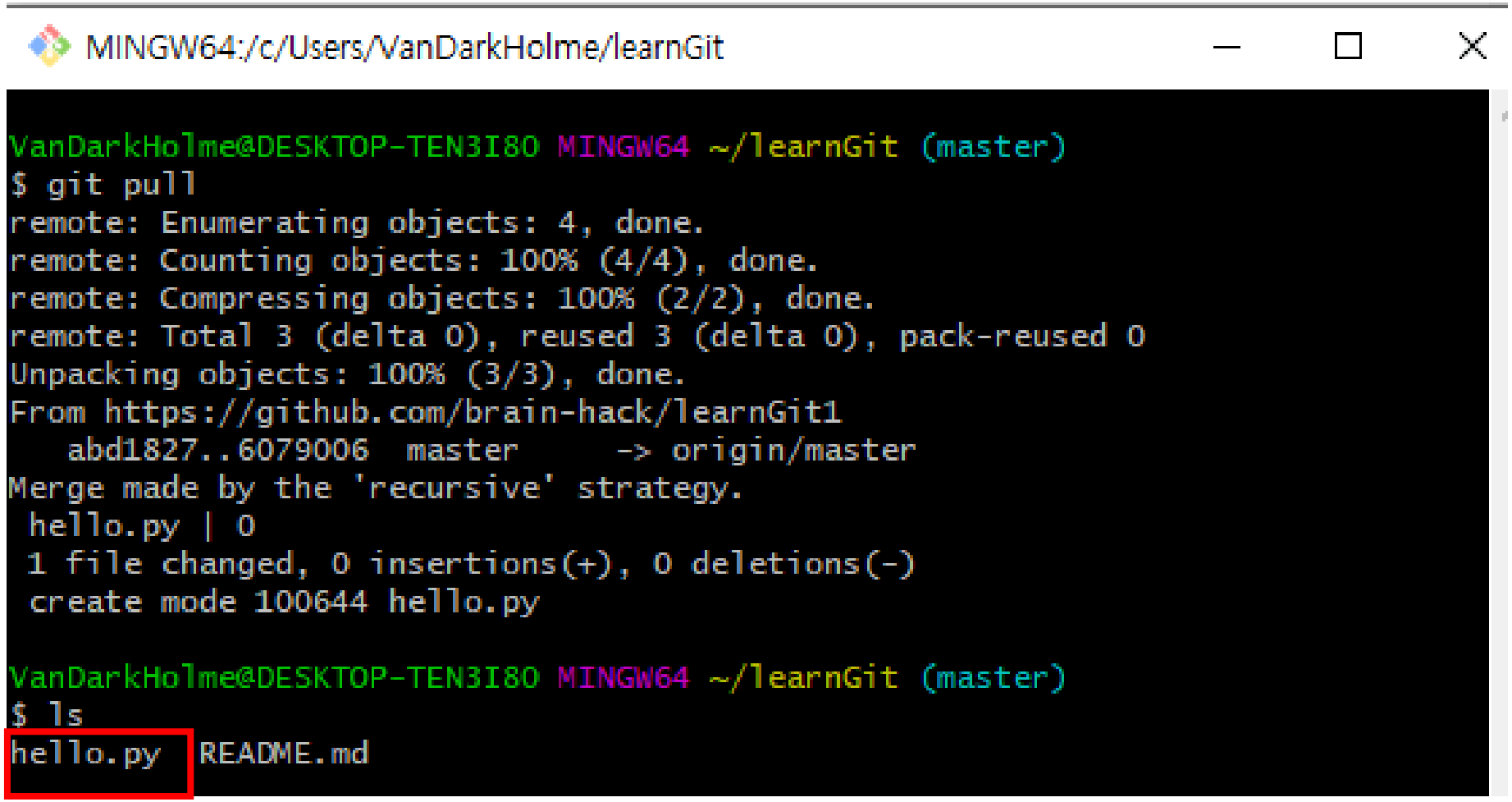
왜 문제가 발생했을까요?


GitHub
commit
hello.py
commit
.gitignore
hello.py
README.md
commit
README.md
.gitignore
rocal repo

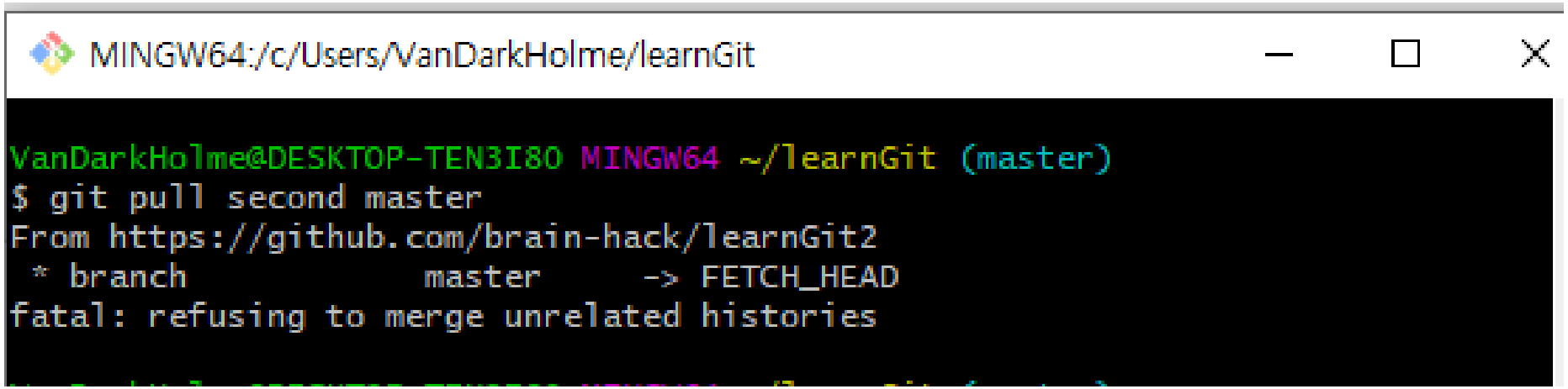
어떻게 해결할까?


GitHub
commit
hello.py
commit
README.md
.gitignore
hello.py
pull
commit
README.md
.gitignore
rocal repo
commit
README.md
.gitignore
hello.py

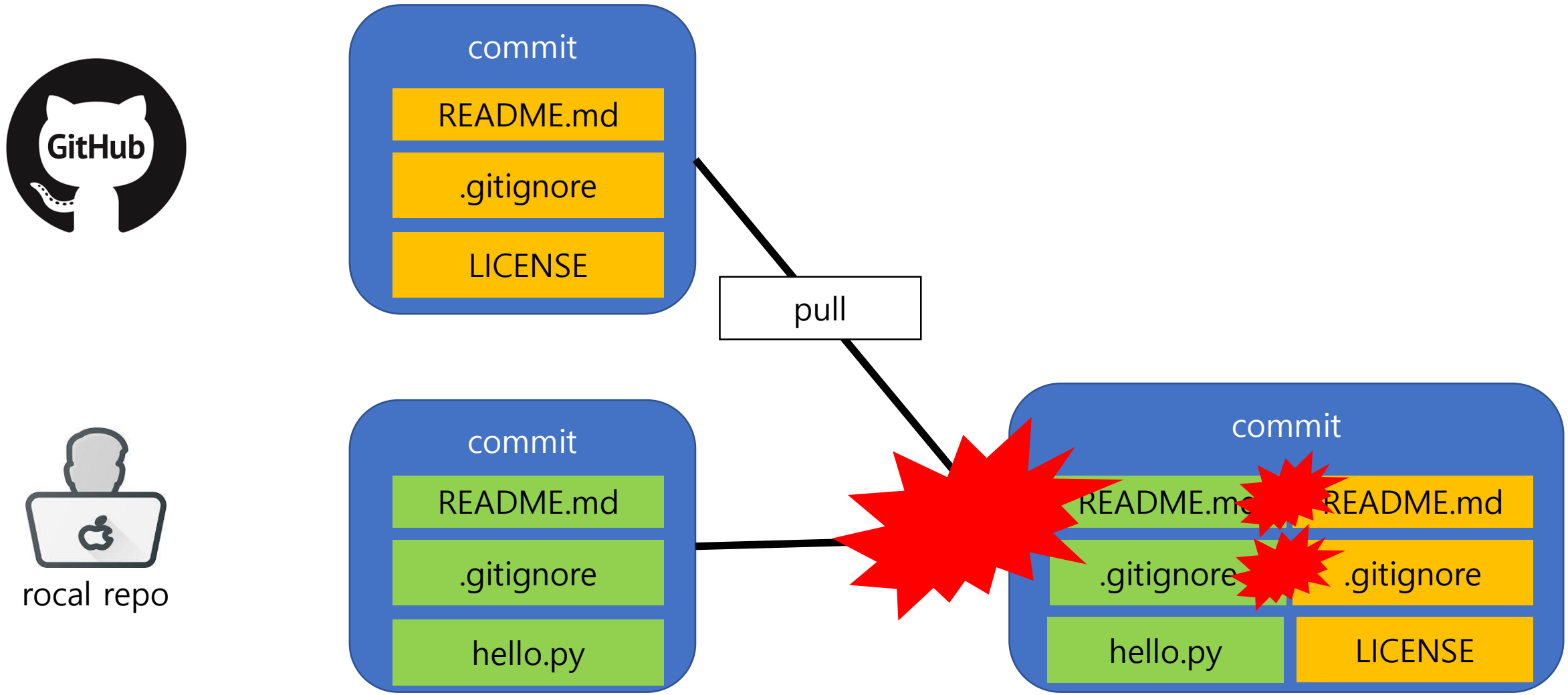
git pull을 통해 remote내용 download



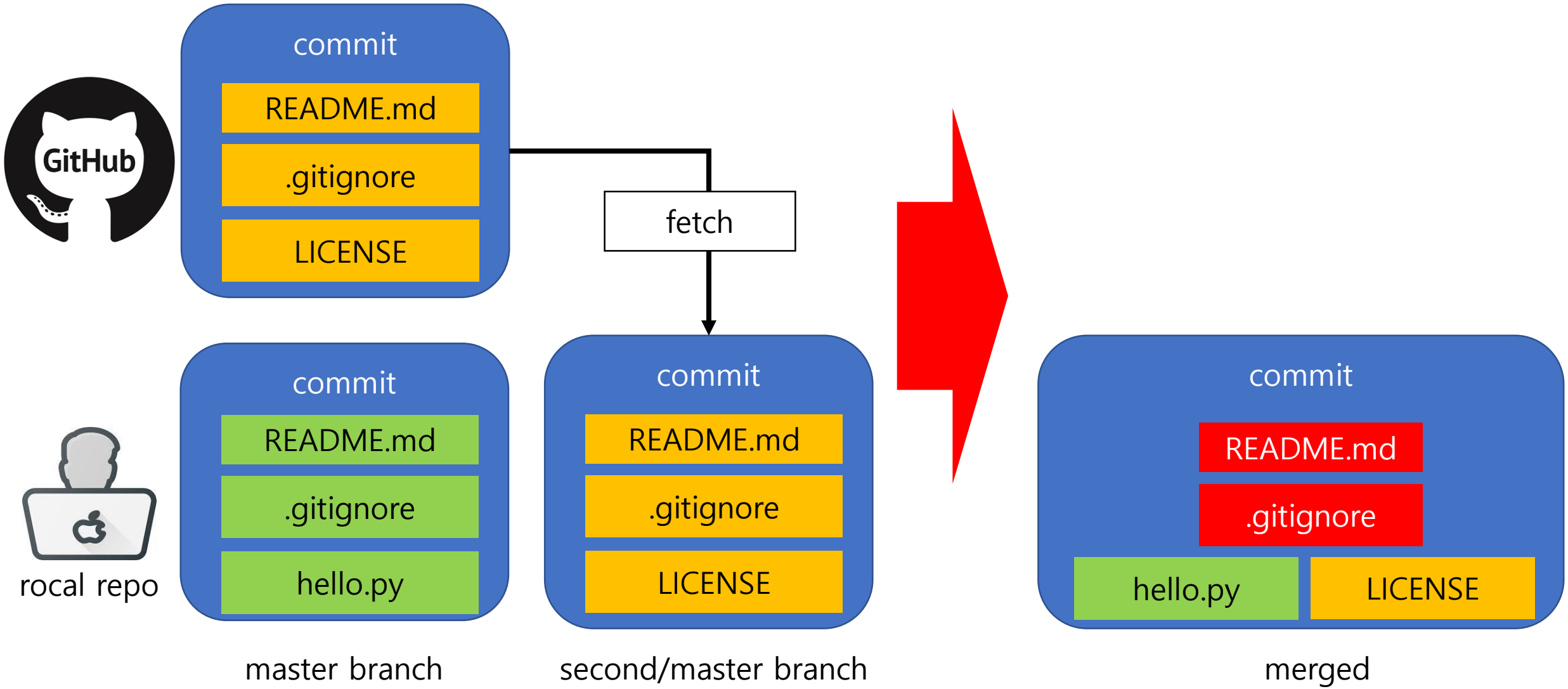
하지만 일이 순탄치 않을 때가.....



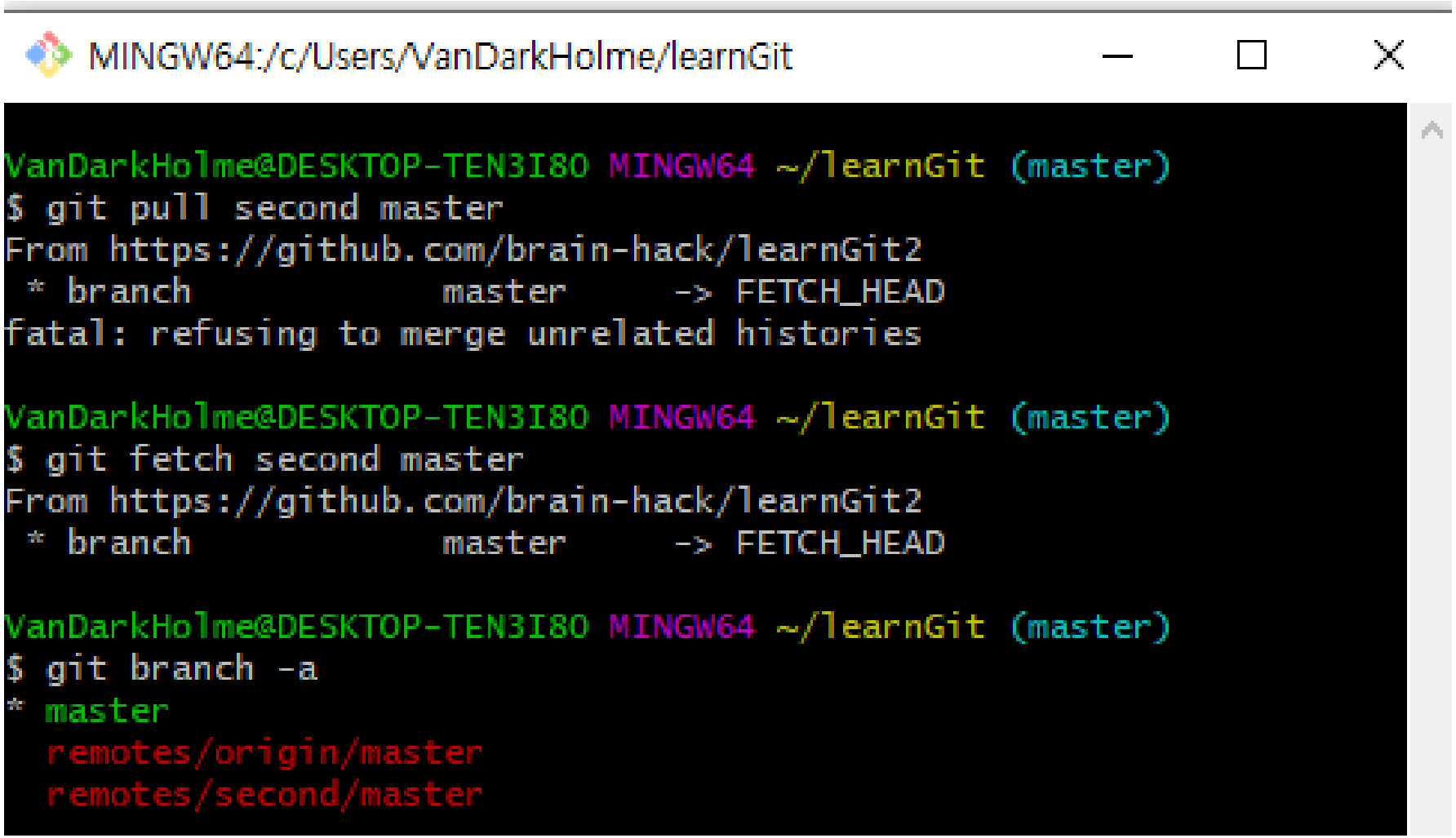
왜 문제가 발생했을까요?


GitHub
commit
README.md
.gitignore
LICENSE
pull
rocal repo
commit
README.md
.gitignore
hello.py
commit
README.md
README.md
.gitignore
.gitignore
hello.py
LICENSE

