

Git 배우기.

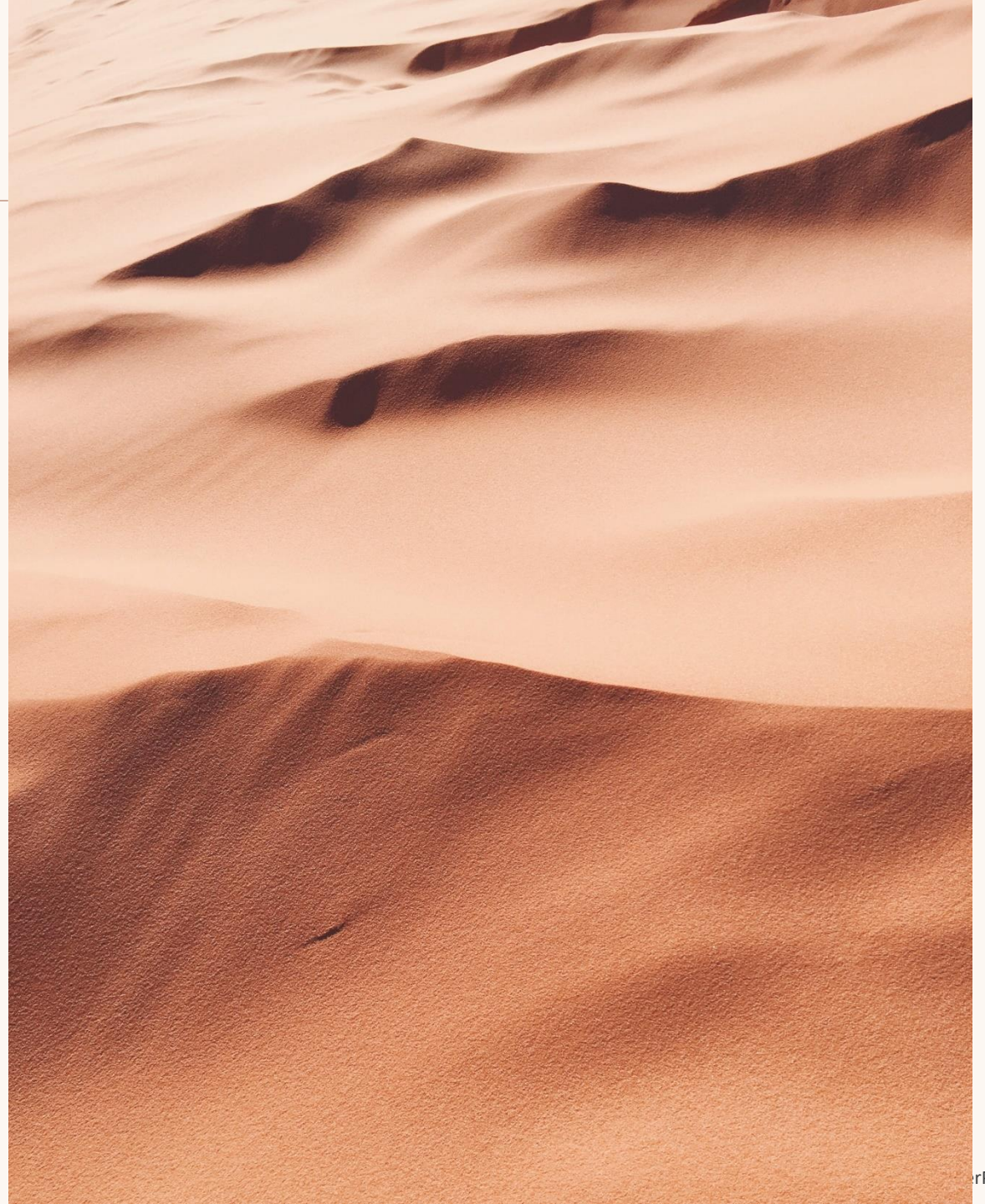
4th class

발표자 : 조형찬

목차 A table of contents.

1 GIT Branch

2 Git 연결과 협업



GIT Branch



git branch 여러 명령어들

https://velog.io/@corone_hi/Git-%EB%AA%85%EB%A0%B9%EC%96%B4

GIT 연결과 협업



이미 local repo가 있는 경우

원격 저장소의 이름
origin : main으로 이용

GIT repo 주소

```
migga@Vivobook_HC MINGW64 /c/Users/migga/Desktop/contest (master)
$ git remote add origin https://github.com/chan-ing/AI_contest_AIng.git~

migga@Vivobook_HC MINGW64 /c/Users/migga/Desktop/contest (master)
$ git remote -v
origin https://github.com/chan-ing/AI_contest_AIng.git~ (fetch)
origin https://github.com/chan-ing/AI_contest_AIng.git~ (push)
```

