



# Git

버전이 되기까지 거쳐가는 세 개의 로컬 공간



모든 변경사항들이 담기는 공간    변경 사항들 중 후보들이 올라가는 공간

## 명령어

clear

xx폴더 만들기 - mkdir xx

폴더 지우기 - rm xx

## 사용자 정보 설정

\$ git config --global user.name "본인이름영어로"

\$ git config --global user.email 본인이메일

git remote add origin 깃주소

## 사용자 정보 설정 확인

\$ git config --list

git init :깃 시작

git status : 현재 깃의 상태

git add 파일명 : 해당 파일을 staging area에 추가 → 파일 내용이 수정 되면 add 다시 해줘야 됨

git rm --cached 파일명.확장자 : 해당 파일을 staging area에 삭제

git add . : 파일을 전부다 추가

git commit -m "메시지" : 깃을 레파지토리에 저장

git log : 커밋 이후 변경 로그를 확인

git commit -am "메시지" : 추가와 커밋 을 동시에 한다

## 깃을 원격저장소에 올리는 작업

git remote add origin 본인의 깃 계정 : 원격저장소를 추가한다

git push origin master : 원격 저장소에 저장한다

## 깃 조회, 되돌리기

git reset --hard HEAD^ : 커밋을 한단계 앞으로 되돌린다(모두 초기화)

git reset --mixed HEAD^ : 커밋을 한단계 앞으로 되돌린다(working directory는 유지)

git reset --soft HEAD^ : 커밋을 한단계 앞으로 되돌린다(working directory , staging area를 유지)

## 브랜치



브랜치를 나누기 전에 최소한 한번이라도 commit이 이루어 져야함(프로젝트 초기세팅)  
깃 나누기 branch

git branch 이름	:브랜치 생성
git branch	:브랜치 확인
git checkout 이름	:브랜치 변경
git branch -D 이름	:브랜치 삭제
git merge 합치는브랜치명	:병합

```

yopy0@DESKTOP-3KT2T40 MINGW64 ~/Desktop/couse/github/example5 (master)
$ git branch
A
* master

yopy0@DESKTOP-3KT2T40 MINGW64 ~/Desktop/couse/github/example5 (master)
$ git branch B

yopy0@DESKTOP-3KT2T40 MINGW64 ~/Desktop/couse/github/example5 (master)
$ git checkout A
Switched to branch 'A'

yopy0@DESKTOP-3KT2T40 MINGW64 ~/Desktop/couse/github/example5 (A)
$ git add .

yopy0@DESKTOP-3KT2T40 MINGW64 ~/Desktop/couse/github/example5 (A)
$ git commit -m "A seconde commit"
[A bb7c68a] A seconde commit
1 file changed, 1 insertion(+)
create mode 100644 AA.txt

yopy0@DESKTOP-3KT2T40 MINGW64 ~/Desktop/couse/github/example5 (A)
$
  
```

```
taeju@DESKTOP-A6CRG0O MINGW64 ~/github/example6 (master)
$ git init
Initialized empty Git repository in C:/Users/taeju/github/example6/.git/
taeju@DESKTOP-A6CRG0O MINGW64 ~/github/example6 (master)
$ git add .

taeju@DESKTOP-A6CRG0O MINGW64 ~/github/example6 (master)
$ git commit -m "first commit"
[master (root-commit) fbd6b39] first commit
1 file changed, 1 insertion(+)
create mode 100644 first.txt

taeju@DESKTOP-A6CRG0O MINGW64 ~/github/example6 (master)
$ git branch A

taeju@DESKTOP-A6CRG0O MINGW64 ~/github/example6 (master)
$ git branch B

taeju@DESKTOP-A6CRG0O MINGW64 ~/github/example6 (master)
$ git checkout A
Switched to branch 'A'

taeju@DESKTOP-A6CRG0O MINGW64 ~/github/example6 (A)
$ notepad

taeju@DESKTOP-A6CRG0O MINGW64 ~/github/example6 (A)
$ notepad a

taeju@DESKTOP-A6CRG0O MINGW64 ~/github/example6 (A)
$

taeju@DESKTOP-A6CRG0O MINGW64 ~/github/example6 (A)
$ git add .

taeju@DESKTOP-A6CRG0O MINGW64 ~/github/example6 (A)
$ git commit -m "A commit"
[A 4aa63f6] A commit
1 file changed, 1 insertion(+)
create mode 100644 a

taeju@DESKTOP-A6CRG0O MINGW64 ~/github/example6 (A)
$ git checkout B
Switched to branch 'B'
```

```
taeju@DESKTOP-A6CRG0O MINGW64 ~/github/example6 (B)
$ notepad b
git add .
git
taeju@DESKTOP-A6CRG0O MINGW64 ~/github/example6 (B)
$ git add .

taeju@DESKTOP-A6CRG0O MINGW64 ~/github/example6 (B)
$ git commit -m "B commit"
[B 3ed0eff] B commit
1 file changed, 1 insertion(+)
create mode 100644 b

taeju@DESKTOP-A6CRG0O MINGW64 ~/github/example6 (B)
$ git status
On branch B
Changes not staged for commit:
(use "git add <file>..." to update what will be committed)
(use "git restore <file>..." to discard changes in working directory)
modified:   b

no changes added to commit (use "git add" and/or "git commit -a")

taeju@DESKTOP-A6CRG0O MINGW64 ~/github/example6 (B)
$ git add .

taeju@DESKTOP-A6CRG0O MINGW64 ~/github/example6 (B)
$ git commit -m "B second commit"
[B 91f4f25] B second commit
1 file changed, 3 insertions(+), 1 deletion(-)

taeju@DESKTOP-A6CRG0O MINGW64 ~/github/example6 (B)
$ git log
commit 91f4f25c20558063814ad9586cd7bbc4550a4b96 (HEAD -> B)
Author: gkekf123 <taeju95@gmail.com>
Date: Thu May 11 11:40:51 2023 +0900
```

```
B second commit
```

```
commit 3ed0effa5c8c14d501290dff2ee3e28c64181456
Author: gkekf123 <taeju95@gmail.com>
Date: Thu May 11 11:40:18 2023 +0900
```