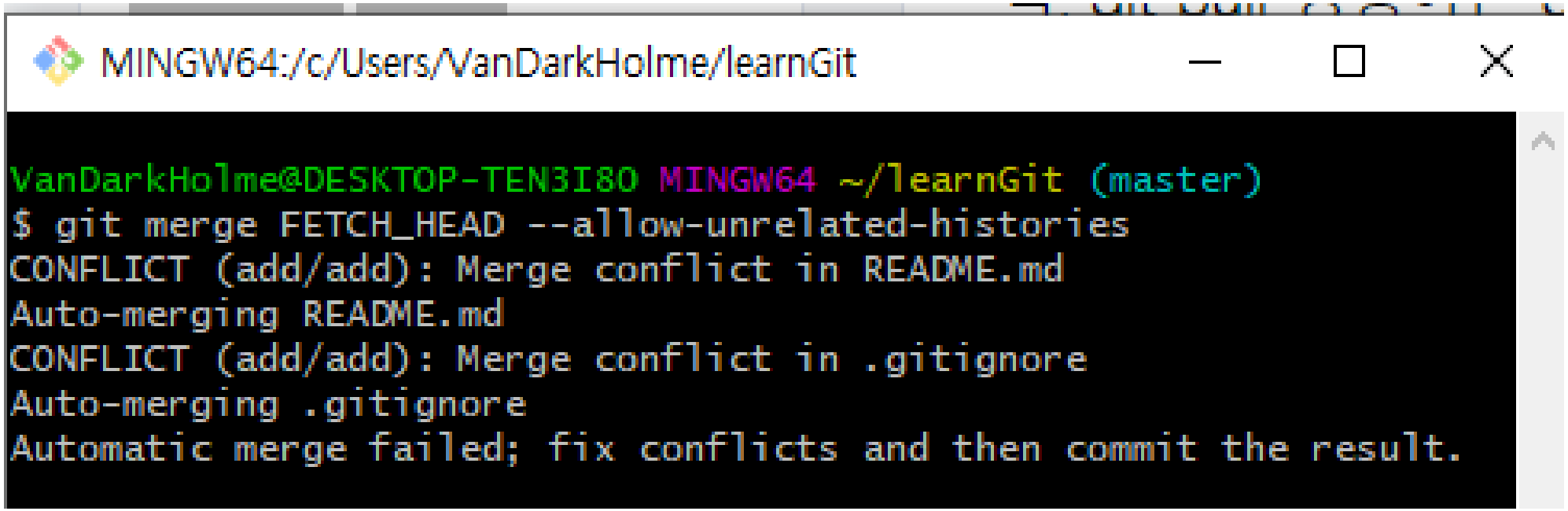
fetch후 수동으로 합친 후 merge



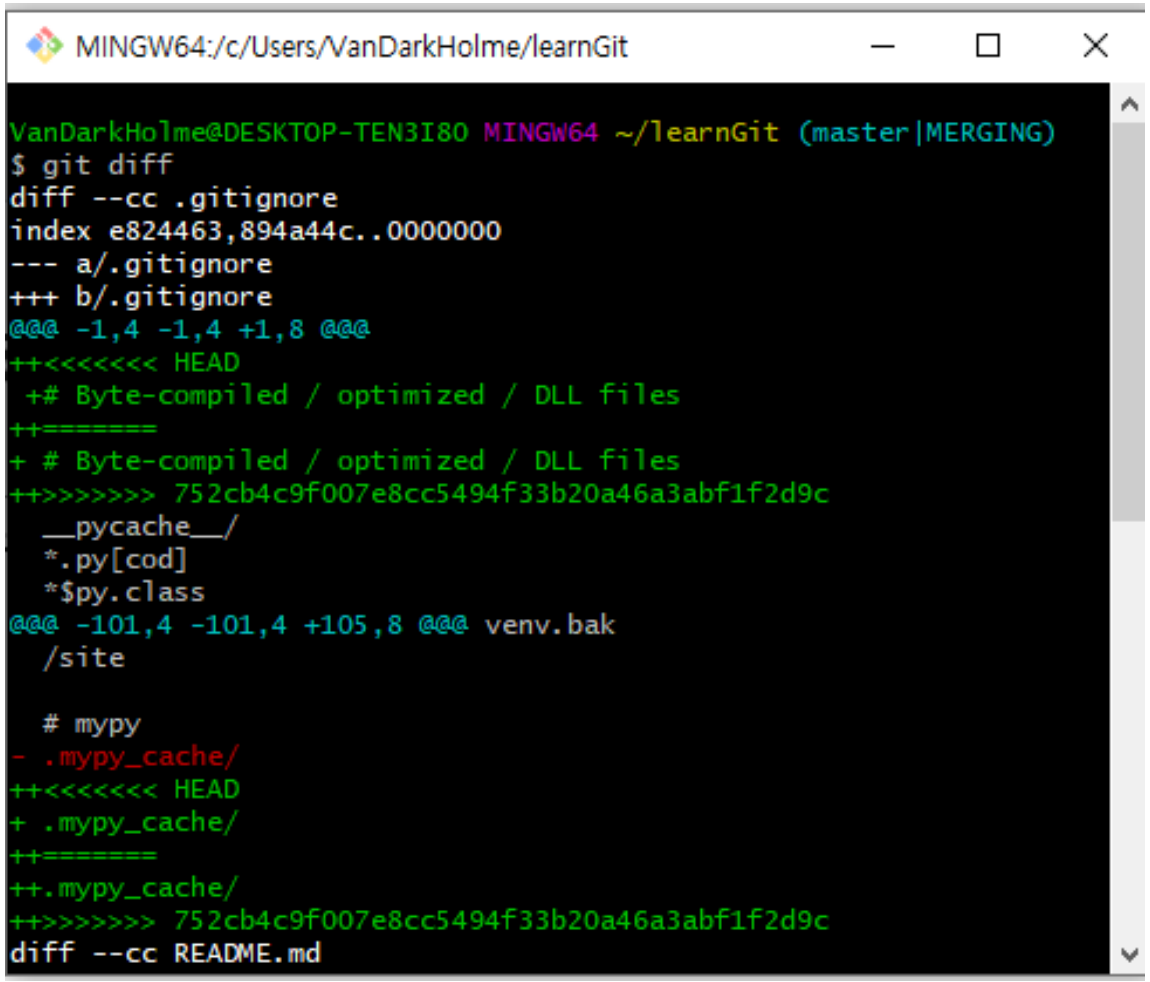
git fetch를 해봅니다



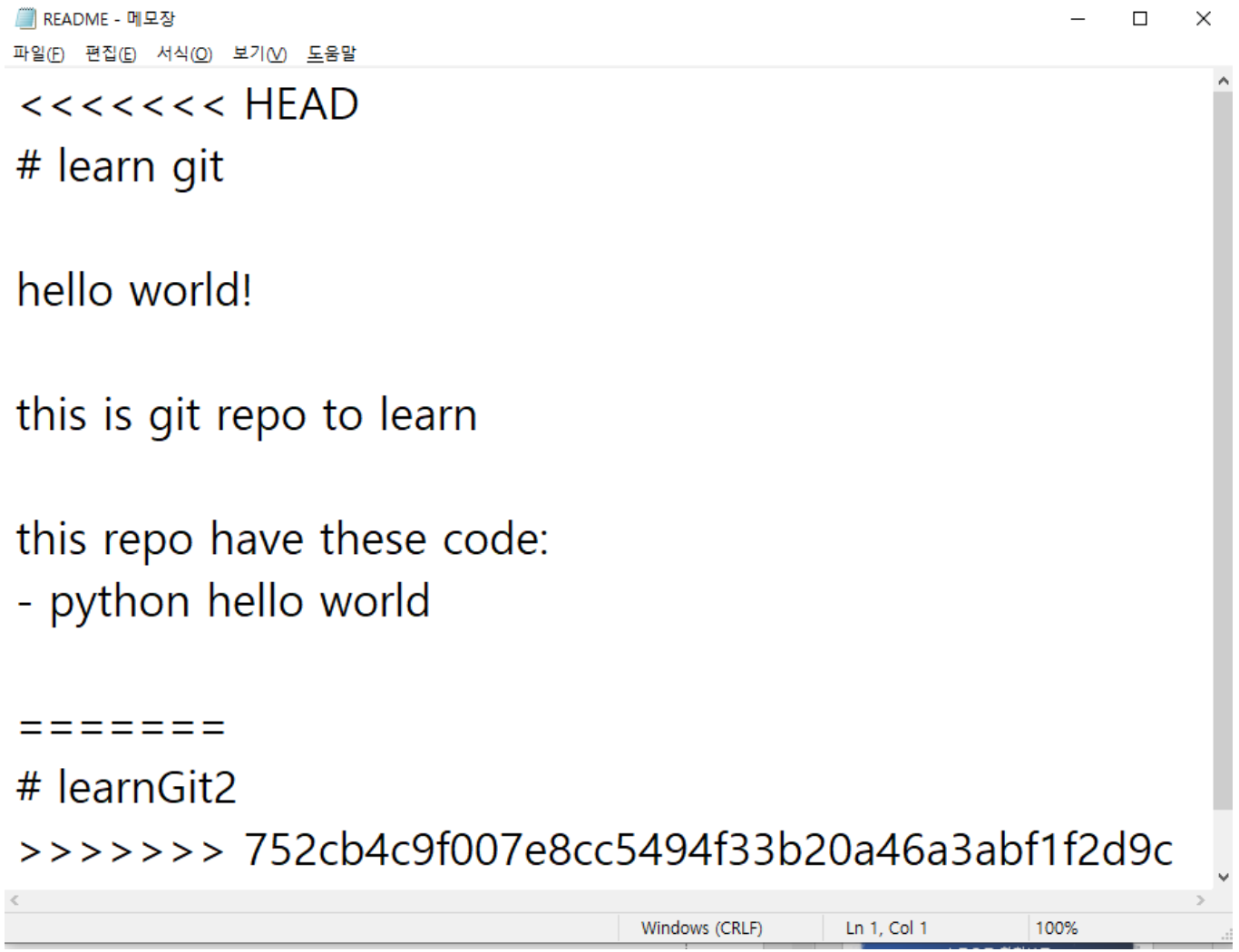
merge를 해볼까요(헬게이트 오픈)



당연히 auto merge는 실패합니다



수동으로 합쳐보죠 –README.md



수동으로 합쳐보죠 –README.md

README - 메모장

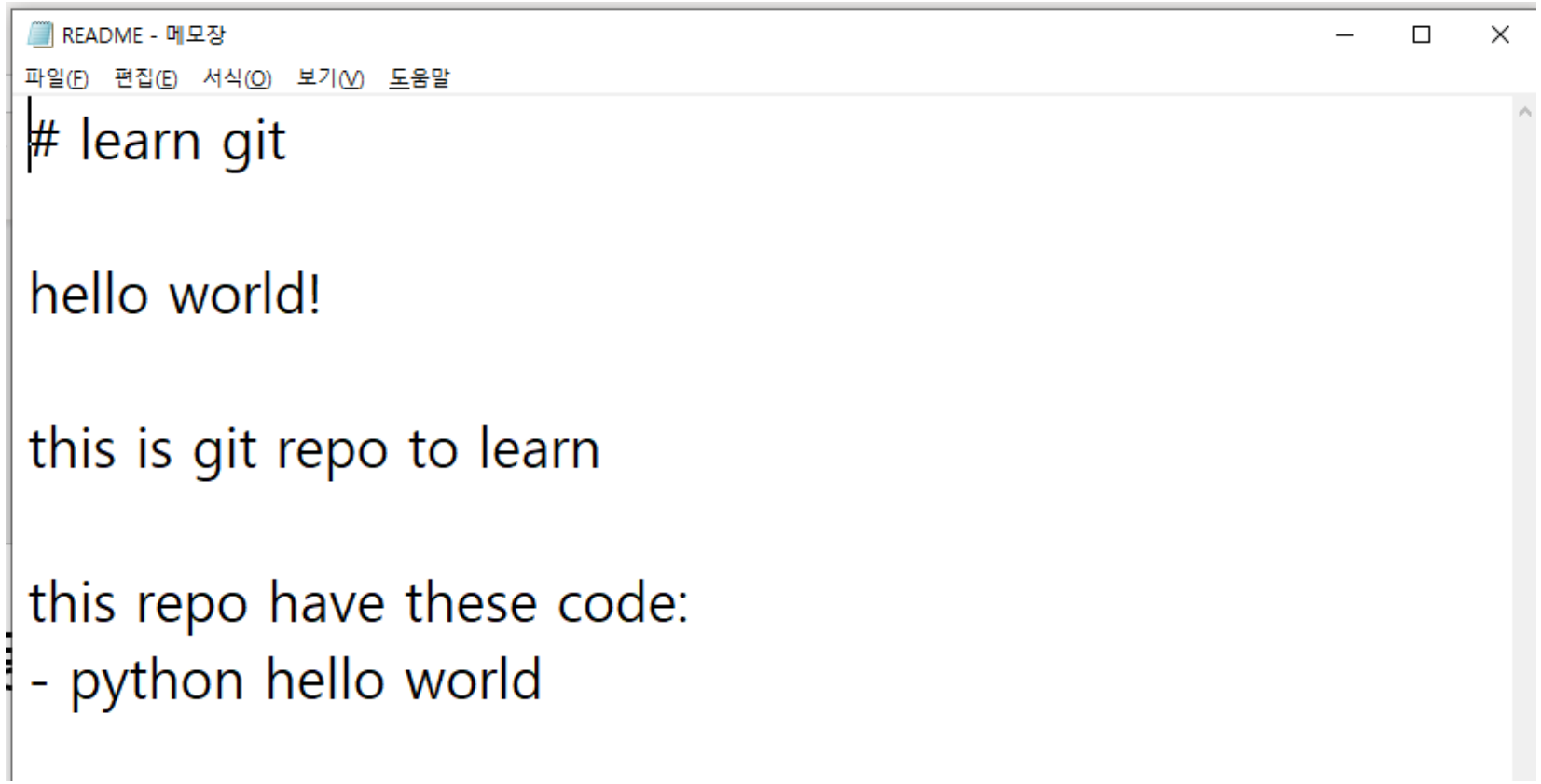
파일(F) 편집(E) 서식(O) 보기(V) 도움말

```
<<<<<<< HEAD
# learn git

hello world!

this is git repo to learn

this repo have these code:
- python hello world

=======
# learnGit2
>>>>>>> 752cb4c9f007e8cc5494f33b20a46a3abf1f2d9c
```