GIT 연결과 협업



이미 local repo가 있는 경우

Branch 이름

```
migga@Vivobook_HC MINGW64 /c/Users/migga/Desktop/contest (leader)
$ git push -u origin leader
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 204 bytes | 204.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'leader' on GitHub by visiting:
remote:   https://github.com/chan-ing/AI_contest_AIng/pull/new/leader
remote:
To https://github.com/chan-ing/AI_contest_AIng.git
 * [new branch]      leader -> leader
branch 'leader' set up to track 'origin/leader'.
```

GIT 연결과 협업



이미 local repo가 있는 경우

```
migga@Vivobook_HC MINGW64 /c/Users/migga/Desktop/contest (leader)
$ git commit -am 'hello'
warning: in the working copy of 'test.txt', LF will be replaced by CRLF the next time Git touches it
[leader 53f7a52] hello
1 file changed, 2 insertions(+)

migga@Vivobook_HC MINGW64 /c/Users/migga/Desktop/contest (leader)
$ git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Writing objects: 100% (3/3), 242 bytes | 242.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/chan-ing/AI_contest_AIng.git
3610607..53f7a52  leader -> leader
```

다음 push부터는 git push만 이용

GIT 연결과 협업



이미 원격 저장소가 있는 경우

현재 디렉토리
복사

```
migga@Vivobook_HC MINGW64 /c/Users/migga/Desktop/temp
$ git clone https://github.com/chan-ing/AI_contest_AIng.git .
Cloning into '.'...
remote: Enumerating objects: 10, done.
remote: Counting objects: 100% (10/10), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 10 (delta 0), reused 6 (delta 0), pack-reused 0
Receiving objects: 100% (10/10), done.
```


GIT 연결과 협업



이미 원격 저장소가 있는 경우

```
migga@Vivobook_HC MINGW64 /c/Users/migga/Desktop/temp
$ git clone https://github.com/chan-ing/AI_contest_AIng.git git_home
Cloning into 'git_home'...
remote: Enumerating objects: 10, done.
remote: Counting objects: 100% (10/10), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 10 (delta 0), reused 6 (delta 0), pack-reused 0
Receiving objects: 100% (10/10), done.
migga@Vivobook_HC MINGW64 /c/Users/migga/Desktop/temp
$ git clone https://github.com/chan-ing/AI_contest_AIng.git git_office
Cloning into 'git_office'...
remote: Enumerating objects: 10, done.
remote: Counting objects: 100% (10/10), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 10 (delta 0), reused 6 (delta 0), pack-reused 0
Receiving objects: 100% (10/10), done.
migga@Vivobook_HC MINGW64 /c/Users/migga/Desktop/temp
$ ls
git_home/  git_office/
```


GIT 연결과 협업

새로운 공간에서 작업할 땐
가장 먼저 'pull'

```

migga@Vivobook_HC MINGW64 /c/Users/migga/Desktop/temp/git_home (ma
in)
$ vi hello.txt

migga@Vivobook_HC MINGW64 /c/Users/migga/Desktop/temp/git_home (ma
in)
$ git add hello.txt
warning: in the working copy of 'hello.txt', LF will be replaced b
y CRLF the next time Git touches it

migga@Vivobook_HC MINGW64 /c/Users/migga/Desktop/temp/git_home (ma
in)
$ git commit hello.txt -m 'working home'
warning: in the working copy of 'hello.txt', LF will be replaced b
y CRLF the next time Git touches it
[main b023202] working home
 1 file changed, 1 insertion(+)
 create mode 100644 hello.txt

migga@Vivobook_HC MINGW64 /c/Users/migga/Desktop/temp/git_home (ma
in)
$ git push
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 20 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 314 bytes | 314.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/chan-ing/AI_contest_AIng.git
 f037904..b023202  main -> main
  
```

```

migga@Vivobook_HC MINGW64 /c/Users/migga/Desktop/temp/git_office
(main)
$ git pull
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 3 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 294 bytes | 21.00 KiB/s, done.
From https://github.com/chan-ing/AI_contest_AIng
 f037904..b023202  main      -> origin/main
Updating f037904..b023202
Fast-forward
 hello.txt | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 hello.txt

migga@Vivobook_HC MINGW64 /c/Users/migga/Desktop/temp/git_office
(main)
$ git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 20 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 257 bytes | 257.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local obje
ct.
To https://github.com/chan-ing/AI_contest_AIng.git
 b023202..b6e3d2c  main -> main
  
```

작업이 끝났다면 'push'
push는 항상 습관화 한다.