B commit

commit fbd6b391a2c75bdbd25e1d688caec3dfa259dedd (master)

Author: gkekf123 [taeju95@gmail.com](mailto:taeju95@gmail.com)

Date: Thu May 11 11:38:02 2023 +0900

first commit

taeju@DESKTOP-A6CRG00 MINGW64 ~/github/example6 (B)

\$ git checkout master

Switched to branch 'master'

taeju@DESKTOP-A6CRG00 MINGW64 ~/github/example6 (master)

\$ git merge A

Updating fbd6b39..4aa63f6

Fast-forward

a | 1 +

1 file changed, 1 insertion(+)

create mode 100644 a

taeju@DESKTOP-A6CRG00 MINGW64 ~/github/example6 (master)

\$ git loig

git: 'loig' is not a git command. See 'git --help'.

The most similar command is

log

taeju@DESKTOP-A6CRG00 MINGW64 ~/github/example6 (master)

\$ git log

commit 4aa63f6cdccf2e57c24237915c99354379a4ba61 (HEAD -> master, A)

Author: gkekf123 [taeju95@gmail.com](mailto:taeju95@gmail.com)

Date: Thu May 11 11:39:20 2023 +0900

A commit

commit fbd6b391a2c75bdbd25e1d688caec3dfa259dedd

Author: gkekf123 [taeju95@gmail.com](mailto:taeju95@gmail.com)

Date: Thu May 11 11:38:02 2023 +0900

```
first commit
```

```
taeju@DESKTOP-A6CRG0O MINGW64 ~/github/example6 (master)
```

```
$ git merge B
```

```
Merge made by the 'ort' strategy.
```

```
b | 3 +++
```

```
1 file changed, 3 insertions(+)
```

```
create mode 100644 b
```

```
taeju@DESKTOP-A6CRG0O MINGW64 ~/github/example6 (master)
```

```
$ git log
```

```
commit 95f522a7bd245f02edcffc99d6b6cc7572fda6eb (HEAD -> master)
```

```
Merge: 4aa63f6 91f4f25
```

```
Author: gkekf123 taeju95@gmail.com
```

```
Date: Thu May 11 11:42:37 2023 +0900
```

```
Merge branch 'B'
```

```
commit 91f4f25c20558063814ad9586cd7bbc4550a4b96 (B)
```

```
Author: gkekf123 taeju95@gmail.com
```

```
Date: Thu May 11 11:40:51 2023 +0900
```

```
B second commit
```

```
commit 3ed0effa5c8c14d501290dff2ee3e28c64181456
```

```
Author: gkekf123 taeju95@gmail.com
```

```
Date: Thu May 11 11:40:18 2023 +0900
```

```
B commit
```

```
commit 4aa63f6cdccf2e57c24237915c99354379a4ba61 (A)
```

```
Author: gkekf123 taeju95@gmail.com
```

```
Date: Thu May 11 11:39:20 2023 +0900
```

```
A commit
```

commit fbd6b391a2c75bdbd25e1d688caec3dfa259dedd

Author: gkekf123 [taeju95@gmail.com](mailto:taeju95@gmail.com)

Date: Thu May 11 11:38:02 2023 +0900

```
first commit
```

(END)

commit 91f4f25c20558063814ad9586cd7bbc4550a4b96 (B)

Author: gkekf123 [taeju95@gmail.com](mailto:taeju95@gmail.com)

Date: Thu May 11 11:40:51 2023 +0900

```
B second commit
```

commit 3ed0effa5c8c14d501290dff2ee3e28c64181456

Author: gkekf123 [taeju95@gmail.com](mailto:taeju95@gmail.com)

Date: Thu May 11 11:40:18 2023 +0900

```
B commit
```

commit 4aa63f6cdccf2e57c24237915c99354379a4ba61 (A)

Author: gkekf123 [taeju95@gmail.com](mailto:taeju95@gmail.com)

Date: Thu May 11 11:39:20 2023 +0900

```
A commit
```

commit fbd6b391a2c75bdbd25e1d688caec3dfa259dedd

Author: gkekf123 [taeju95@gmail.com](mailto:taeju95@gmail.com)

Date: Thu May 11 11:38:02 2023 +0900

```
first commit
```

~

~

(END)

commit 95f522a7bd245f02edcffc99d6b6cc7572fda6eb (HEAD -> master)

Merge: 4aa63f6 91f4f25



Author: gkekf123 [taeju95@gmail.com](mailto:taeju95@gmail.com)  
Date: Thu May 11 11:42:37 2023 +0900