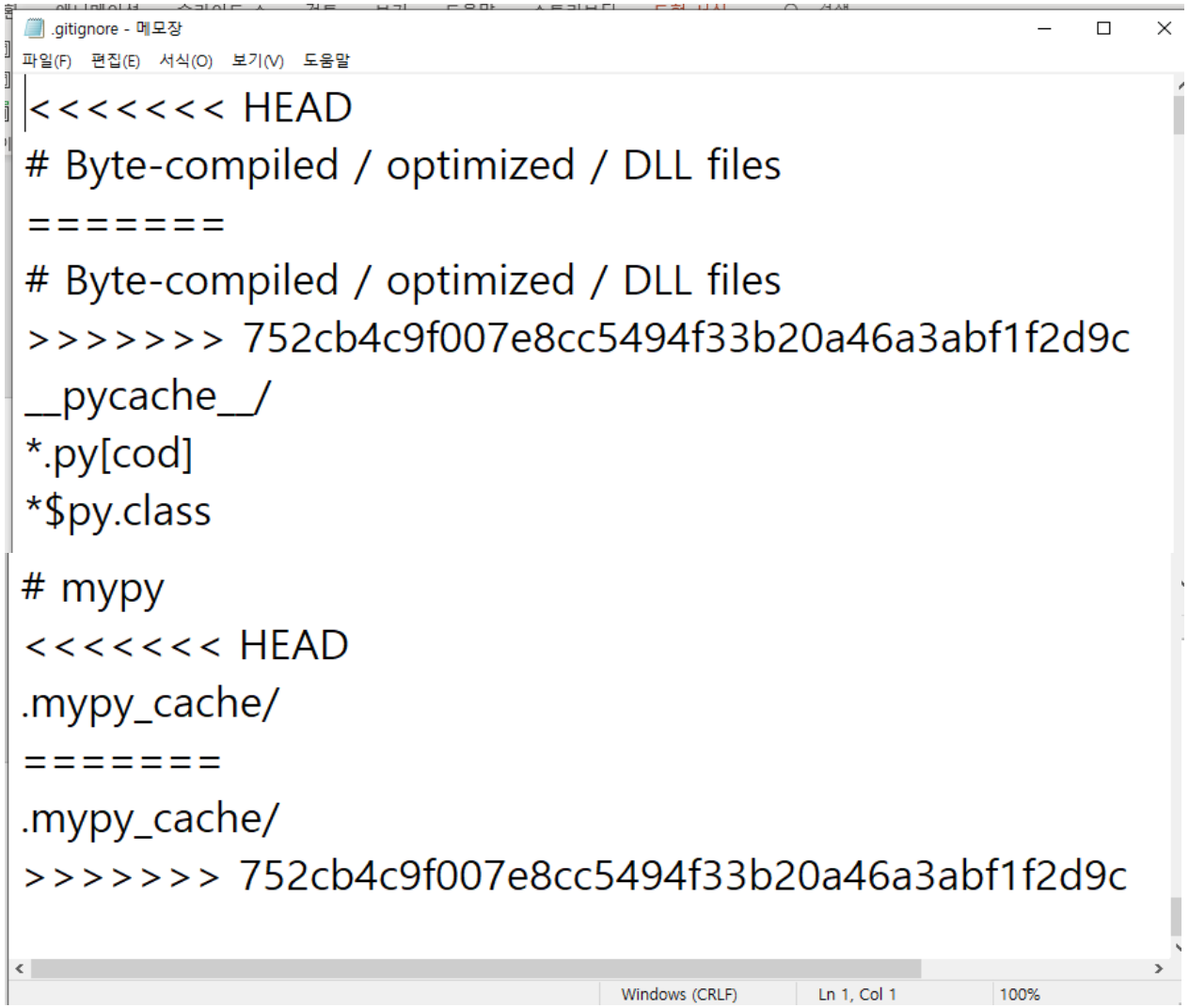
선택을 합니다
이걸 날려버리죠

Windows (CRLF) Ln 1, Col 1 100%

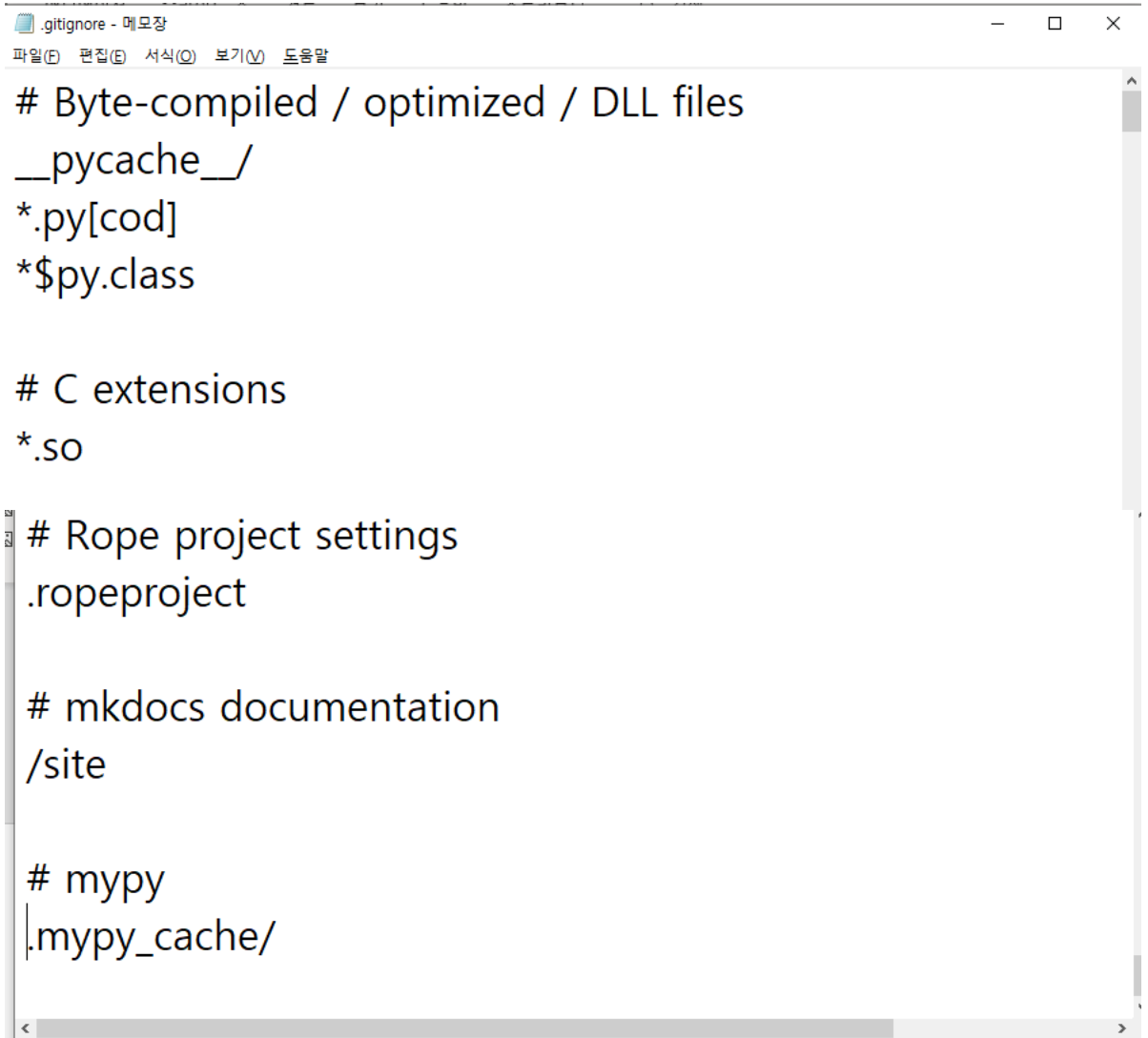
수동으로 합쳐보죠 –README.md



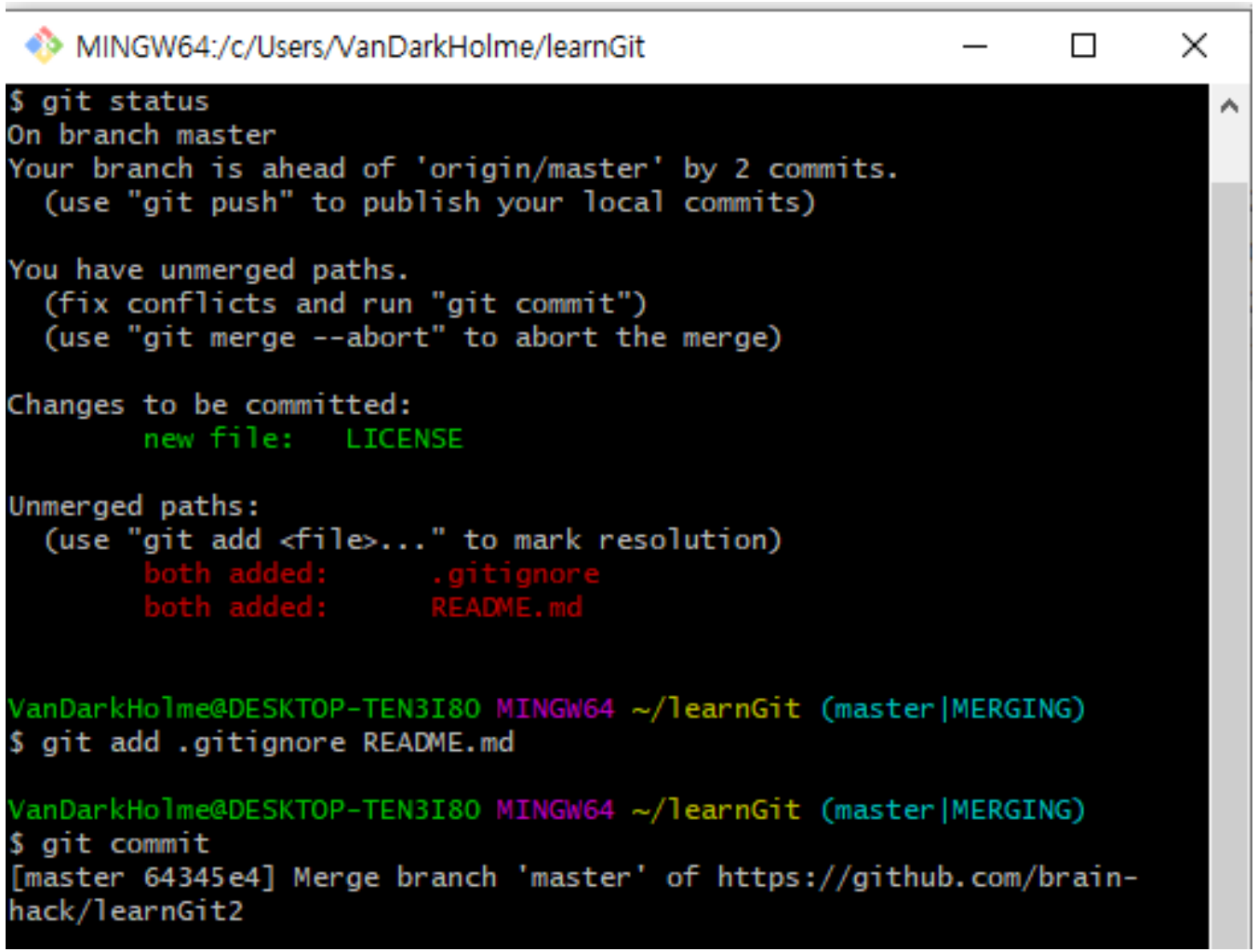
수동으로 합쳐보죠 - .gitignore



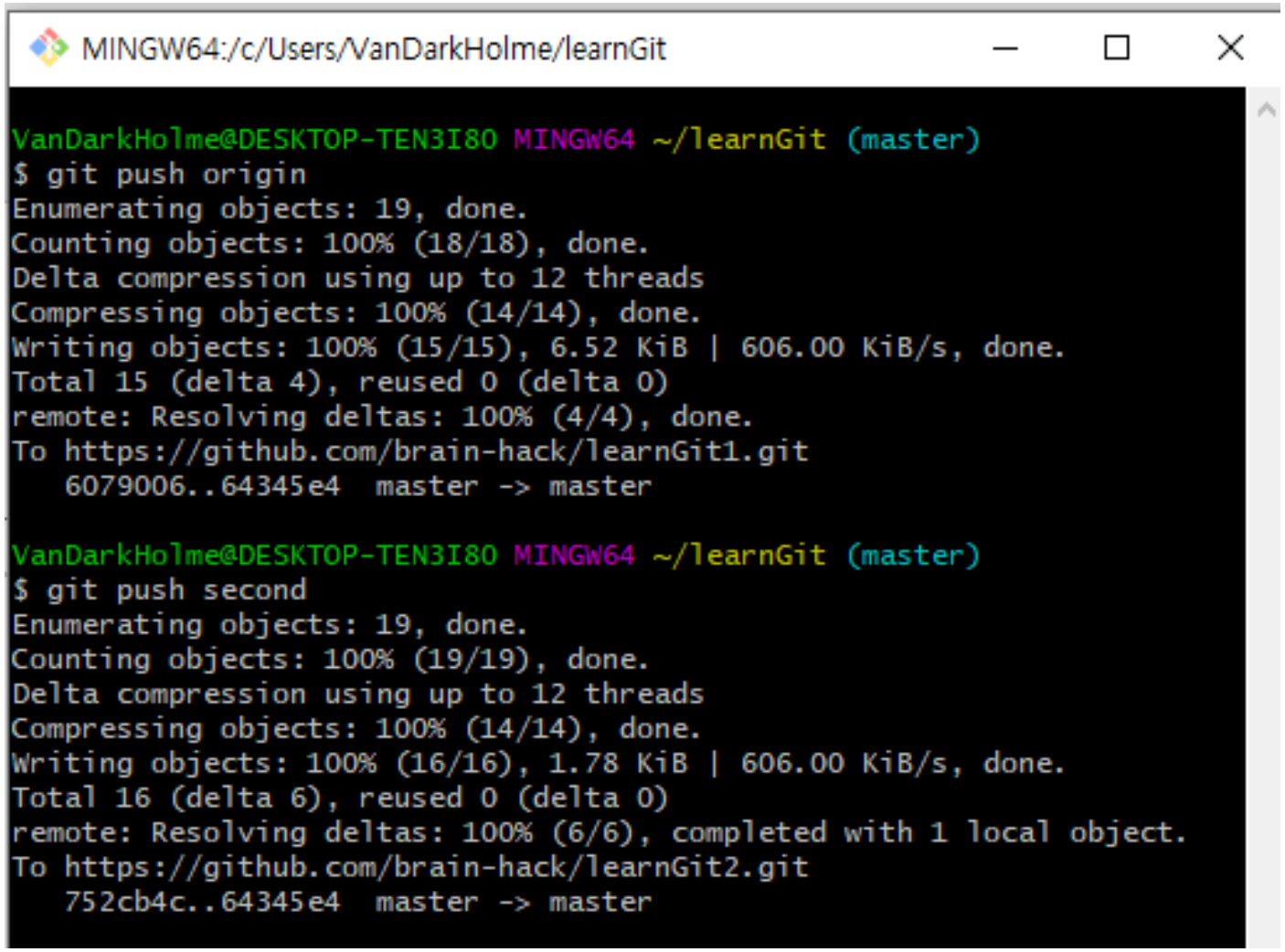
수동으로 합쳐보죠 - .gitignore



충돌 제거한 내용을 commit



새로 merge한 내역 push



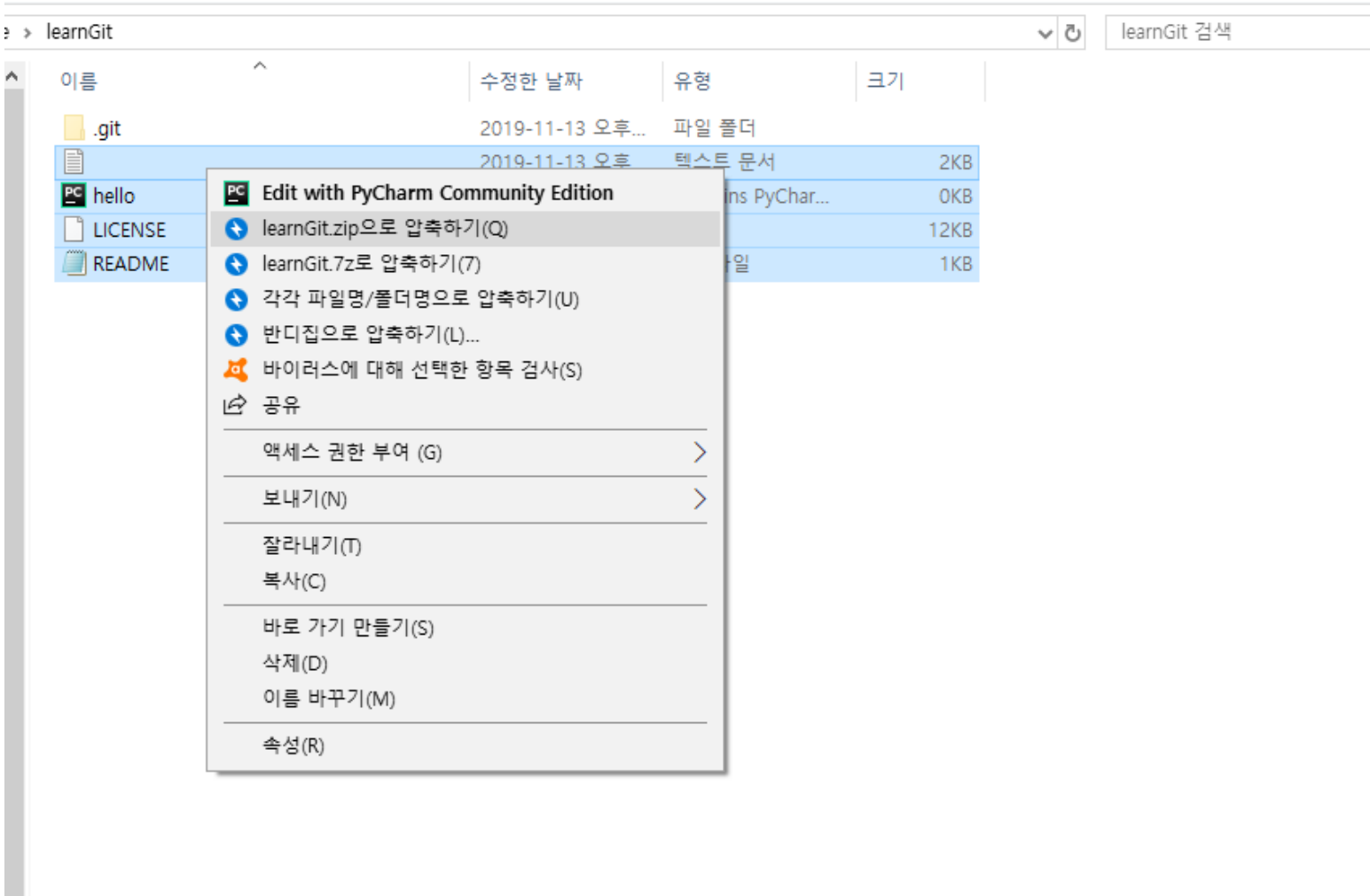
remote가 업데이트 됐으니 받아야겠죠?

MINGW64:/c/Users/VanDarkHolme/learnGit1

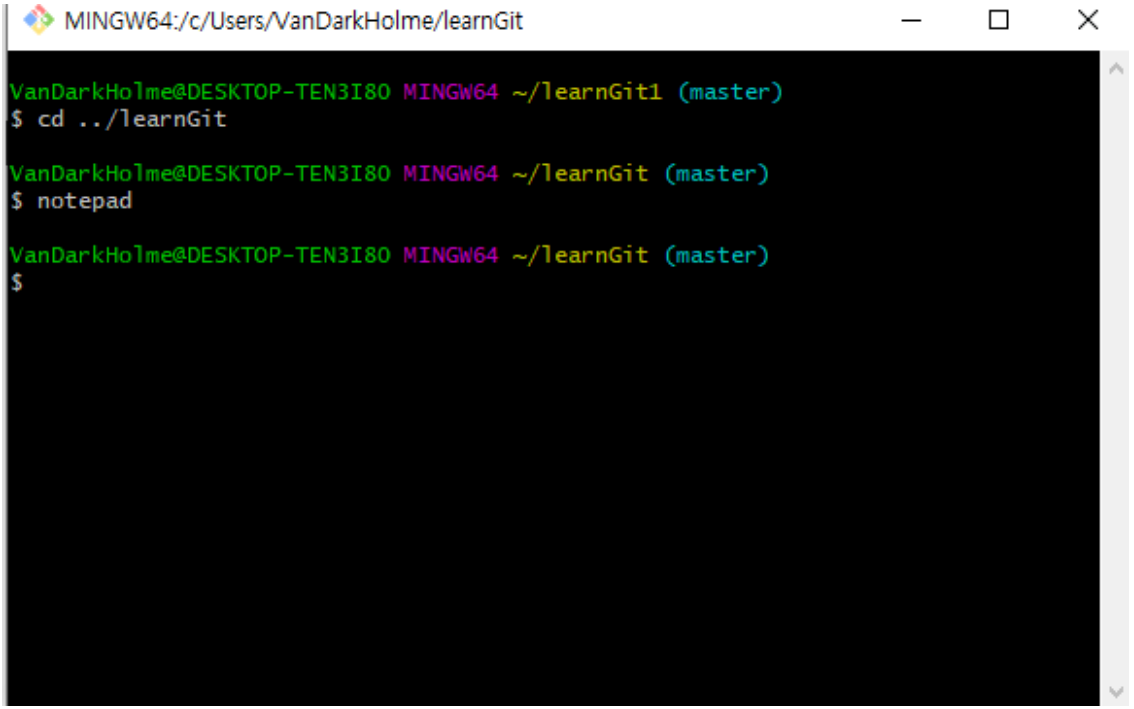
```
VanDarkHolme@DESKTOP-TEN3I80 MINGW64 ~/learnGit (master)
$ cd ../learnGit1

VanDarkHolme@DESKTOP-TEN3I80 MINGW64 ~/learnGit1 (master)
$ git pull
remote: Enumerating objects: 19, done.
remote: Counting objects: 100% (18/18), done.
remote: Compressing objects: 100% (10/10), done.
remote: Total 15 (delta 4), reused 15 (delta 4), pack-reused 0
Unpacking objects: 100% (15/15), done.
From https://github.com/brain-hack/learnGit1
   6079006..64345e4  master     -> origin/master
Updating 6079006..64345e4
Fast-forward
 .gitignore | 104 ++++++++++++++++++++++++++++++++
 LICENSE    | 201 +++++++++++++++++++++++++++++++++++++++++++++++++++++
+++++++++
 README.md  |   8 ++-
 3 files changed, 311 insertions(+), 2 deletions(-)
 create mode 100644 .gitignore
 create mode 100644 LICENSE
```

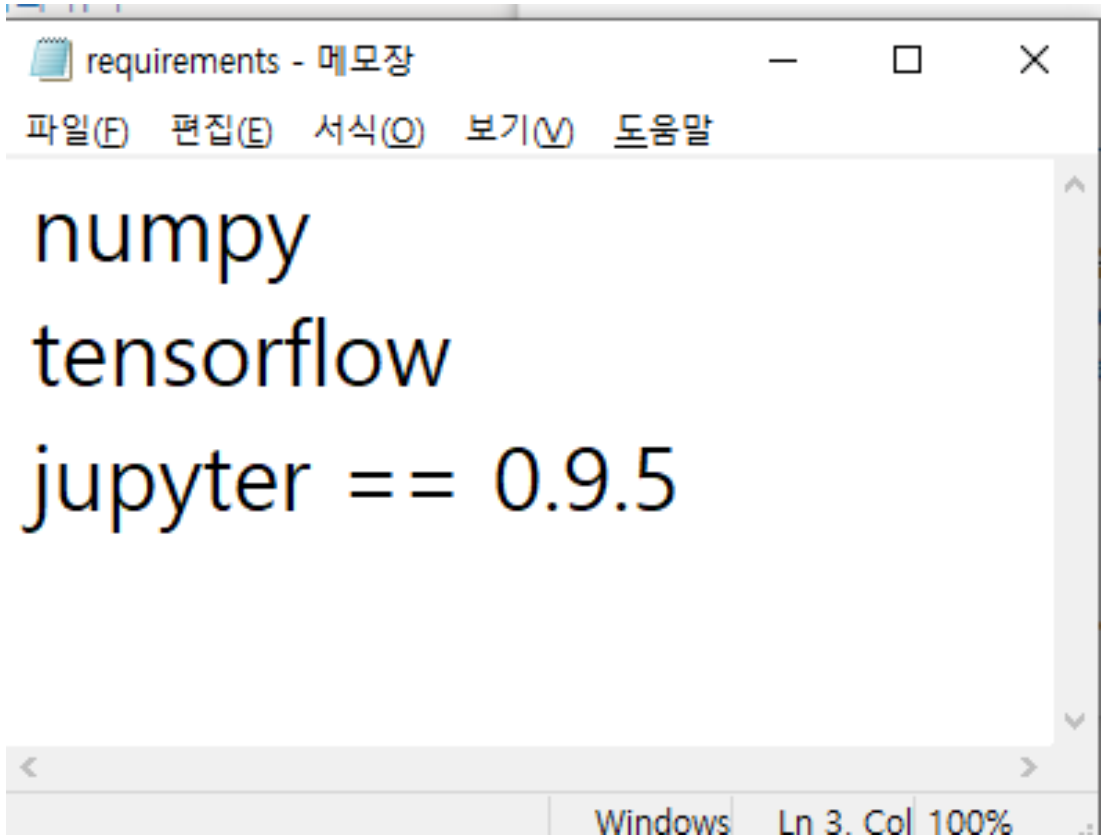
local learnGit : 우선 압축파일을 만들어볼까요

(이 파일은 커밋에 안 넣을 겁니다)

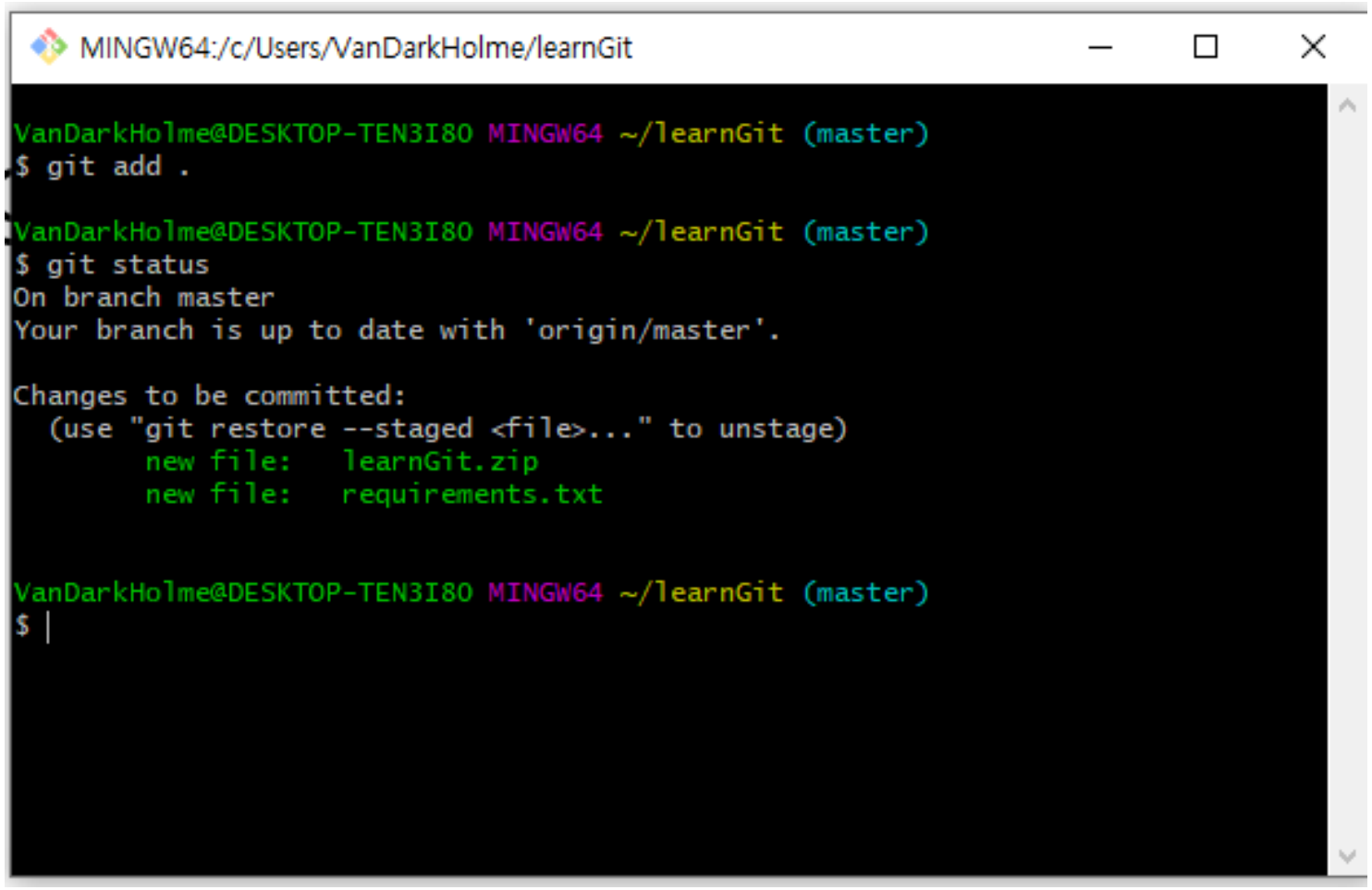


local learnGit : requirements 파일을 추가합니다.

