

# 2023-10-18

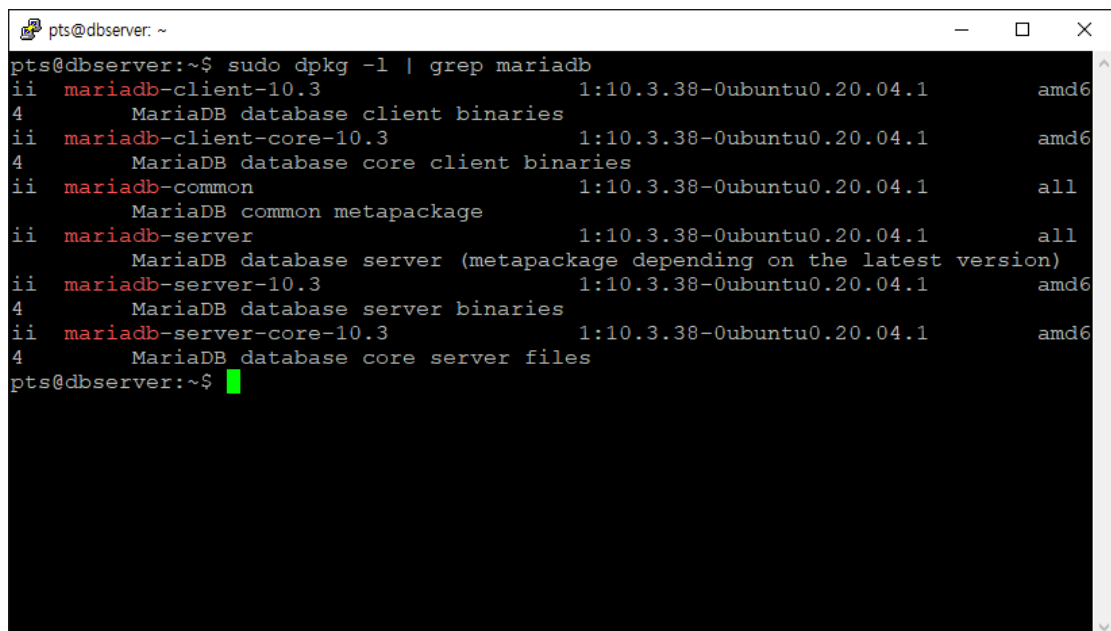
apt-get → 설치목록

- update → 목록
- upgrade → 세부내용

sudo apt-get install 패키지명

ex)sudo apt-get install mariadb-server

- remove → 삭제 패키지 (설정 파일을 남겨 둠)
- purge → 삭제 패키지 (설정 파일도 제거)
- list → 조건 목록



```
pts@dbserver: ~  
pts@dbserver:~$ sudo dpkg -l | grep mariadb  
ii  mariadb-client-10.3          1:10.3.38-0ubuntu0.20.04.1      amd64  
4   MariaDB database client binaries  
ii  mariadb-client-core-10.3     1:10.3.38-0ubuntu0.20.04.1      amd64  
4   MariaDB database core client binaries  
ii  mariadb-common               1:10.3.38-0ubuntu0.20.04.1      all  
   MariaDB common metapackage  
ii  mariadb-server               1:10.3.38-0ubuntu0.20.04.1      all  
   MariaDB database server (metapackage depending on the latest version)  
ii  mariadb-server-10.3          1:10.3.38-0ubuntu0.20.04.1      amd64  
4   MariaDB database server binaries  
ii  mariadb-server-core-10.3     1:10.3.38-0ubuntu0.20.04.1      amd64  
4   MariaDB database core server files  
pts@dbserver:~$
```

