

Linux & Ubuntu

WeGo & LIMO

목 차

1. OS (운영 체제)
2. Linux
3. Linux Filesystem
4. Ubuntu
5. Ubuntu Installation
6. Linux Command
7. Etc

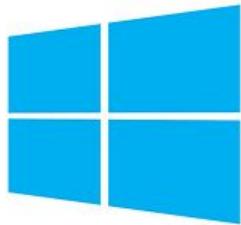
01

Operating
System

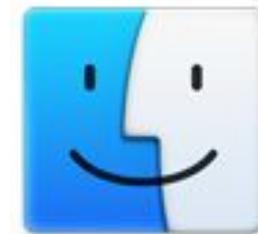
OS
(운영 체제)

01 OS (운영 체제)

- OS : Operating System (운영 체제)
 - 시스템 하드웨어를 관리 (CPU, RAM 등의 하드웨어에 명령 전달)
 - 사용자에게 편의성을 제공 (GUI 같은 환경을 제공하여, 사용하기 쉽게 함)
 - 컴퓨터의 성능을 높여주며, 사용자에게 편의성을 제공하는 컴퓨터의 하드웨어 관리를 주로 수행하는 소프트웨어
 - 사용자가 하드웨어에 대해 몰라도, 컴퓨터를 쉽게 사용할 수 있도록 도와주는 역할을 수행



Windows 10

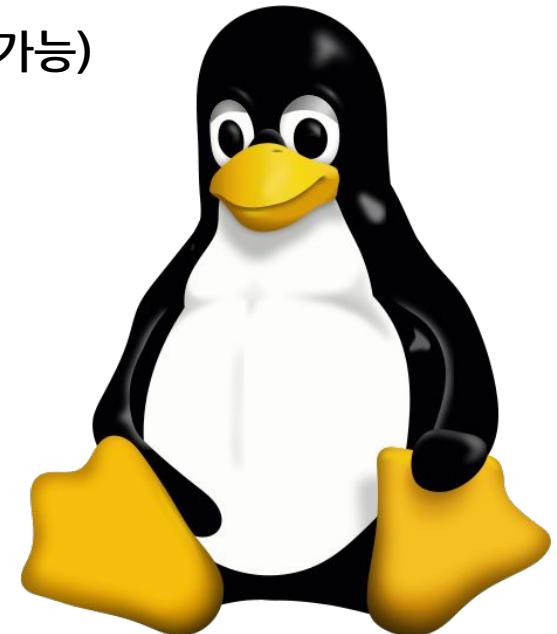


02

Linux

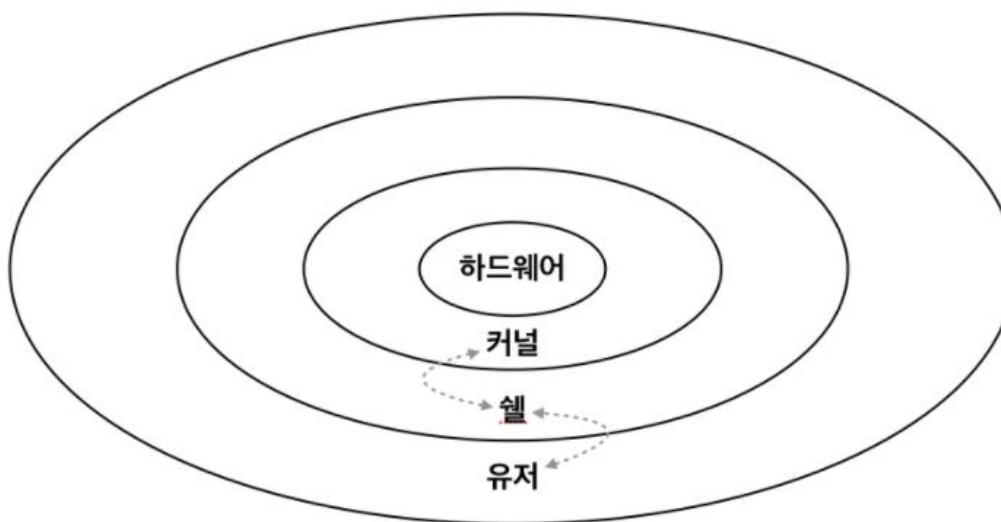
02 Linux

- Linux
 - PC에서 사용하는 운영체제 중 하나
 - 오픈 소스 운영 체제 (누구나 자유롭게 수정 및 사용이 가능)
 - 하드웨어 리소스를 관리하는 Linux 커널과 나머지 구성 소프트웨어 패키지를 포함
 - 용도에 따른 다양한 배포판이 존재
 - 서버 - 페도라 (레드햇 계열)
 - 모바일 기기 - 안드로이드
 - 일반 - 우분투 (데비안 계열)



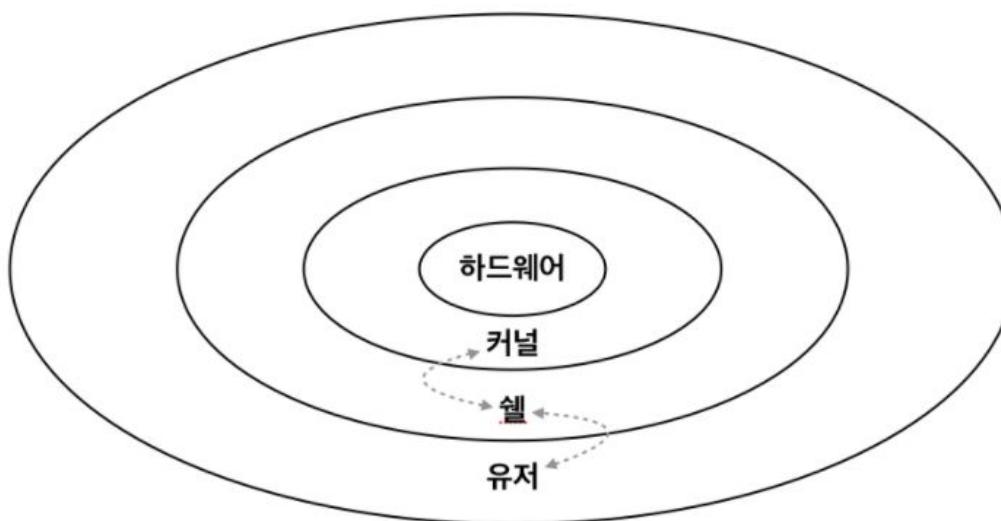
02 Linux

- Linux Kernel
 - 하드웨어와 가장 가까운 프로그램
 - 사용자의 명령을 받아서, 하드웨어에 명령 전달
 - 하드웨어를 직접 제어하는 것이 위험할 수 있으므로, 커널을 이용하여 제어



02 Linux

- Linux Shell
 - 커널과 응용프로그램 사이에 존재
 - 응용 프로그램의 명령어를 해석하여, 커널에 전달하는 역할
 - 명령어 해석기의 역할을 수행
 - 다양한 쉘의 종류가 존재 (sh, bash, csh, zsh, fish)



02 Linux

- Terminal

- 텍스트를 통해 Shell에 명령을 내릴 수 있도록 해주는 역할
 - 실제 사용자 - Shell 사이의 입출력을 전달하는 역할

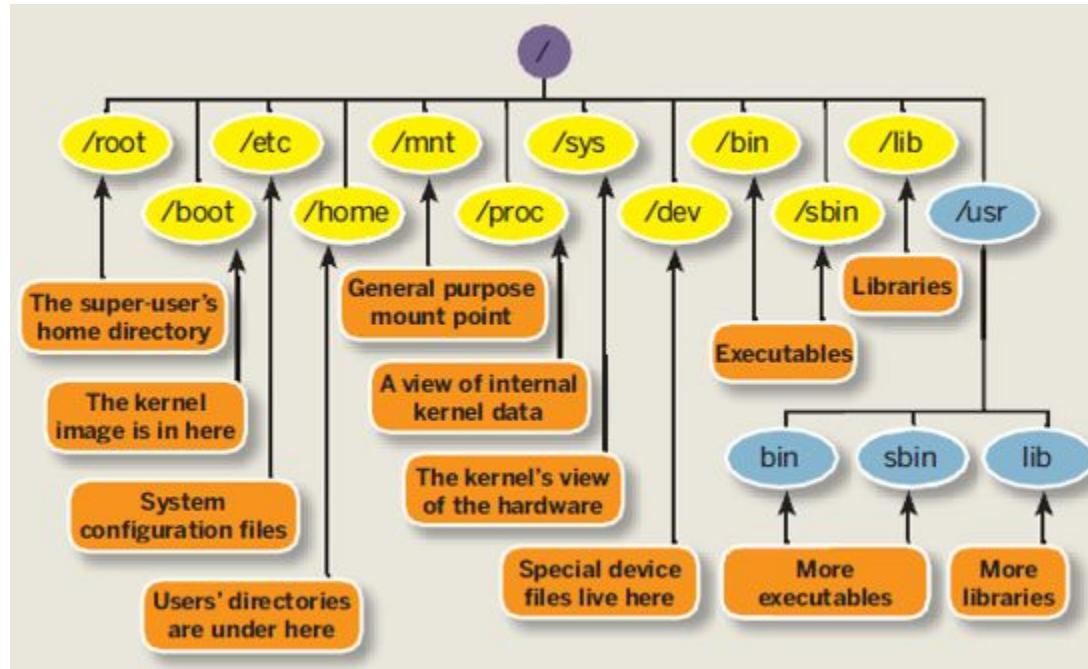


03

Linux Filesystem

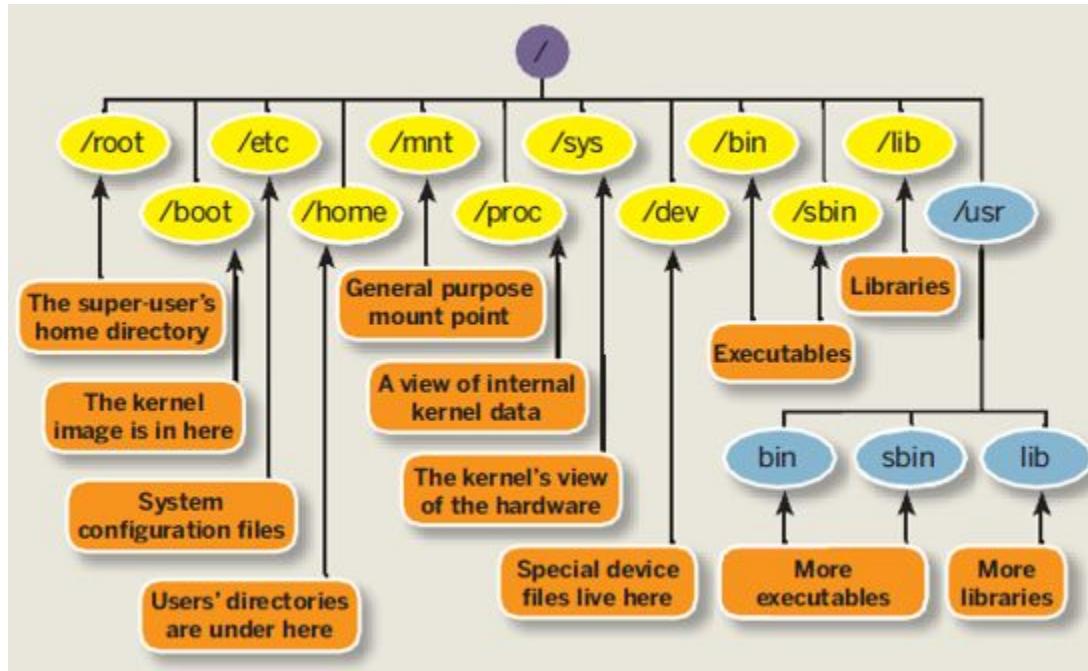
04 Linux Filesystem

- 파일 시스템 계층 구조 (Linux Filesystem Hierarchy Standard)
 - 파일이나 자료를 쉽게 검색 및 찾을 수 있도록 보관 또는 조직하는 구조
 - 최상위 폴더를 / (root) 로 하는 형태가 표준 계층 구조



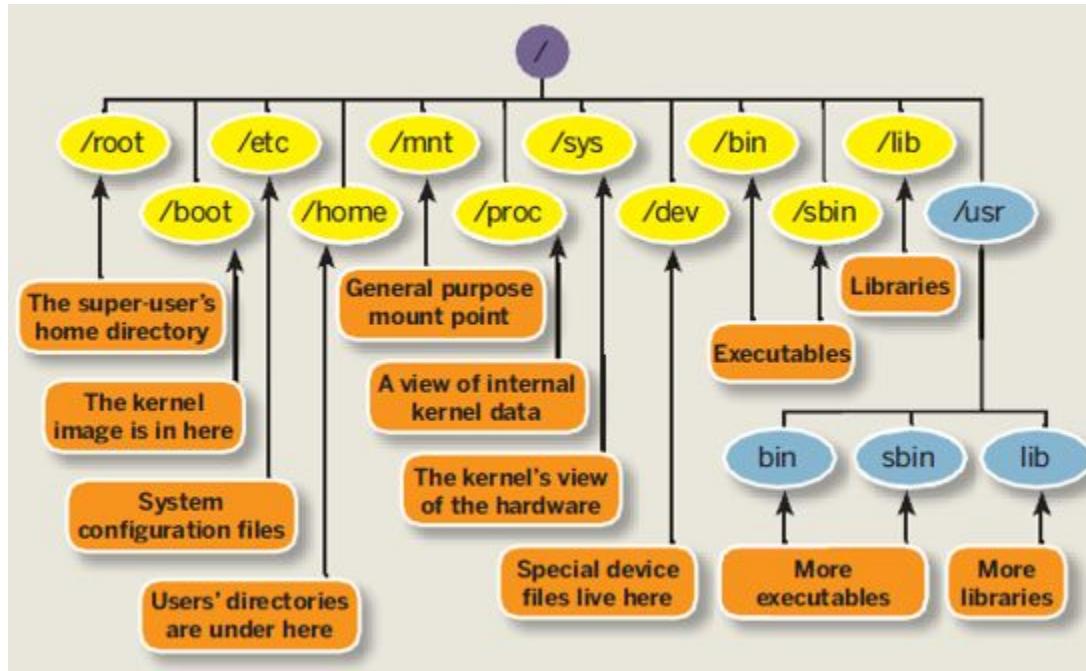
04 Linux Filesystem

- /root
 - 슈퍼유저 (관리자)의 홈 디렉토리 (기본 폴더 위치)
 - 접근해서 파일을 수정하면 위험할 수 있음



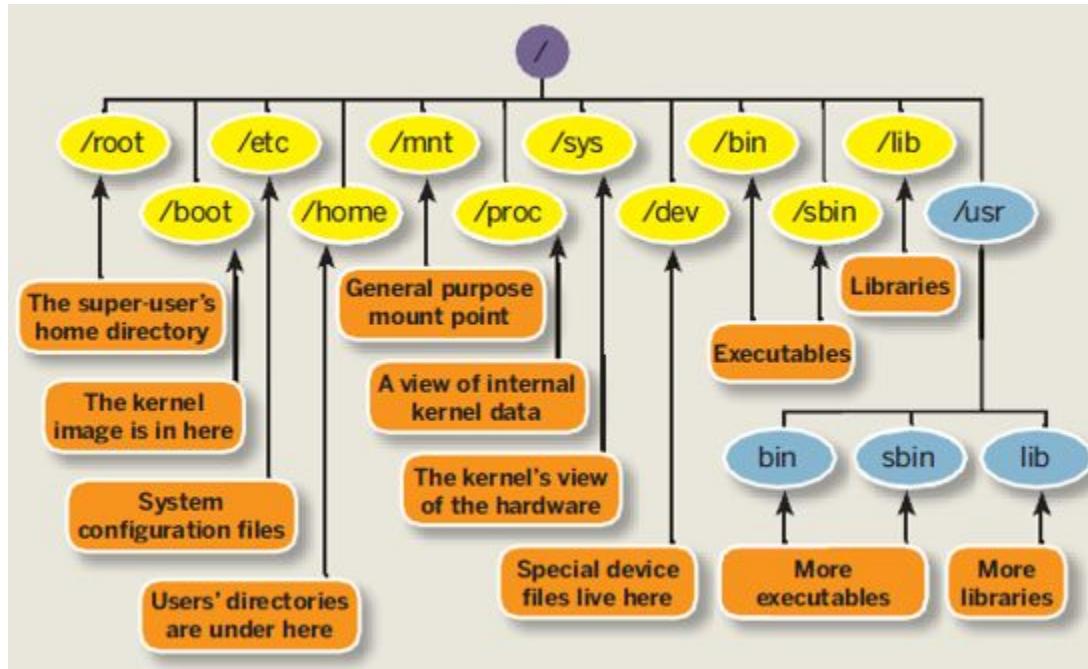
04 Linux Filesystem

- /boot
 - 리눅스 커널을 보관하고 있는 폴더
 - 접근해서 파일을 수정하면 위험할 수 있음



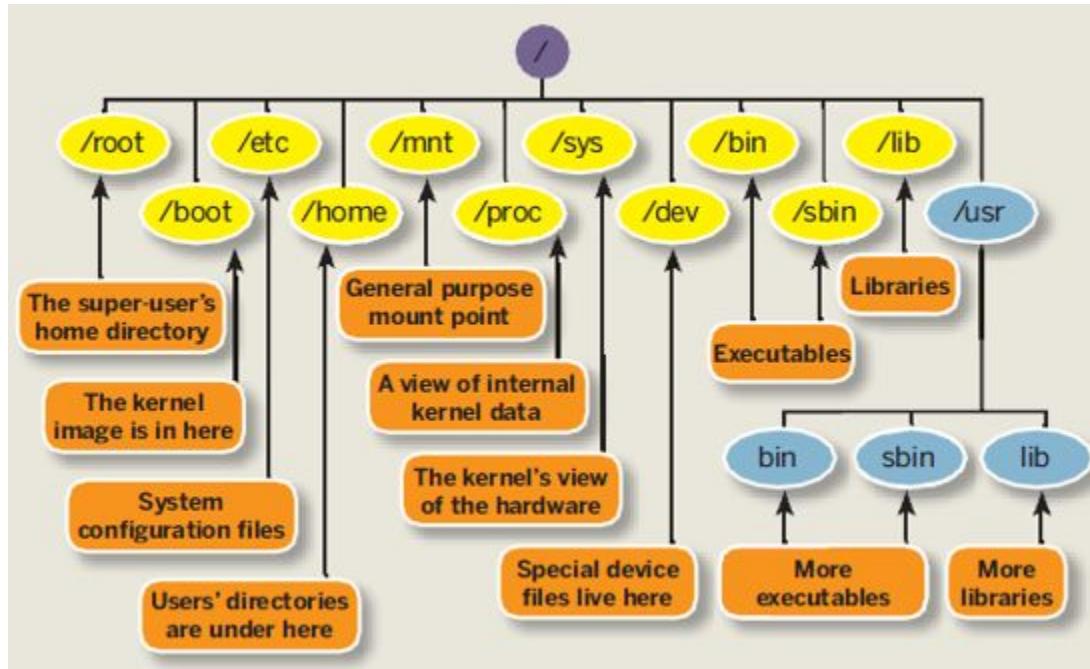
04 Linux Filesystem

- /etc
 - 시스템 설정 파일들이 모여있는 폴더
 - 시스템 부팅 시, 실행하는 서비스 파일등도 포함됨
 - 수정해도 크게 문제가 되지는 않는 경우가 많음



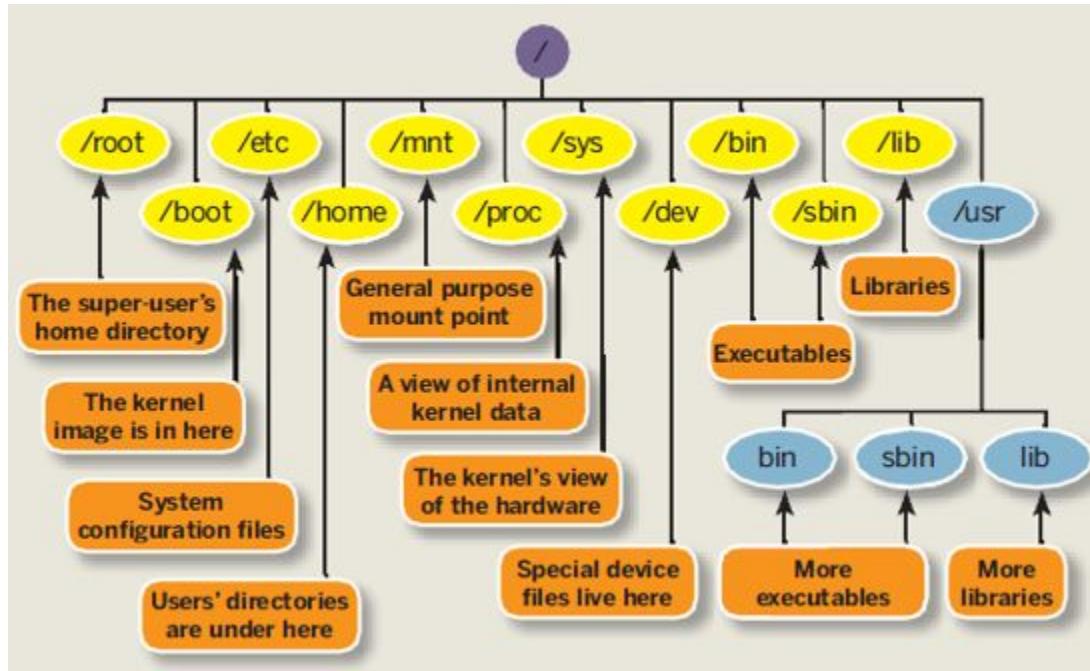
04 Linux Filesystem

- /home
 - 일반적으로 사용하는 위치
 - 각 User가 사용할 수 있는 공간이 제공됨
 - 수정해도 크게 문제가 되지 않음



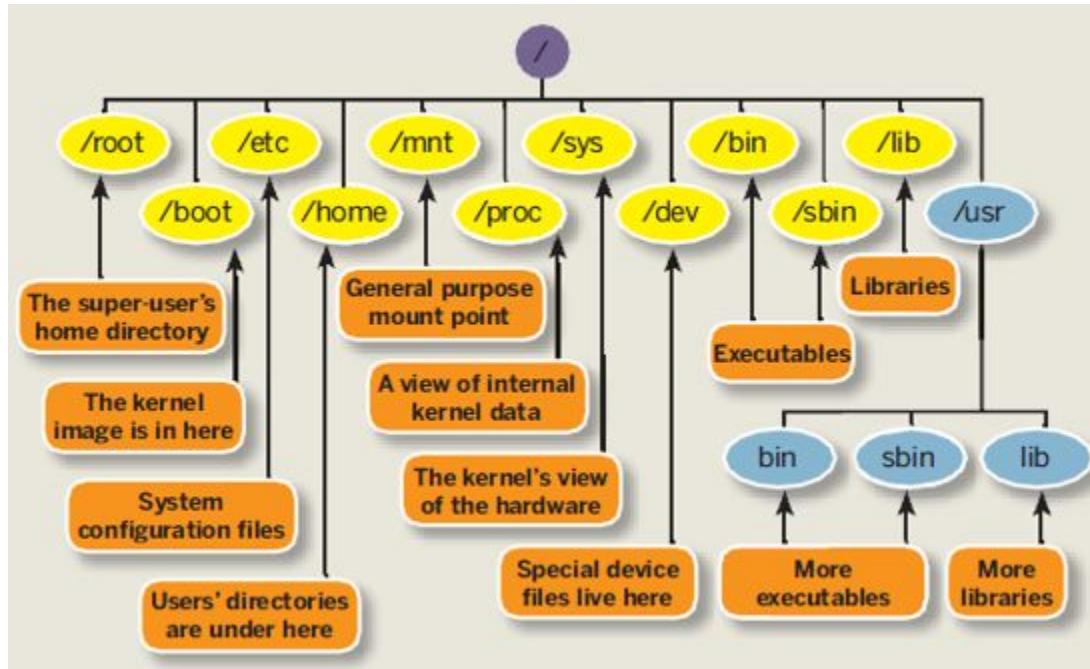
04 Linux Filesystem

- /mnt
 - USB 저장 장치 및 외장하드 등이 마운팅되는 위치
 - 수정하면, 연결된 저장 장치 및 외장 하드에 있는 파일을 수정할 수 있음