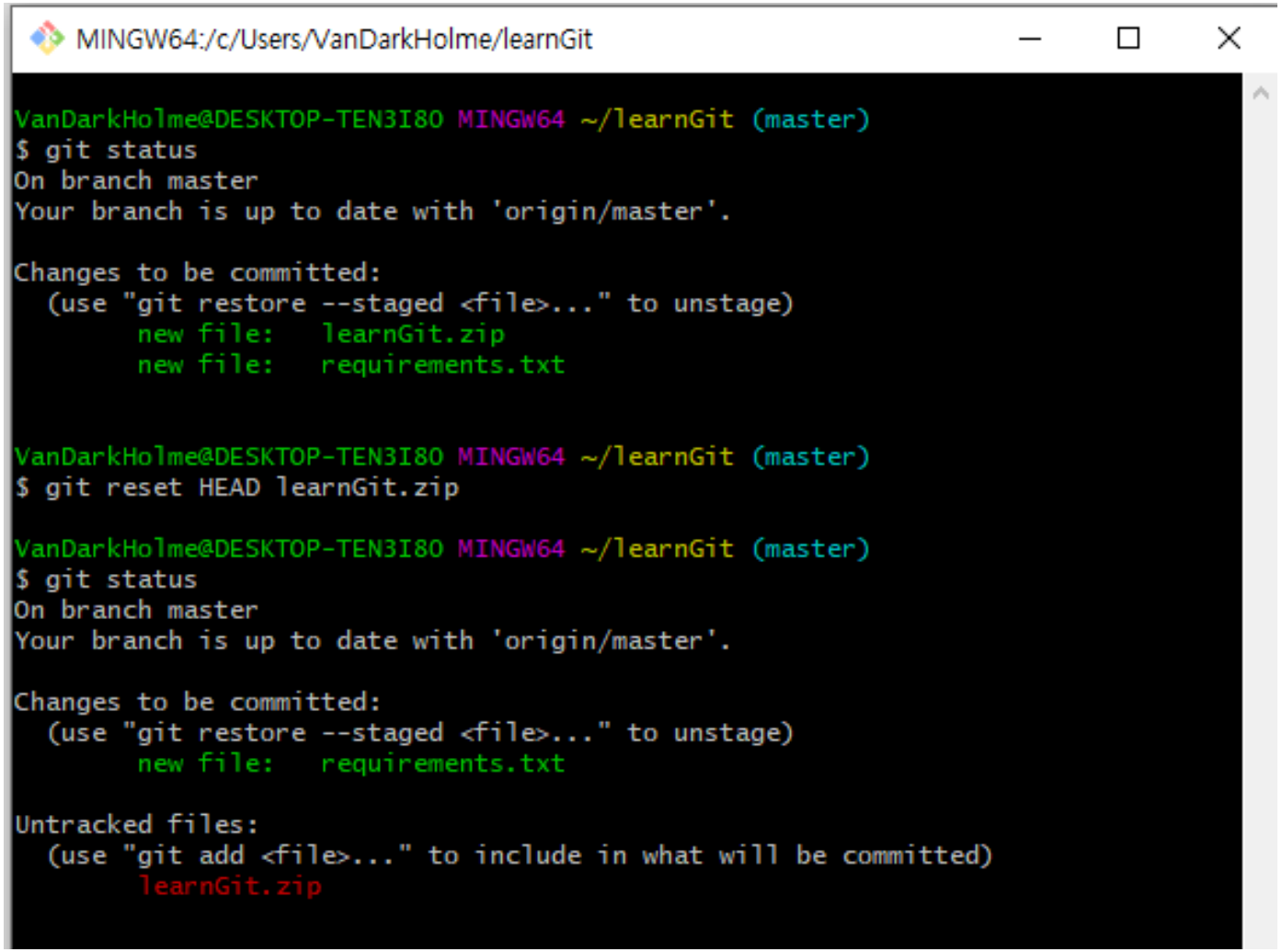




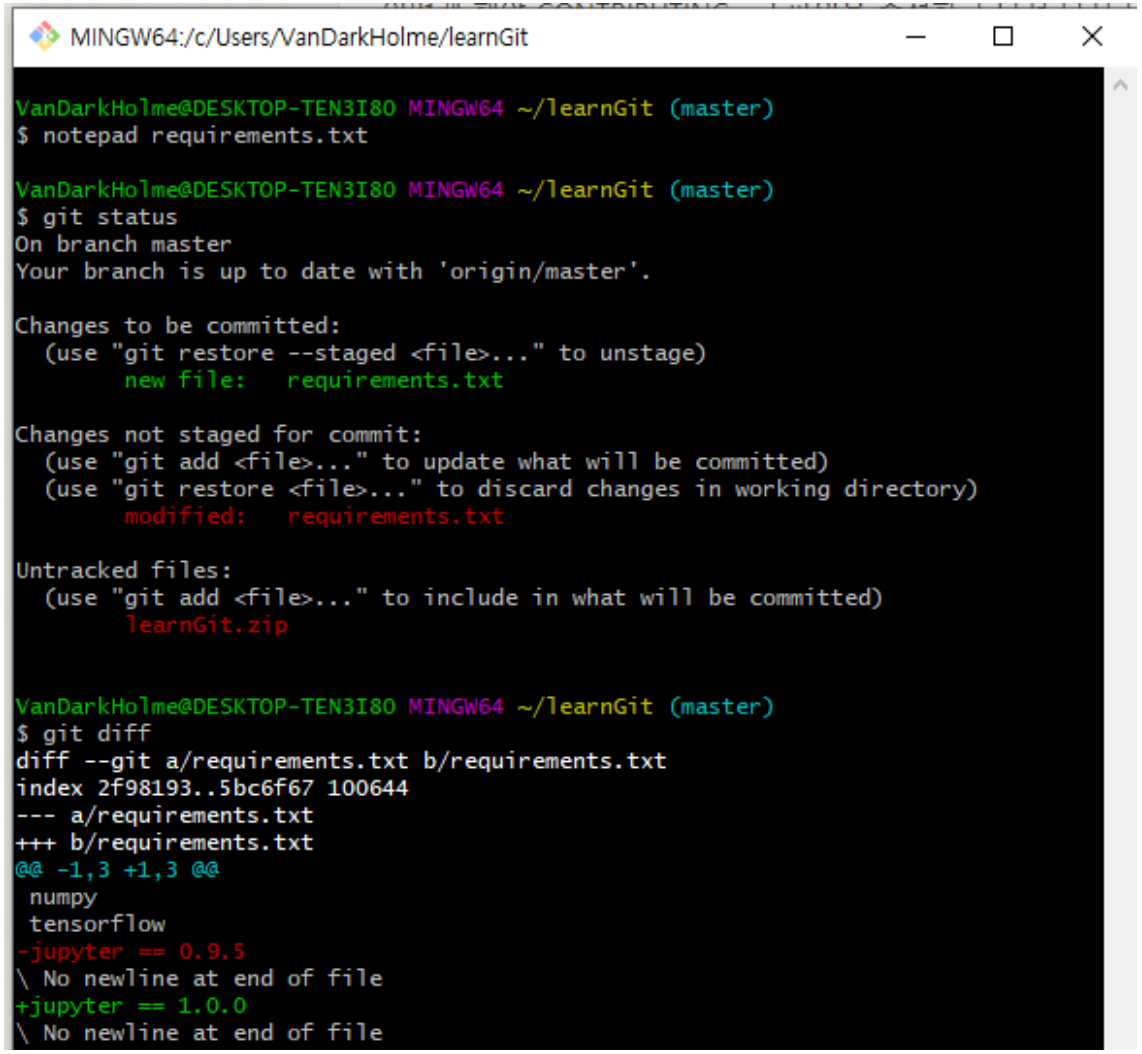
git add를 해볼까요

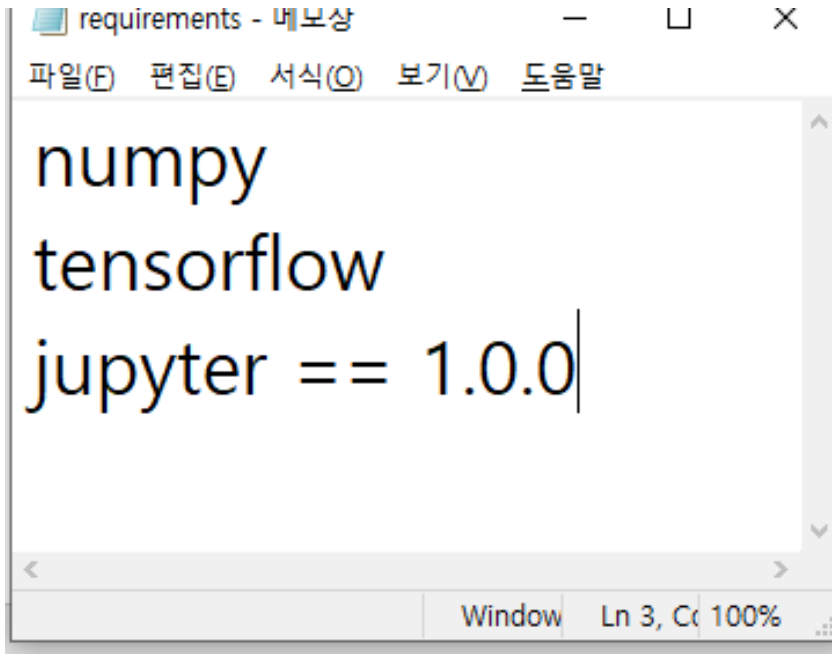


오 이런 실수로 압축파일도 추가했네요

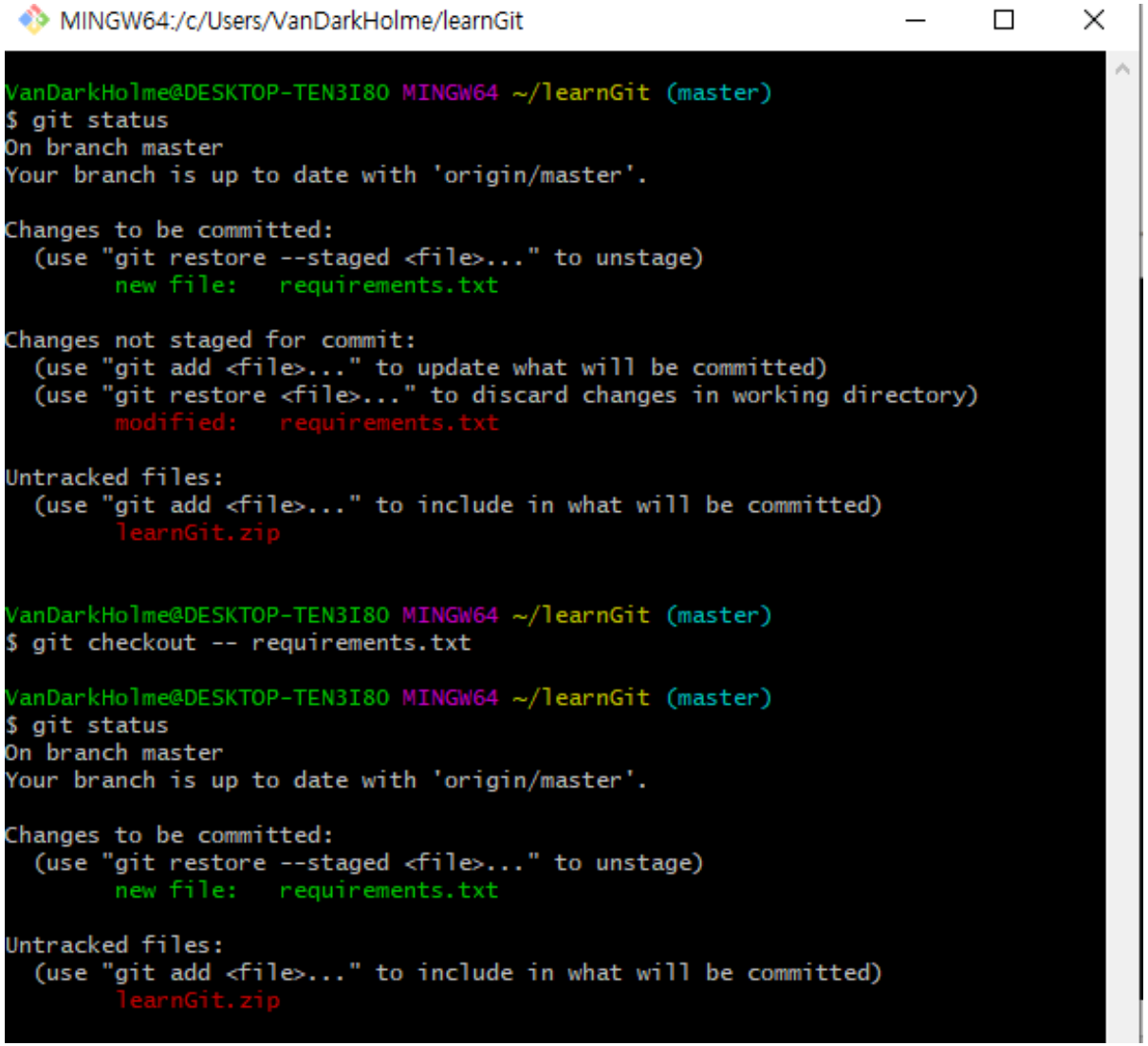


requirements의 주피터 버전을 1.0.0으로

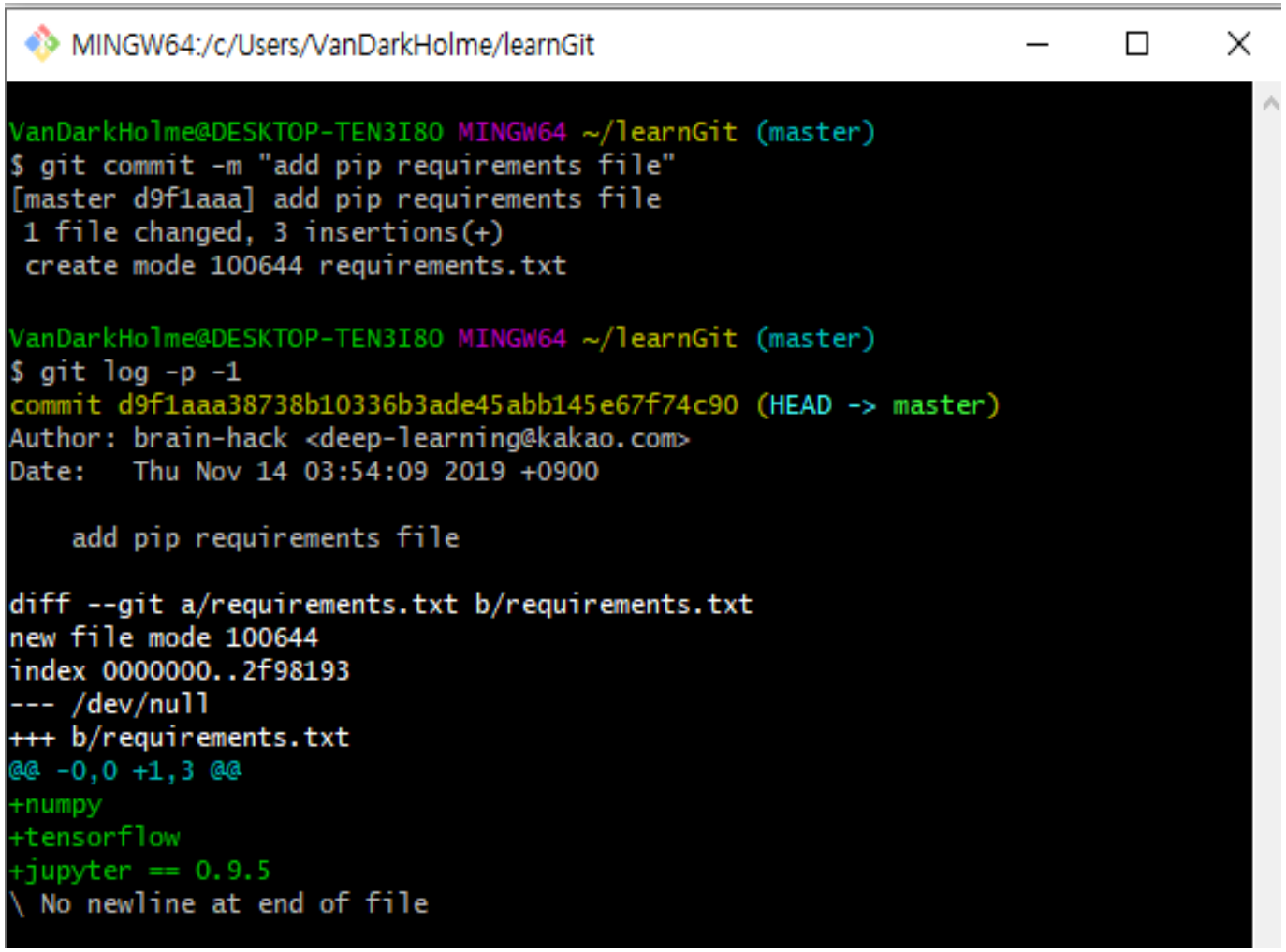




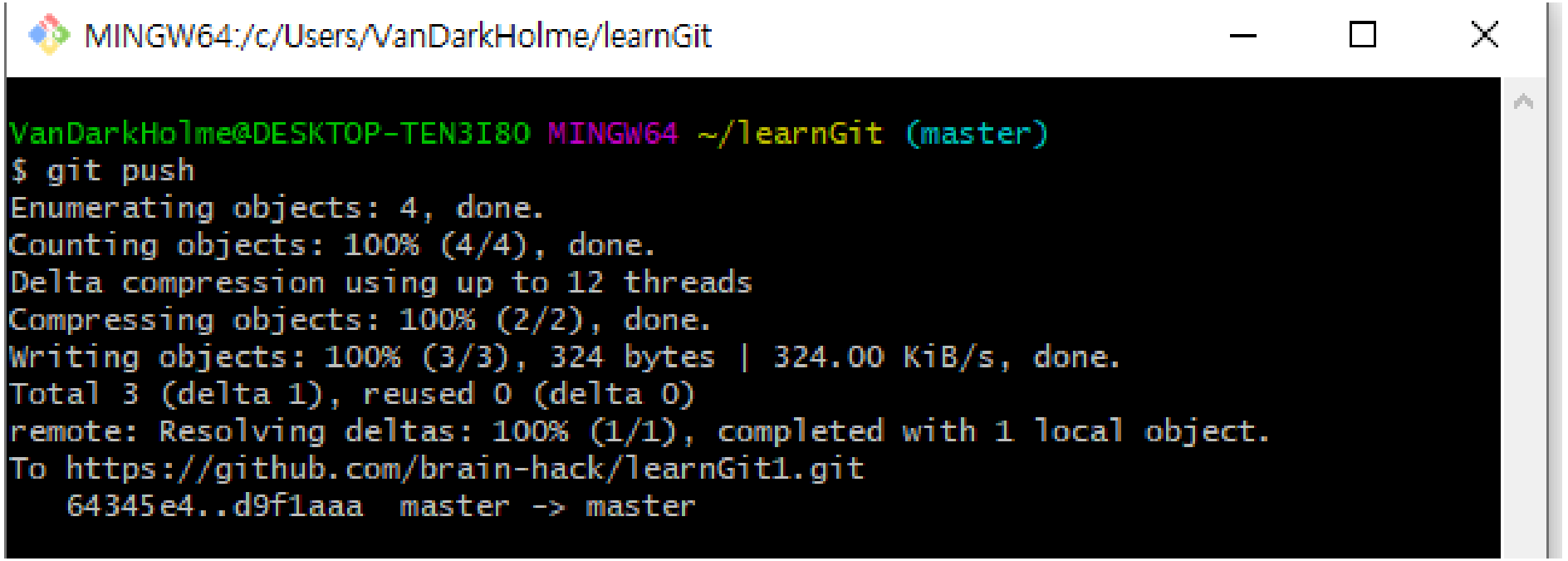
앗 jupyter 1.0.0에서는 원하는 기능이 안됩니다.



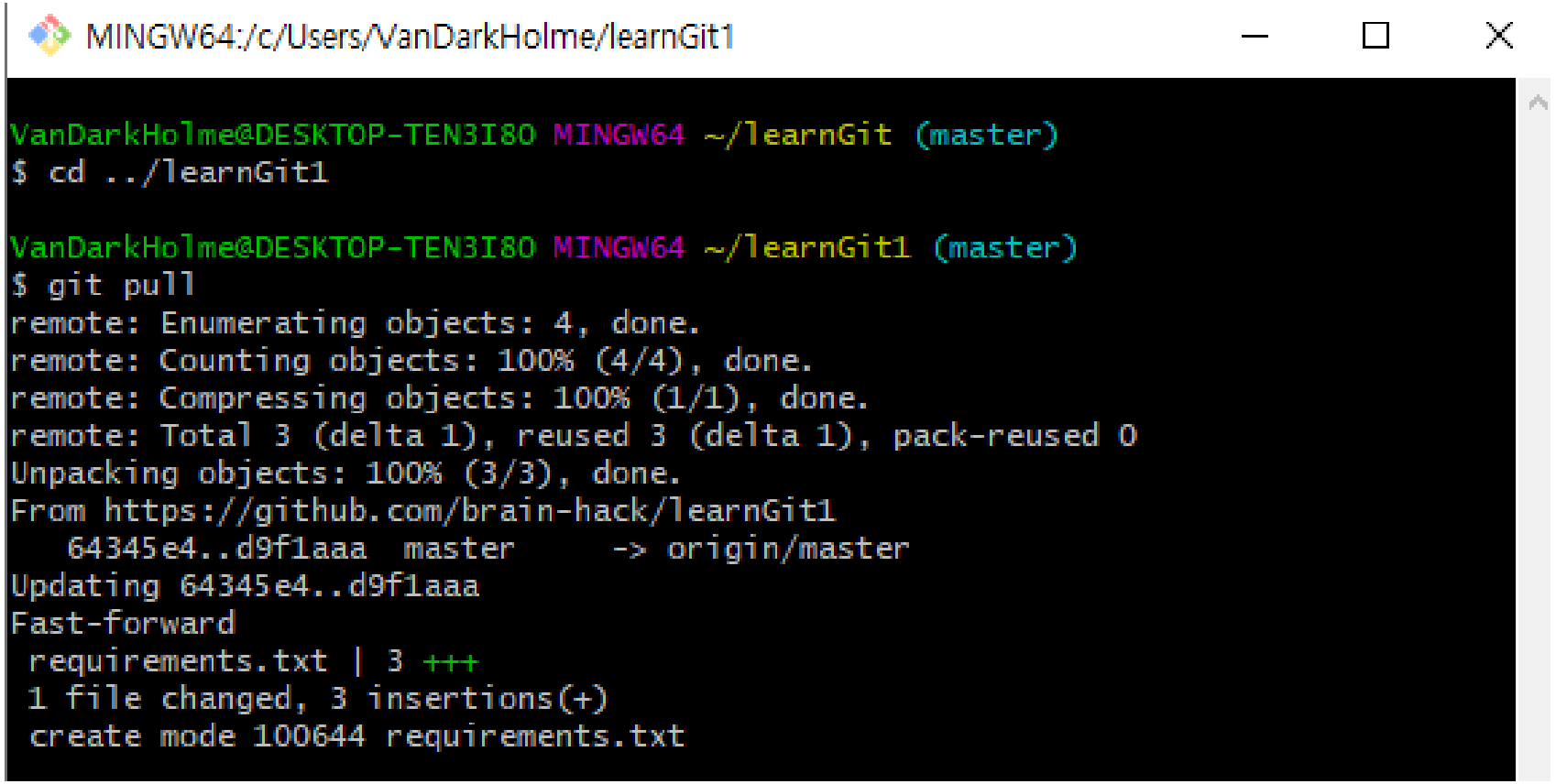
커밋하기


MINGW64:/c/Users/VanDarkHolme/learnGit
VanDarkHolme@DESKTOP-TEN3I80 MINGW64 ~/learnGit (master)
$ git commit -m "add pip requirements file"
[master d9f1aaa] add pip requirements file
1 file changed, 3 insertions(+)
create mode 100644 requirements.txt
VanDarkHolme@DESKTOP-TEN3I80 MINGW64 ~/learnGit (master)
$ git log -p -1
commit d9f1aaa38738b10336b3ade45abb145e67f74c90 (HEAD -> master)
Author: brain-hack <deep-learning@kakao.com>
Date: Thu Nov 14 03:54:09 2019 +0900
add pip requirements file
diff --git a/requirements.txt b/requirements.txt
new file mode 100644
index 0000000..2f98193
--- /dev/null
+++ b/requirements.txt
@@ -0,0 +1,3 @@
+numpy
+tensorflow
+jupyter == 0.9.5
\ No newline at end of file