Merge branch 'B'

commit 91f4f25c20558063814ad9586cd7bbc4550a4b96 (B)  
Author: gkekf123 [taeju95@gmail.com](mailto:taeju95@gmail.com)  
Date: Thu May 11 11:40:51 2023 +0900

B second commit

commit 3ed0effa5c8c14d501290dff2ee3e28c64181456  
Author: gkekf123 [taeju95@gmail.com](mailto:taeju95@gmail.com)  
Date: Thu May 11 11:40:18 2023 +0900

B commit

commit 4aa63f6cdccf2e57c24237915c99354379a4ba61 (A)  
Author: gkekf123 [taeju95@gmail.com](mailto:taeju95@gmail.com)  
Date: Thu May 11 11:39:20 2023 +0900

A commit

commit fbd6b391a2c75bdbd25e1d688caec3dfa259dedd  
Author: gkekf123 [taeju95@gmail.com](mailto:taeju95@gmail.com)  
Date: Thu May 11 11:38:02 2023 +0900

first commit

~

taeju@DESKTOP-A6CRG00 MINGW64 ~/github/example6 (master)  
\$ git branch -D A  
Deleted branch A (was 4aa63f6).  
  
taeju@DESKTOP-A6CRG00 MINGW64 ~/github/example6 (master)  
\$ git branch -D B

Deleted branch B (was 91f4f25).

taeju@DESKTOP-A6CRG0O MINGW64 ~/github/example6 (master)

\$ git log

commit 95f522a7bd245f02edcffc99d6b6cc7572fda6eb (HEAD -> master)

Merge: 4aa63f6 91f4f25

Author: gkekf123 [taeju95@gmail.com](mailto:taeju95@gmail.com)

Date: Thu May 11 11:42:37 2023 +0900

Merge branch 'B'

commit 91f4f25c20558063814ad9586cd7bbc4550a4b96

Author: gkekf123 [taeju95@gmail.com](mailto:taeju95@gmail.com)

Date: Thu May 11 11:40:51 2023 +0900

B second commit

commit 3ed0effa5c8c14d501290dff2ee3e28c64181456

Author: gkekf123 [taeju95@gmail.com](mailto:taeju95@gmail.com)

Date: Thu May 11 11:40:18 2023 +0900

B commit

commit 4aa63f6cdccf2e57c24237915c99354379a4ba61

Author: gkekf123 [taeju95@gmail.com](mailto:taeju95@gmail.com)

Date: Thu May 11 11:39:20 2023 +0900

A commit

commit fbd6b391a2c75bdbd25e1d688caec3dfa259dedd

Author: gkekf123 [taeju95@gmail.com](mailto:taeju95@gmail.com)

Date: Thu May 11 11:38:02 2023 +0900

first commit

```
taeju@DESKTOP-A6CRG0O MINGW64 ~/github/example6 (master)
$
```

## 원격 저장소와 상호작용

1. 원격저장소에 조회(추가)하기  
\$ git remote add <단축이름> 주소
2. 원격저장소에 조회(추가)하기  
(나의 레파지토리의 master브랜치를 origin의 master브랜치로 push해라)  
\$ git push origin master
3. 원격저장소에서 얻어와서 합치기  
(origin을 나의 레파지토리 master브랜치를 갖고와라)  
\$ git pull (origin master)
4. 원격 저장소에서 얻기  
\$ git fetch (origin master)
5. 원격 저장소에서 복사하기  
\$ git clone 주소

pull vs fetch 차이

pull = fetch + merge 기능

git remote -v :깃의 현재 원격 저장소를 확인한다  
git remote remove 단축이름:깃의 현재 원격 저장소를  
삭제한다

## 실습

```
taeju@DESKTOP-A6CRG0O MINGW64 ~/github/DemoProject (master)
$ git init
Initialized empty Git repository in C:/Users/taeju/github/DemoProject/.git/
taeju@DESKTOP-A6CRG0O MINGW64 ~/github/DemoProject (master)
$ git add .
```

```
taeju@DESKTOP-A6CRG0O MINGW64 ~/github/DemoProject (master)
$ git commit -m "first commit"
[master (root-commit) 516d97c] first commit
1 file changed, 1 insertion(+)
create mode 100644 first

taeju@DESKTOP-A6CRG0O MINGW64 ~/github/DemoProject (master)
$ git log
commit 516d97c2a48ed8ce15425f283bc34bf3f6cf0b20 (HEAD -> master)
Author: gkekf123 <taeju95@gmail.com>
Date: Thu May 11 12:20:22 2023 +0900
```

```
first commit
```

```
taeju@DESKTOP-A6CRG0O MINGW64 ~/github/DemoProject (master)
$ git remote add xxx https://github.com/gkekf123/DemoProject.git

taeju@DESKTOP-A6CRG0O MINGW64 ~/github/DemoProject (master)
$ git remote -v
xxx https://github.com/gkekf123/DemoProject.git (fetch)
xxx https://github.com/gkekf123/DemoProject.git (push)

taeju@DESKTOP-A6CRG0O MINGW64 ~/github/DemoProject (master)
$ git remote remove
usage: git remote remove <name>

taeju@DESKTOP-A6CRG0O MINGW64 ~/github/DemoProject (master)
$ git remote remove xxx

taeju@DESKTOP-A6CRG0O MINGW64 ~/github/DemoProject (master)
$ git remote -v

taeju@DESKTOP-A6CRG0O MINGW64 ~/github/DemoProject (master)
$
```