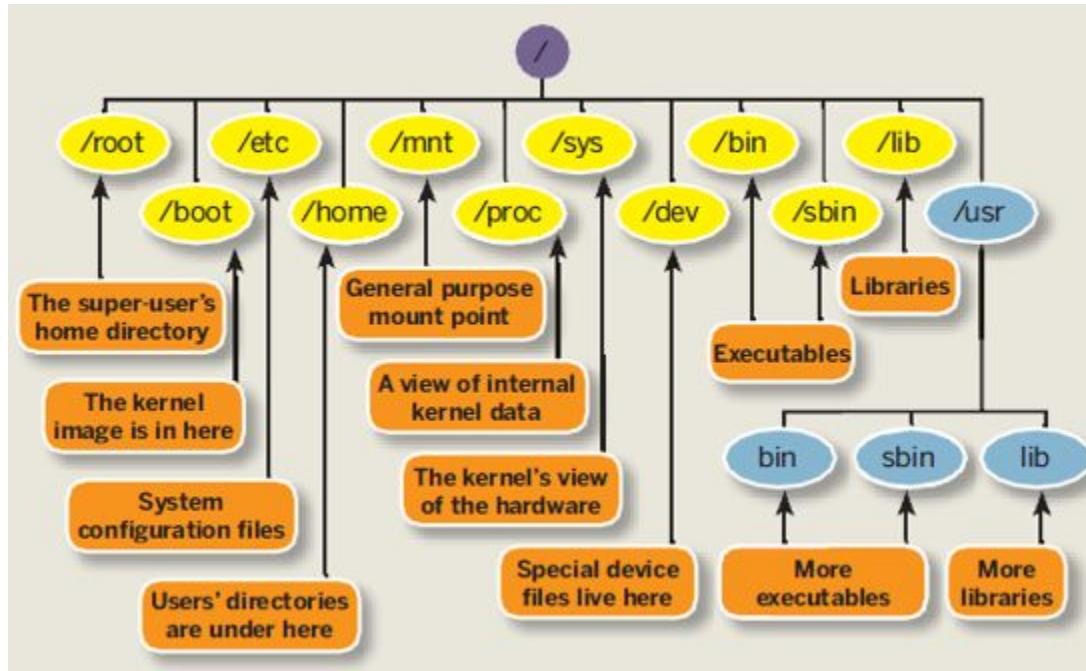
04 Linux Filesystem

- /proc
 - 커널을 통해 관리되는 폴더
 - 커널을 통해 볼 수 있는 가상의 파일들이 존재
 - 접근해서 파일을 수정하면 위험할 수 있음



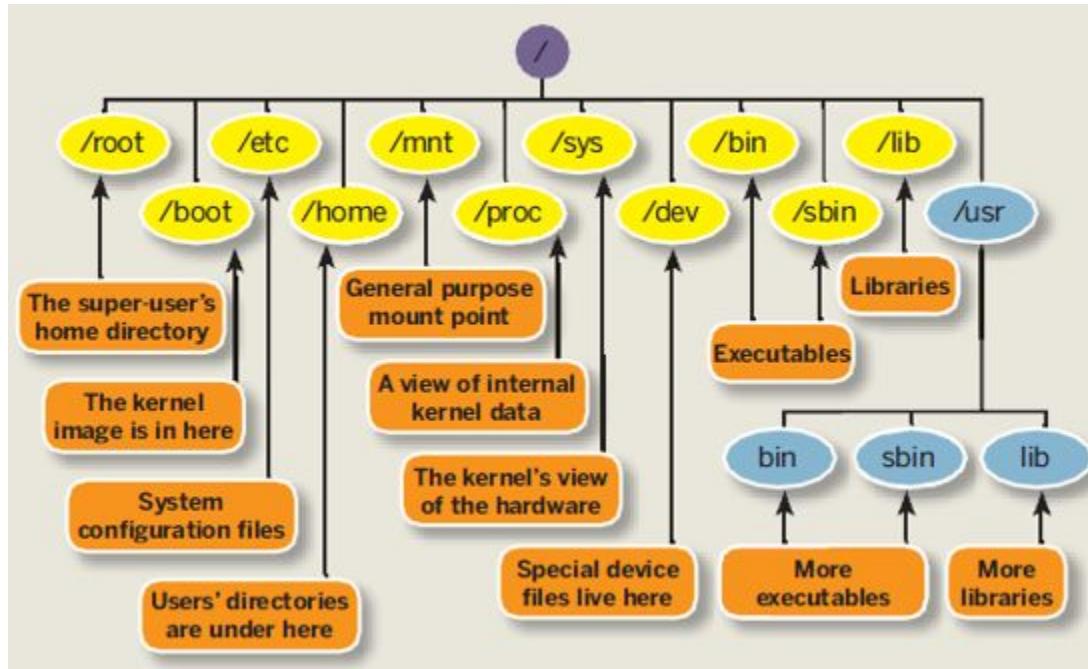
04 Linux Filesystem

- /sys
 - 커널을 통해 확인할 수 있는 하드웨어 등의 파일
 - 접근해서 파일을 수정하면 위험할 수 있음



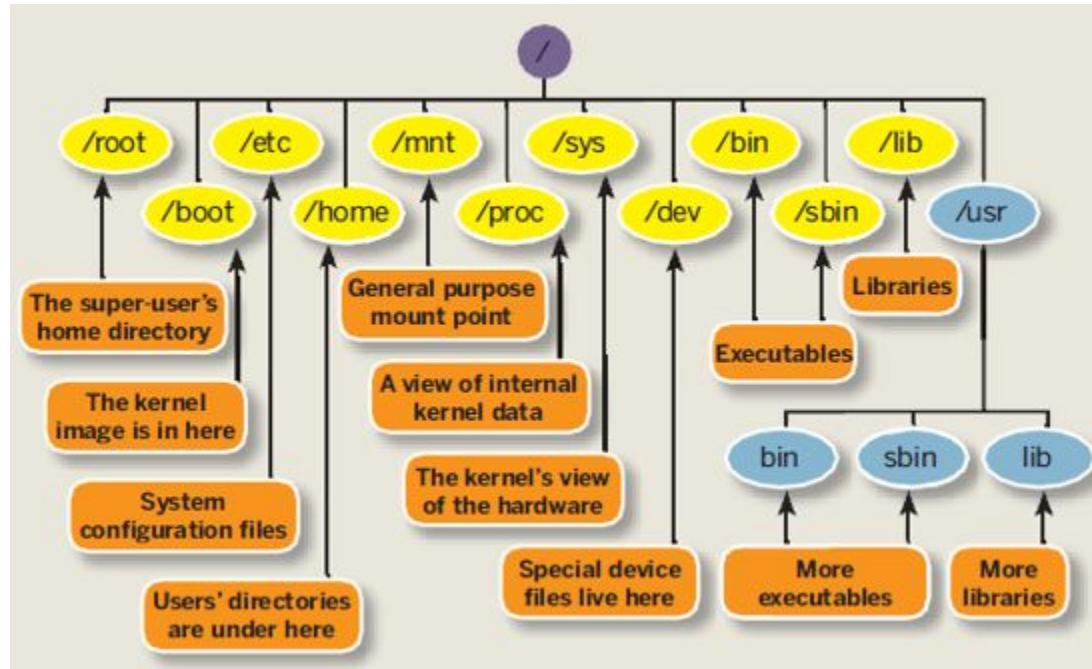
04 Linux Filesystem

- /dev
 - 연결된 장치를 확인할 수 있는 폴더
 - 리눅스에서는 장치들도 파일로서 관리되고 있음
 - 접근해서 파일을 수정하면 위험할 수 있음



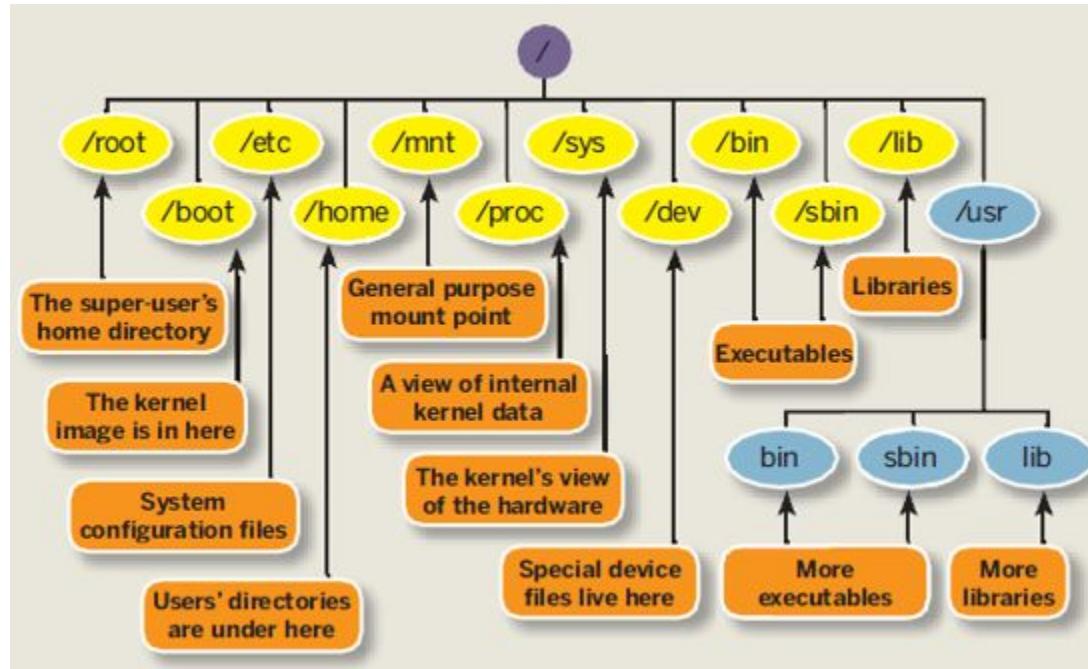
04 Linux Filesystem

- /bin
 - 실행할 수 있는 파일들이 위치한 폴더
 - 일반적으로 실행하는 명령어들의 Binary 파일들이 존재
 - 접근해서 파일을 수정하면 터미널 명령어를 사용하지 못할 수 있음



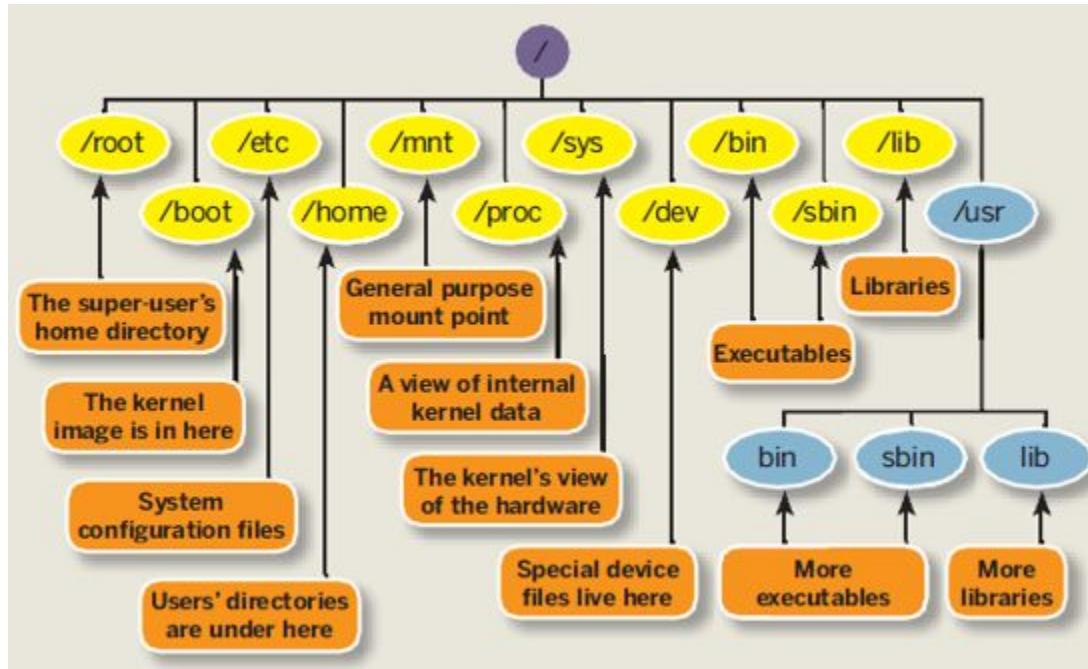
04 Linux Filesystem

- /sbin
 - 시스템 Binary 파일들이 존재
 - 시스템을 실행하기에 필수적인 실행 파일들이 존재
 - 접근해서 파일을 수정하면 문제가 될 수 있음



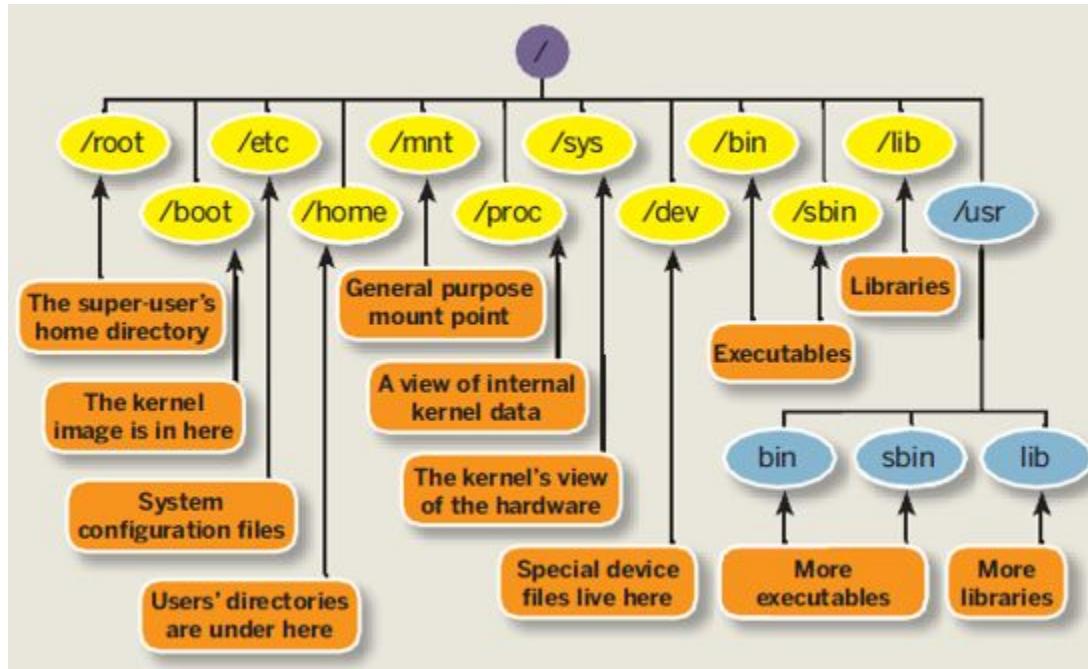
04 Linux Filesystem

- /lib
 - 시스템에서 사용하는 라이브러리 파일들이 존재
 - .so 또는 .a 파일들이 존재하는 폴더



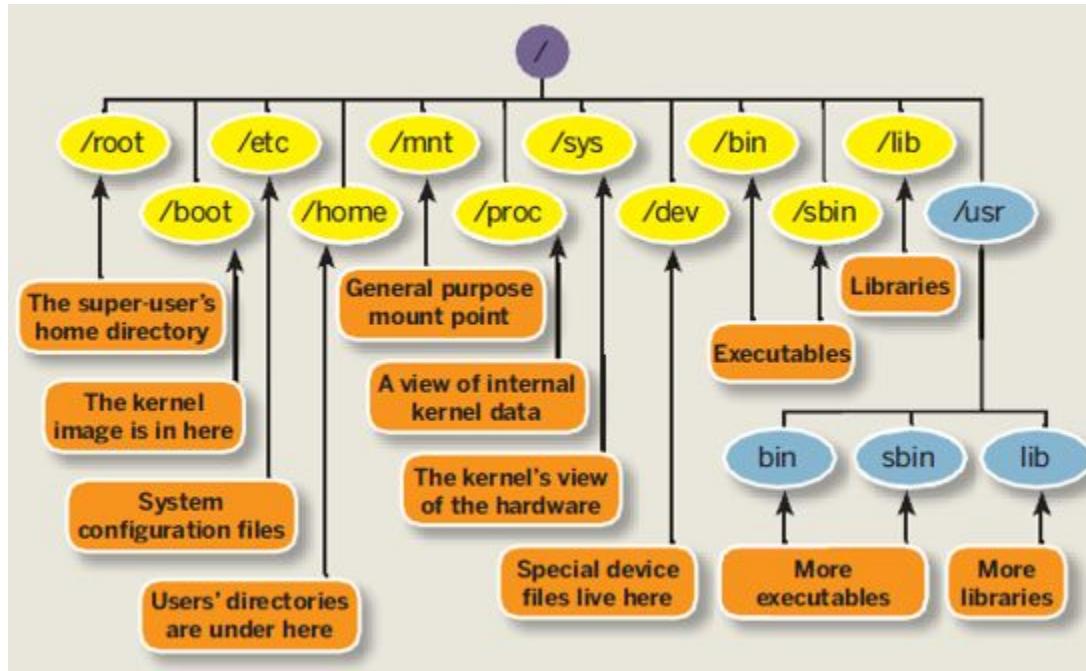
04 Linux Filesystem

- /usr
 - 일반 사용자가 사용할 수 있는 실행 파일, 라이브러리 등을 포함한 폴더
 - 수정하면 사용자의 명령 등을 사용할 수 없을 수 있음



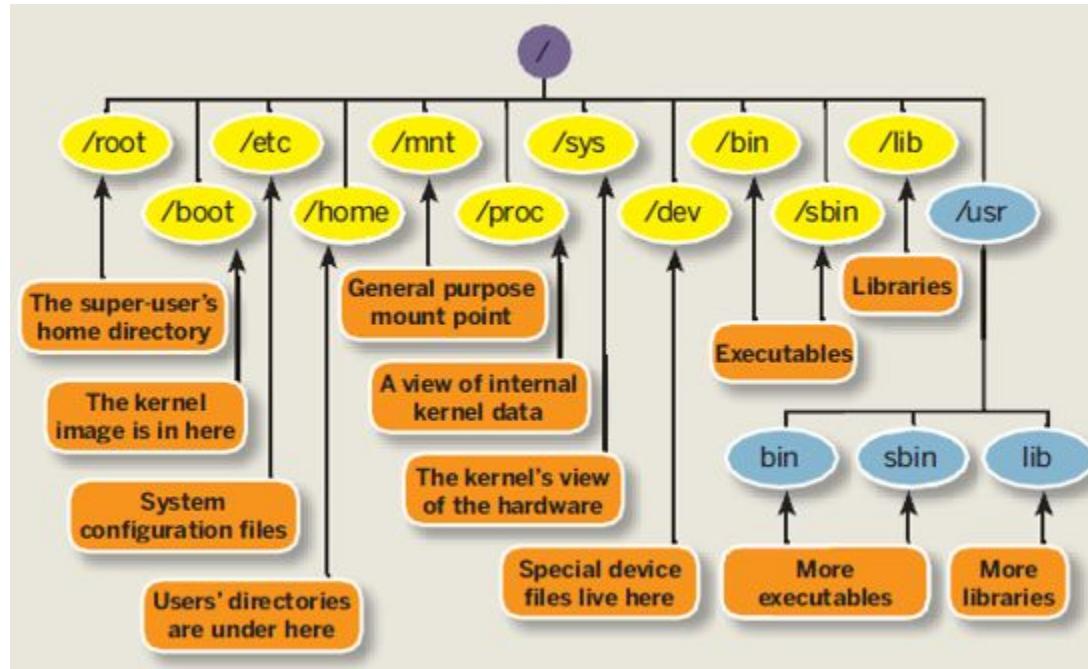
04 Linux Filesystem

- /tmp
 - 임시 파일들이 존재하는 폴더
 - 지우더라도 크게 문제 되지 않음



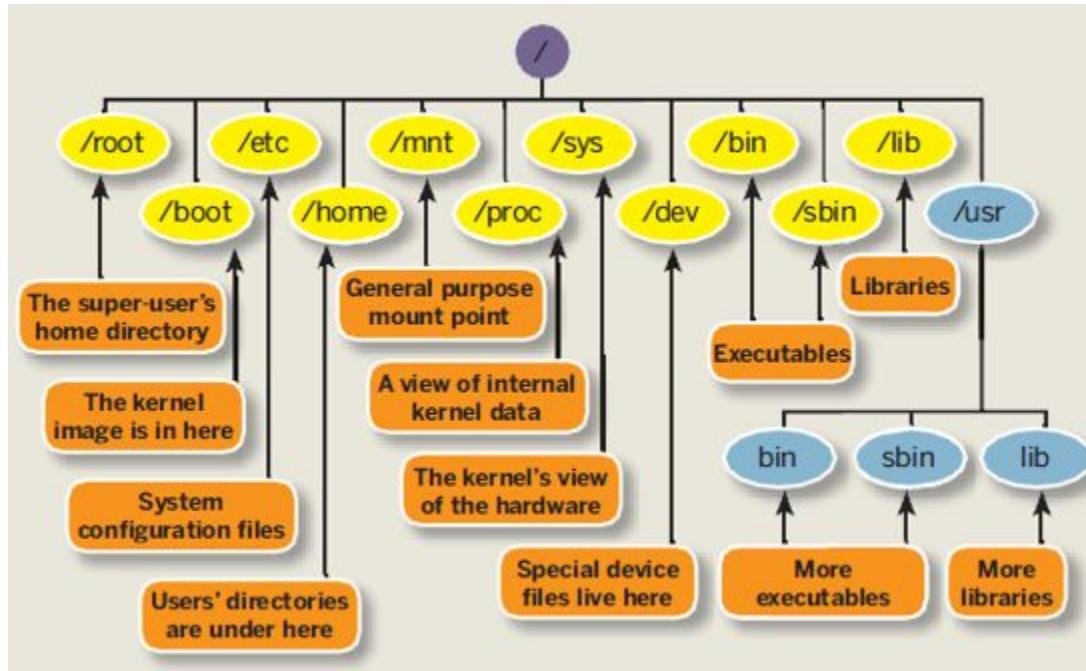
04 Linux Filesystem

- /opt
 - 필수적인 파일들이 아닌 Optional한 파일들이 위치한 폴더
 - 필수적이지 않은 소프트웨어를 설치하면 /opt 폴더에 설치되는 경우가 있음
 - 지워도 문제되지 않음



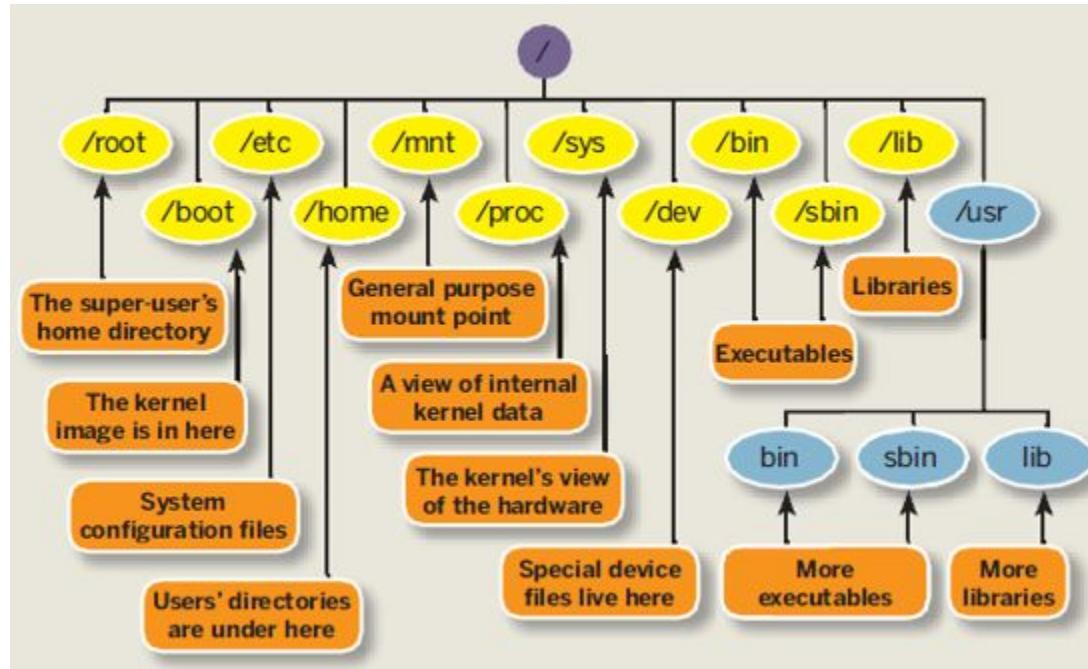
04 Linux Filesystem

- /var
 - 시스템에서 변경되는 값들이 존재하는 폴더
 - 잠금 파일, 로그 파일 등이 포함된 폴더



04 Linux Filesystem

- /lost+found
 - 파일 시스템 복구를 위한 파일들이 위치한 폴더
 - 지워도 크게 문제되지 않는다.



