

## 리눅스 명령어 모음

```
cd [name] : Change Directory to [name]
cd .. : Change directory to upper directory
cd ~ : Changes directory to home directory
mkdir [name] : Creates a [name] directory
pwd : displays the current working directory
ls : displays every files in the current directory
rmdir : removes directories
mv A / B : Change the name of file from A to B
touch [name] : Creates a new file with name
cat : (Concatenate) Displays file contents on Terminal
clear : clear the terminal (literally)
cp [source] [destination] : copy files or directoris from source to destination
mv [source] [destination] : move files or directories from source to destination
rm [filename] OR rm -r directory : remove files or directories. ('-r' keyword : del
chmod ____ : changes the file access permission.
/*
<chmod>
User : The owner of the file
Group : Users who are members of the file's group
Others: All other users except for user and group

r/w/x : Permission to Read/Write(modify)/Excute file.
Read =4
Write = 2
Execute = 1

ex) 755 : User can R/W/E, Group can Read/Execute, Others can Read/Execute.
*/
chmod 755 script.sh // User : rwx, Group : r-x, Others : r-x
```

# Git?

- 분산 버전 관리 시스템

## git command

```
git init : The first thing you should do to use git in your folder
git add [filename]: move changes from '작업폴더' to 'Staging Area'
Git commit -m [filename] : Move changes from 'Staging Area' to 'Repository'
git push [filename] : upload file from local Repository to Github Repository.
git clone [link]: brings git repository to my local computer
git pull [filename] : bring file from repository to local.
git switch [branchname] : switches branch
git branch [branchname] : creates branch
git merge [합칠 branch]: merges branch to the branch you're on.
```

git

- add & commit : local에서 repository로 push하기 전 단계. 누가 어떤 코드를 수정했는지 한 눈에 알 수 있게 해줌. 만약 잘못된 코드가 있다면, 이전에 남긴 commit을 확인하는 과정을 통해 어떤 코드가 문제를 발생시켰는지 빠르게 확인 가능.
- commit : 개발자의 일기장 같은 느낌.

## Branch : Repository 전체의 버전 관리를 위해 존재함.

- Why do we need branch?
  - If user is using 'ver 1.0.0', It is hard to change codes for ver '1.0.1' because users are using that.
  - ex) Make a 'Develop' Branch from a 'Master Branch' and fix codes using 'Develop' branch
    - 개발자는 develop branch에서 코드 수정하다가 나중에 master branch로 병합함.
  - 'git

## git flow : branch 관리 방법중 하나

- Master - 제품으로 출시 되는 브랜치
- Develop - 다음 출시 버전을 개발하는 브랜치
- Feature - 기능을 개발하는 브랜치
- Release - 이번 출시 버전을 준비하는 브랜치
- Hotfix - 출시 버전에서 발생한 버그를 수정하는 브랜치

Gcc 이용 컴파일

gcc -E text.c : 전처리까지만 실행한 결과를 파일에 출력 (너무 길어서 생략)

```

casey@casey:~/2024-Whois-SummerSecurityEdu/HW1_SSE_1주차_임찬우$ gcc -E test.c
# 1 "test.c"
# 1 "<built-in>"
# 1 "<command-line>"
# 31 "<command-line>"
# 1 "/usr/include/stdc-predef.h" 1 3 4
# 32 "<command-line>" 2
# 1 "test.c"
# 1 "/usr/include/stdio.h" 1 3 4
# 27 "/usr/include/stdio.h" 3 4
# 1 "/usr/include/aarch64-linux-gnu/bits/libc-header-start.h" 1 3 4
# 33 "/usr/include/aarch64-linux-gnu/bits/libc-header-start.h" 3 4
# 1 "/usr/include/features.h" 1 3 4
# 461 "/usr/include/features.h" 3 4
# 1 "/usr/include/aarch64-linux-gnu/sys/cdefs.h" 1 3 4
# 452 "/usr/include/aarch64-linux-gnu/sys/cdefs.h" 3 4
# 1 "/usr/include/aarch64-linux-gnu/bits/wordsize.h" 1 3 4
# 453 "/usr/include/aarch64-linux-gnu/sys/cdefs.h" 2 3 4
# 1 "/usr/include/aarch64-linux-gnu/bits/long-double.h" 1 3 4
# 454 "/usr/include/aarch64-linux-gnu/sys/cdefs.h" 2 3 4
# 462 "/usr/include/features.h" 2 3 4
# 485 "/usr/include/features.h" 3 4
# 1 "/usr/include/aarch64-linux-gnu/gnu/stubs.h" 1 3 4

# 1 "/usr/include/aarch64-linux-gnu/bits/wordsize.h" 1 3 4
# 6 "/usr/include/aarch64-linux-gnu/gnu/stubs.h" 2 3 4

# 1 "/usr/include/aarch64-linux-gnu/gnu/stubs-lp64.h" 1 3 4
# 9 "/usr/include/aarch64-linux-gnu/gnu/stubs.h" 2 3 4

```

gcc -c test.c : 오브젝트 파일 생성.

```

casey@casey:~/2024-Whois-SummerSecurityEdu/HW1_SSE_1주차_임찬우$ gcc -c test.c
casey@casey:~/2024-Whois-SummerSecurityEdu/HW1_SSE_1주차_임찬우$ ls
2024-Whois-SummerSecurityEdu  HW1_SSE_1주차_임찬우  test  test.c  test.o

```

gcc test.o : 오브젝트 파일을 이용한 실행 파일 생성. (a.out)

```

casey@casey:~/2024-Whois-SummerSecurityEdu/HW1_SSE_1주차_임찬우$ gcc test.o
casey@casey:~/2024-Whois-SummerSecurityEdu/HW1_SSE_1주차_임찬우$ ls
2024-Whois-SummerSecurityEdu  HW1_SSE_1주차_임찬우  test.c  test.o
a.out  test

```