## 1. 원격저장소에 조회(추가)하기

**\$ git remote add <단축이름> 주소**

```
taeju@DESKTOP-A6CRG0O MINGW64 ~/github/DemoProject (master)
$ git remote add origin https://github.com/gkekf123/DemoProject.git
```

```
taeju@DESKTOP-A6CRG0O MINGW64 ~/github/DemoProject (master)
$ git remote -v
origin https://github.com/gkekf123/DemoProject.git (fetch)
origin https://github.com/gkekf123/DemoProject.git (push)
taeju@DESKTOP-A6CRG0O MINGW64 ~/github/DemoProject (master)
```

## 2. 원격저장소에 조회(추가)하기

(나의 레파지토리의 master브랜치를 origin의 master브랜치로 push해라)

**\$ git push origin master**

\$ git push origin master

Enumerating objects: 3, done.

Counting objects: 100% (3/3), done.

Writing objects: 100% (3/3), 234 bytes | 234.00 KiB/s, done.

Total 3 (delta 0), reused 0 (delta 0), pack-reused 0

To <https://github.com/gkekf123/DemoProject.git>

- [new branch] master -> master

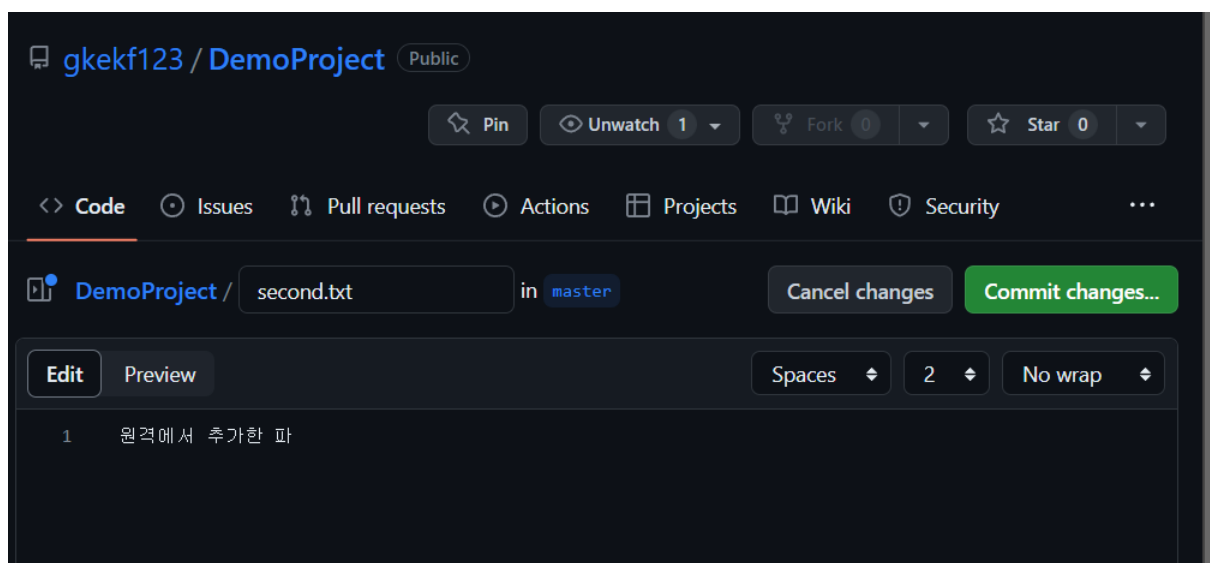
taeju@DESKTOP-A6CRG0O MINGW64 ~/github/DemoProject (master)