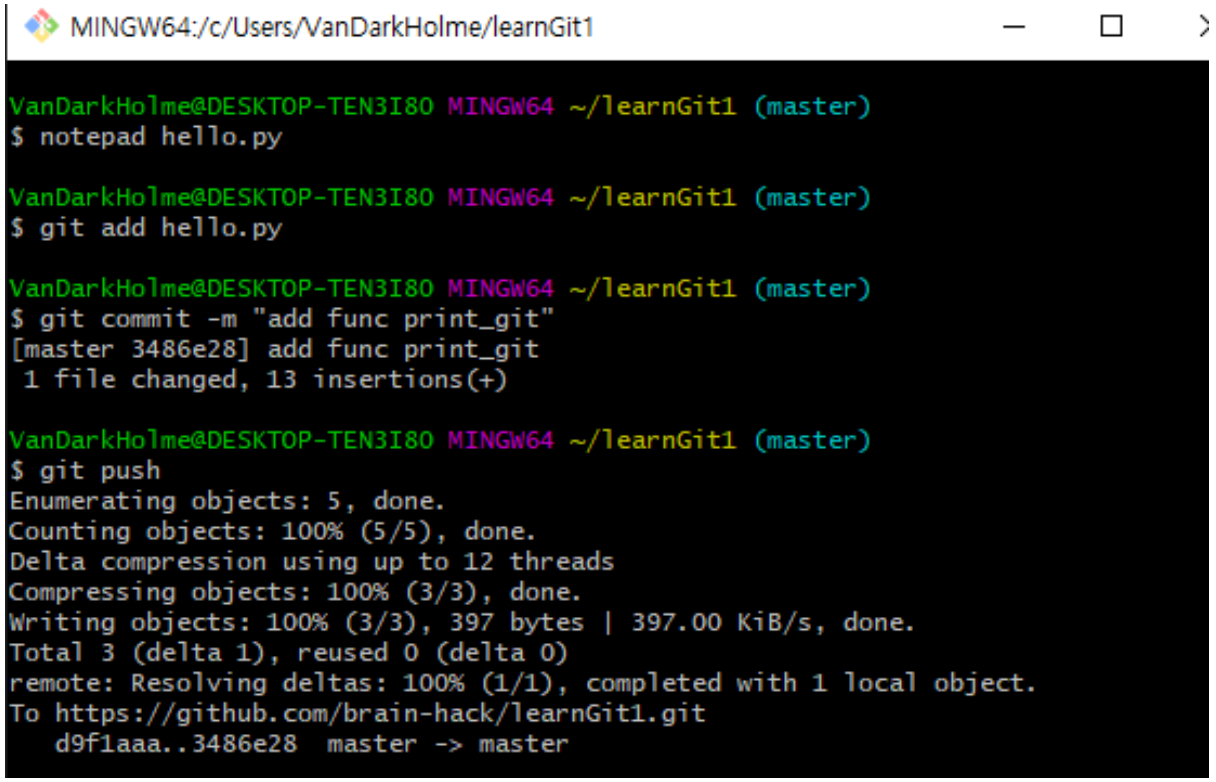
push하기

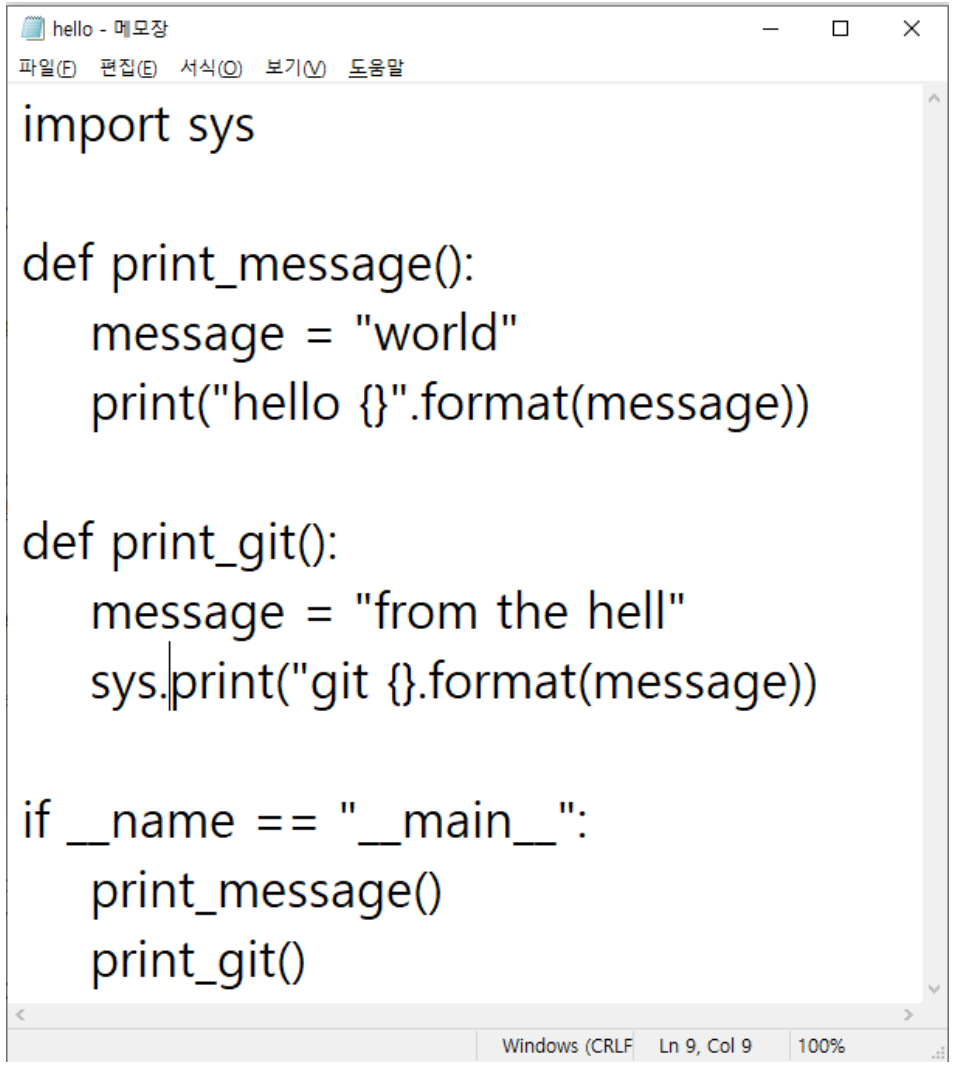

MINGW64:/c/Users/VanDarkHolme/learnGit
VanDarkHolme@DESKTOP-TEN3I8O MINGW64 ~/learnGit (master)
$ git push
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 12 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 324 bytes | 324.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/brain-hack/learnGit1.git
64345e4..d9f1aaa master -> master