04 Ubuntu

04 Ubuntu

- Ubuntu
 - 리눅스 커널을 기반으로 한, 리눅스 배포판
 - 리눅스 배포판 중 가장 많이 사용되며, Debian 계열
 - 다른 배포판에 비해 편리한 설치 및 유지보수, 사용자 친화적인 사용법 및 인터페이스를 특징으로 가진다.
 - APT 를 이용하여 소프트웨어 설치, 관리, 제거를 할 수 있음



04 Ubuntu

- Ubuntu 버전
 - 매년 4월 및 10월에 정규 버전 출시
 - 짹수 해 4월에 LTS (Long Term Support) 버전이 출시
 - 정규 버전은 6개월마다 배포되며, 배포일 기준 9개월간 업데이트 및 지원
 - LTS 버전은 2년마다 배포되며, 배포일 기준 최대 5년까지 업데이트 및 지원

18.04 LTS	Bionic Beaver	2018년 4월 26일	2023년 4월	4.15
18.10	Cosmic Cuttlefish	2018년 10월 18일	2019년 7월 18일	4.18
19.04	Disco Dingo	2019년 4월 18일	2020년 1월 23일	5.0
19.10	Eoan Ermine	2019년 10월 17일	2020년 7월	5.3
20.04 LTS	Focal Fossa	2020년 4월 23일	2025년 4월	5.4
21.04	Hirsute Hippo	2021년 4월 22일	2022년 1월	5.11
21.10	Impish Indri	2021년 10월 14일	2022년 7월	5.13

05

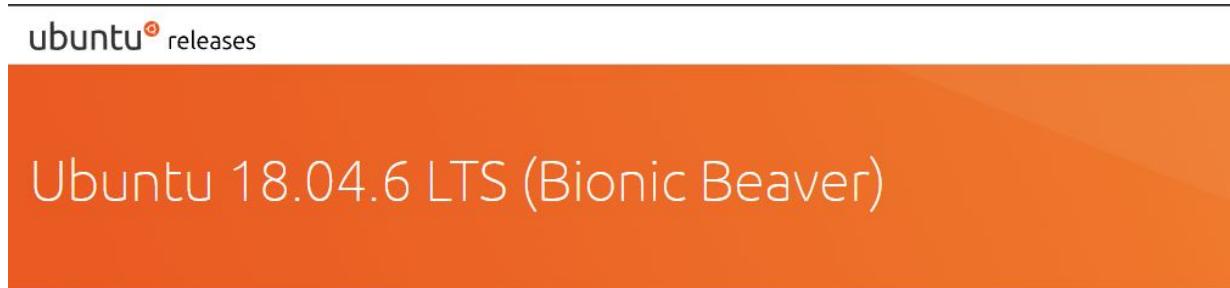
Ubuntu Installation

05 Ubuntu Installation

- USB 준비 (4GB 이상), Windows 가 설치된 PC 준비

1. Ubuntu image 파일 다운로드

- a. <https://releases.ubuntu.com/18.04> 에 접속
- b. Ubuntu Image (64 bit) 다운로드



Select an image

Ubuntu is distributed on three types of images described below.

클릭

Desktop image

The desktop image allows you to try Ubuntu without changing your computer at all, and at your option to install it permanently later. This type of image is what most people will want to use. You will need at least 1024MiB of RAM to install from this image.

64-bit PC (AMD64) desktop image

Choose this if you have a computer based on the AMD64 or EM64T architecture (e.g., Athlon64, Opteron, EM64T Xeon, Core 2). Choose this if you are at all unsure.

05 Ubuntu Installation

- USB 준비 (4GB 이상), Windows 가 설치된 PC 준비
2. Rufus 프로그램 다운로드
 - a. <https://rufus.ie/ko/> 접속
 - b. rufus 프로그램 다운로드



05 Ubuntu Installation

- USB 준비 (4GB 이상), Windows 가 설치된 PC 준비

3. Ubuntu 설치 USB 만들기

a. USB를 PC에 연결

b. Rufus 실행



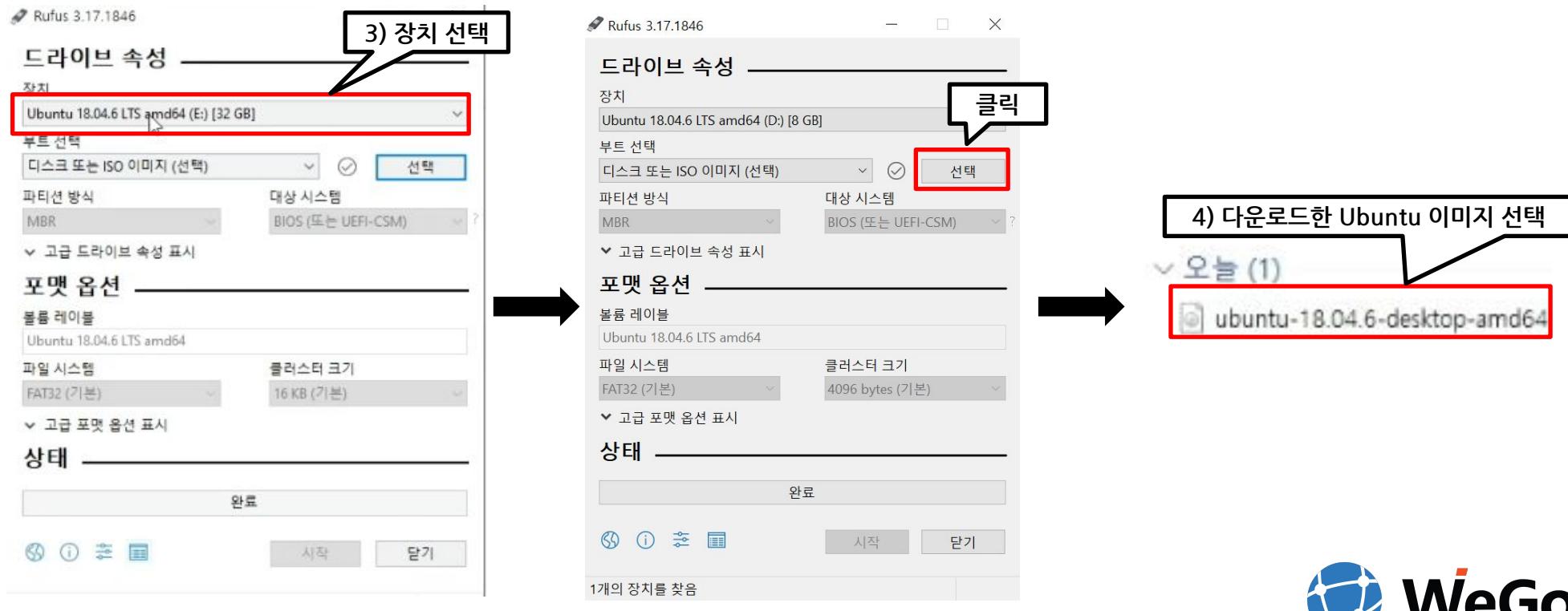
WeGo

05 Ubuntu Installation

- USB 준비 (4GB 이상), Windows 가 설치된 PC 준비

4. Ubuntu 설치 USB 만들기

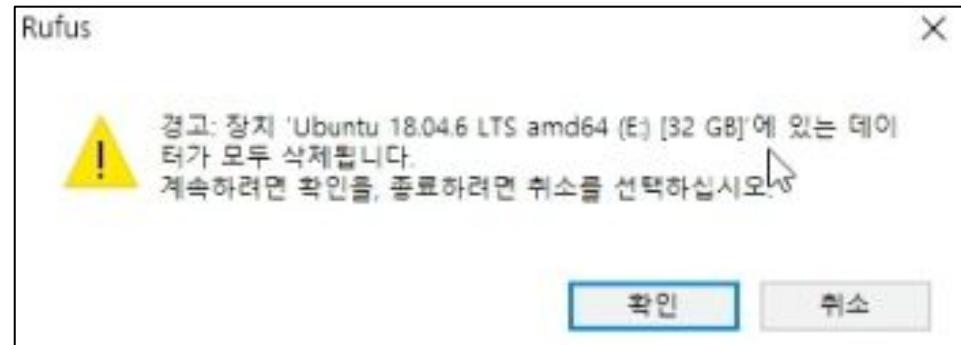
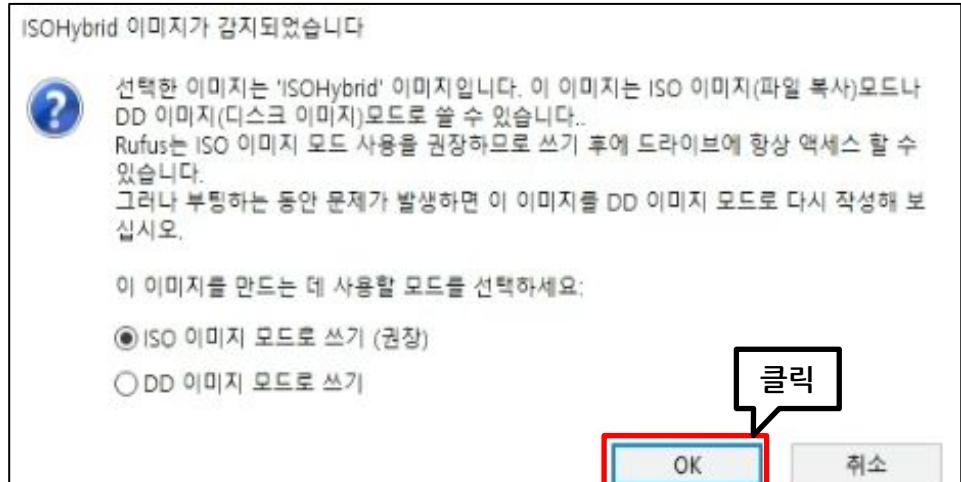
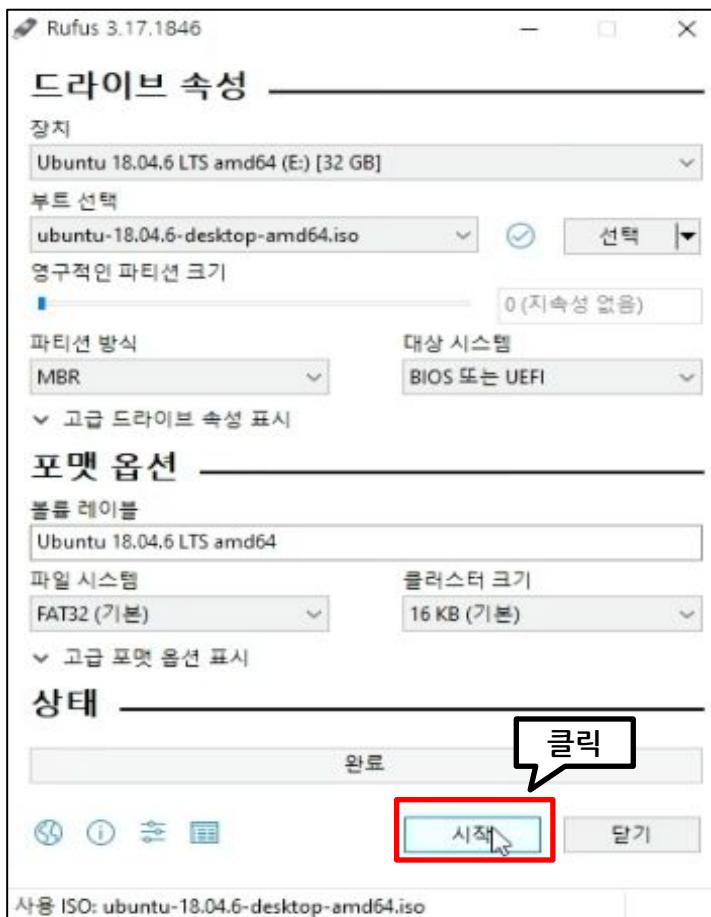
- 장치 선택 → 연결한 USB 선택
- 부트 선택 → 다운로드한 Ubuntu 이미지 선택



05 Ubuntu Installation

- USB 준비 (4GB 이상), Windows 가 설치된 PC 준비

5. USB 포맷 시작 (**주의**. USB 데이터가 모두 삭제되니 미리 백업하십시오)

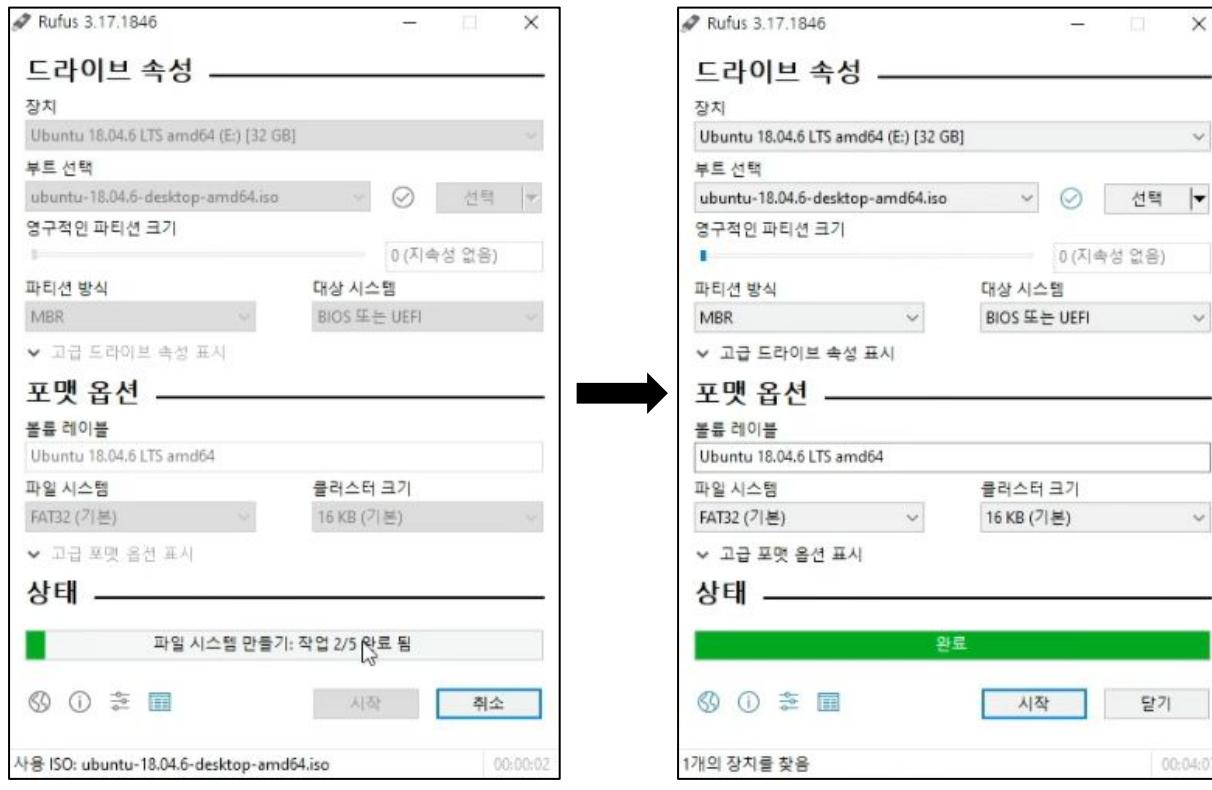


05 Ubuntu Installation

- USB 준비 (4GB 이상), Windows 가 설치된 PC 준비

6. USB 포맷 시작

- a. 포맷이 완료되고, 부팅 USB가 완성됩니다.
- b. 완성된 후, Ubuntu를 설치할 PC에 USB를 연결한 후, 재부팅합니다.



05 Ubuntu Installation

- USB 준비 (4GB 이상), Windows 가 설치된 PC 준비

7. BIOS 설정 진입

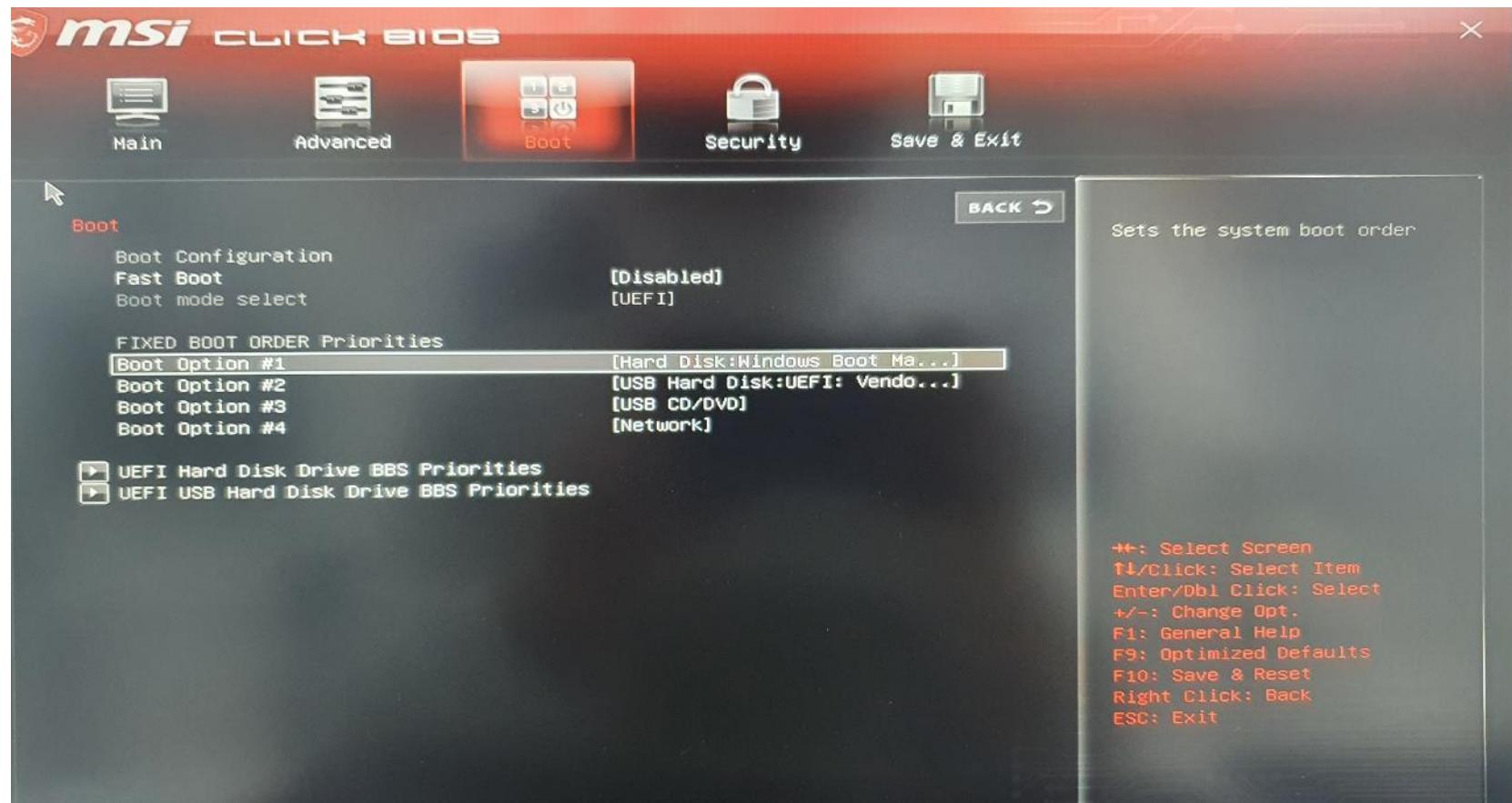
- a. 컴퓨터 부팅화면에서 특정 키를 입력하여 BIOS에 진입합니다.
(제조사 별로 BIOS 진입 방법이 다르므로, 확인하여 진입합니다.)