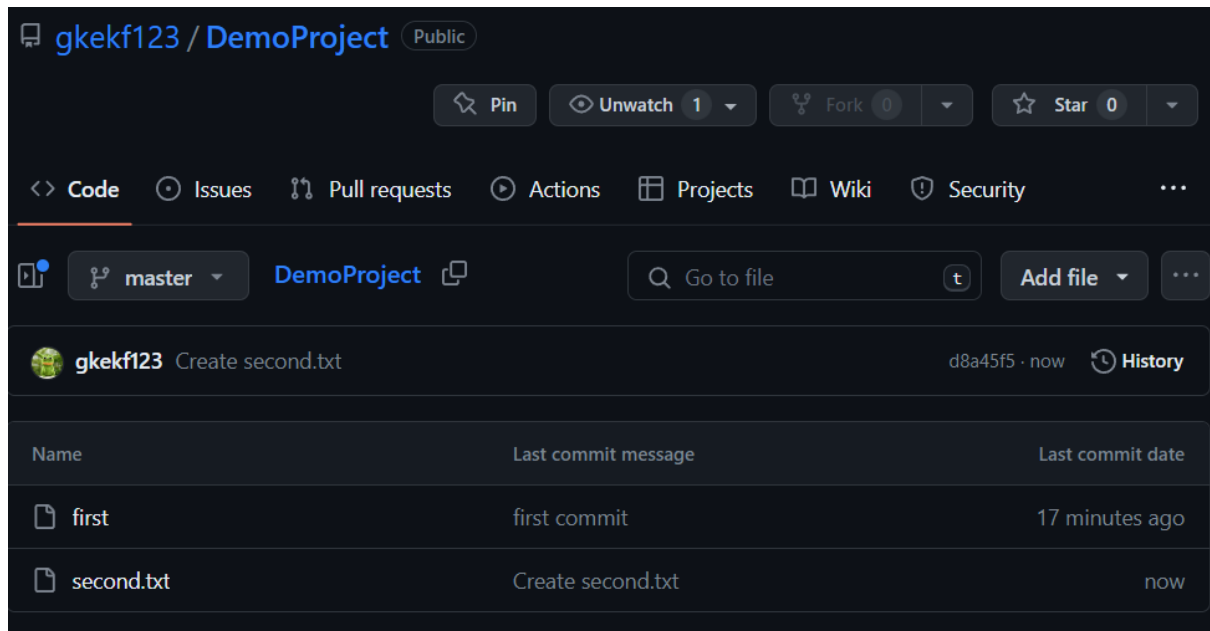
## 3. 원격저장소에서 얻어와서 합치기

(origin을 나의 레파지토리 master브랜치를 갖고와라)

**\$ git pull (origin master)**

깃헙에서 크리에이트 뉴 파일 생성 후 커밋





- 로컬과 원격이 달라짐 로컬에는 first만 있음 → pull 해줘야 된다

taeju@DESKTOP-A6CRG0O MINGW64 ~/github/DemoProject (master)

\$ git pull origin master

remote: Enumerating objects: 4, done.

remote: Counting objects: 100% (4/4), done.

remote: Compressing objects: 100% (2/2), done.

remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0

Unpacking objects: 100% (3/3), 685 bytes | 57.00 KiB/s, done.

From <https://github.com/gkekf123/DemoProject>

branch master -> FETCH\_HEAD

516d97c..d8a45f5 master -> origin/master

Updating 516d97c..d8a45f5

Fast-forward

second.txt | 1 +

1 file changed, 1 insertion(+)

create mode 100644 second.txt

4. 원격 저장소에서 얻기 - fetch로임시적으로 가져옴(수동으로 merge해야 된다)

\$ git fetch (origin master)

taeju@DESKTOP-A6CRG0O MINGW64 ~/github/DemoProject (master)

\$ git fetch origin master

remote: Enumerating objects: 5, done.  
remote: Counting objects: 100% (5/5), done.  
remote: Compressing objects: 100% (3/3), done.  
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0  
Unpacking objects: 100% (3/3), 720 bytes | 55.00 KiB/s, done.  
From <https://github.com/gkekf123/DemoProject>

branch master -> FETCH\_HEAD

d8a45f5..c1aedfd master -> origin/master

taeju@DESKTOP-A6CRG0O MINGW64 ~/github/DemoProject (master)

\$ git checkout FETCH\_HEAD

Note: switching to 'FETCH\_HEAD'.

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 switching back to a branch.

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

git switch -c <new-branch-name>

Or undo this operation with:

git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at c1aedfd Update second.txt

→ second.txt 파일에 수정 내용이 들어간다

taeju@DESKTOP-A6CRG0O MINGW64 ~/github/DemoProject ((c1aedfd...))

\$ git checkout master

Previous HEAD position was c1aedfd Update second.txt

Switched to branch 'master'

→ second.txt 파일 원복

taeju@DESKTOP-A6CRG0O MINGW64 ~/github/DemoProject (master)

\$ git checkout FETCH\_HEAD

Note: switching to 'FETCH\_HEAD'.

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 switching back to a branch.

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

```
git switch -c <new-branch-name>
```

Or undo this operation with:

```
git switch -
```

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at c1aedfd Update second.txt

→ 새로운 branch를 만들어야 된다

```
taeju@DESKTOP-A6CRG0O MINGW64 ~/github/DemoProject ((c1aedfd...))
```

```
$ git switch -c test
```

Switched to a new branch 'test'

```
taeju@DESKTOP-A6CRG0O MINGW64 ~/github/DemoProject (test)
```

→ 임시 branch

```
taeju@DESKTOP-A6CRG0O MINGW64 ~/github/DemoProject (test)
```

```
$ git checkout master
```

Switched to branch 'master'

```
taeju@DESKTOP-A6CRG0O MINGW64 ~/github/DemoProject (master)
```

```
$ git merge test
```

Updating d8a45f5..c1aedfd

Fast-forward

second.txt | 4 +++-

1 file changed, 3 insertions(+), 1 deletion(-)

```
taeju@DESKTOP-A6CRG0O MINGW64 ~/github/DemoProject (master)
```

→ master로 합친다

```
taeju@DESKTOP-A6CRG0O MINGW64 ~/github/DemoProject (master)
```

```
$ git branch
```

master

test

```
taeju@DESKTOP-A6CRG0O MINGW64 ~/github/DemoProject (master)
```

```
$ git branch -D test
```

Deleted branch test (was c1aedfd).

```
taeju@DESKTOP-A6CRG0O MINGW64 ~/github/DemoProject (master)
```