local repo : learnGit1 작업하기

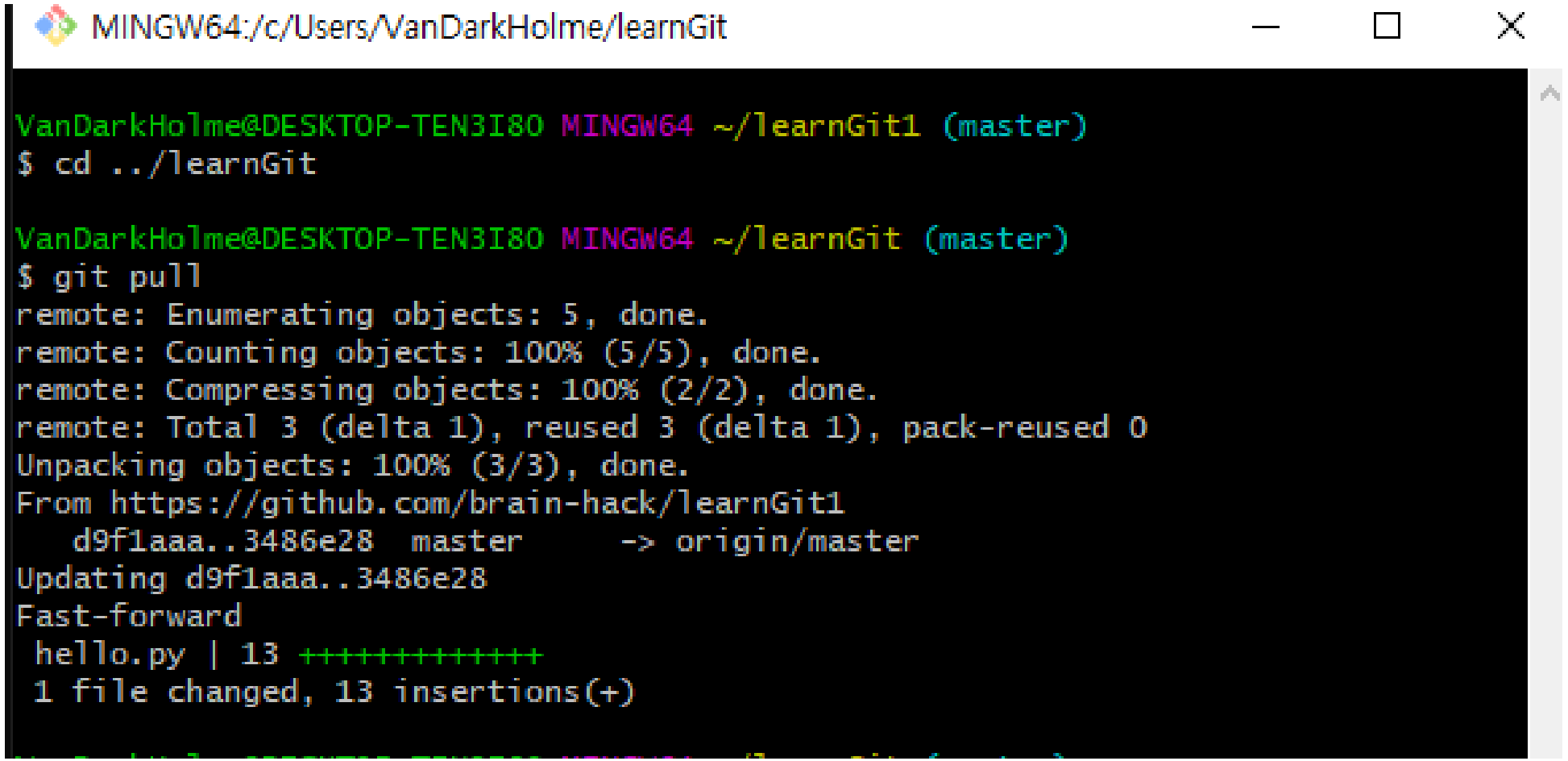


local repo : learnGit1 작업하기

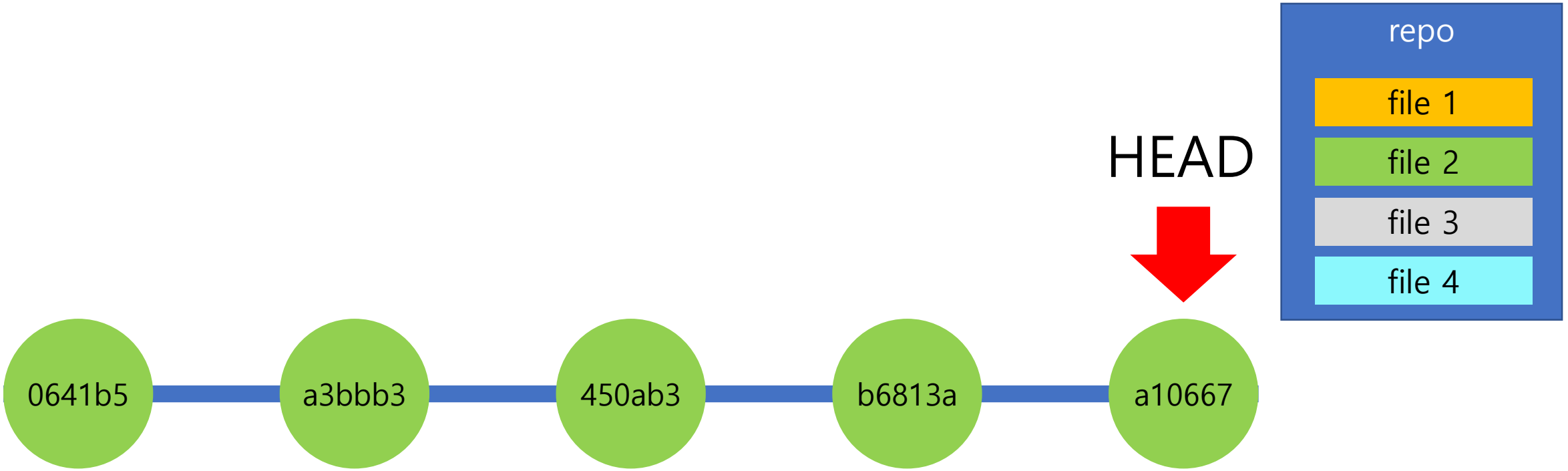




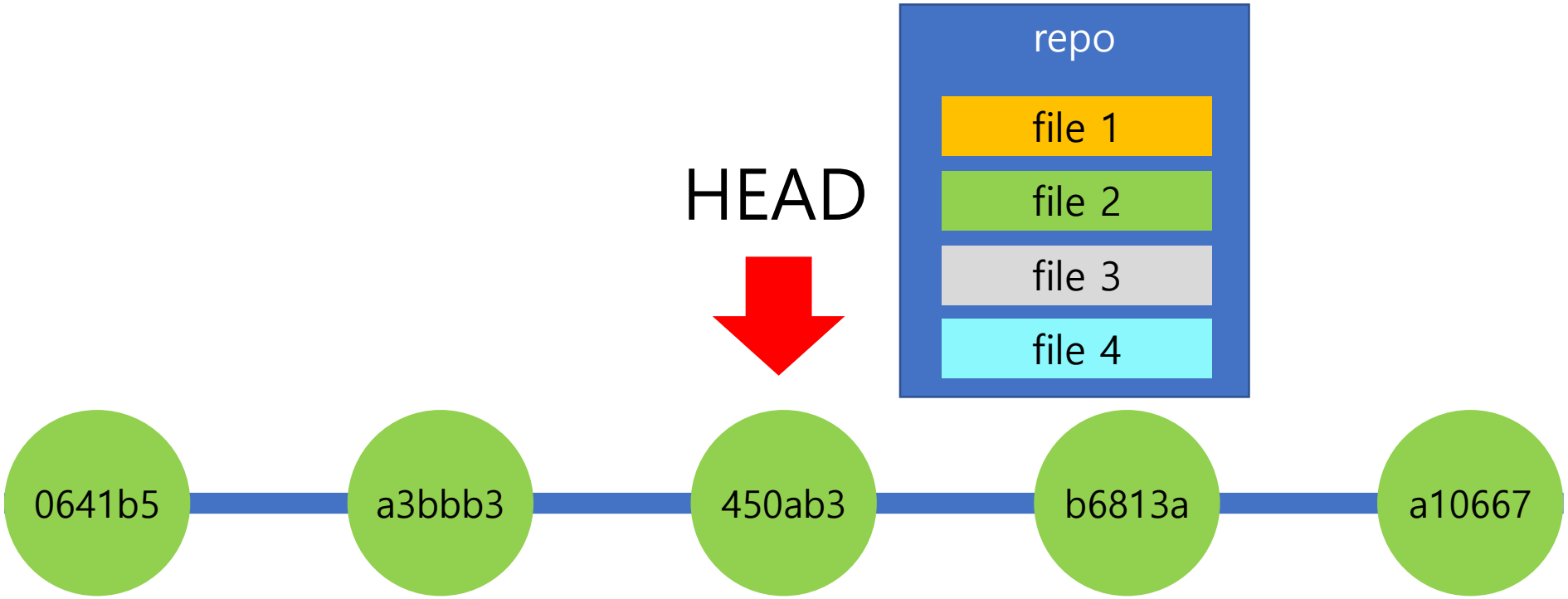
local repo : learnGit pull받기



커밋 되돌리기

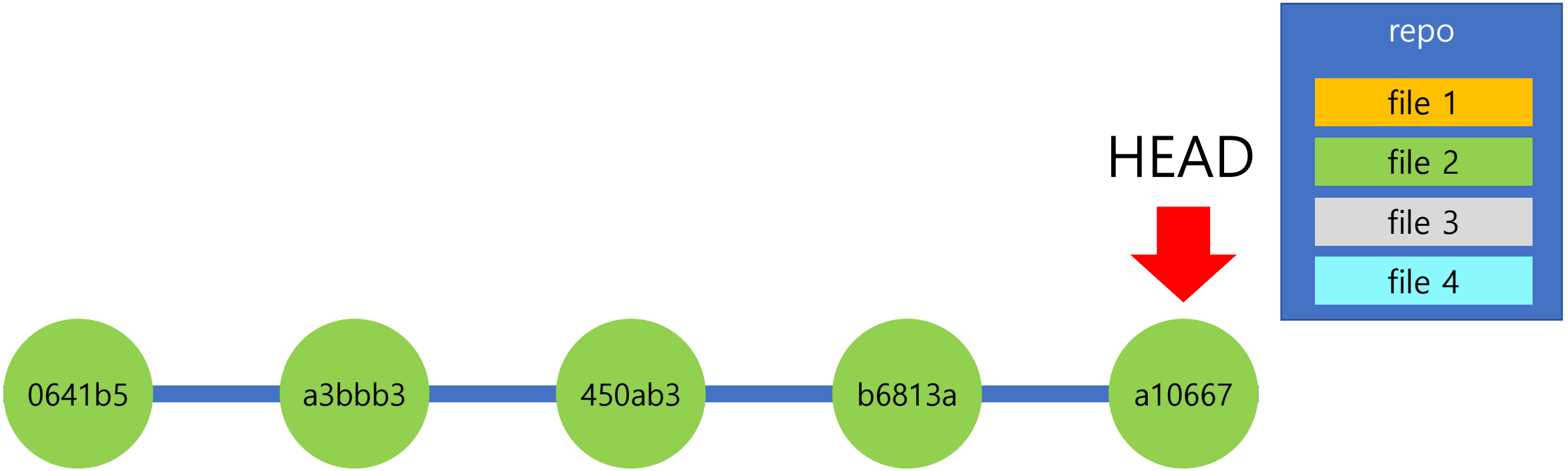

repo
file 1
file 2
file 3
file 4
HEAD
0641b5
a3bbb3
450ab3
b6813a
a10667

커밋 되돌리기

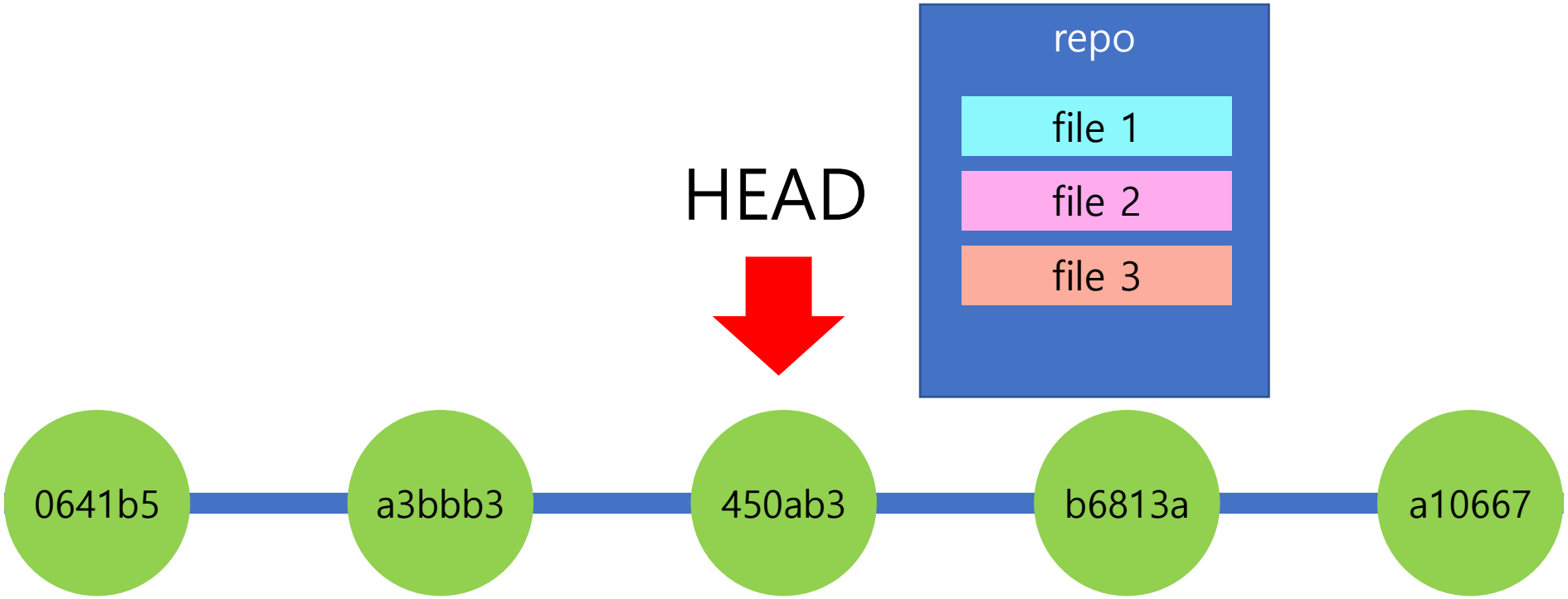



```
$ git reset --soft 450ab3
```

커밋 되돌리기

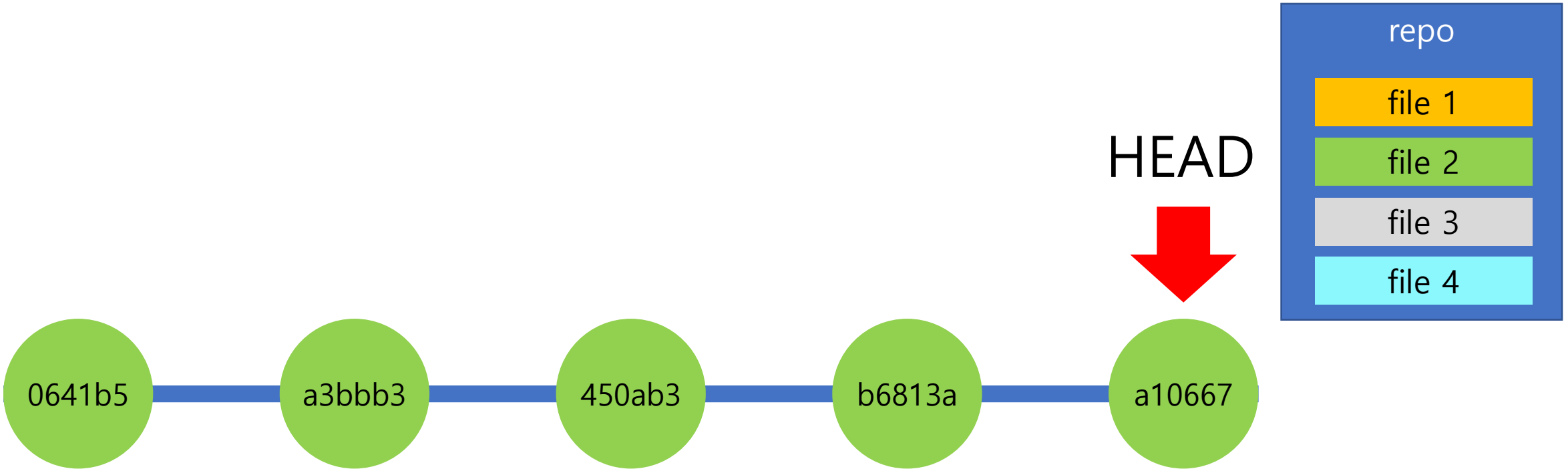

repo
file 1
file 2
file 3
file 4
HEAD
0641b5
a3bbb3
450ab3
b6813a
a10667

커밋 되돌리기

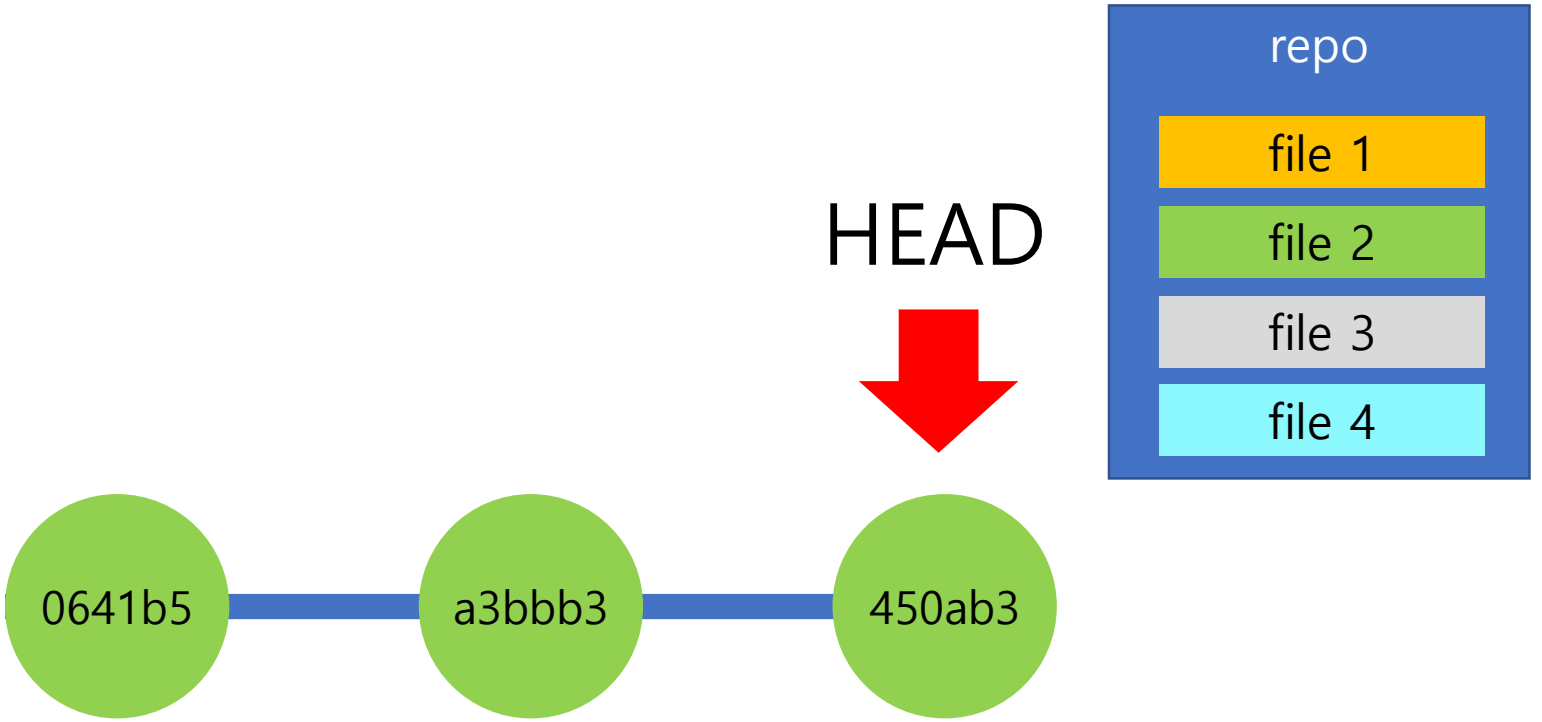



```
$ git reset --mixed 450ab3
```

커밋 되돌리기


repo
file 1
file 2
file 3
file 4
HEAD
0641b5
a3bbb3
450ab3
b6813a
a10667

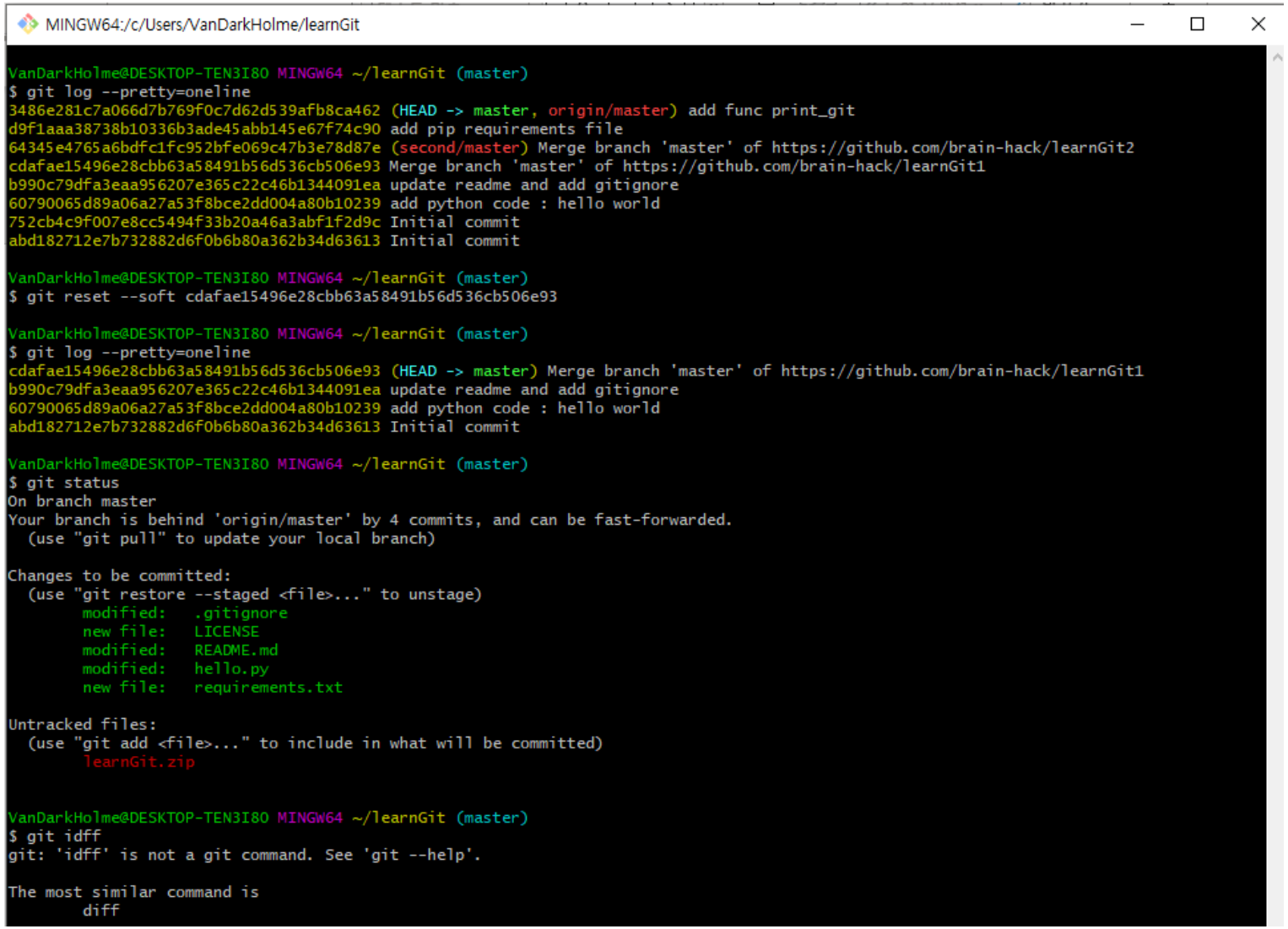
커밋 되돌리기



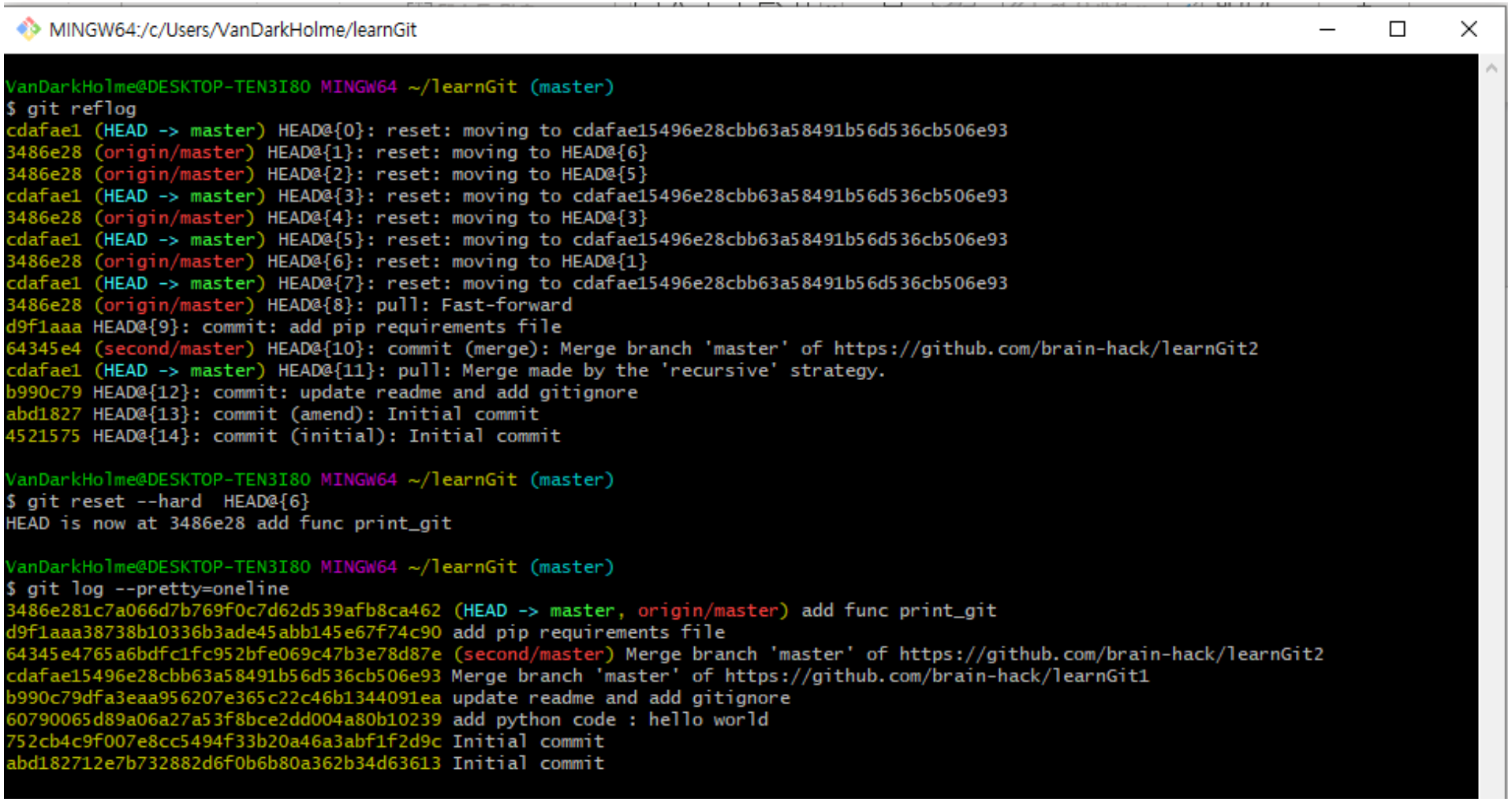
unstaged

```
$ git reset --hard 450ab3
```

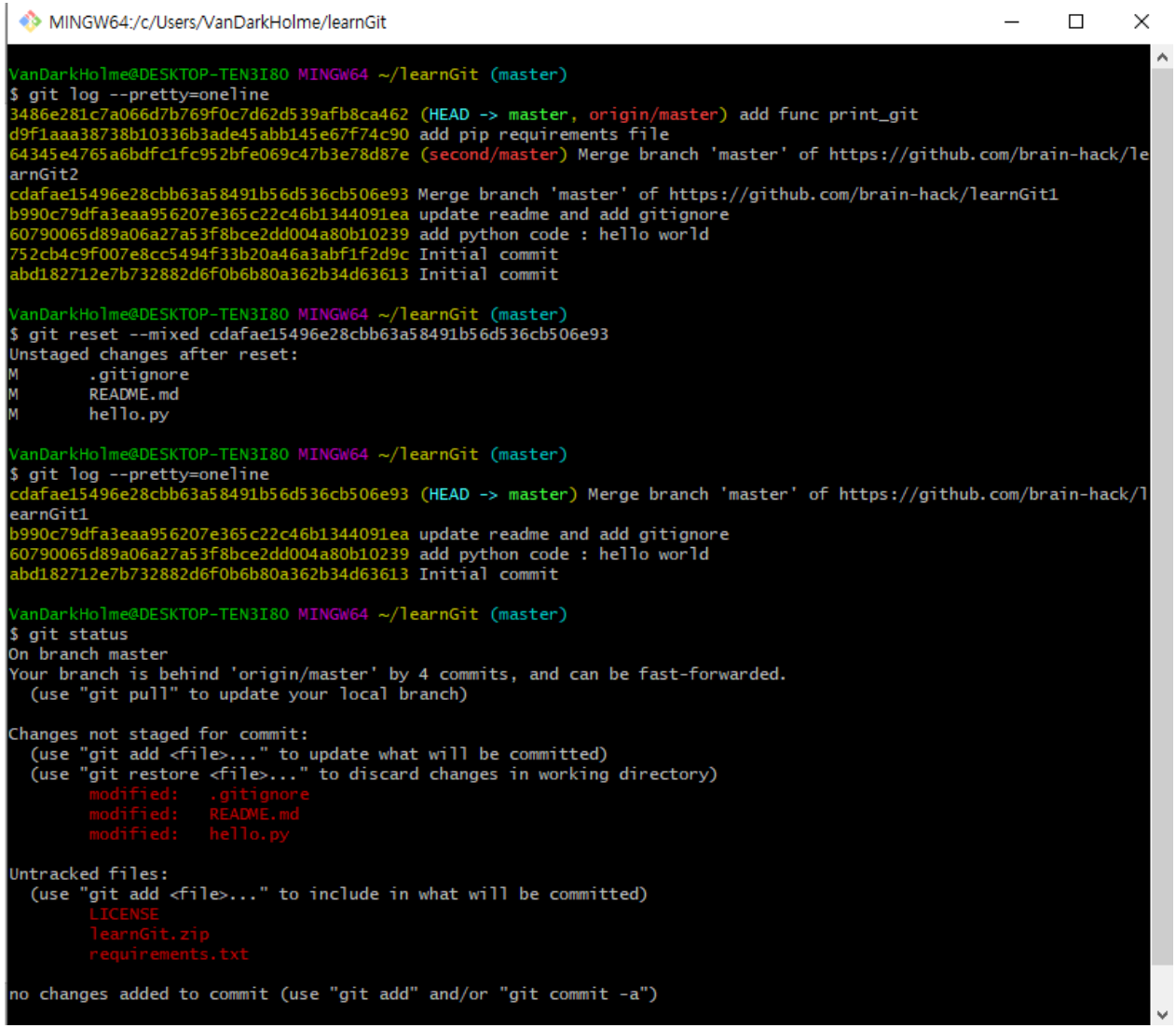
커밋 되돌리기 reset --soft



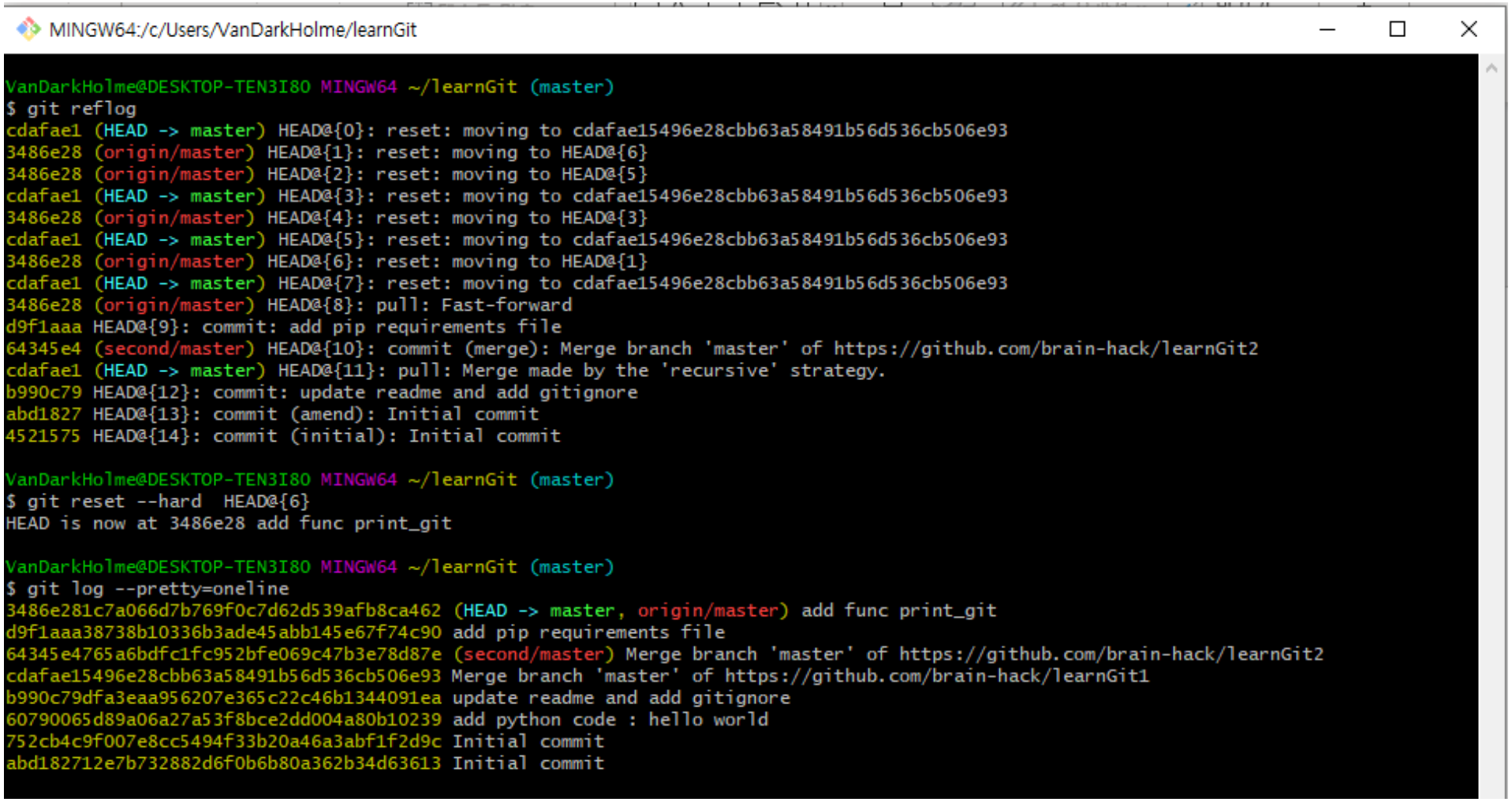
reset 취소


MINGW64:/c/Users/VanDarkHolme/learnGit
VanDarkHolme@DESKTOP-TEN3I8O MINGW64 ~/learnGit (master)
$ git reflog
cdafae1 (HEAD -> master) HEAD@{0}: reset: moving to cdafae15496e28cbb63a58491b56d536cb506e93
3486e28 (origin/master) HEAD@{1}: reset: moving to HEAD@{6}
3486e28 (origin/master) HEAD@{2}: reset: moving to HEAD@{5}
cdafae1 (HEAD -> master) HEAD@{3}: reset: moving to cdafae15496e28cbb63a58491b56d536cb506e93
3486e28 (origin/master) HEAD@{4}: reset: moving to HEAD@{3}
cdafae1 (HEAD -> master) HEAD@{5}: reset: moving to cdafae15496e28cbb63a58491b56d536cb506e93
3486e28 (origin/master) HEAD@{6}: reset: moving to HEAD@{1}
cdafae1 (HEAD -> master) HEAD@{7}: reset: moving to cdafae15496e28cbb63a58491b56d536cb506e93
3486e28 (origin/master) HEAD@{8}: pull: Fast-forward
d9f1aaa HEAD@{9}: commit: add pip requirements file
64345e4 (second/master) HEAD@{10}: commit (merge): Merge branch 'master' of https://github.com/brain-hack/learnGit2
cdafae1 (HEAD -> master) HEAD@{11}: pull: Merge made by the 'recursive' strategy.
b990c79 HEAD@{12}: commit: update readme and add gitignore
abd1827 HEAD@{13}: commit (amend): Initial commit
4521575 HEAD@{14}: commit (initial): Initial commit
VanDarkHolme@DESKTOP-TEN3I8O MINGW64 ~/learnGit (master)
$ git reset --hard HEAD@{6}
HEAD is now at 3486e28 add func print_git
VanDarkHolme@DESKTOP-TEN3I8O MINGW64 ~/learnGit (master)
$ git log --pretty=oneline
3486e281c7a066d7b769f0c7d62d539afb8ca462 (HEAD -> master, origin/master) add func print_git
d9f1aaa38738b10336b3ade45abb145e67f74c90 add pip requirements file
64345e4765a6bdfc1fc952bfe069c47b3e78d87e (second/master) Merge branch 'master' of https://github.com/brain-hack/learnGit2
cdafae15496e28cbb63a58491b56d536cb506e93 Merge branch 'master' of https://github.com/brain-hack/learnGit1
b990c79dfa3eaa956207e365c22c46b1344091ea update readme and add gitignore
60790065d89a06a27a53f8bce2dd004a80b10239 add python code : hello world
752cb4c9f007e8cc5494f33b20a46a3abf1f2d9c Initial commit
abd182712e7b732882d6f0b6b80a362b34d63613 Initial commit