05 Ubuntu Installation

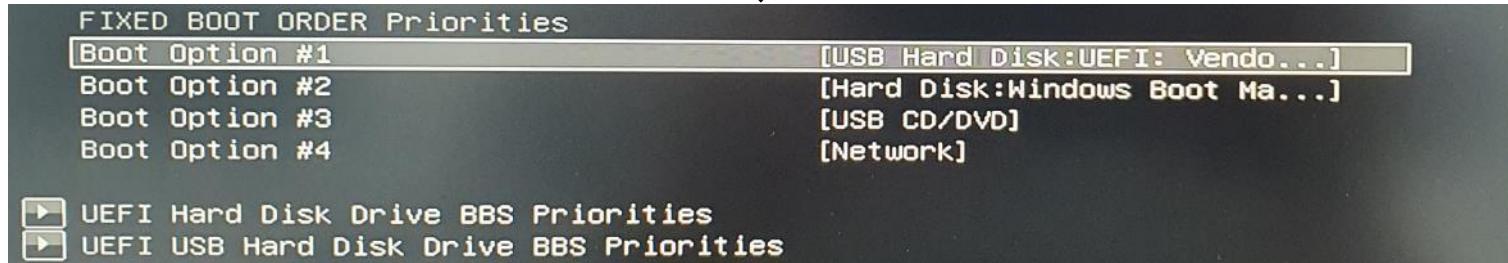
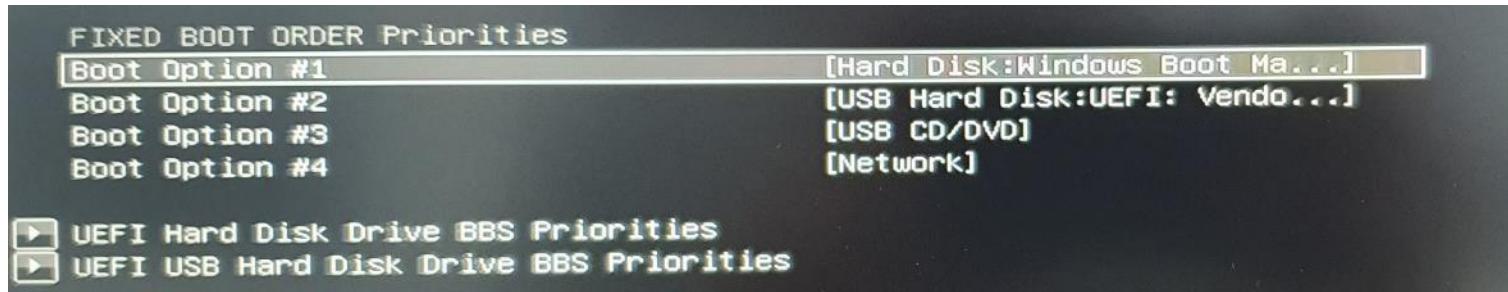
- USB 준비 (4GB 이상), Windows 가 설치된 PC 준비

8. Boot 옵션으로 이동



05 Ubuntu Installation

- USB 준비 (4GB 이상), Windows 가 설치된 PC 준비
9. 부팅 우선순위 변경 (첫 번째 옵션이 Ubuntu 설치 USB가 되도록 설정)

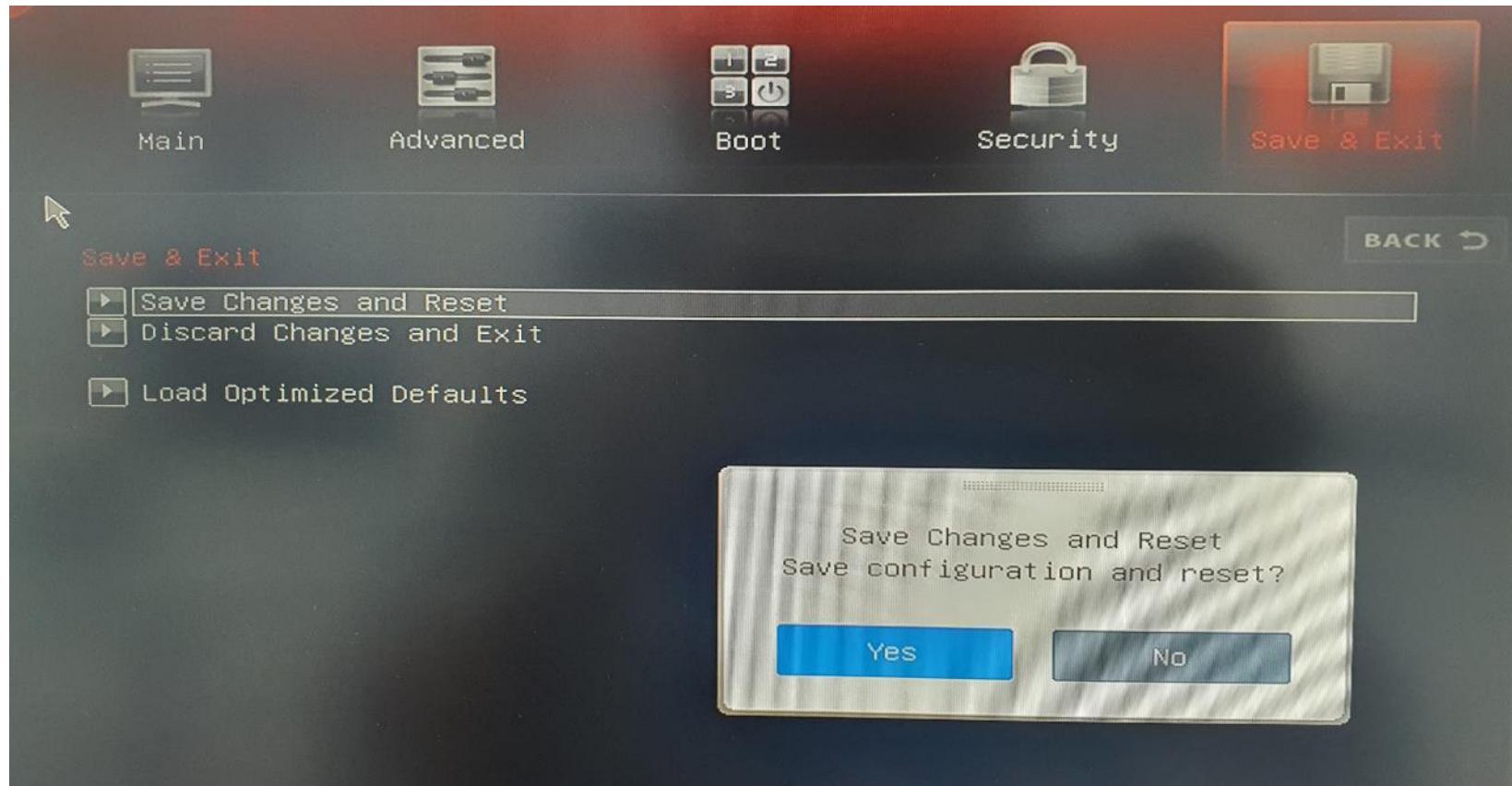


WeGo

05 Ubuntu Installation

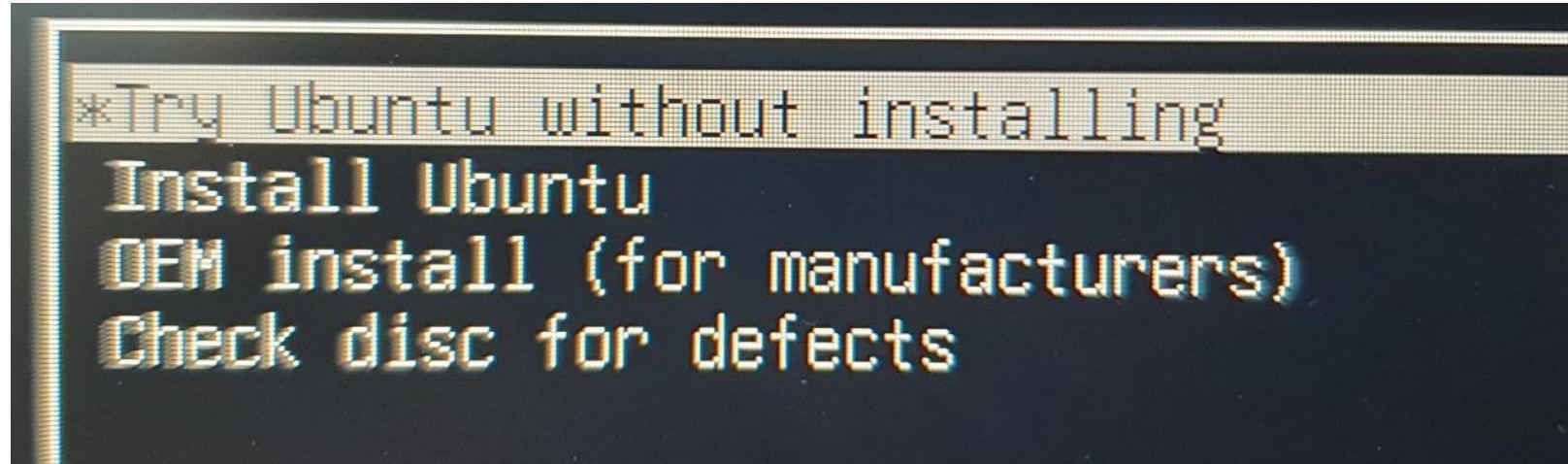
- USB 준비 (4GB 이상), Windows 가 설치된 PC 준비

10. 저장 후 나가기



05 Ubuntu Installation

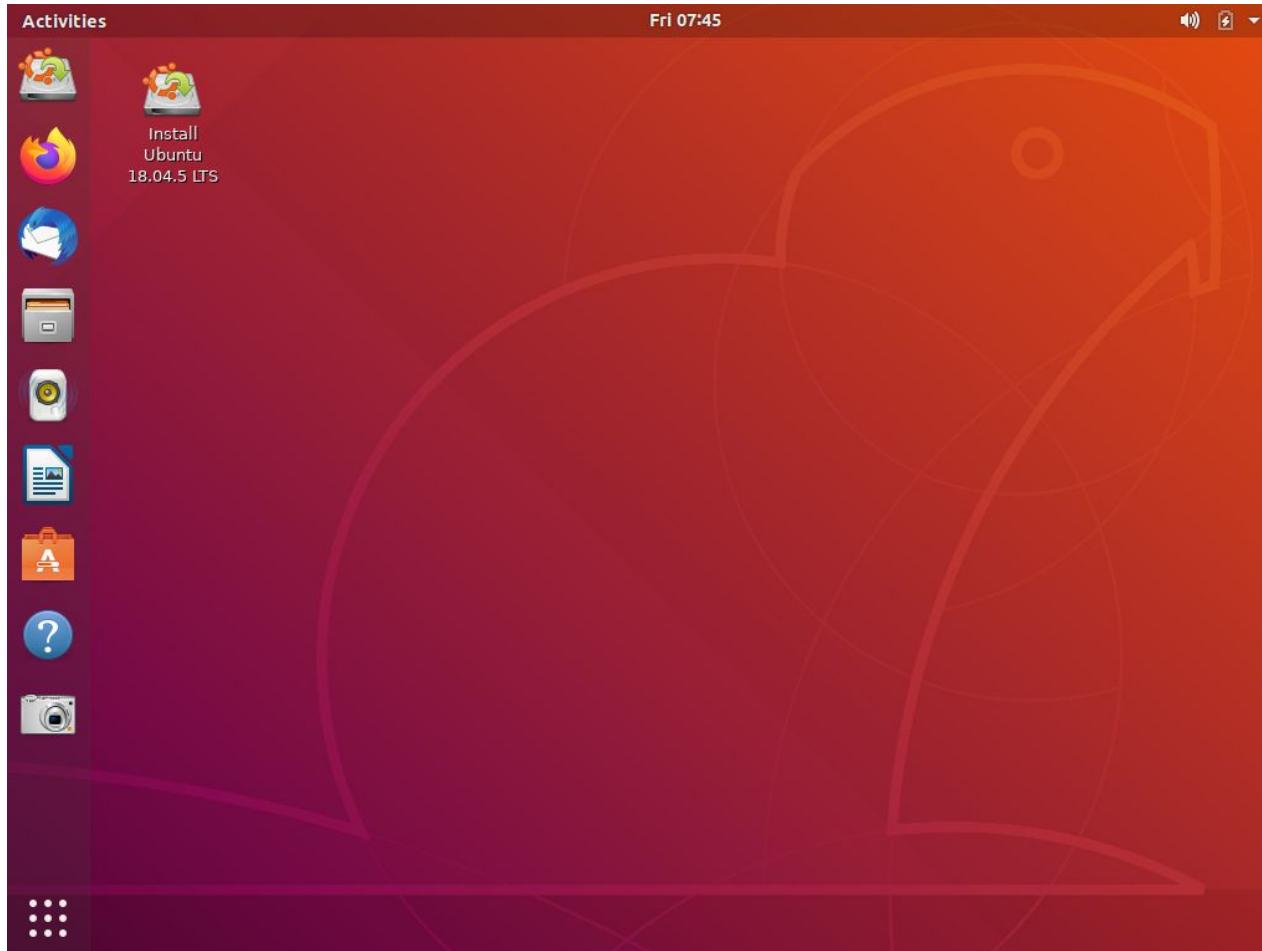
- USB 준비 (4GB 이상), Windows 가 설치된 PC 준비
11. 재부팅 시 Ubuntu USB로 부팅이 되면 다음 옵션을 선택
- a. Try Ubuntu without installing 옵션 선택



05 | Ubuntu Installation

- USB 준비 (4GB 이상), Windows 가 설치된 PC 준비

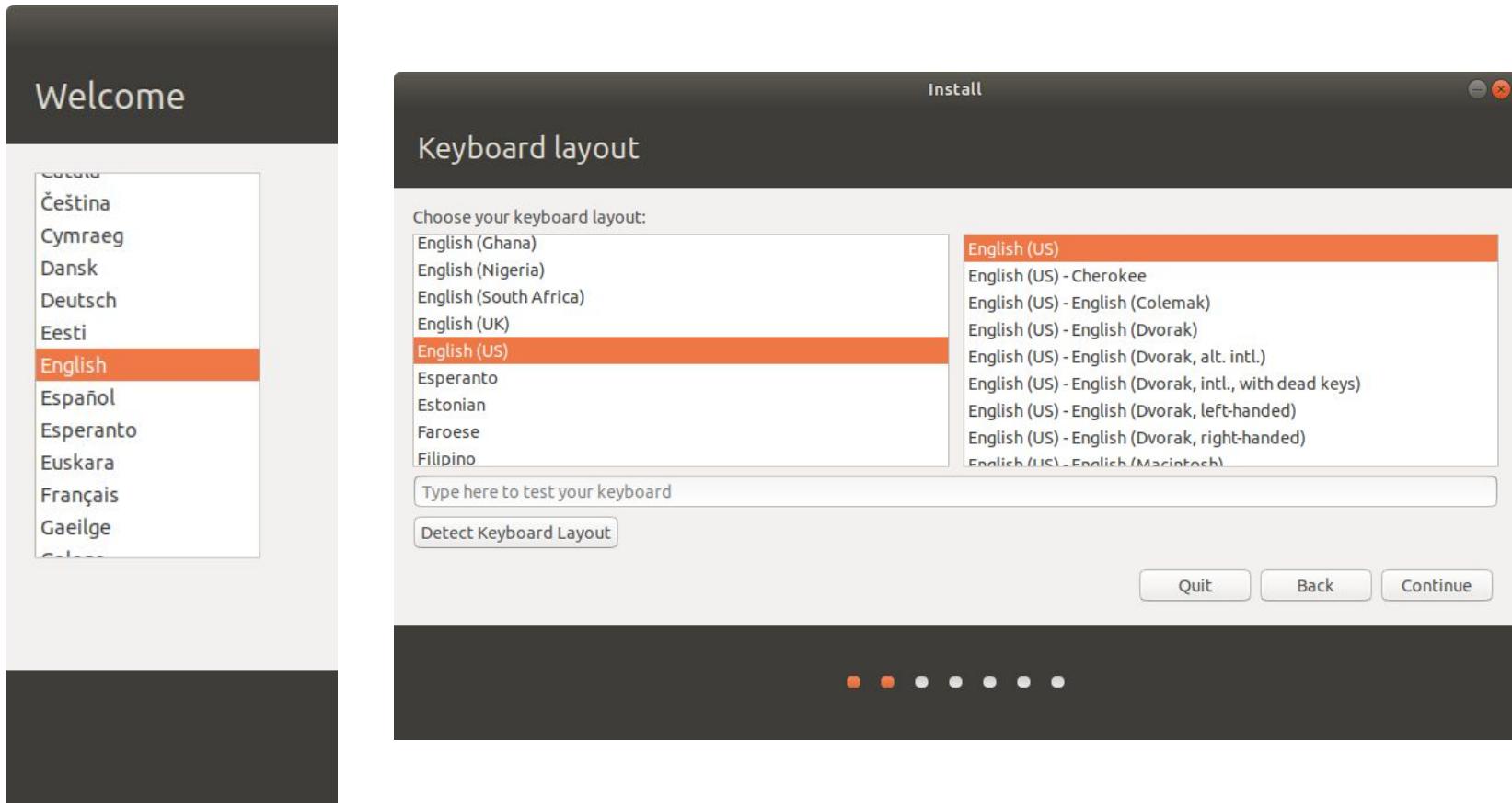
12. 바탕화면에 Install Ubuntu 아이콘을 더블클릭하여 실행



05 Ubuntu Installation

- USB 준비 (4GB 이상), Windows 가 설치된 PC 준비

13. 언어 선택 시 English 옵션 선택하기

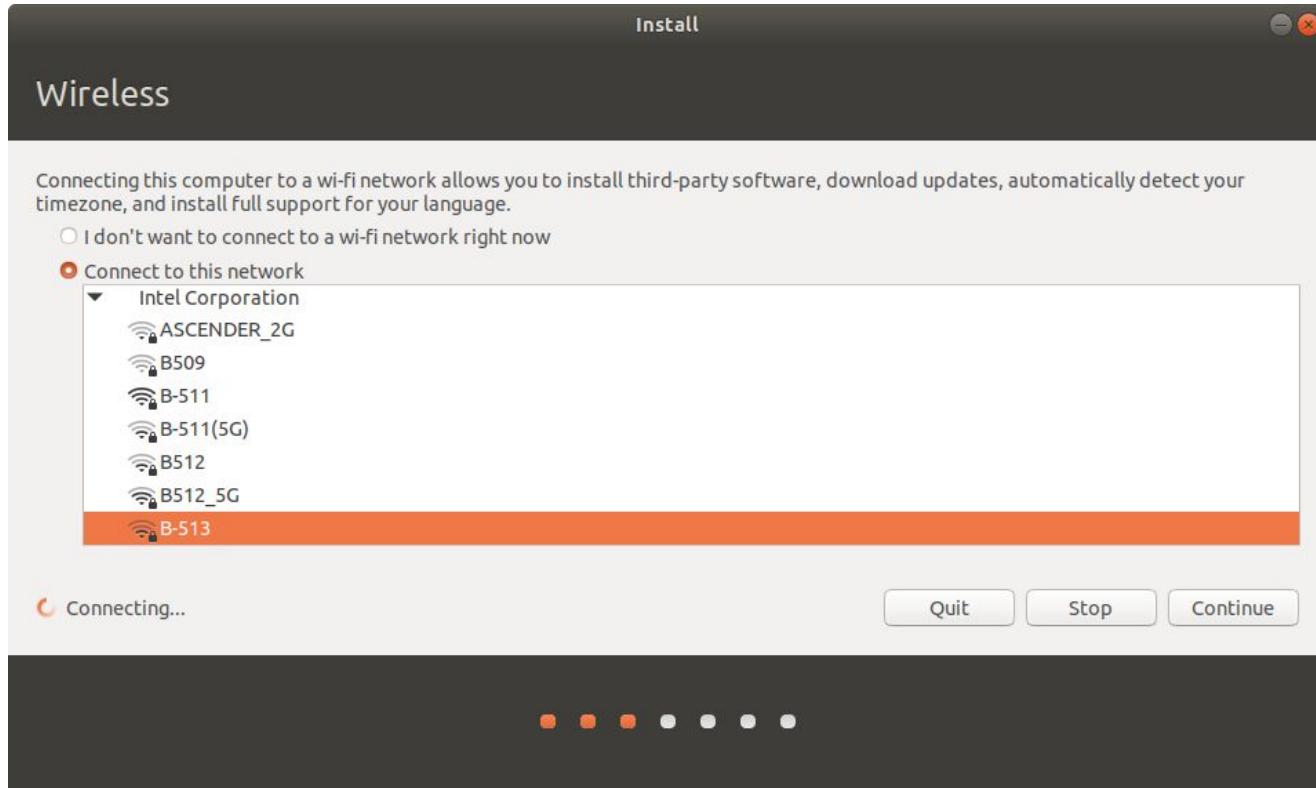


05 Ubuntu Installation

- USB 준비 (4GB 이상), Windows 가 설치된 PC 준비

14. 인터넷 연결하기

- a. 연결할 수 없는 상태라면 생략 가능
- b. 연결된 것을 확인 후 진행

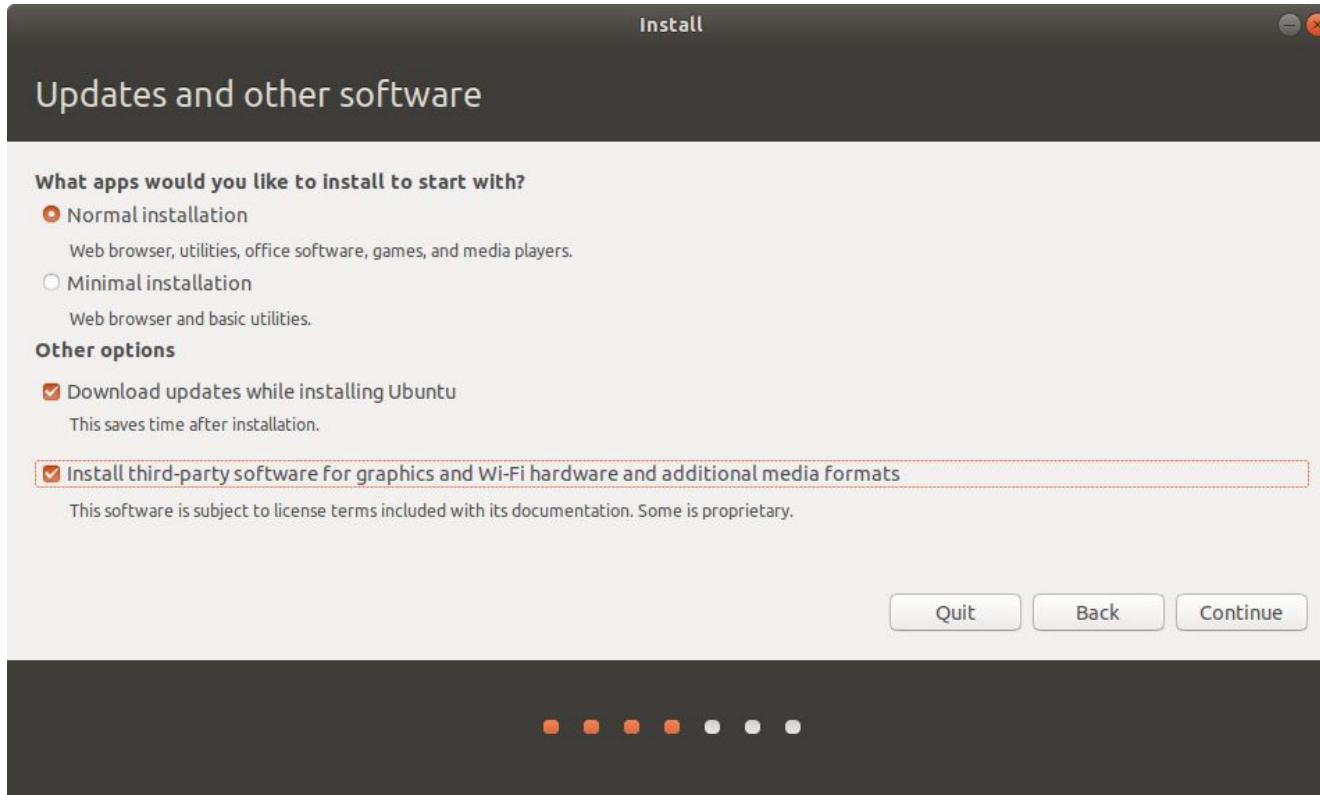


05 Ubuntu Installation

- USB 준비 (4GB 이상), Windows 가 설치된 PC 준비

15. 업데이트 및 소프트웨어 설치 옵션 제외

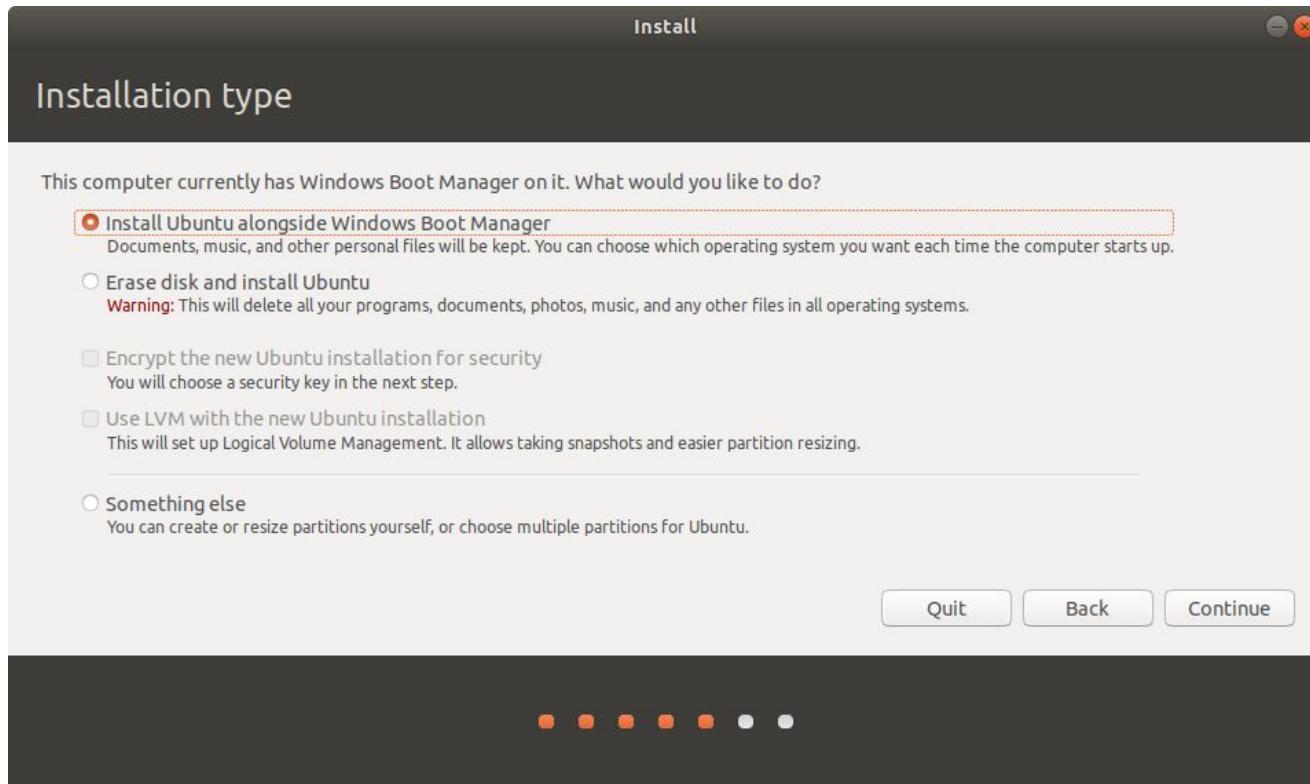
- a. Install third-party software for graphics and Wi-Fi hardware and additional media formats **체크 해제**