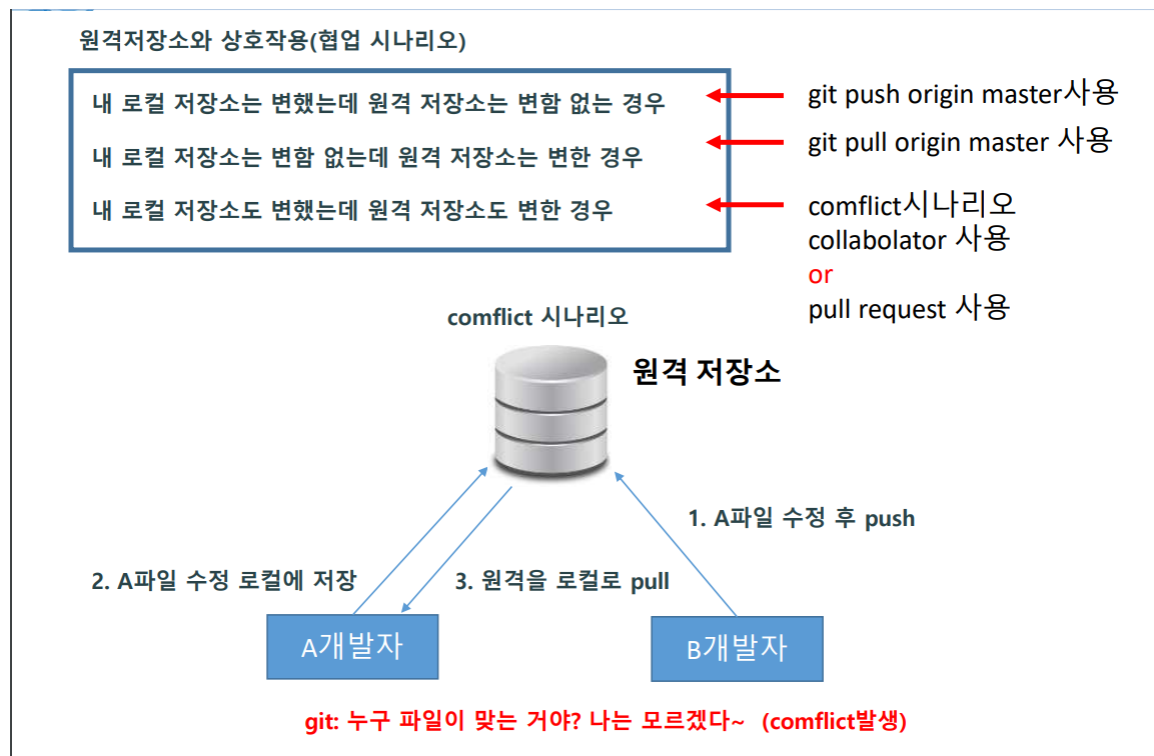


→ test branch는 필요 없으니 삭제

## 5. 원격 저장소에서 복사하기

\$ git clone 주소

## 원격 저장소와 상호작용



## conflict해소절차

- > 같은 파일을 동시에 수정할 때 발생한다.
- > 1. pull을 당겨오면 conflict가 발생한다 (git status로 확인가능)
- > 2. 일단 내가 쓰던 코드를 add -> commit한다(commit을 안하는 경우 단순 에러만 발생)
- > 3. 다시 한번 pull을 한다 (conflict발생)
- > 4. conflict 파일 목록을 확인하고 파일을 확인하고 코드를 수정한다
- > 5. 다시 한번 add -> commit 후 push한다

taeju@DESKTOP-A6CRG00 MINGW64 ~/github/DemoProject (master)

\$ git pull origin master

remote: Enumerating objects: 4, done.

remote: Counting objects: 100% (4/4), done.  
remote: Compressing objects: 100% (2/2), done.  
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0  
Unpacking objects: 100% (3/3), 713 bytes | 33.00 KiB/s, done.  
From <https://github.com/gkekf123/DemoProject>

- branch master -> FETCH\_HEAD  
c1aedfd..882acbf master -> origin/master  
Updating c1aedfd..882acbf  
Fast-forward  
fourth.txt | 1 +  
1 file changed, 1 insertion(+)  
create mode 100644 fourth.txt

taeju@DESKTOP-A6CRG00 MINGW64 ~/github/DemoProject (master)

taeju@DESKTOP-A6CRG00 MINGW64 ~/github/DemoProject (master)

\$ git log

commit 882acbf6a7cbbab0221980d4d7d1408f65dec35 (HEAD -> master,  
origin/master, origin/HEAD)

Author: TAEJU LEE [92421686+gkekf123@users.noreply.github.com](mailto:92421686+gkekf123@users.noreply.github.com)

Date: Thu May 11 14:32:39 2023 +0900

Create fourth.txt

commit c1aedfd7d69a9ecc421e03f1d78c09b292032b84

Author: TAEJU LEE [92421686+gkekf123@users.noreply.github.com](mailto:92421686+gkekf123@users.noreply.github.com)

Date: Thu May 11 12:44:13 2023 +0900

Update second.txt

commit d8a45f58bdb4fc3bf2e19233ff79668b18d1afea

Author: TAEJU LEE [92421686+gkekf123@users.noreply.github.com](mailto:92421686+gkekf123@users.noreply.github.com)

Date: Thu May 11 12:37:50 2023 +0900

Create second.txt

commit 516d97c2a48ed8ce15425f283bc34bf3f6cf0b20

Author: gkekf123 [taeju95@gmail.com](mailto:taeju95@gmail.com)

Date: Thu May 11 12:20:22 2023 +0900

```
first commit
```

taeju@DESKTOP-A6CRG00 MINGW64 ~/github/DemoProject (master)