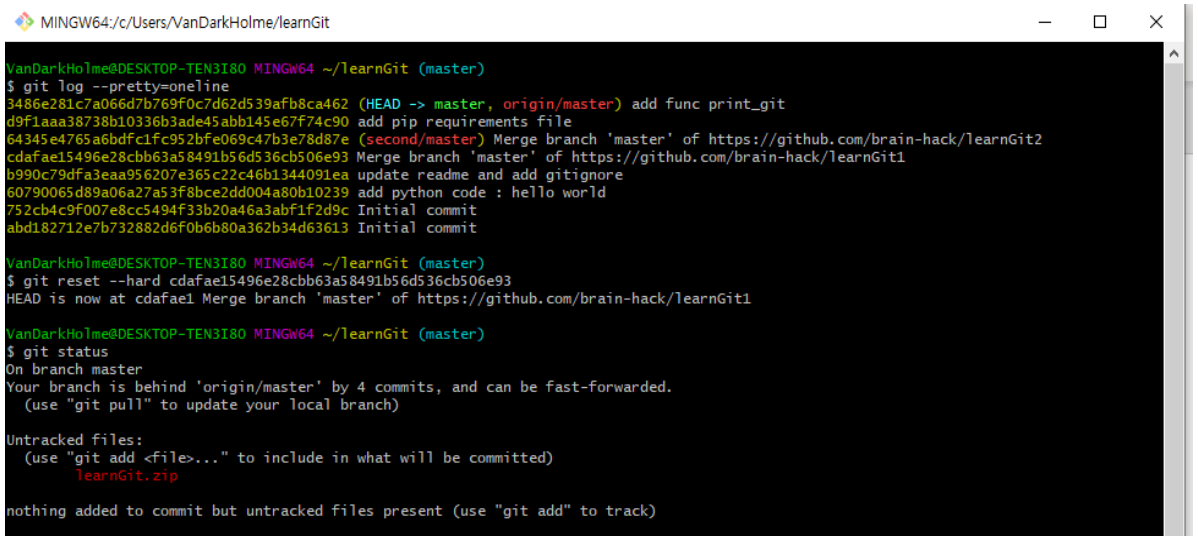
커밋 되돌리기 reset --mixed

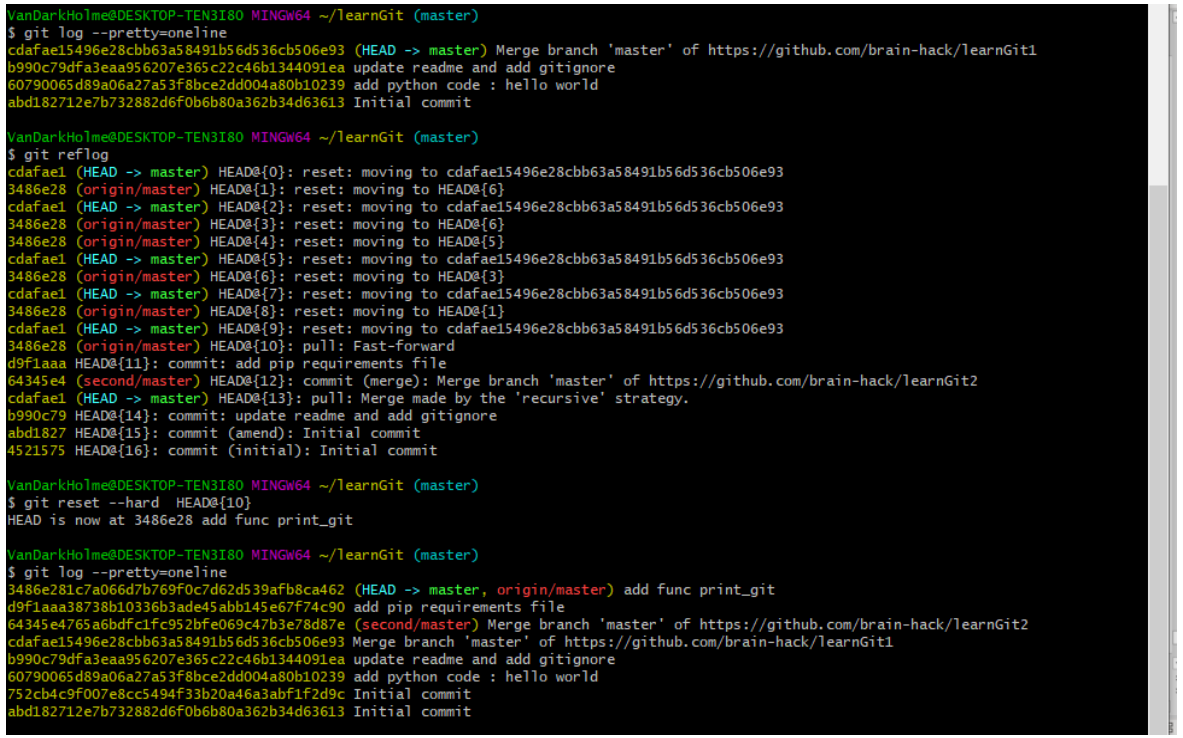


reset 취소


MINGW64:/c/Users/VanDarkHolme/learnGit
VanDarkHolme@DESKTOP-TEN3I8O MINGW64 ~/learnGit (master)
$ git reflog
cdafae1 (HEAD -> master) HEAD@{0}: reset: moving to cdafae15496e28cbb63a58491b56d536cb506e93
3486e28 (origin/master) HEAD@{1}: reset: moving to HEAD@{6}
3486e28 (origin/master) HEAD@{2}: reset: moving to HEAD@{5}
cdafae1 (HEAD -> master) HEAD@{3}: reset: moving to cdafae15496e28cbb63a58491b56d536cb506e93
3486e28 (origin/master) HEAD@{4}: reset: moving to HEAD@{3}
cdafae1 (HEAD -> master) HEAD@{5}: reset: moving to cdafae15496e28cbb63a58491b56d536cb506e93
3486e28 (origin/master) HEAD@{6}: reset: moving to HEAD@{1}
cdafae1 (HEAD -> master) HEAD@{7}: reset: moving to cdafae15496e28cbb63a58491b56d536cb506e93
3486e28 (origin/master) HEAD@{8}: pull: Fast-forward
d9f1aaa HEAD@{9}: commit: add pip requirements file
64345e4 (second/master) HEAD@{10}: commit (merge): Merge branch 'master' of https://github.com/brain-hack/learnGit2
cdafae1 (HEAD -> master) HEAD@{11}: pull: Merge made by the 'recursive' strategy.
b990c79 HEAD@{12}: commit: update readme and add gitignore
abd1827 HEAD@{13}: commit (amend): Initial commit
4521575 HEAD@{14}: commit (initial): Initial commit
VanDarkHolme@DESKTOP-TEN3I8O MINGW64 ~/learnGit (master)
$ git reset --hard HEAD@{6}
HEAD is now at 3486e28 add func print_git
VanDarkHolme@DESKTOP-TEN3I8O MINGW64 ~/learnGit (master)
$ git log --pretty=oneline
3486e281c7a066d7b769f0c7d62d539afb8ca462 (HEAD -> master, origin/master) add func print_git
d9f1aaa38738b10336b3ade45abb145e67f74c90 add pip requirements file
64345e4765a6bdfc1fc952bfe069c47b3e78d87e (second/master) Merge branch 'master' of https://github.com/brain-hack/learnGit2
cdafae15496e28cbb63a58491b56d536cb506e93 Merge branch 'master' of https://github.com/brain-hack/learnGit1
b990c79dfa3eaa956207e365c22c46b1344091ea update readme and add gitignore
60790065d89a06a27a53f8bce2dd004a80b10239 add python code : hello world
752cb4c9f007e8cc5494f33b20a46a3abf1f2d9c Initial commit
abd182712e7b732882d6f0b6b80a362b34d63613 Initial commit

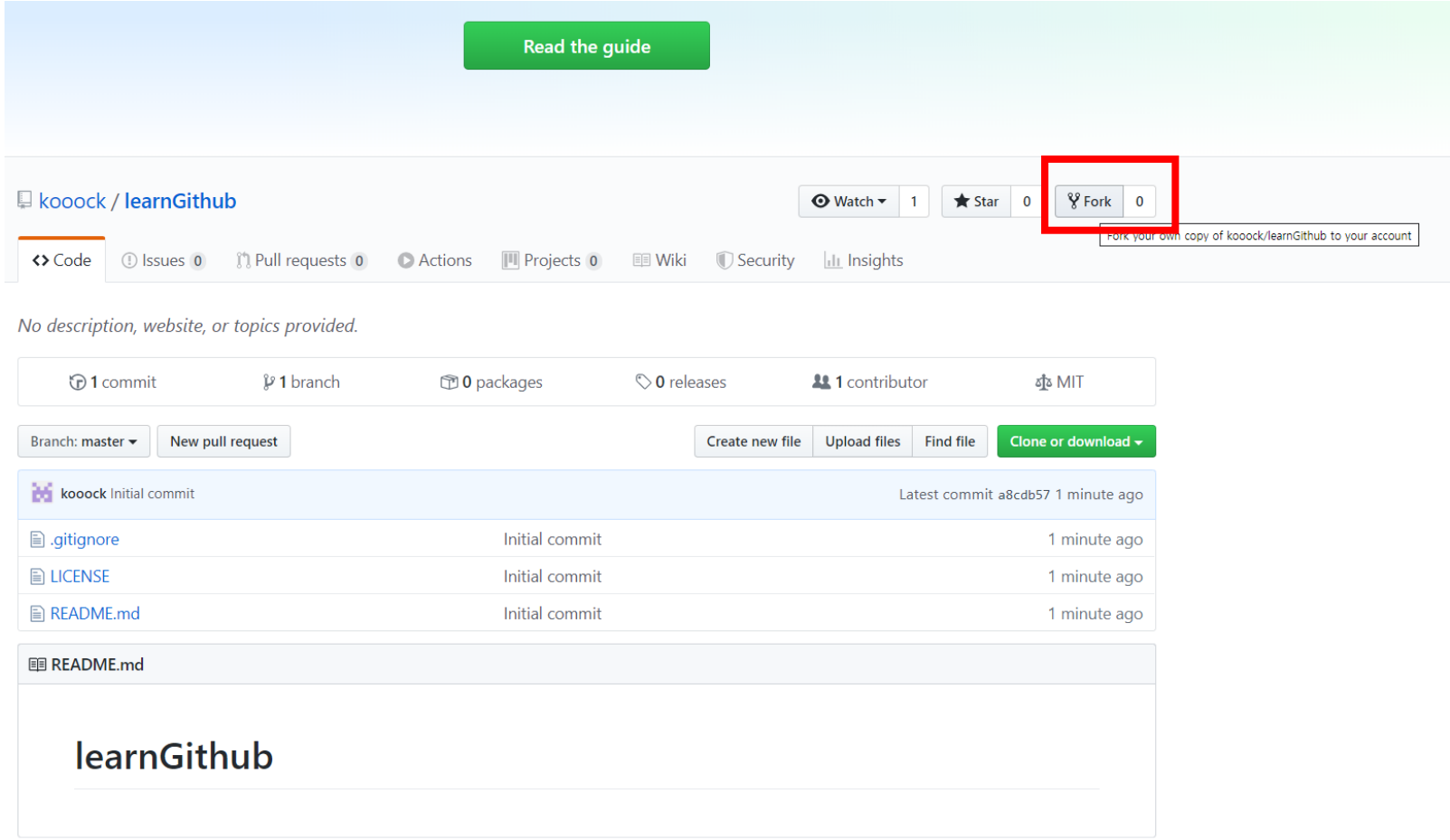
사실 reset hard도 복구가 가능하다



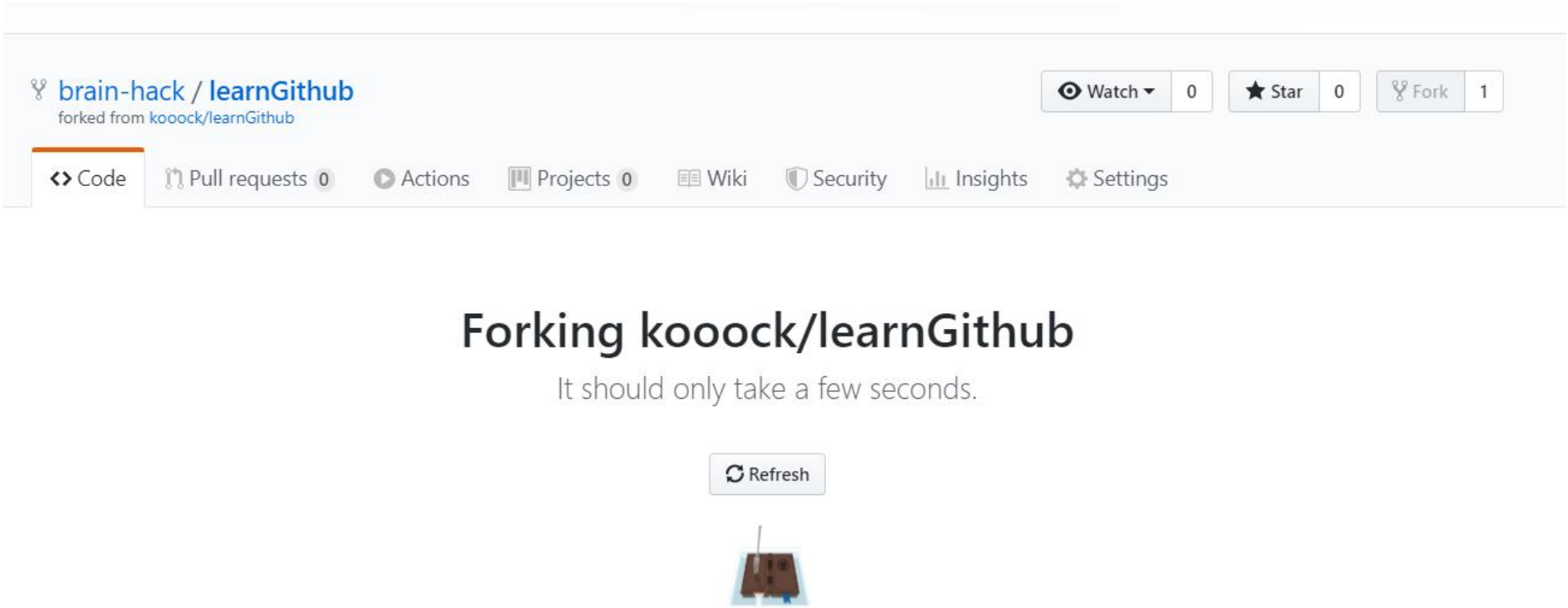


fork를 해봅시다

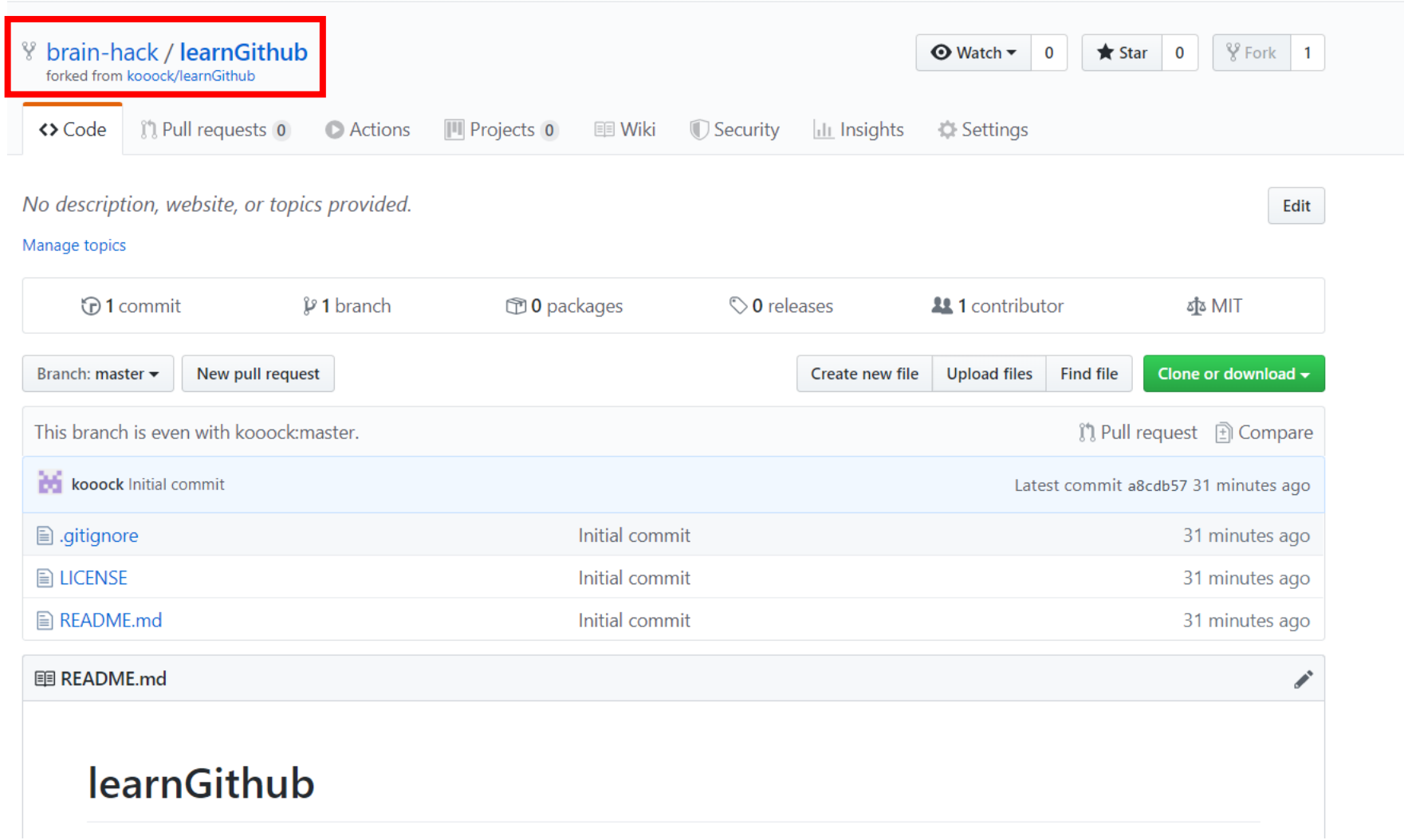
https://github.com/kooock/learnGithub



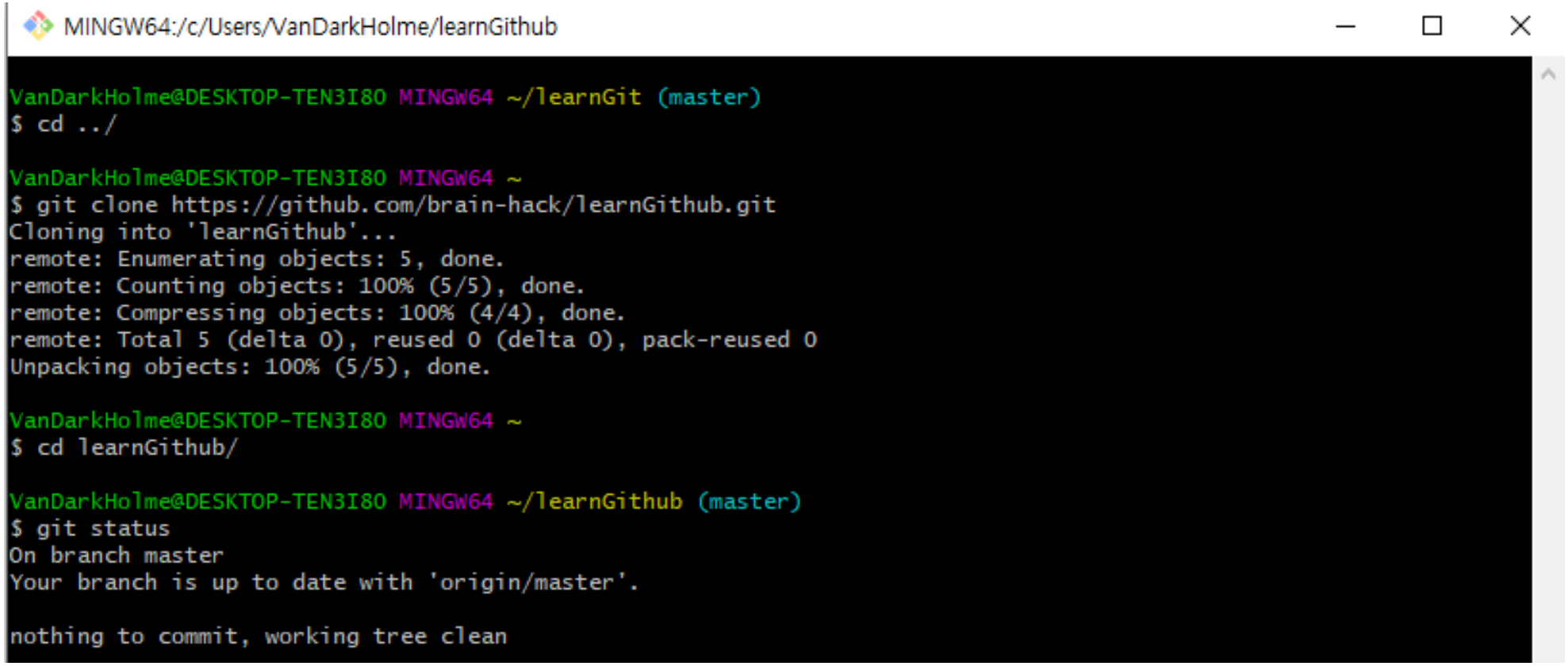
fork를 해봅시다


brain-hack / learnGithub
forked from kooock/learnGithub
Watch 0
Star 0
Fork 1
Code
Pull requests 0
Actions
Projects 0
Wiki
Security
Insights
Settings
Forking kooock/learnGithub
It should only take a few seconds.
Refresh