05 Ubuntu Installation

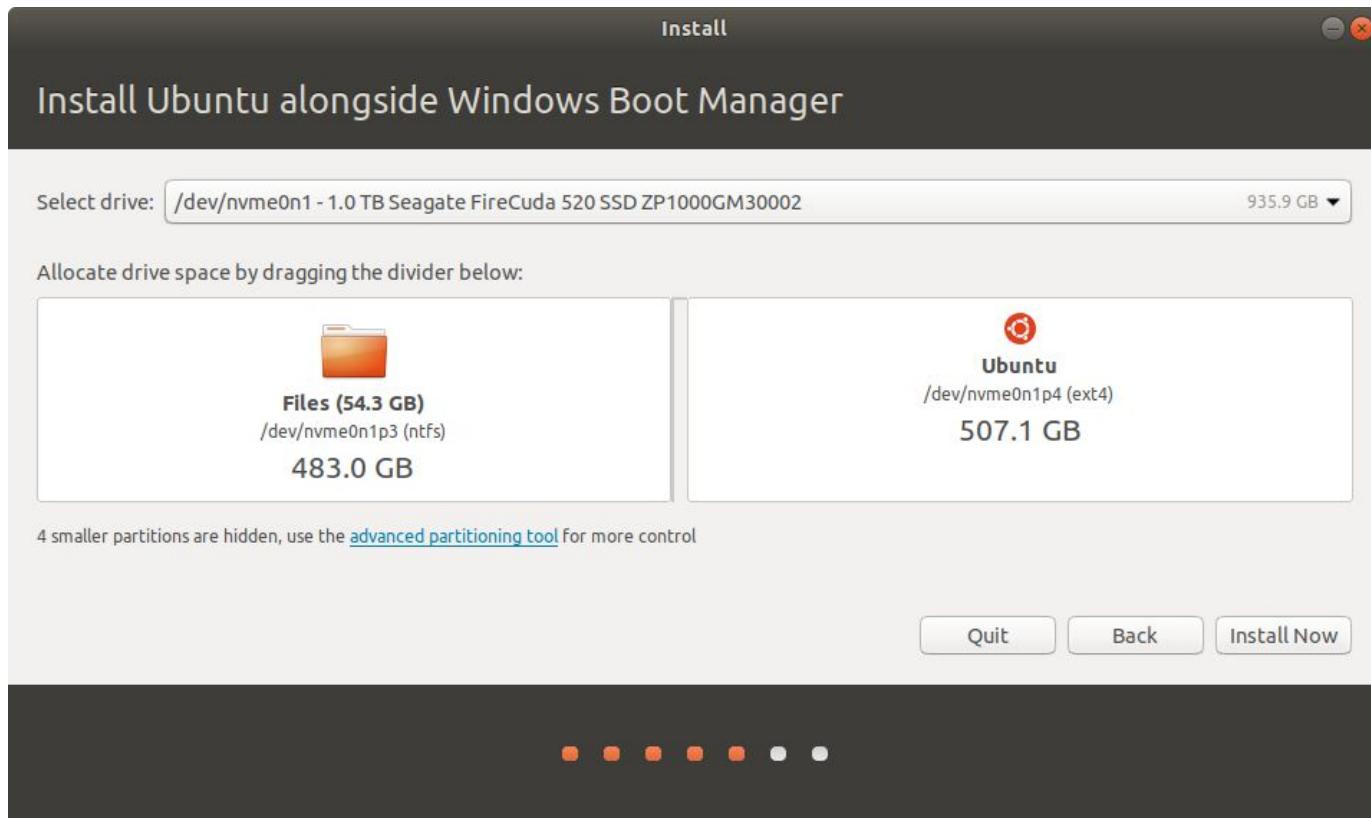
- USB 준비 (4GB 이상), Windows 가 설치된 PC 준비

16. Installation type에서 **Install Ubuntu alongside Windows Boot Manager** 선택



05 Ubuntu Installation

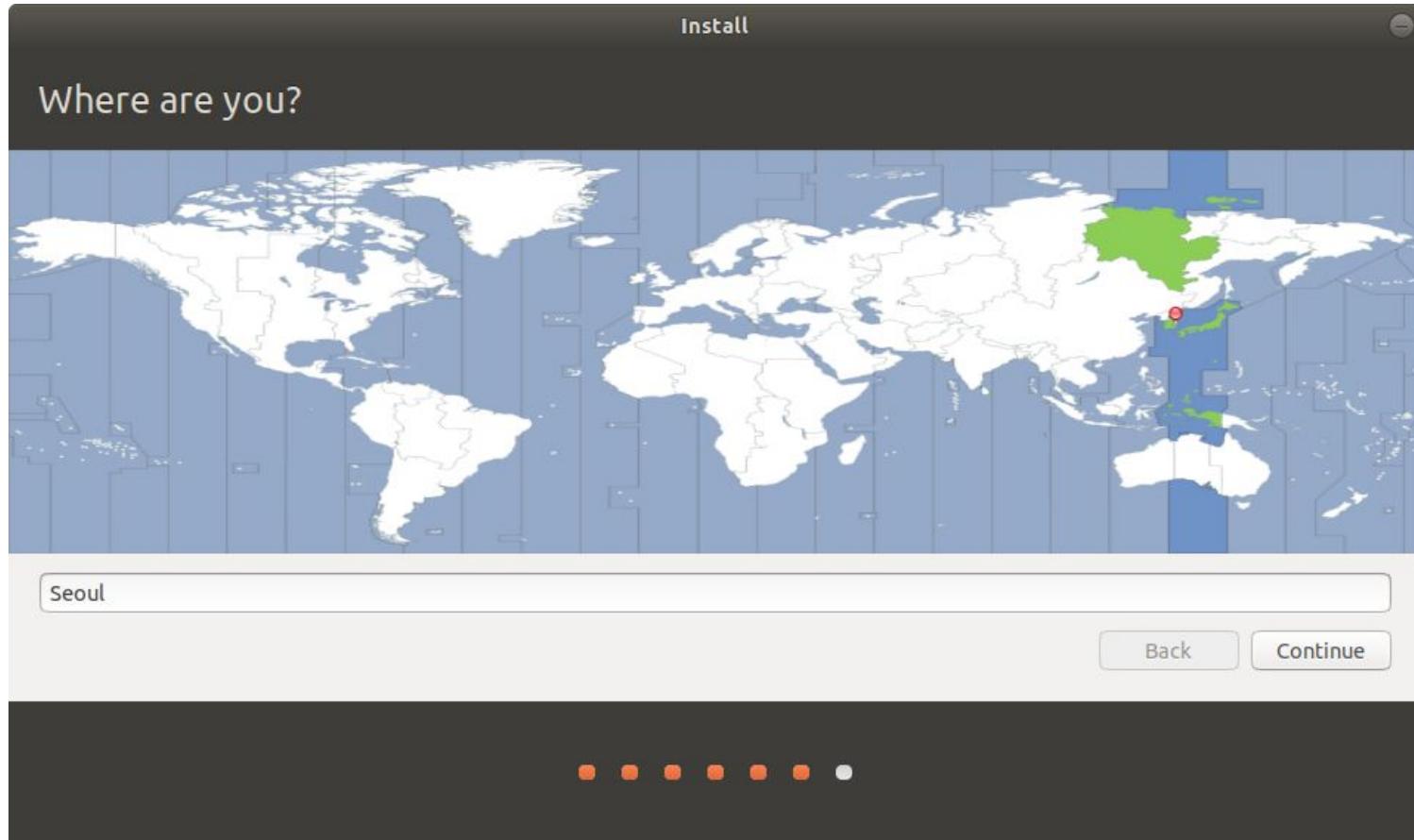
- USB 준비 (4GB 이상), Windows 가 설치된 PC 준비
17. Ubuntu에 할당할 저장공간 지정 (진행 시 나오는 창에서 Continue 선택)



05 Ubuntu Installation

- USB 준비 (4GB 이상), Windows 가 설치된 PC 준비

18. 지역 Seoul로 선택

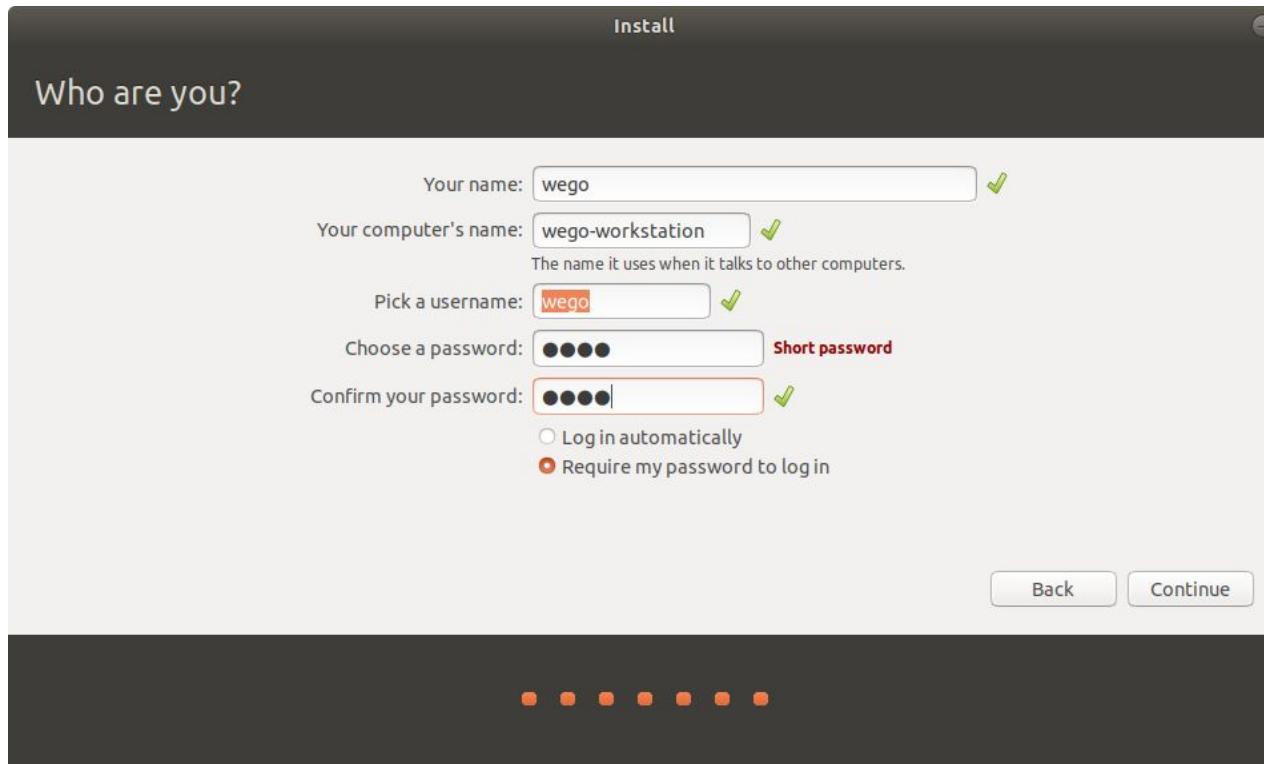


05 Ubuntu Installation

- USB 준비 (4GB 이상), Windows 가 설치된 PC 준비

19. 로그인 정보 입력

- a. 영어로 작성
- b. 공백(space) 사용 시 에러 발생할 수 있으므로, ‘-’ 또는 ‘_’로 대체

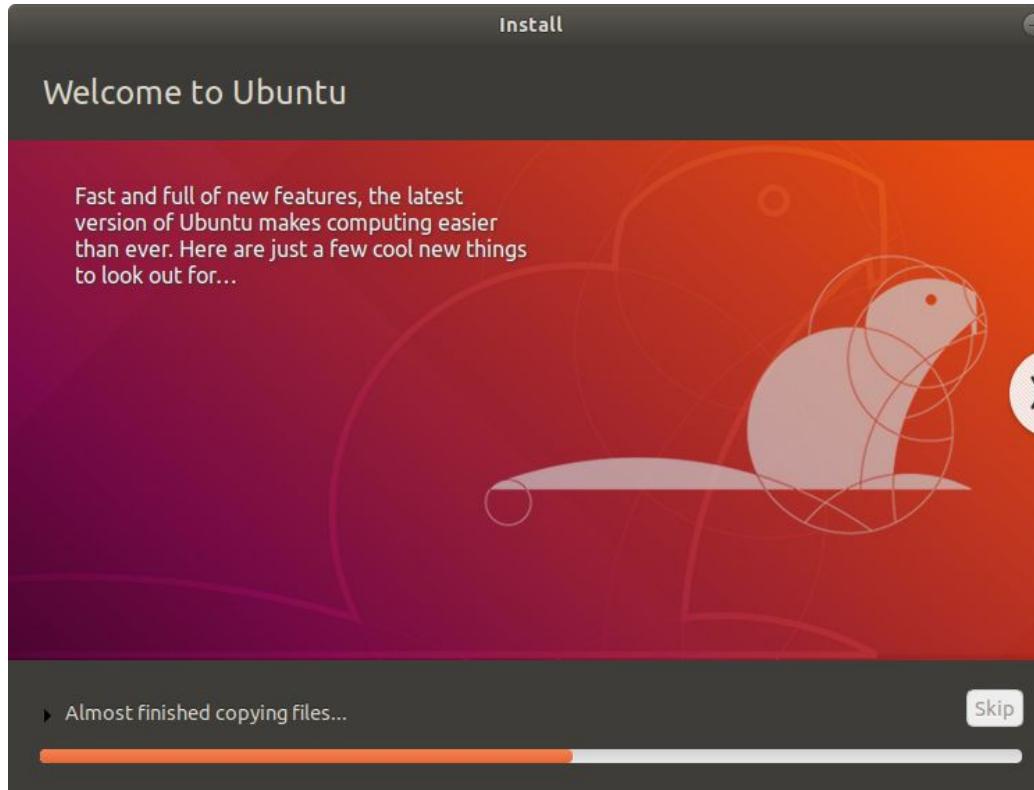


05 Ubuntu Installation

- USB 준비 (4GB 이상), Windows 가 설치된 PC 준비

20. Ubuntu 설치 시작

- a. 설치가 완료되면 나오는 창에서 Restart Now를 클릭
- b. 재부팅이 시작되면, 설치 USB를 빼기

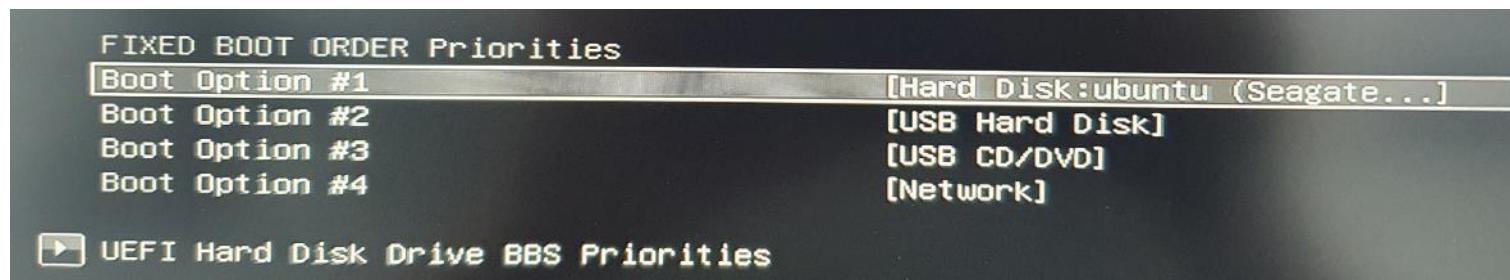
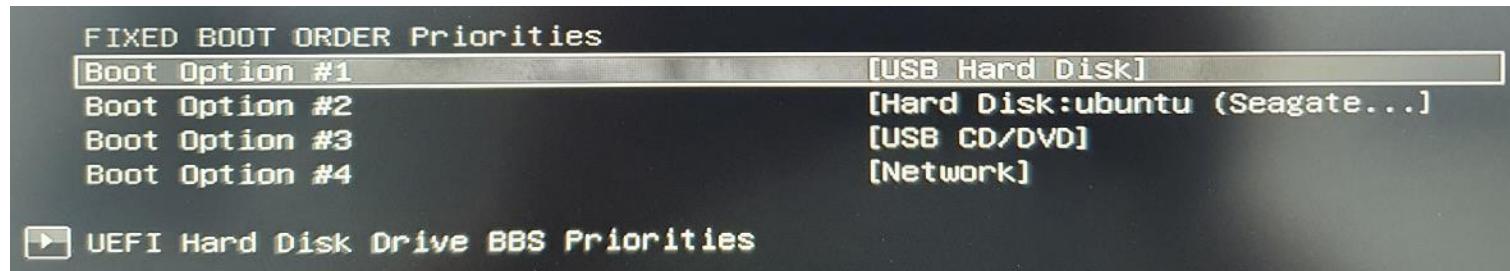


05 Ubuntu Installation

- USB 준비 (4GB 이상), Windows 가 설치된 PC 준비

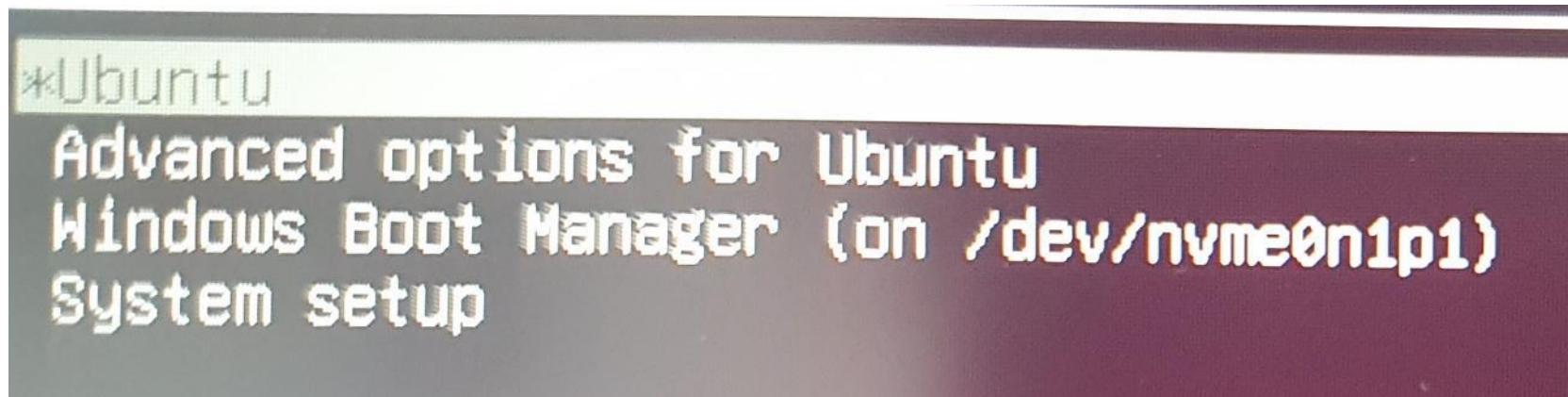
21. 부팅 우선 순위 변경

a. 첫 번째 옵션이 Ubuntu가 되도록 변경



05 Ubuntu Installation

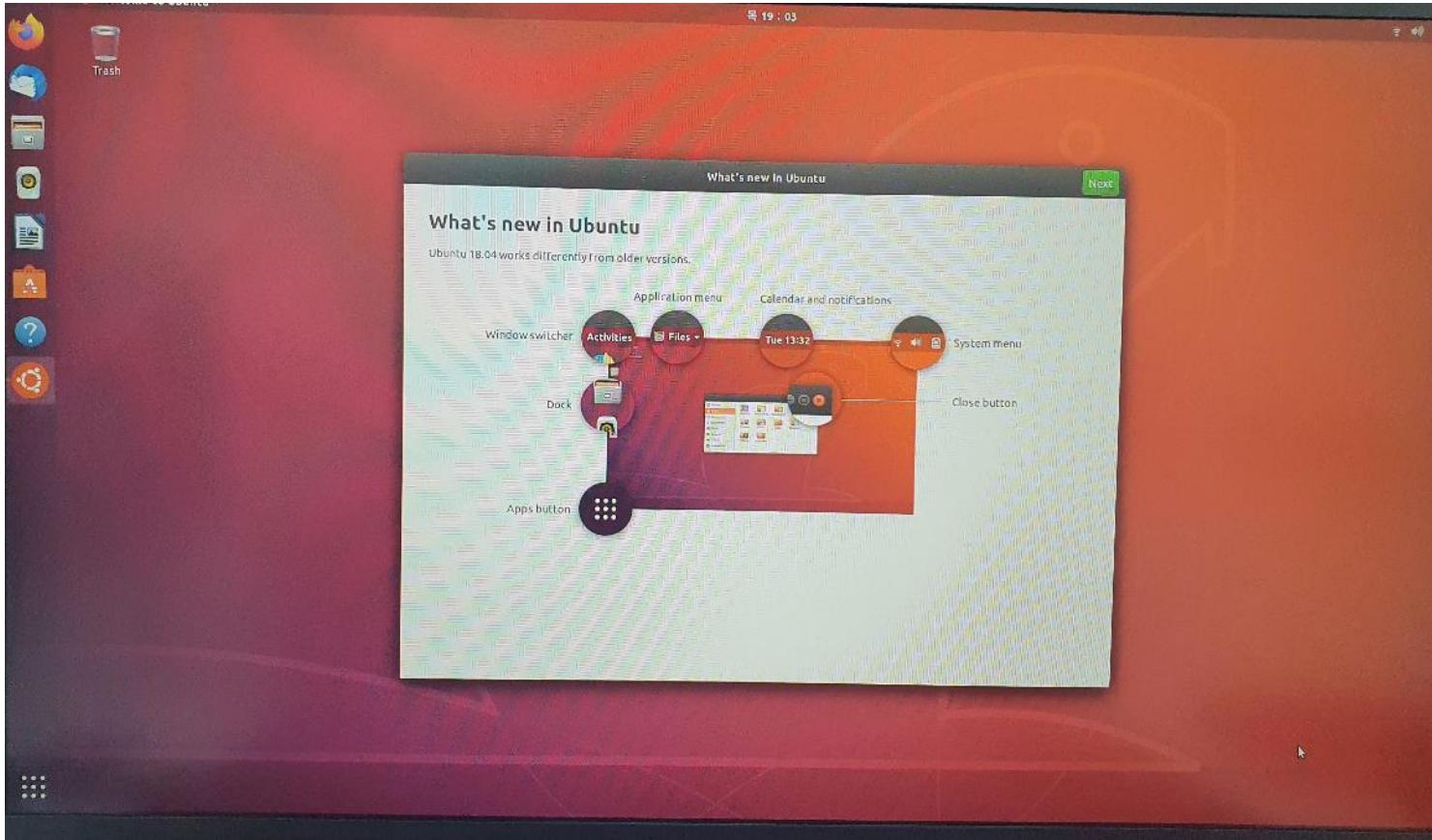
- USB 준비 (4GB 이상), Windows 가 설치된 PC 준비
22. 저장 후, 재부팅 시, 'Ubuntu' 를 선택하여 부팅



05 Ubuntu Installation

- USB 준비 (4GB 이상), Windows 가 설치된 PC 준비

22. 로그인 후, Ubuntu 진입 완료



06

Linux Command

06 Linux Command

- Bash(Bourne-Again Shell) 란?
 - 현재 리눅스의 표준 쉘
 - Alias : 명령어 및 단축 기능
 - 연산 기능
 - History 기능
 - 자동 이름 완성 기능
 - 프롬프트 제어 기능 등이 존재
 - Linux Command를 Bash에 입력하여 명령을 전달
 - 이후의 설명은 Bash를 기준으로 설명합니다.

```
wego@wego-GF63-Thin-10SCXR:~$ uname -r  
5.4.0-90-generic  
wego@wego-GF63-Thin-10SCXR:~$ echo Hello World  
Hello World  
wego@wego-GF63-Thin-10SCXR:~$ echo Hello Linux  
Hello Linux  
wego@wego-GF63-Thin-10SCXR:~$ █
```



06 Linux Command

- Terminal
 - Linux에서 Terminal을 열기 위해서 “윈도우”키를 누른 후, Terminal을 검색하여 열 수 있다.
 - Terminal은 Tilix Terminal을 기준으로 설명드리겠습니다.