GIT 연결과 협업



git pull과 git fetch는 뭐가 다를까?

```
e52bf7d (origin/master) 7
416efac (HEAD -> master) 6
90cc120 5
dfd9906 4
264e038 3
9f6227a 2
fafb885 1
(END)
```

git fetch를 하게 되면, 원격 저장소에는 변화가 있지만, 지역 저장소에는 변화 X.
(git pull(fetch + merge)은 모두 같아 진다.)

원격 저장소로부터 병합을 시키지 않기 때문에, 로컬과 원격 저장소의 차이를 볼 수 있다.

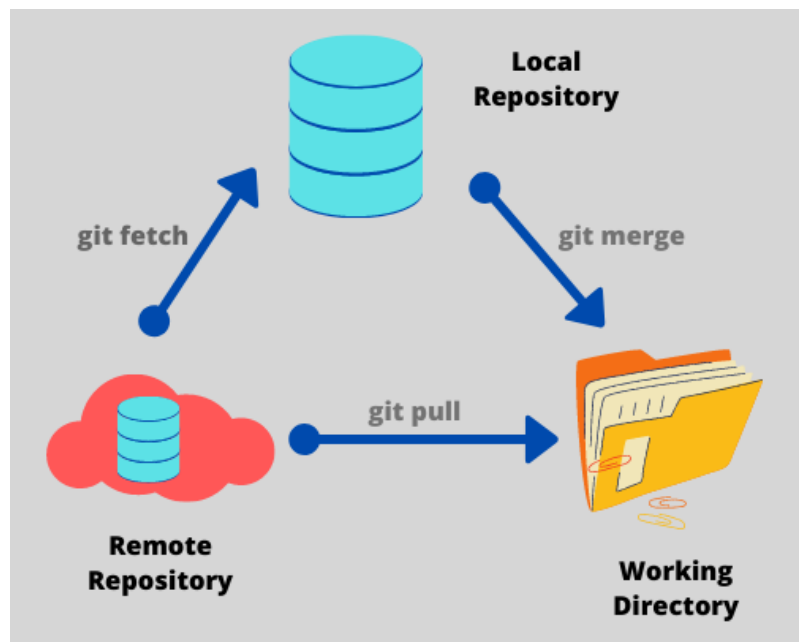
```
dev/git/home master ✓
▶ git merge origin/master
```

git merge로 지역 저장소에 반영 할 수 있다.

GIT 연결과 협업

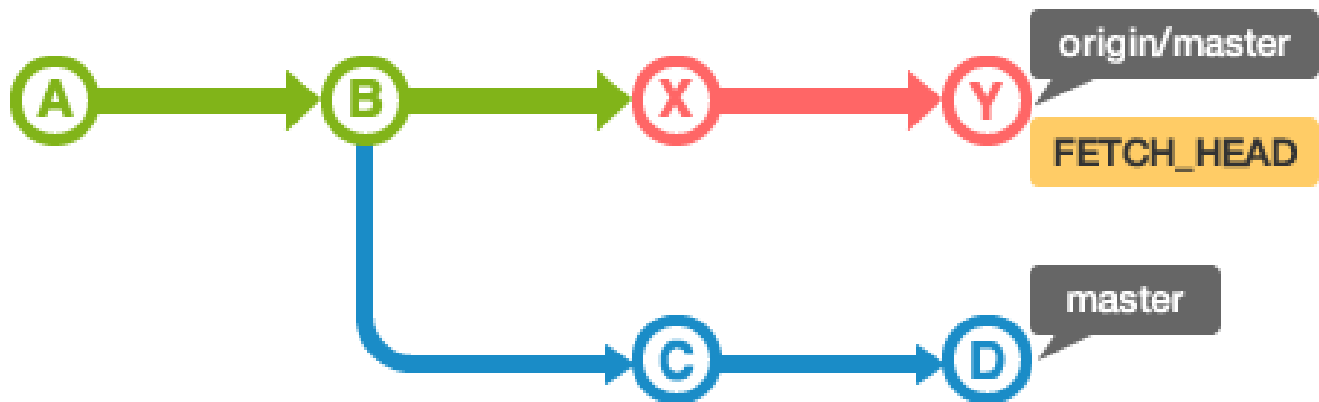


git pull과 git fetch는 뭐가 다를까?



Fetch로 가져온 최신 커밋 정보는, **origin master**가 아닌 **FETCH_HEAD**라는 브랜치로 가져온다.

'FETCH_HEAD' 브랜치를 merge 하거나 다시 pull 을 실행하면 됩니다.



GIT 연결과 협업



초대된 repo를 fork 하여, pull request 보내기

<https://inpa.tistory.com/entry/GIT-%E2%9A%A1%EF%B8%8F-%EA%B9%83%ED%97%99-PRPull-Request-%EB%B3%B4%EB%82%B4%EB%8A%94-%EB%B0%A9%EB%B2%95-folk-issue>

꽃.