\$ git reset --mixed HEAD^

taeju@DESKTOP-A6CRG00 MINGW64 ~/github/DemoProject (master)

\$ git log

commit c1aedfd7d69a9ecc421e03f1d78c09b292032b84 (HEAD -> master)

Author: TAEJU LEE [92421686+gkekf123@users.noreply.github.com](mailto:92421686+gkekf123@users.noreply.github.com)

Date: Thu May 11 12:44:13 2023 +0900

```
Update second.txt
```

commit d8a45f58bdb4fc3bf2e19233ff79668b18d1afea

Author: TAEJU LEE [92421686+gkekf123@users.noreply.github.com](mailto:92421686+gkekf123@users.noreply.github.com)

Date: Thu May 11 12:37:50 2023 +0900

```
Create second.txt
```

commit 516d97c2a48ed8ce15425f283bc34bf3f6cf0b20

Author: gkekf123 [taeju95@gmail.com](mailto:taeju95@gmail.com)

Date: Thu May 11 12:20:22 2023 +0900

```
first commit
```

taeju@DESKTOP-A6CRG00 MINGW64 ~/github/DemoProject (master)

\$ git add .

taeju@DESKTOP-A6CRG00 MINGW64 ~/github/DemoProject (master)

\$ git commit -m "create third.txt"

[master f8d1573] create third.txt

1 file changed, 1 insertion(+)

create mode 100644 third.txt

taeju@DESKTOP-A6CRG00 MINGW64 ~/github/DemoProject (master)

\$ git log

commit f8d15738d660448abf0a012e65b87ec48b431374 (HEAD -> master)

Author: gkekf123 <taeju95@gmail.com>

Date: Thu May 11 14:37:21 2023 +0900

```
create third.txt
```

commit c1aedfd7d69a9ecc421e03f1d78c09b292032b84

Author: TAEJU LEE <92421686+gkekf123@users.noreply.github.com>

Date: Thu May 11 12:44:13 2023 +0900

```
Update second.txt
```

commit d8a45f58bdb4fc3bf2e19233ff79668b18d1afea

Author: TAEJU LEE <92421686+gkekf123@users.noreply.github.com>

Date: Thu May 11 12:37:50 2023 +0900

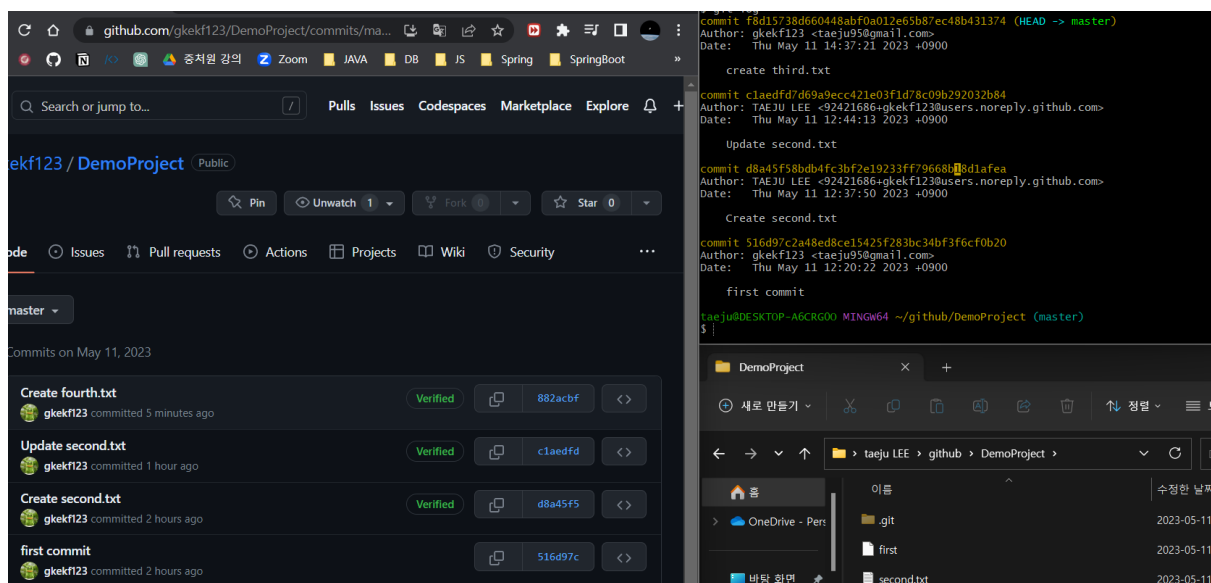
```
Create second.txt
```

commit 516d97c2a48ed8ce15425f283bc34bf3f6cf0b20

Author: gkekf123 <taeju95@gmail.com>

Date: Thu May 11 12:20:22 2023 +0900

```
first commit
```



→ 커밋 차이가 난다

git pull origin master

```
Merge branch 'master' of https://github.com/gkekf123/DemoProject
# Please enter a commit message to explain why this merge is necessary,
# especially if it merges an updated upstream into a topic branch.
#
# Lines starting with '#' will be ignored, and an empty message aborts
# the commit.
~
~
~
~
~
~
~
~
~
```

→ 원격과 로컬의 로그가 달라서 이런창이 나온다

→ esc shift :

taeju@DESKTOP-A6CRG00 MINGW64 ~/github/DemoProject (master)