06 Linux Command

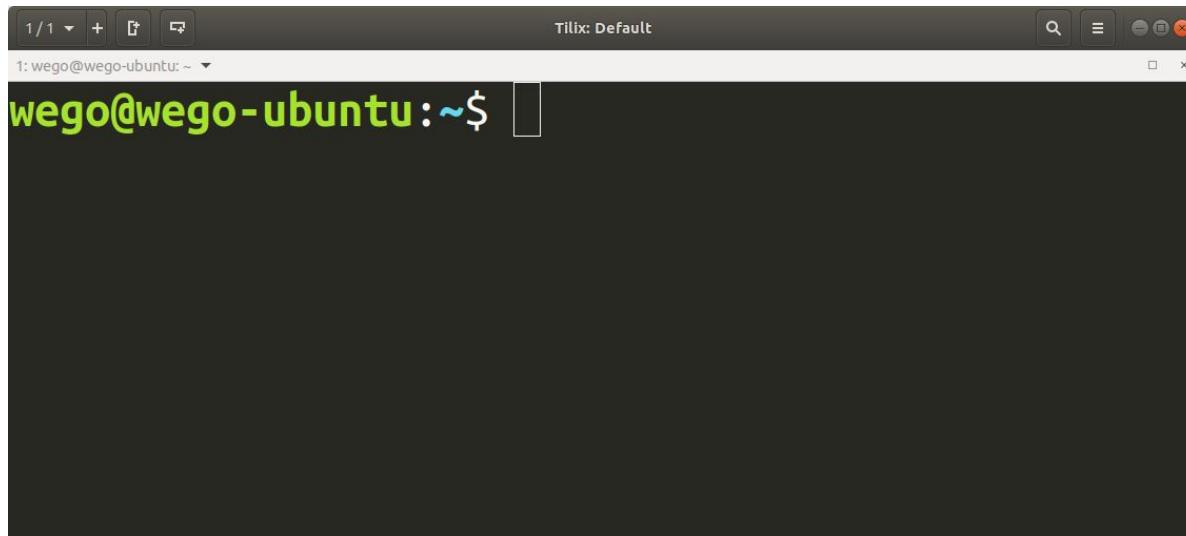
- 터미널을 실행하는 단축키는 Ctrl + Alt + T 를 입력하면 볼 수 있다.
- 우선 터미널을 열어서 Tilix 터미널을 설치하겠습니다.
- Ctrl + Alt + T 를 입력하여, 기본 터미널을 열고, 아래 명령어를 입력합니다.
- 명령어의 \$ 표시는 리눅스 터미널에 입력한다는 것을 의미합니다.
 - \$ sudo apt install tilix
- 위 명령어를 분해해서 보면
 - sudo → 관리자 권한으로 이후의 명령어를 실행한다는 의미
 - apt install → apt 패키지 관리자로 패키지를 설치하겠다는 의미
 - tilix → 설치하려는 파일의 이름
 - 예를 들어 hello_world 라는 파일을 설치한다면 아래와 같이 입력하면 된다.
 - \$ sudo apt install hello_world

```
wego@wego-ubuntu:~$ sudo apt install tilix
```

06 Linux Command

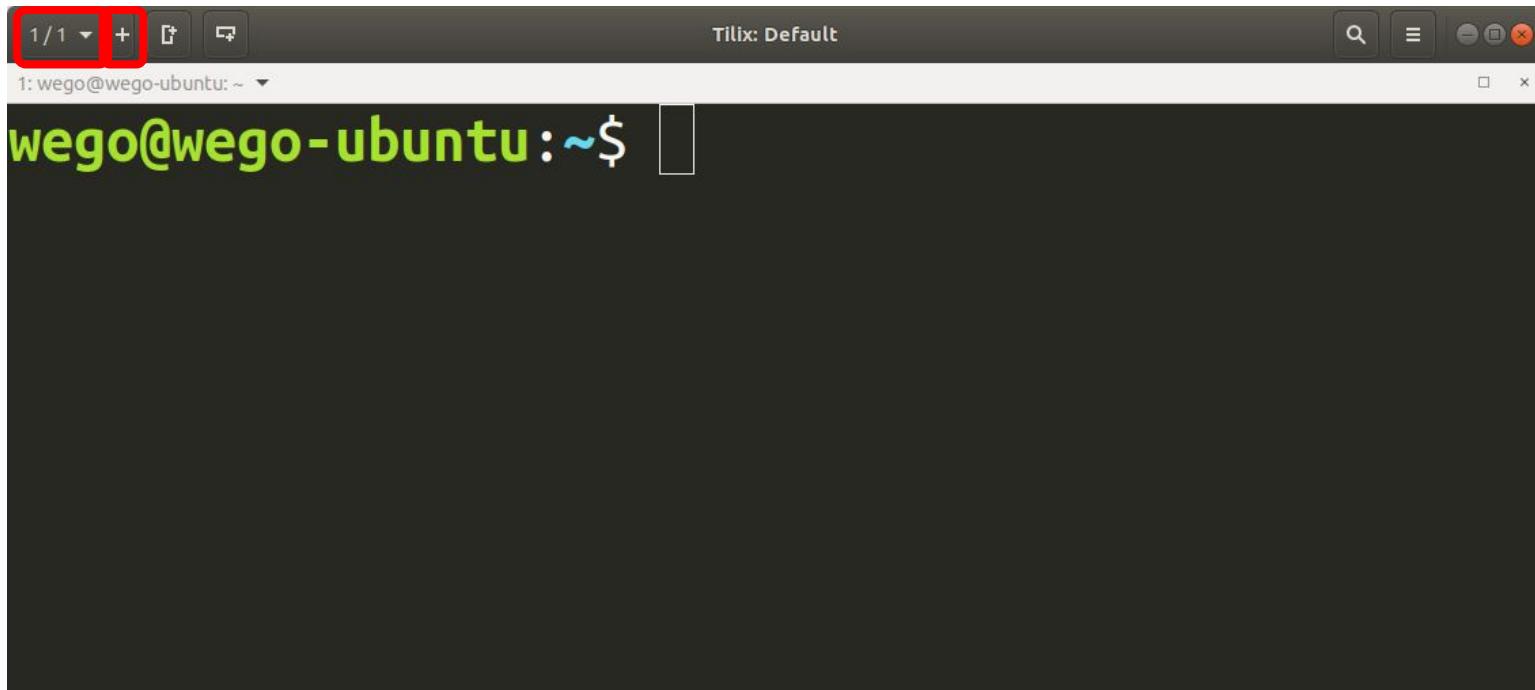
- Tilix가 정상적으로 설치되었다면 Ctrl + Alt + T 를 입력하여 터미널을 실행하였을 때, 기본 Terminal이 아닌, Tilix Terminal로 실행이 됩니다.
- 기본 터미널로 실행 되지 않을 경우는 아래의 링크를 참고하시면 됩니다.

<https://websetnet.net/ko/how-to-change-the-default-terminal-in-ubuntu/>



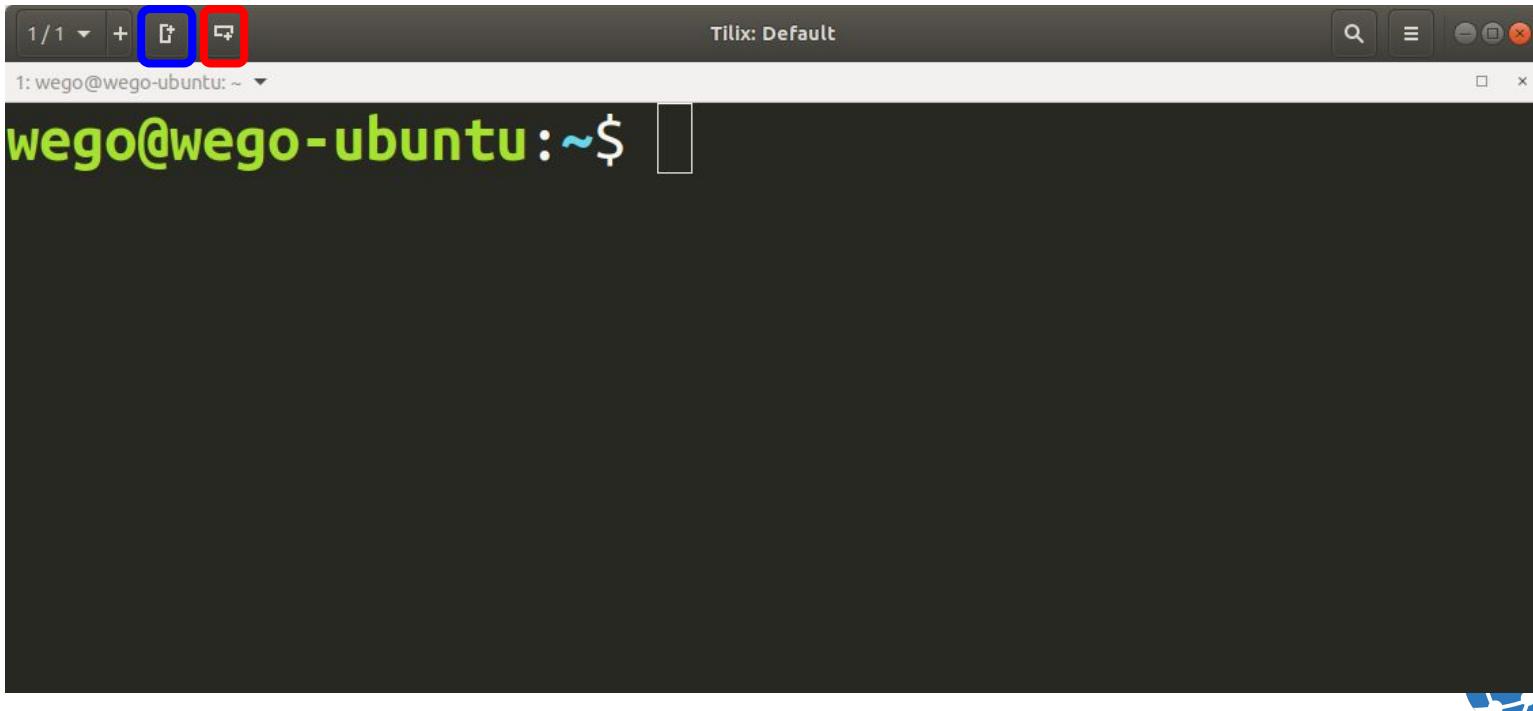
06 Linux Command

- Tilix 터미널의 경우, 상하, 좌우로 분할하여 사용할 수 있으며, 여러 개의 탭을 추가하여 사용할 수 있습니다.
- 상단의 + 버튼을 클릭하여 새로운 탭을 추가할 수 있습니다.
- 새로운 탭의 전환은 좌측 상단의 1/2 □ 을 클릭하여, 전환할 수 있습니다.



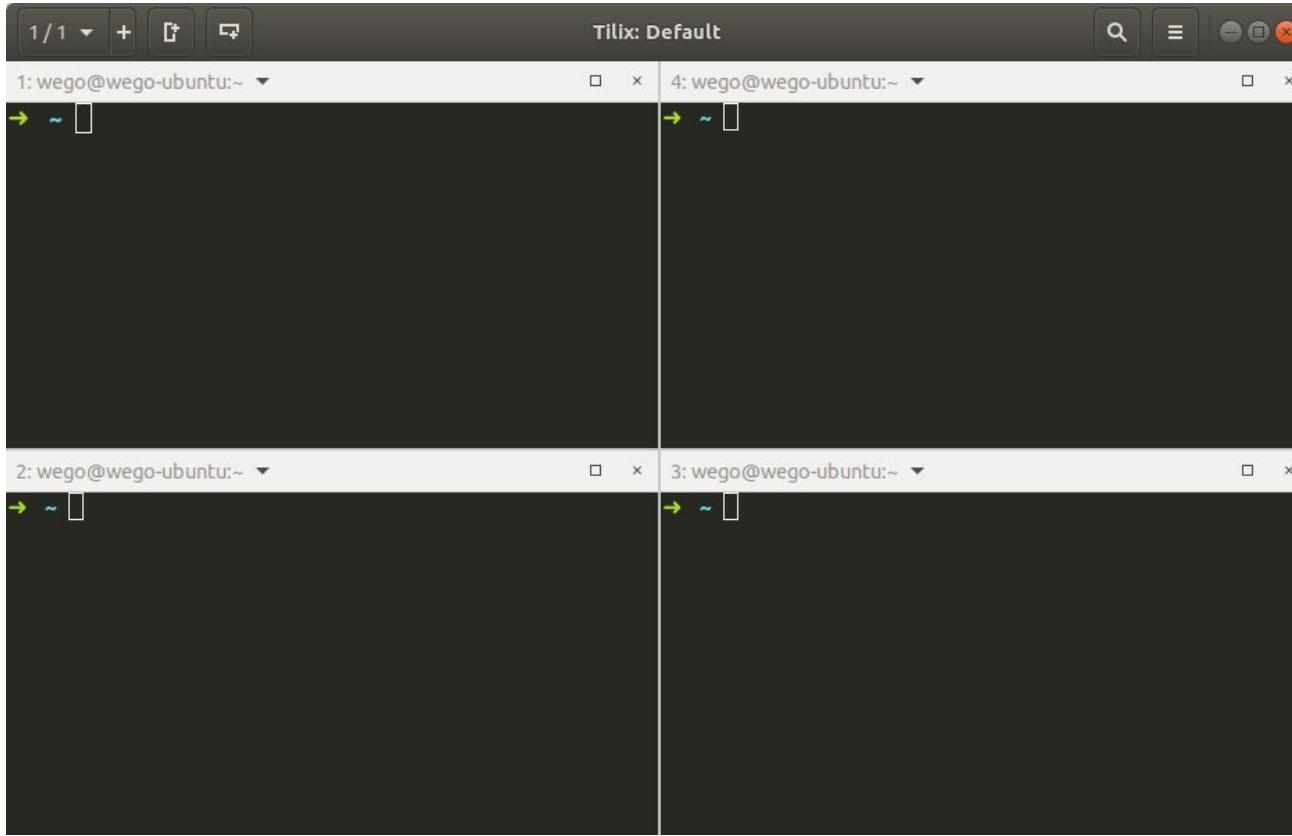
06 Linux Command

- Tilix 터미널의 경우, 상하, 좌우로 분할하여 사용할 수 있으며, 여러 개의 탭을 추가하여 사용할 수 있습니다.
- 하나의 탭에서 터미널을 상하, 좌우로 분할할 수 있습니다.
- 파란색 버튼은 터미널을 좌우로 분할합니다.
- 빨간색 버튼은 터미널을 상하로 분할합니다.



06 Linux Command

- Tilix 터미널의 경우, 상하, 좌우로 분할하여 사용할 수 있으며, 여러 개의 탭을 추가하여 사용할 수 있습니다.
- 분할된 터미널 사이의 이동은 Alt + 방향키로 할 수 있습니다.



06 Linux Command

- 터미널을 처음 실행하였을 때, 기본적으로 터미널에 출력되는 내용은 다음과 같이 정리할 수 있습니다.
- <<USER_NAME>>@<<PC_NAME>>:<<Working_Directory>>\$
- 따라서, 다음과 같이 출력될 수 있습니다.
- wego@wego-ubuntu:~\$
- wego는 사용자의 이름인 <<USER_NAME>>을 의미합니다.
- wego-ubuntu는 PC의 이름인 <<PC_NAME>>을 의미합니다.
- ~은 현재 작업 중인 폴더인, <<Working_Directory>>를 의미합니다.
- ~은 현재 사용자의 Home Directory를 의미하며, 절대 경로로는 /home/<<USER_NAME>>/ 으로 나타내어집니다.

06 Linux Command

- 터미널 사용 시 <TAP> 키를 입력하면, 자동 완성 기능을 제공합니다.
- <TAP> 키의 경우, 리눅스 명령어 및 파일, 폴더 등의 내용을 자동 완성 시 유용하게 사용할 수 있습니다.
- 예를 들어, cd Docu<TAP> 입력 시, cd Documents/ 까지 자동 완성이 됩니다.
- 모든 명령어를 직접 입력하는 것 보다, <TAP>을 적당히 활용하여, 자동완성 기능을 이용하는 것이 실행도 빠르며, 오타를 줄일 수 있는 좋은 방법입니다.

06 Linux Command

- 터미널 사용 중, 특정 명령어는 멈추기 전까지 지속적으로 실행되는 명령어들도 있습니다.
- 이와 같은 명령어들을 실행 중, 멈추려면 Ctrl + c 를 입력하면 멈추고 종료할 수 있습니다.
- Ctrl + z 명령어의 경우, 종료가 아닌, 일시 정지의 의미를 가지고 있으며, 일시 정지 하였을 경우, fg라고 입력하면 다시 실행할 수 있습니다.
- Ctrl + z 명령어는 자주 사용하지 않으므로, 실행 종료를 원할 때는 Ctrl + c 를 입력하여 종료하면 됩니다.

06 Linux Command

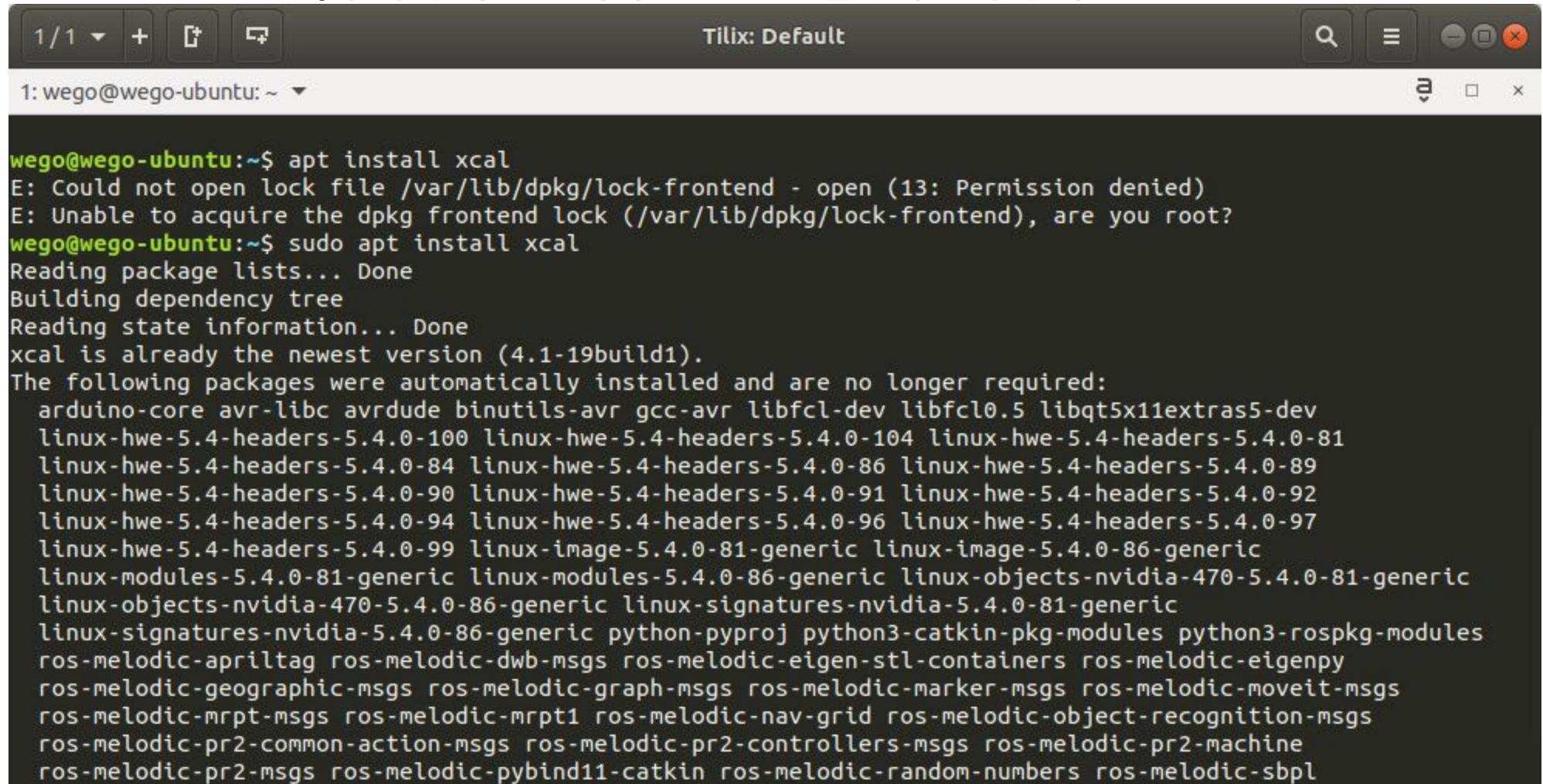
- 터미널에서 작업을 진행 시에는 작업하는 위치의 확인 및 이동이 중요합니다.
- 이를 위해서 사용하는 경로 표현 방법으로는 절대 및 상대 경로가 있습니다.
 - 절대 경로는 / 를 기준으로 표현한 파일의 위치
 - /usr/bin/python , /usr/include/header 등으로 표현
 - ~의 경우 /home/<<USER_NAME>>의 폴더이므로,
~을 기준으로 한 폴더 또한, 절대경로가 될 수 있습니다.
 - 상대 경로는 현재 작업 폴더 (Terminal 기준) 를 기준으로 표현한 파일의 위치
 - bin/python , ./include/header, ../bin/cat 등으로 표현할 수 있음
 - 경로의 시작이 아무 것도 없거나, ./로 시작하면 현재 작업 폴더 하위의 파일
 - ../ 로 시작하면 현재 작업 폴더 상위의 파일을 의미

06 Linux Command

- sudo - 특정 명령어를 관리자 권한으로 실행하는 명령어 ★★★★★★
 - 사용 방법 ([]는 생략 가능)
 - sudo <<사용할 명령어>>
 - <<사용할 명령어>> 부분에 사용할 명령어를 입력
 - 특정 명령어를 관리자 권한(root 권한)으로 실행하며, 특정 명령어를 root 권한으로 실행할 수 없는 경우도 있음
 - 사용 예시
 - sudo apt install xcal