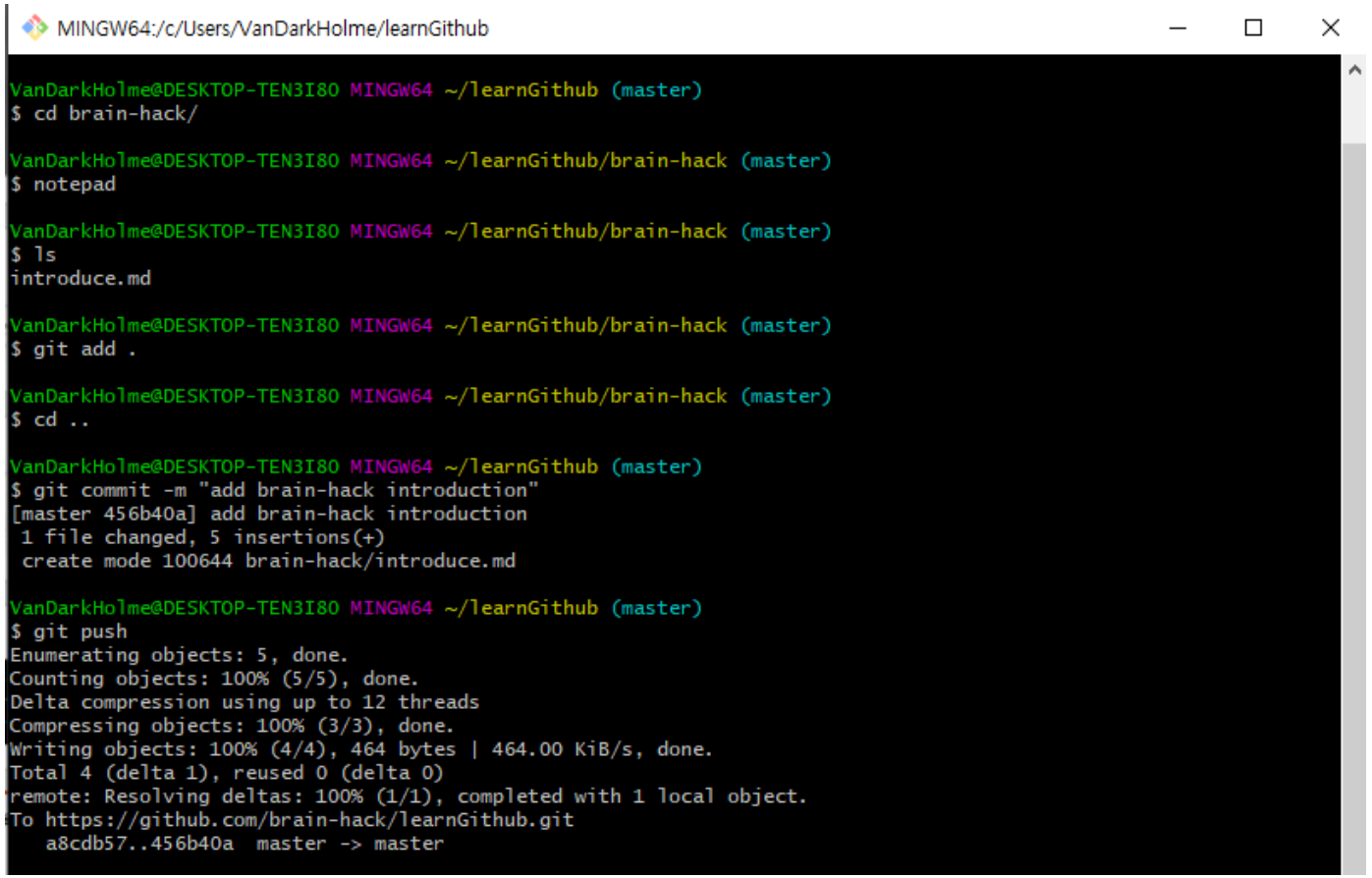
fork를 해봅시다

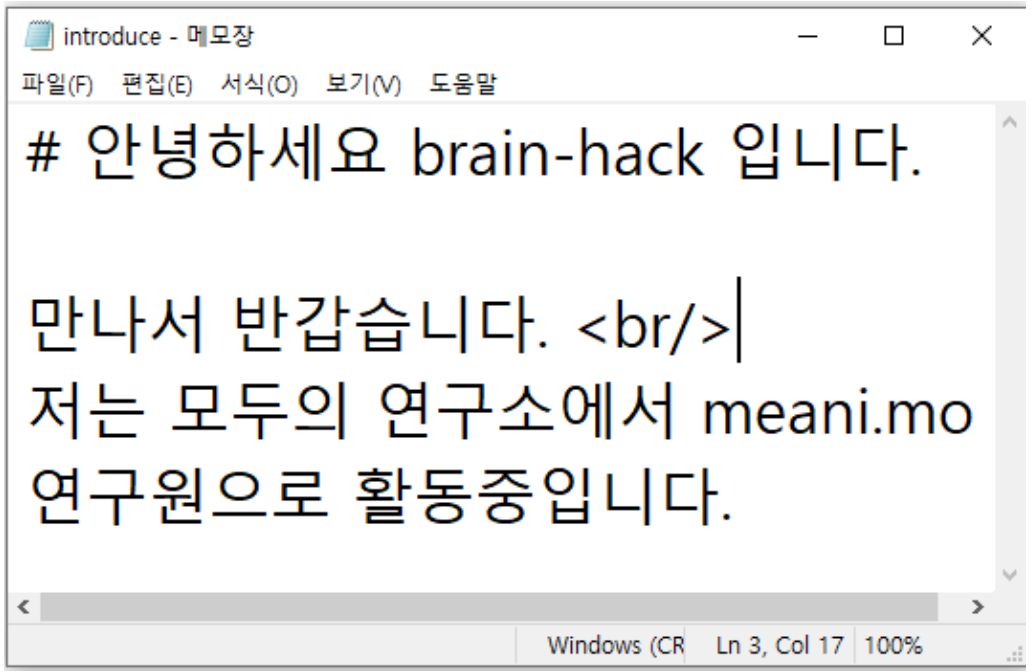

brain-hack / learnGithub
forked from kooock/learnGithub
Watch 0
Star 0
Fork 1
Code
Pull requests 0
Actions
Projects 0
Wiki
Security
Insights
Settings
No description, website, or topics provided.
Edit
Manage topics
1 commit
1 branch
0 packages
0 releases
1 contributor
MIT
Branch: master
New pull request
Create new file
Upload files
Find file
Clone or download
This branch is even with kooock:master.
Pull request
Compare
kooock Initial commit
Latest commit a8cdb57 31 minutes ago
.gitignore
Initial commit
31 minutes ago
LICENSE
Initial commit
31 minutes ago
README.md
Initial commit
31 minutes ago
README.md
learnGithub

clone을 해봅시다

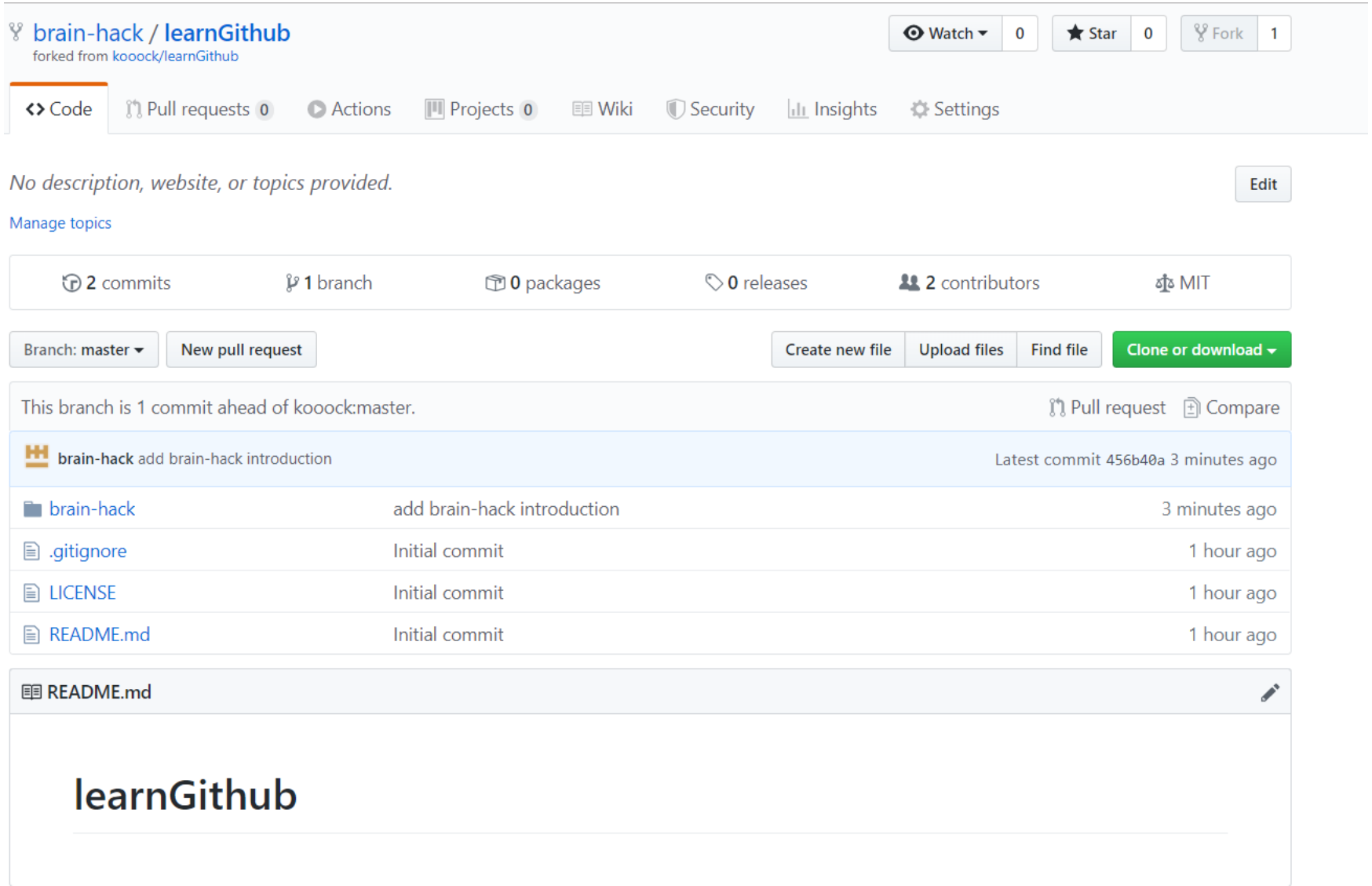

MINGW64:/c/Users/VanDarkHolme/learnGithub
VanDarkHolme@DESKTOP-TEN3I8O MINGW64 ~/learnGit (master)
$ cd ../
VanDarkHolme@DESKTOP-TEN3I8O MINGW64 ~
$ git clone https://github.com/brain-hack/learnGithub.git
Cloning into 'learnGithub'...
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (5/5), done.
VanDarkHolme@DESKTOP-TEN3I8O MINGW64 ~
$ cd learnGithub/
VanDarkHolme@DESKTOP-TEN3I8O MINGW64 ~/learnGithub (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.
nothing to commit, working tree clean

local repo에서 작업을 해볼까요?

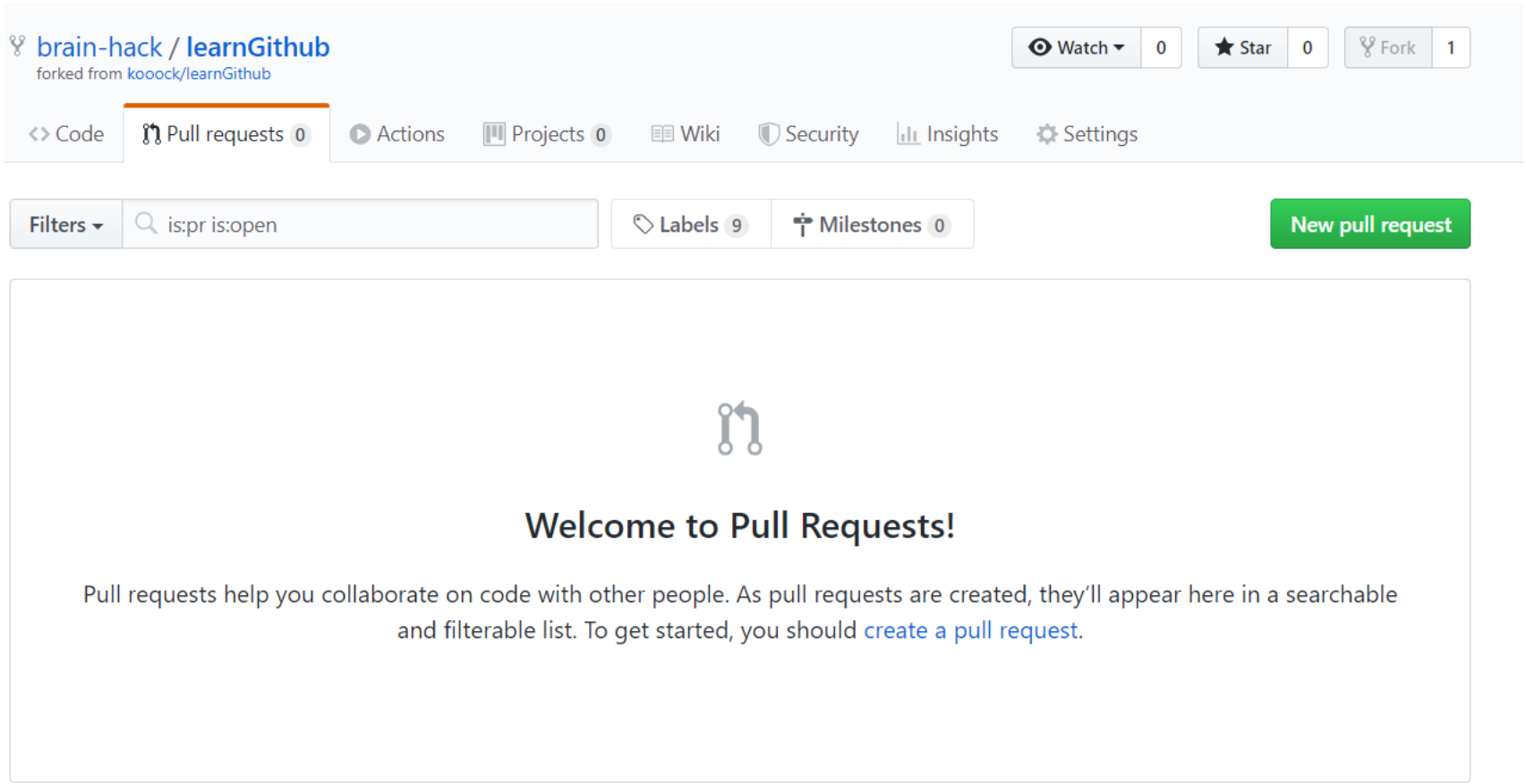




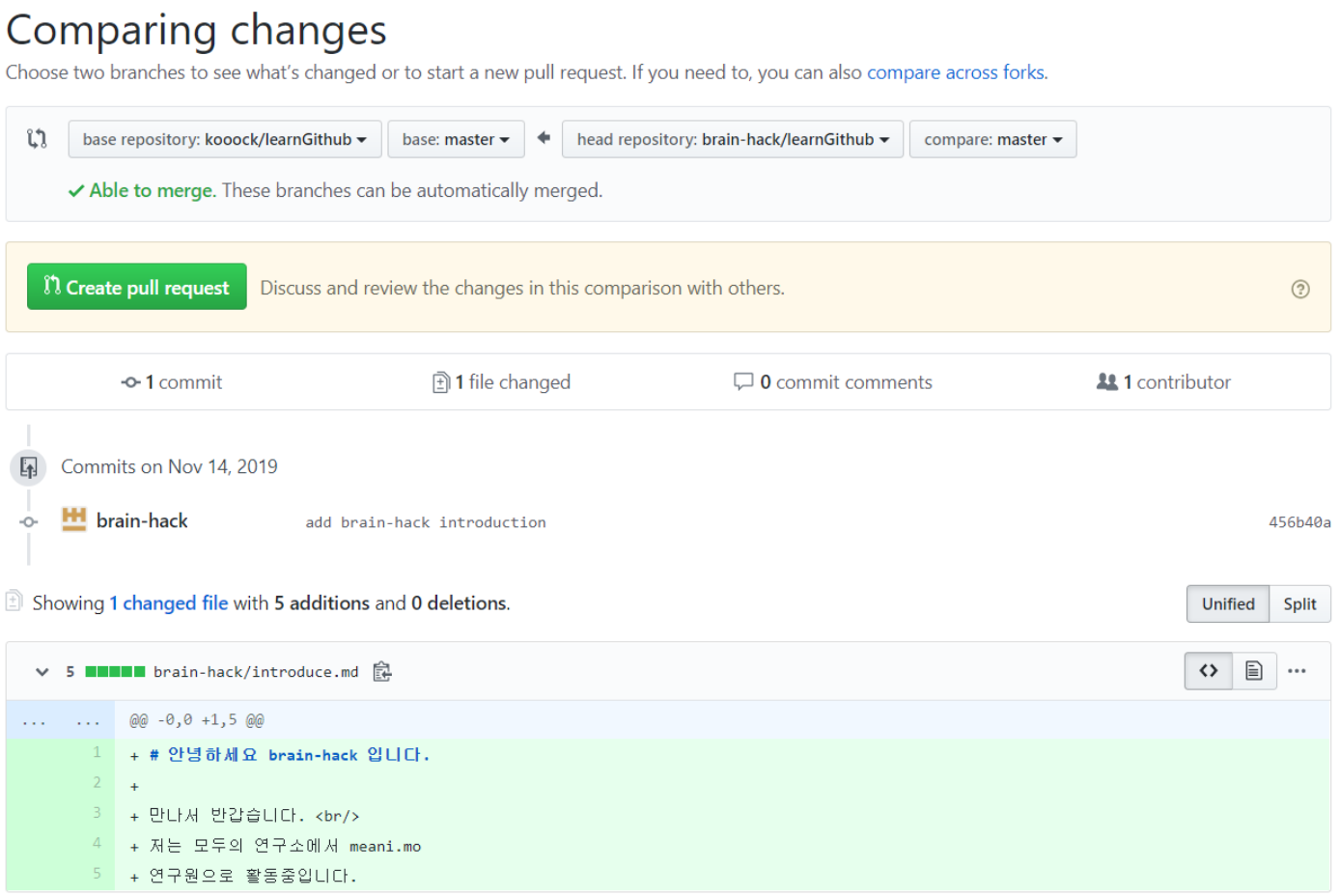
remote repository 확인



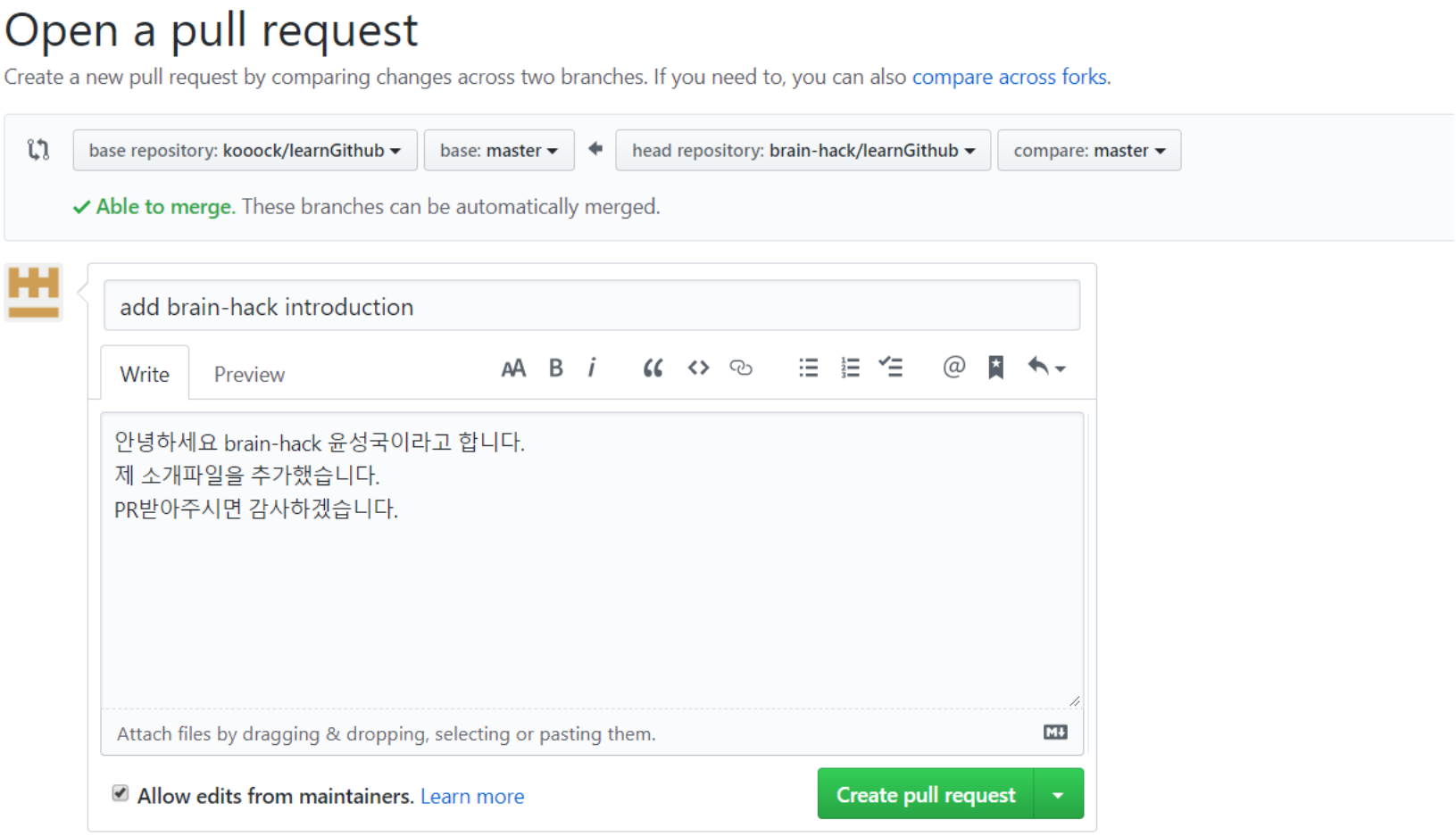
pull request 만들기



pull request 만들기



pull request 만들기



pull request 확인