sudo dpkg -l | grep mariadb → 설치한 총 패키지 리스트

- sudo apt-get purge -y 복불

```
pts@dbserver: ~  
pts@dbserver:~$ ls -al  
total 40  
drwxr-xr-x 4 pts pts 4096 Oct 17 13:52 .  
drwxr-xr-x 3 root root 4096 Oct 16 16:28 ..  
-rw----- 1 pts pts 39 Oct 16 17:05 .bash_history  
-rw-r--r-- 1 pts pts 220 Feb 25 2020 .bash_logout  
-rw-r--r-- 1 pts pts 3771 Feb 25 2020 .bashrc  
drwx----- 2 pts pts 4096 Oct 16 16:28 .cache  
-rw----- 1 pts pts 9 Oct 17 13:52 .mysql_history  
-rw-r--r-- 1 pts pts 807 Feb 25 2020 .profile  
drwx----- 2 pts pts 4096 Oct 16 16:28 .ssh  
-rw-r--r-- 1 pts pts 0 Oct 16 16:28 .sudo_as_admin_successful  
-rw----- 1 pts pts 728 Oct 17 13:52 .viminfo  
pts@dbserver:~$
```

. → 나 자신

.. → 위

.불음 → 숨김

sudo apt-get install tree

[Dos]

md - 디렉토리 생성

rd - 디렉토리 제거

cd - 디렉토리 이동

cd .. - 상위폴더로 이동

dir - 디렉토리 보기

dir test01 - 경로 지정 가능

dir /? - 도움말

copy con(원본) test01.txt(대상) - 만들고 내용입력

원본에서 입력한 내용을 대상으로 복사

ex) copy con test01.txt

1st

2st

컨트롤+c

copy : 복사

con : 콘솔

copy test01.txt test02.txt - 복사

copy test01.txt c:\ex\test01

[와일드 카드 문자]

copy t\*.txt test01 - t로 시작하는 모든 txt를 test01 디렉토리로 복사

t\* - t로 시작하는 모든 파일명

\*t - t로 끝나는

t? - t로 시작하는 두글자만, t도 포함한다

\*.txt - 모든 txt

move test01.txt c:\ex\test01

[리눅스]

mkdir test01 - 리눅스 디렉토리 생성 (md)

rmdir test01 - 리눅스 디렉토리 제거 ( rd)

cd - 디렉토리 이동

cd ..

ls -al - 디렉토리 보여줌 (dir)

touch - 여러 파일을 만듦

ex) touch test01.txt test02.txt test03.txt

>test01.txt - 명령어가 가장 짧다, 한개만 가능

cat test01.txt - 파일의 내용을 보여줌

cat > test01.txt - 텍스트 입력

cat >> test01.txt - 텍스트 추가 입력

vim test01.txt - 에디터

cp \*.\* test01 - 복사

```
pts@dbserver: ~  
pts@dbserver:~$ ps  
  PID TTY          TIME CMD  
 2693 pts/0        00:00:00 bash  
 2898 pts/0        00:00:00 ps  
pts@dbserver:~$ kill 2898  
-bash: kill: (2898) - No such process  
pts@dbserver:~$ ps  
  PID TTY          TIME CMD  
 2693 pts/0        00:00:00 bash  
 2899 pts/0        00:00:00 ps  
pts@dbserver:~$
```

ps - 작동중인 프로그램/ 사용 명령

ps - 작동중인 프로그램/ e 전체 사용명령

kill 2693 - 작동중인 프로그램을 죽여버림 (pid번호 입력)

\*터미널은 실시간으로 받지 않으므로 재실행 해야됨

java 환경변수 설정 한 이유 → 우선권 부여 받아 다른 프로그램보다 상위 실행권을 획득하기 위해

ex) A B 동시 요청시 A or B에 우선권 부여/ 우선권을 미리 환경변수로 획득시키는것

git설치 - 전부 next - cmd - git-work에 exm폴더 생성 - cd exm이동

-git config —list - cd exm - git init 입력

git config —global user.name “pts”

git config —global user.email “wastyno100@gmail.com”

.git dir → /ah - rd .git /s 로 삭제 (필요하면)

```
명령 프롬프트
switch      Switch branches
tag         Create, list, delete or verify a tag object signed with GPG

collaborate (see also: git help workflows)
fetch       Download objects and refs from another repository
pull        Fetch from and integrate with another repository or a local branch
push        Update remote refs along with associated objects

'git help -a' and 'git help -g' list available subcommands and some
concept guides. See 'git help <command>' or 'git help <concept>'
to read about a specific subcommand or concept.
See 'git help git' for an overview of the system.

c:\git-work>git config --list
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
http.sslbackend=openssl
http.sslcainfo=C:/Program Files/Git/mingw64/etc/ssl/certs/ca-bundle.crt
core.autocrlf=true
core.fscache=true
core.symlinks=false
pull.rebase=false
credential.helper=manager
credential.https://dev.azure.com.usehttppath=true
init.defaultbranch=master

c:\git-work>
```