06 Linux Command

- sudo - 특정 명령어를 관리자 권한으로 실행하는 명령어



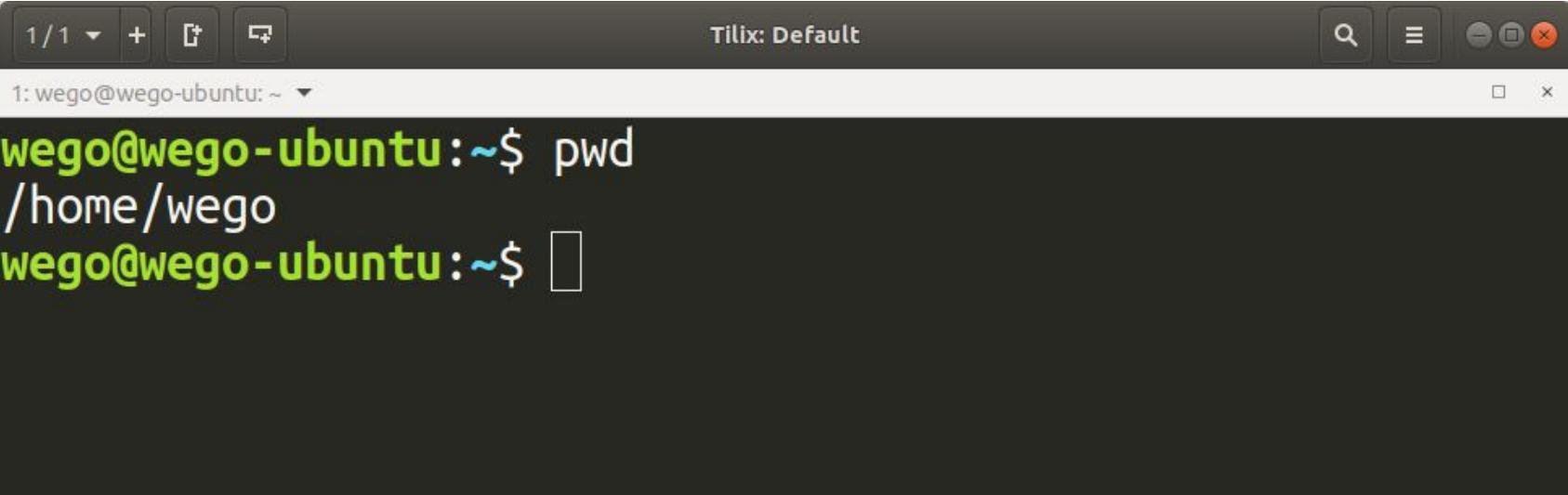
```
1/1 + 🔍 1: wego@wego-ubuntu: ~ Tilix: Default
wego@wego-ubuntu:~$ apt install xcal
E: Could not open lock file /var/lib/dpkg/lock-frontend - open (13: Permission denied)
E: Unable to acquire the dpkg frontend lock (/var/lib/dpkg/lock-frontend), are you root?
wego@wego-ubuntu:~$ sudo apt install xcal
Reading package lists... Done
Building dependency tree
Reading state information... Done
xcal is already the newest version (4.1-19build1).
The following packages were automatically installed and are no longer required:
  arduino-core avr-libc avrdude binutils-avr gcc-avr libfcl-dev libfcl0.5 libqt5x11extras5-dev
  linux-hwe-5.4-headers-5.4.0-100 linux-hwe-5.4-headers-5.4.0-104 linux-hwe-5.4-headers-5.4.0-81
  linux-hwe-5.4-headers-5.4.0-84 linux-hwe-5.4-headers-5.4.0-86 linux-hwe-5.4-headers-5.4.0-89
  linux-hwe-5.4-headers-5.4.0-90 linux-hwe-5.4-headers-5.4.0-91 linux-hwe-5.4-headers-5.4.0-92
  linux-hwe-5.4-headers-5.4.0-94 linux-hwe-5.4-headers-5.4.0-96 linux-hwe-5.4-headers-5.4.0-97
  linux-hwe-5.4-headers-5.4.0-99 linux-image-5.4.0-81-generic linux-image-5.4.0-86-generic
  linux-modules-5.4.0-81-generic linux-modules-5.4.0-86-generic linux-objects-nvidia-470-5.4.0-81-generic
  linux-objects-nvidia-470-5.4.0-86-generic linux-signatures-nvidia-5.4.0-81-generic
  linux-signatures-nvidia-5.4.0-86-generic python-pyproj python3-catkin-pkg-modules python3-rosPKG-modules
  ros-melodic-apriltag ros-melodic-dwb-msgs ros-melodic-eigen-stl-containers ros-melodic-eigenpy
  ros-melodic-geographic-msgs ros-melodic-graph-msgs ros-melodic-marker-msgs ros-melodic-moveit-msgs
  ros-melodic-mrpt-msgs ros-melodic-mrpt1 ros-melodic-nav-grid ros-melodic-object-recognition-msgs
  ros-melodic-pr2-common-action-msgs ros-melodic-pr2-controllers-msgs ros-melodic-pr2-machine
  ros-melodic-pr2-msgs ros-melodic-pybind11-catkin ros-melodic-random-numbers ros-melodic-sbpl
```

06 Linux Command

- apt (advanced package installer) - 패키지 설치 관리자 ★★★★★★
 - 사용 방법 ([]는 생략 가능)
 - sudo apt install <<PACKAGE_NAME>>
 - 필요한 패키지를 설치할 수 있도록 해주는 관리자
 - sudo apt install <<PACKAGE_NAME>>으로 패키지를 설치할 수 있습니다.
 - 설치된 패키지들 중, 업데이트할 수 있는 내용이 있는지 확인할 때는 sudo apt update 명령어를 이용하여 할 수 있습니다.
 - 업데이트가 있는 패키지를 실제 업그레이드를 할 때는 sudo apt upgrade 명령어를 이용하여 업그레이드하여 설치할 수 있습니다.

06 Linux Command

- **pwd (Print Working Directory)** - 현재 작업 중인 폴더 출력 ★★★★
 - 사용 방법 ([]는 생략 가능)
 - pwd
 - 현재 작업 중인 폴더를 출력해줍니다.
 - 윗쪽에서 설명드렸던, \$ 표시 앞에 적혀있는 경로를 통해서도 확인이 가능합니다
 - 현재 작업 중인 폴더를 절대 경로로 터미널에 출력해줍니다.



The screenshot shows a terminal window titled "Tilix: Default". The window has a dark theme with light-colored text. At the top, there are various icons for file operations like opening, saving, and closing. The title bar also shows the window name and some status indicators. The main area of the terminal shows the command line interface. The prompt "1: wego@wego-ubuntu: ~" is visible at the top left. Below it, the command "wego@wego-ubuntu:~\$ pwd" is typed, followed by its output "/home/wego". The command is entered again below, resulting in another "/home/wego" output. The terminal ends with a blank line and a small square icon representing the cursor or a placeholder for input.

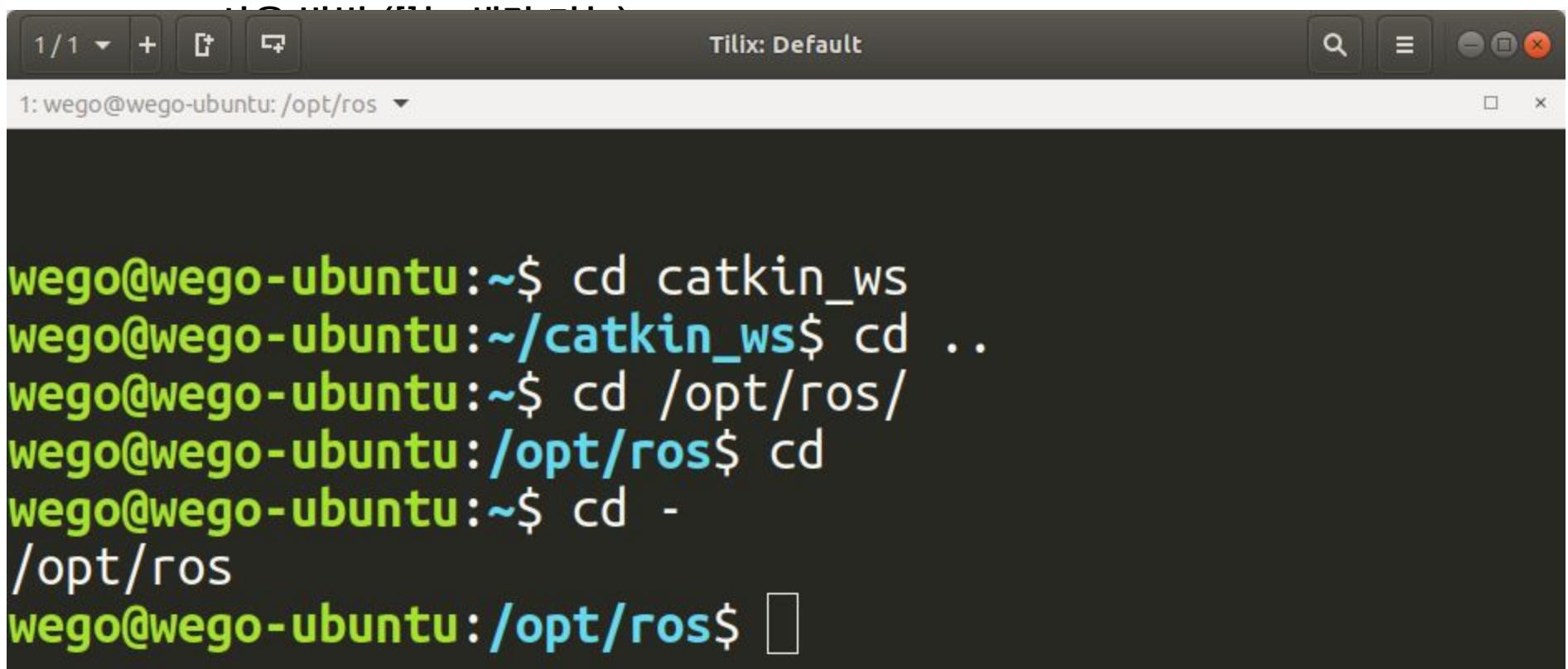
```
wego@wego-ubuntu:~$ pwd
/home/wego
wego@wego-ubuntu:~$
```

06 Linux Command

- cd (Change Directory) - 현재 작업 중인 폴더를 변경 ★★★★★★
 - 사용 방법 ([]는 생략 가능)
 - cd [<<이동할 경로>>]
 - 이동할 경로 부분에는 절대 경로 및 상대 경로가 모두 들어갈 수 있습니다.
 - 이동 경로를 생략할 시, ~로 이동됩니다. (홈폴더)
 - 사용 예시 (절대 경로 및 상대 경로로 작업 폴더 변경)
 - cd ~/catkin_ws
 - cd /opt/ros/melodic/share/
 - cd catkin_ws/src
 - cd ../../opt/ros/melodic
 - cd - (이전 폴더로 이동)

06 Linux Command

- cd (Change Directory) - 현재 작업 중인 폴더를 변경



The screenshot shows a terminal window titled "Tilix: Default" with a dark theme. The window title bar includes standard icons for file operations and a search bar. The terminal prompt is "1: wego@wego-ubuntu: /opt/ros ~". The user has entered the following commands:

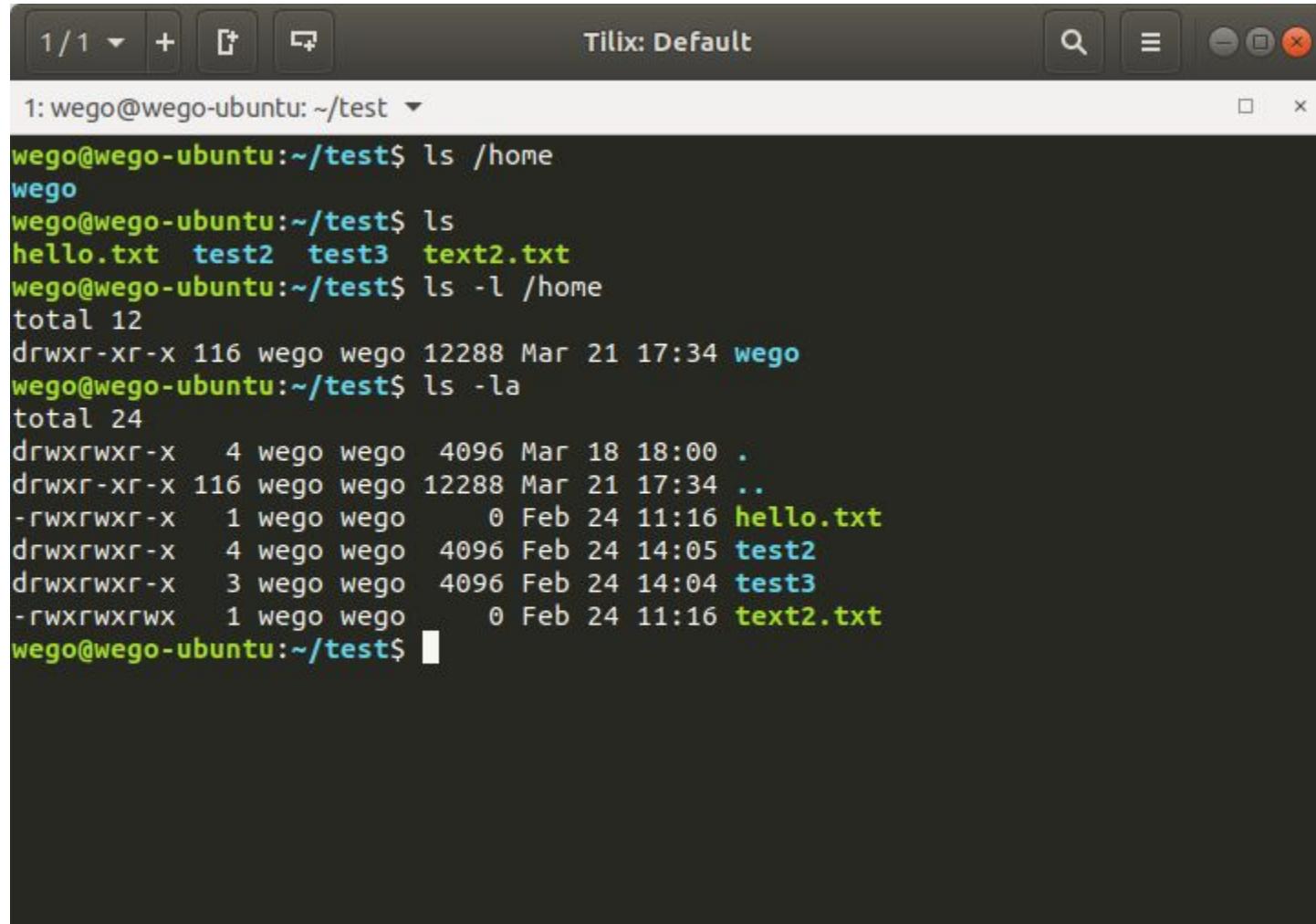
```
wego@wego-ubuntu:~$ cd catkin_ws
wego@wego-ubuntu:~/catkin_ws$ cd ..
wego@wego-ubuntu:~$ cd /opt/ros/
wego@wego-ubuntu:/opt/ros$ cd -
wego@wego-ubuntu:~$ cd -
wego@wego-ubuntu:/opt/ros$ 
```

06 Linux Command

- ls (list contents) - 폴더 내부의 파일 및 폴더 리스트 출력 ★★★★★★
 - 사용 방법 ([]는 생략 가능)
 - ls [-l] [-a] [<<확인할 폴더>>]
 - 확인할 폴더 부분에는 절대 경로 및 상대 경로가 모두 들어갈 수 있습니다.
 - -l 옵션 추가 시, 리스트로, 상세 내용을 출력해줍니다.
 - -a 옵션 추가 시, .으로 시작하는 숨김 파일들도 출력해줍니다.
 - 확인할 폴더 부분을 생략할 시, 현재 작업 폴더 내부의 목록을 출력합니다.
 - 사용 예시
 - ls
 - ls /home/wego/catkin_ws/
 - ls ../../opt/ros/

06 Linux Command

- ls (list directory contents) - 폴더 내부의 파일 및 폴더 리스트 출력



```
1: wego@wego-ubuntu:~/test
wego@wego-ubuntu:~/test$ ls /home
wego
wego@wego-ubuntu:~/test$ ls
hello.txt  test2  test3  text2.txt
wego@wego-ubuntu:~/test$ ls -l /home
total 12
drwxr-xr-x 116 wego wego 12288 Mar 21 17:34 wego
wego@wego-ubuntu:~/test$ ls -la
total 24
drwxrwxr-x    4 wego wego  4096 Mar 18 18:00 .
drwxr-xr-x 116 wego wego 12288 Mar 21 17:34 ..
-rwxrwxr-x    1 wego wego     0 Feb 24 11:16 hello.txt
drwxrwxr-x    4 wego wego  4096 Feb 24 14:05 test2
drwxrwxr-x    3 wego wego  4096 Feb 24 14:04 test3
-rwxrwxrwx    1 wego wego     0 Feb 24 11:16 text2.txt
wego@wego-ubuntu:~/test$
```

06 Linux Command

- ls (list directory contents) - 폴더 내부의 파일 및 폴더 리스트 출력

```
wego@wego-ubuntu:~/test$ ls -la
total 24
drwxrwxr-x  4 wego wego  4096 Mar 18 18:00 .
drwxr-xr-x 116 wego wego 12288 Mar 21 17:34 ..
-rw-rw-r-x  1 wego wego     0 Feb 24 11:16 hello.txt
drwxrwxr-x  4 wego wego  4096 Feb 24 14:05 test2
drwxrwxr-x  3 wego wego  4096 Feb 24 14:04 test3
-rw-rw-rwx  1 wego wego     0 Feb 24 11:16 text2.txt
```

파일 및 권한	내부 링크 개수	파일 소유자	파일 소유 그룹	용량	최종 수정 날짜	파일 이름
drwxrwxr-x 4 wego wego 4096 Mar 18 18:00 .	4	wego	wego	4096	Mar 18 18:00	.

- 파일 및 권한

- 맨 앞 d는 디렉토리, -는 일반 파일을 의미
- 이 후, **rwxrwxrwx**는 3자리씩 끊어서, User, Group, Others의 파일 권한을 의미
- r는 읽기, w는 쓰기, x는 실행 권한을 의미합니다.

06 Linux Command

- ls (list directory contents) - 폴더 내부의 파일 및 폴더 리스트 출력

```
wego@wego-ubuntu:~/test$ ls -la
total 24
drwxrwxr-x  4 wego wego  4096 Mar 18 18:00 .
drwxr-xr-x 116 wego wego 12288 Mar 21 17:34 ..
-rw-rw-r-x  1 wego wego     0 Feb 24 11:16 hello.txt
drwxrwxr-x  4 wego wego  4096 Feb 24 14:05 test2
drwxrwxr-x  3 wego wego  4096 Feb 24 14:04 test3
-rw-rw-rwx  1 wego wego     0 Feb 24 11:16 text2.txt
```

파일 및 권한	내부 링크 개수	파일 소유자	파일 소유 그룹	용량	최종 수정 날짜	파일 이름
drwxrwxr-x 4 wego wego 4096 Mar 18 18:00 .	4	wego	wego	4096	Mar 18 18:00	.
drwxr-xr-x 116 wego wego 12288 Mar 21 17:34 ..	116	wego	wego	12288	Mar 21 17:34	..
-rw-rw-r-x 1 wego wego 0 Feb 24 11:16 hello.txt	1	wego	wego	0	Feb 24 11:16	hello.txt
drwxrwxr-x 4 wego wego 4096 Feb 24 14:05 test2	4	wego	wego	4096	Feb 24 14:05	test2
drwxrwxr-x 3 wego wego 4096 Feb 24 14:04 test3	3	wego	wego	4096	Feb 24 14:04	test3
-rw-rw-rwx 1 wego wego 0 Feb 24 11:16 text2.txt	1	wego	wego	0	Feb 24 11:16	text2.txt

- 내부 링크 개수

- 일반 파일일 경우 1
- 폴더일 경우, 내부에 포함하고 있는 폴더 개수에 따라서 폴더 개수 + 2가 일반적

06 Linux Command

- ls (list directory contents) - 폴더 내부의 파일 및 폴더 리스트 출력

```
wego@wego-ubuntu:~/test$ ls -la
total 24
drwxrwxr-x  4 wego wego  4096 Mar 18 18:00 .
drwxr-xr-x 116 wego wego 12288 Mar 21 17:34 ..
-rw-rw-r-x  1 wego wego     0 Feb 24 11:16 hello.txt
drwxrwxr-x  4 wego wego  4096 Feb 24 14:05 test2
drwxrwxr-x  3 wego wego  4096 Feb 24 14:04 test3
-rw-rw-rwx  1 wego wego     0 Feb 24 11:16 text2.txt
```

파일 및 권한	내부 링크 개수	파일 소유자	파일 소유 그룹	용량	최종 수정 날짜	파일 이름

- 파일 소유자

- 보통 파일을 생성한 User의 이름으로 소유자가 등록됨
- root 권한의 파일의 경우, 일반 명령어로는 권한 변경 등이 불가능 할 수 있음
- sudo 입력을 넣어서 수정하면 수정 가능

06 Linux Command

- ls (list directory contents) - 폴더 내부의 파일 및 폴더 리스트 출력

```
wego@wego-ubuntu:~/test$ ls -la
total 24
drwxrwxr-x  4 wego wego  4096 Mar 18 18:00 .
drwxr-xr-x 116 wego wego 12288 Mar 21 17:34 ..
-rw-rw-r-x  1 wego wego     0 Feb 24 11:16 hello.txt
drwxrwxr-x  4 wego wego  4096 Feb 24 14:05 test2
drwxrwxr-x  3 wego wego  4096 Feb 24 14:04 test3
-rw-rw-rwx  1 wego wego     0 Feb 24 11:16 text2.txt
```

파일 및 권한	내부 링크 개수	파일 소유자	파일 소유 그룹	용량	최종 수정 날짜	파일 이름

- 파일 소유 그룹
 - 파일 소유자와 크게 다를 바 없음

06 Linux Command

- ls (list directory contents) - 폴더 내부의 파일 및 폴더 리스트 출력

```
wego@wego-ubuntu:~/test$ ls -la
total 24
drwxrwxr-x  4 wego wego  4096 Mar 18 18:00 .
drwxr-xr-x 116 wego wego 12288 Mar 21 17:34 ..
-rw-rw-r-x  1 wego wego     0 Feb 24 11:16 hello.txt
drwxrwxr-x  4 wego wego  4096 Feb 24 14:05 test2
drwxrwxr-x  3 wego wego  4096 Feb 24 14:04 test3
-rw-rw-rwx  1 wego wego     0 Feb 24 11:16 text2.txt
```

파일 및 권한	내부 링크 개수	파일 소유자	파일 소유 그룹	용량	최종 수정 날짜	파일 이름

- 용량
 - Byte 단위의 폴더 용량을 의미

06 Linux Command

- ls (list directory contents) - 폴더 내부의 파일 및 폴더 리스트 출력

```
wego@wego-ubuntu:~/test$ ls -la
total 24
drwxrwxr-x  4 wego wego  4096 Mar 18 18:00 .
drwxr-xr-x 116 wego wego 12288 Mar 21 17:34 ..
-rwxrwxr-x  1 wego wego     0 Feb 24 11:16 hello.txt
drwxrwxr-x  4 wego wego  4096 Feb 24 14:05 test2
drwxrwxr-x  3 wego wego  4096 Feb 24 14:04 test3
-rwxrwxrwx  1 wego wego     0 Feb 24 11:16 text2.txt
```

파일 및 권한	내부 링크 개수	파일 소유자	파일 소유 그룹	용량	최종 수정 날짜	파일 이름

- 최종 수정 날짜
 - 월, 일, 년으로 출력되며, 올해 수정된 파일의 경우, 년 부분에 수정한 시간이 표시됨

06 Linux Command

- ls (list directory contents) - 폴더 내부의 파일 및 폴더 리스트 출력

```
wego@wego-ubuntu:~/test$ ls -la
total 24
drwxrwxr-x  4 wego wego  4096 Mar 18 18:00 .
drwxr-xr-x 116 wego wego 12288 Mar 21 17:34 ..
-rwxrwxr-x  1 wego wego     0 Feb 24 11:16 hello.txt
drwxrwxr-x  4 wego wego  4096 Feb 24 14:05 test2
drwxrwxr-x  3 wego wego  4096 Feb 24 14:04 test3
-rwxrwxrwx  1 wego wego     0 Feb 24 11:16 text2.txt
```