\$ git log

commit 1ccc07531baba217ae9f7600baddece9487cc25b (HEAD -> master)

Merge: f8d1573 882acbf

Author: gkekf123 [taeju95@gmail.com](mailto:taeju95@gmail.com)

Date: Thu May 11 14:39:08 2023 +0900

```
Merge branch 'master' of <https://github.com/gkekf123/DemoProject>
```

commit f8d15738d660448abf0a012e65b87ec48b431374

Author: gkekf123 [taeju95@gmail.com](mailto:taeju95@gmail.com)

Date: Thu May 11 14:37:21 2023 +0900

```
create third.txt
```

commit 882acbf6a7cbbab0221980d4d7d1408f65dec35 (origin/master,  
origin/HEAD)

Author: TAEJU LEE [92421686+gkekf123@users.noreply.github.com](mailto:92421686+gkekf123@users.noreply.github.com)

Date: Thu May 11 14:32:39 2023 +0900

```
Create fourth.txt
```

```
commit c1aedfd7d69a9ecc421e03f1d78c09b292032b84
```

```
Author: TAEJU LEE <92421686+gkekf123@users.noreply.github.com>
```

```
Date: Thu May 11 12:44:13 2023 +0900
```

```
Update second.txt
```

```
commit d8a45f58bdb4fc3bf2e19233ff79668b18d1afea
```

```
Author: TAEJU LEE <92421686+gkekf123@users.noreply.github.com>
```

```
Date: Thu May 11 12:37:50 2023 +0900
```

```
Create second.txt
```

```
commit 516d97c2a48ed8ce15425f283bc34bf3f6cf0b20
```

```
Author: gkekf123 <taeju95@gmail.com>
```

```
Date: Thu May 11 12:20:22 2023 +0900
```

```
first commit
```

(END)

q

```
taeju@DESKTOP-A6CRG00 MINGW64 ~/github/DemoProject (master)
```

```
$ git push origin master
```

```
Enumerating objects: 7, done.
```

```
Counting objects: 100% (7/7), done.
```

```
Delta compression using up to 16 threads
```

```
Compressing objects: 100% (4/4), done.
```

```
Writing objects: 100% (5/5), 526 bytes | 526.00 KiB/s, done.
```

```
Total 5 (delta 2), reused 0 (delta 0), pack-reused 0
```

```
remote: Resolving deltas: 100% (2/2), completed with 1 local object.
```

```
To https://github.com/gkekf123/DemoProject.git
```

```
882acbf..1ccc075 master -> master
```

→ 원격과 로컬의 커밋 상태가 같아진다

로컬 first 내용 추가 / 원격 first 내용 추가 후 커밋 = 에러발생 ⇒ 커밋을 다시해줘야된다

taeju@DESKTOP-A6CRG00 MINGW64 ~/github/DemoProject (master)

\$ git pull origin master

remote: Enumerating objects: 4, done.

remote: Counting objects: 100% (4/4), done.

remote: Compressing objects: 100% (3/3), done.

remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0

Unpacking objects: 100% (3/3), 805 bytes | 21.00 KiB/s, done.

From <https://github.com/gkekf123/DemoProject>

branch master -> FETCH\_HEAD

1ccc075..cb1f1f6 master -> origin/master

error: The following untracked working tree files would be overwritten by merge:

first.txt

Please move or remove them before you merge.

Aborting

Updating 1ccc075..cb1f1f6

taeju@DESKTOP-A6CRG00 MINGW64 ~/github/DemoProject (master)

\$ git log

commit 1ccc07531baba217ae9f7600baddece9487cc25b (HEAD -> master)

Merge: f8d1573 882acbf

Author: gkekf123 [taeju95@gmail.com](mailto:taeju95@gmail.com)

Date: Thu May 11 14:39:08 2023 +0900

```
Merge branch 'master' of <https://github.com/gkekf123/DemoProject>
```

commit f8d15738d660448abf0a012e65b87ec48b431374

Author: gkekf123 [taeju95@gmail.com](mailto:taeju95@gmail.com)

Date: Thu May 11 14:37:21 2023 +0900

```
create third.txt
```

commit 882acbf6a7cbbab0221980d4d7d1408f65dec35

Author: TAEJU LEE [92421686+gkekf123@users.noreply.github.com](mailto:92421686+gkekf123@users.noreply.github.com)

Date: Thu May 11 14:32:39 2023 +0900

```
Create fourth.txt
```

---

commit c1aedfd7d69a9ecc421e03f1d78c09b292032b84  
Author: TAEJU LEE [92421686+gkekf123@users.noreply.github.com](mailto:92421686+gkekf123@users.noreply.github.com)  
Date: Thu May 11 12:44:13 2023 +0900

Update second.txt

commit d8a45f58bdb4fc3bf2e19233ff79668b18d1afea  
Author: TAEJU LEE [92421686+gkekf123@users.noreply.github.com](mailto:92421686+gkekf123@users.noreply.github.com)  
Date: Thu May 11 12:37:50 2023 +0900

Create second.txt

commit 516d97c2a48ed8ce15425f283bc34bf3f6cf0b20  
Author: gkekf123 [taeju95@gmail.com](mailto:taeju95@gmail.com)  
Date: Thu May 11 12:20:22 2023 +0900

first commit

(END)

q