(git config —list 입력시)

git이 설치 되어있는 최초의 환경설정 내용

etc : 설정파일이 들어가는 디렉토리 (기억해놓기)

- gitconfig 파일 열어서 master 이름을 바꾸면 내 이름으로 나올수 있다.

git status

```
Untracked files:
  (use "git add <file>..." to include in what will be committed)
1a.txt
```

(빨강 - 변경은 됐는데 올리지 않을 파일)

git add . (git add 1a.txt도 가능)

```
Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
new file:   1a.txt
```

(초록 - 저장소에 저장할 준비가 됐다는 뜻)

git commit -m "10-18" (.git에 등록)

-m : 메세지 이름

github 저장소 만듦

```
echo "# exm" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/wastyno100/Training_Daily_Note.git
git push -u origin main
```

```
git remote add origin https://github.com/wastyno100/eeee.git
git branch -M main
git push -u origin main
```

git init → local git 시작 github 저장소와 연결되는 곳. 두번 할 필요없다

```
git add README.md
```

```
git status
```

```
git add .
```

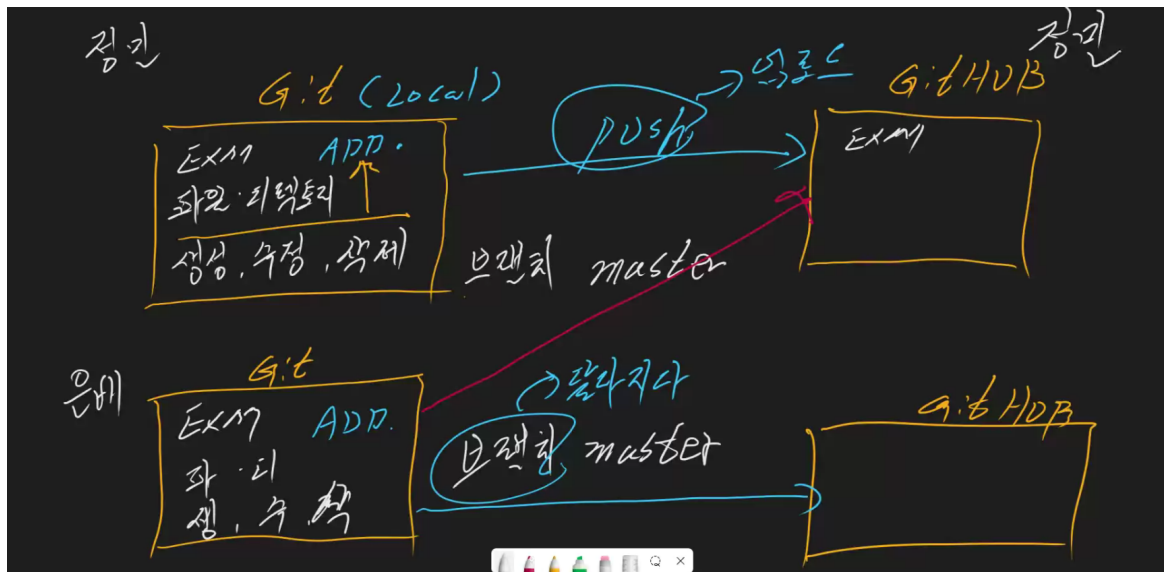
```
git status
```

```
git commit -m "10-18-01 커밋메세지 이유 적기"
```

```
git branch -M main
```

```
git remote add origin https://github.com/wastyno100/exm.git
```

```
git push -u origin main
```



다른 사람 git에 허밋 하려면 주인과는 다른 이름을 지정해야된다.

내 허가ip 21~30