파일 및 권한	내부 링크 개수	파일 소유자	파일 소유 그룹	용량	최종 수정 날짜	파일 이름

- 파일 이름

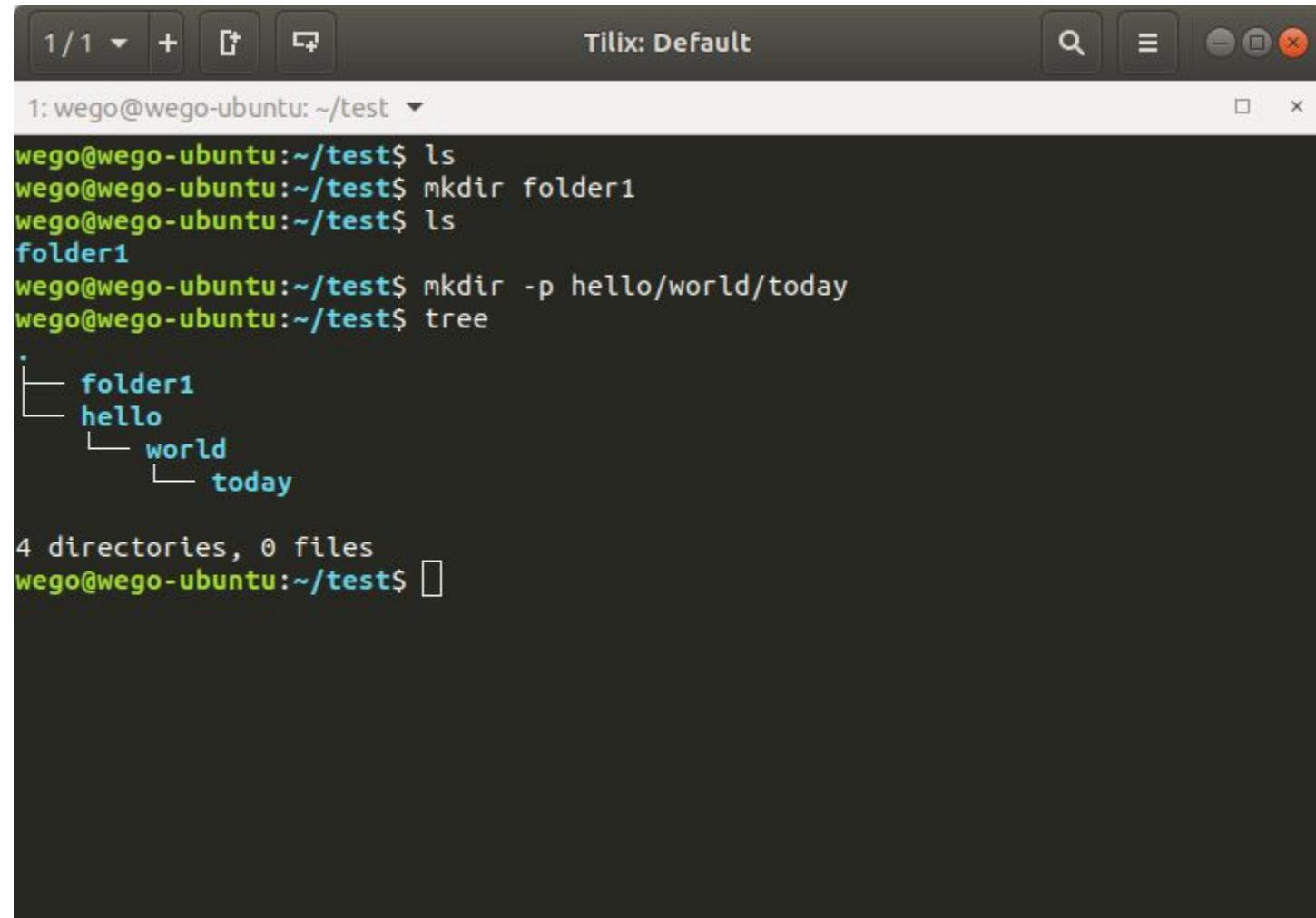
- 파일 이름을 나타냄
- 하늘색으로 나타내어진 파일은 폴더임을 나타냅니다
- 연두색으로 나타내어진 파일은 실행권한이 있는 파일임을 나타냅니다.

06 Linux Command

- **mkdir (make directory)** - 폴더를 만드는 명령어 ★★★★
 - 사용 방법 ([]는 생략 가능)
 - `mkdir [-p] <<FOLDER_NAME>>`
 - `<<FOLDER_NAME>>` 부분을 생성하려는 폴더의 경로로 입력합니다.
 - 여러 개의 폴더를 순서대로, 만들 경우, -p 옵션을 붙여야합니다. (상위 폴더부터 순서대로 생성해주는 옵션)
 - 사용 예시
 - `mkdir folder1`
 - `mkdir -p hello/world/today`

06 Linux Command

- mkdir (make directory) - 폴더를 만드는 명령어



The screenshot shows a terminal window titled "Tilix: Default". The terminal content is as follows:

```
1: wego@wego-ubuntu:~/test ▾
wego@wego-ubuntu:~/test$ ls
• wego@wego-ubuntu:~/test$ mkdir folder1
wego@wego-ubuntu:~/test$ ls
• folder1
wego@wego-ubuntu:~/test$ mkdir -p hello/world/today
wego@wego-ubuntu:~/test$ tree
:
•   └── folder1
      └── hello
          └── world
              └── today

4 directories, 0 files
wego@wego-ubuntu:~/test$
```

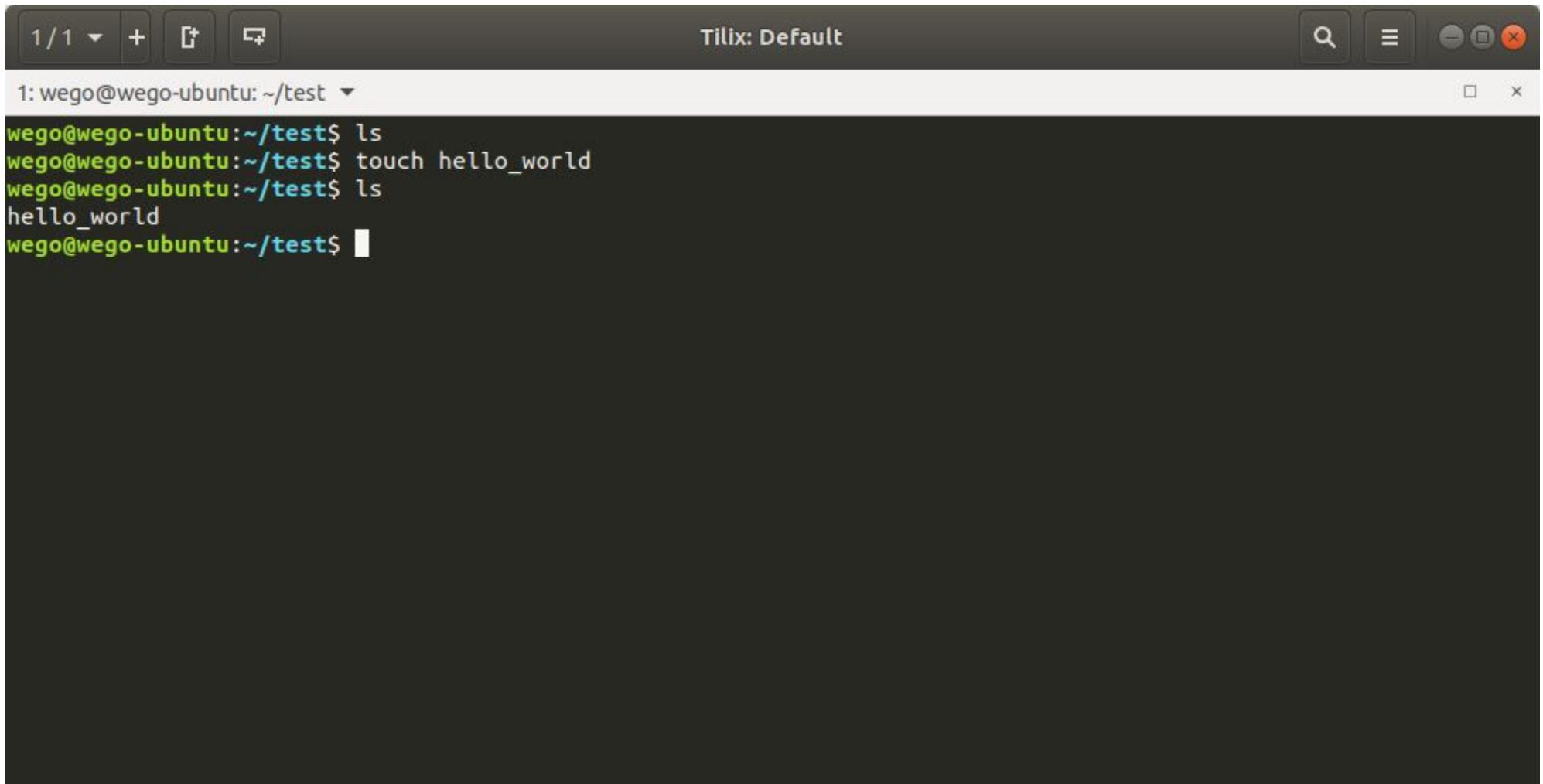
The terminal shows the creation of a directory named "folder1" and a nested directory structure "hello/world/today". The "tree" command is used to display the directory hierarchy.

06 Linux Command

- touch - 파일의 최종 수정 시간을 변경, 또는 새 파일을 생성하는 명령어★★★
 - 사용 방법 ([]는 생략 가능)
 - touch <<FILE_NAME>>
 - <<FILE_NAME>> 부분을 생성 또는 수정 시간을 바꾸려는 파일을 입력합니다.
 - 사용 예시
 - touch hello_world

06 Linux Command

- touch - 파일의 최종 수정 시간을 변경, 또는 새 파일을 생성하는 명령어



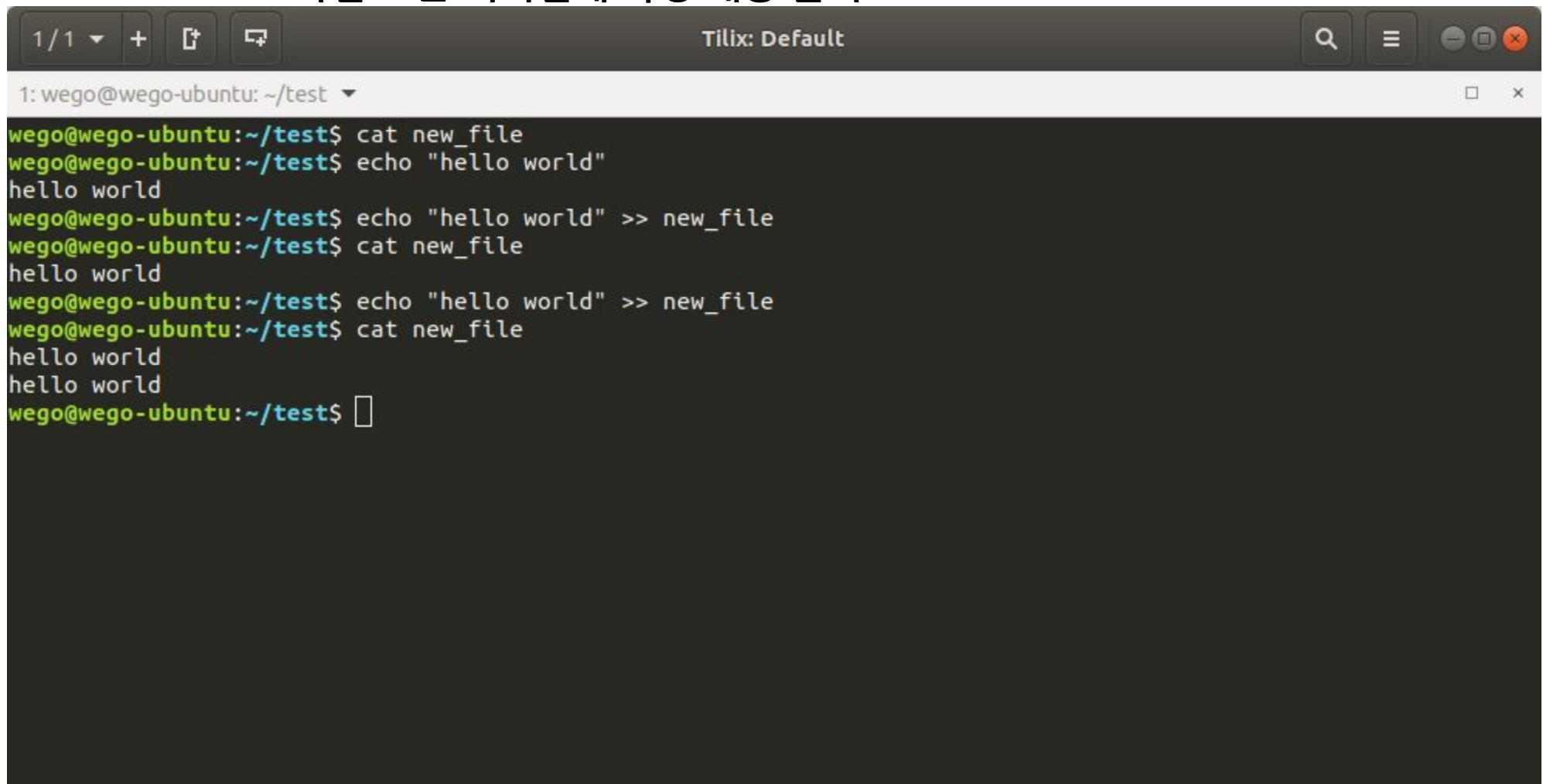
```
1: wego@wego-ubuntu:~/test
wego@wego-ubuntu:~/test$ ls
wego@wego-ubuntu:~/test$ touch hello_world
wego@wego-ubuntu:~/test$ ls
hello_world
wego@wego-ubuntu:~/test$
```

06 Linux Command

- echo - 파일 또는 터미널에 특정 내용 출력★★★
 - 사용 방법 ([]는 생략 가능, {}는 중 하나 선택)
 - echo “<<출력하려는 내용>>” [>> <<FILE_NAME>>]
 - 출력하려는 내용만 입력 시, 터미널에 출력
 - FILE_NAME 입력 시, 기존 내용 뒤에 추가로 출력하려는 내용이 입력됨
 - 사용 예시
 - echo “hello”
 - echo “hello world” >> new_file

06 Linux Command

- echo - 파일 또는 터미널에 특정 내용 출력



The screenshot shows a terminal window titled "Tilix: Default". The terminal window has a dark background and light-colored text. It displays the following command-line session:

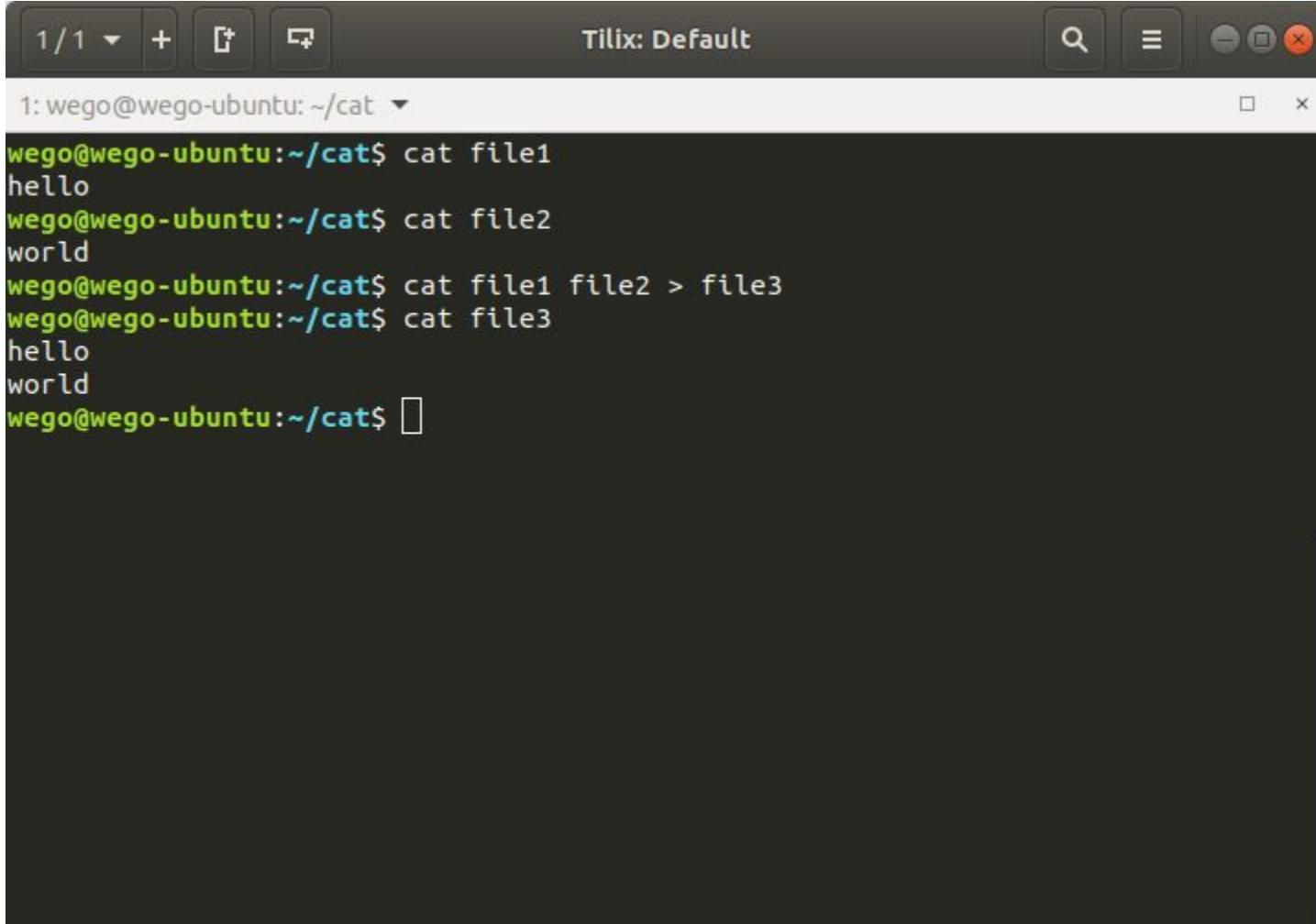
```
1: wego@wego-ubuntu:~/test
wego@wego-ubuntu:~/test$ cat new_file
wego@wego-ubuntu:~/test$ echo "hello world"
hello world
wego@wego-ubuntu:~/test$ echo "hello world" >> new_file
wego@wego-ubuntu:~/test$ cat new_file
hello world
wego@wego-ubuntu:~/test$ echo "hello world" >> new_file
wego@wego-ubuntu:~/test$ cat new_file
hello world
hello world
wego@wego-ubuntu:~/test$ 
```

06 Linux Command

- cat (concatenate files) - 파일 내부의 내용을 출력해주는 명령어 ★★
 - 사용 방법 ([]는 생략 가능)
 - cat <<FILENAME>>
 - 확인할 파일을 입력하면 내용을 확인할 수 있습니다.
 - 이에 추가적으로, cat FILE1 FILE2 > FILE3 과 같은 형식으로 입력하면, FILE1과 FILE2의 내용을 합쳐, FILE3으로 만들 수 있습니다.
 - 사용 예시
 - cat file1
 - cat file2
 - cat file1 file2 > file3

06 Linux Command

- cat (concatenate files) - 파일 내부의 내용을 출력해주는 명령어

- The terminal window shows the following session:

```
1: wego@wego-ubuntu: ~/cat
wego@wego-ubuntu:~/cat$ cat file1
hello
wego@wego-ubuntu:~/cat$ cat file2
world
wego@wego-ubuntu:~/cat$ cat file1 file2 > file3
wego@wego-ubuntu:~/cat$ cat file3
hello
world
wego@wego-ubuntu:~/cat$
```

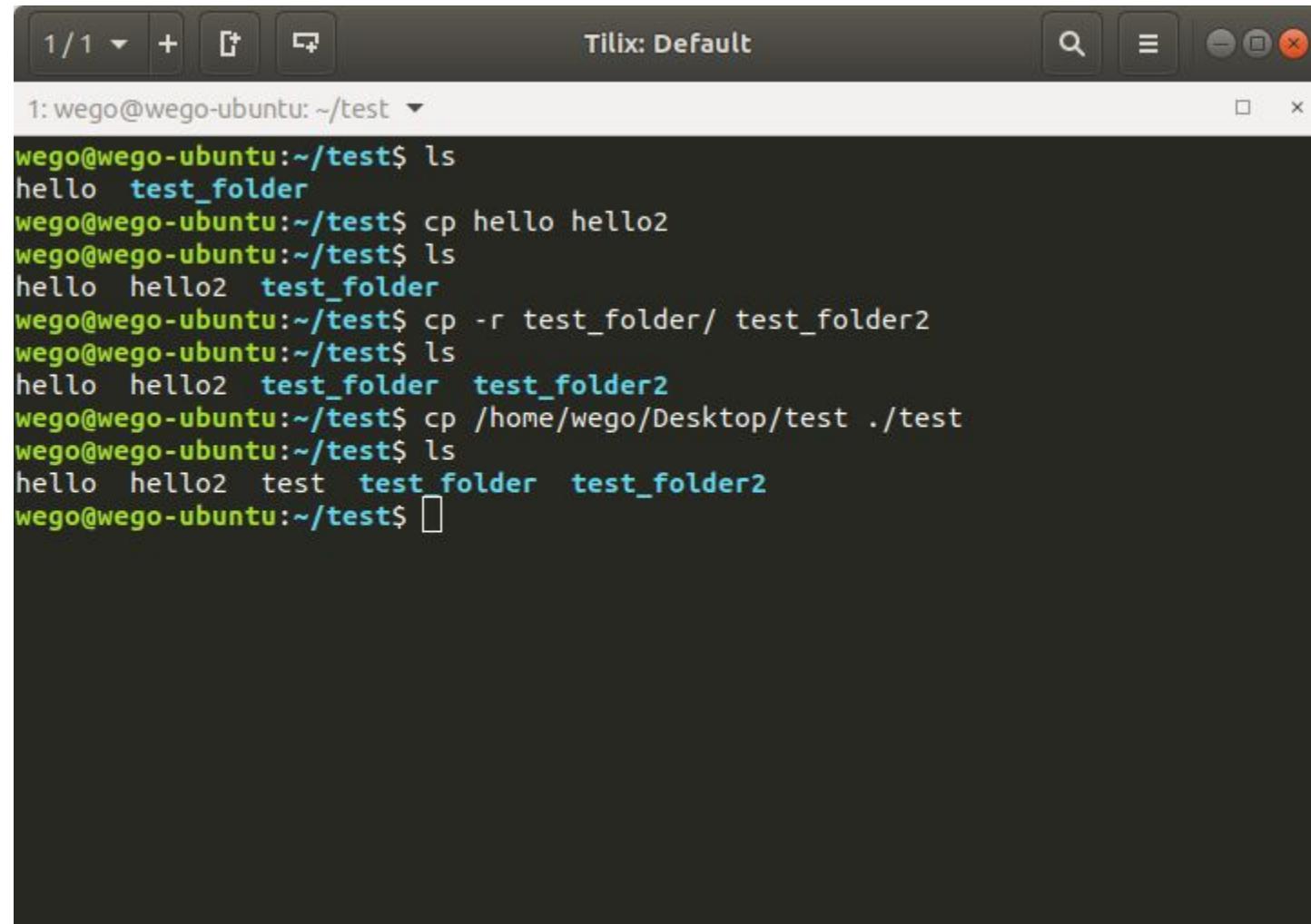
E1과

06 Linux Command

- cp (copy) - 파일 또는 폴더를 복사하는 명령어★★
 - 사용 방법 ([]는 생략 가능)
 - cp [-r] <<ORIGINAL_FILE>> <<DESTINATION_FILE>>
 - ORIGINAL_FILE에 복사하려는 파일의 경로를 입력합니다.
 - DESTINATION_FILE에 붙여넣으려는 위치의 경로를 입력합니다.
 - 해당 경로는 상대 경로 및 절대 경로를 모두 사용할 수 있습니다.
 - -r 옵션은 폴더를 복사할 때 사용합니다.
 - 사용 예시
 - cp hello hello2
 - cp -r test_folder test_folder2
 - cp /home/wego/Desktop/test ./test

06 Linux Command

- cp (copy) - 파일 또는 폴더를 복사하는 명령어



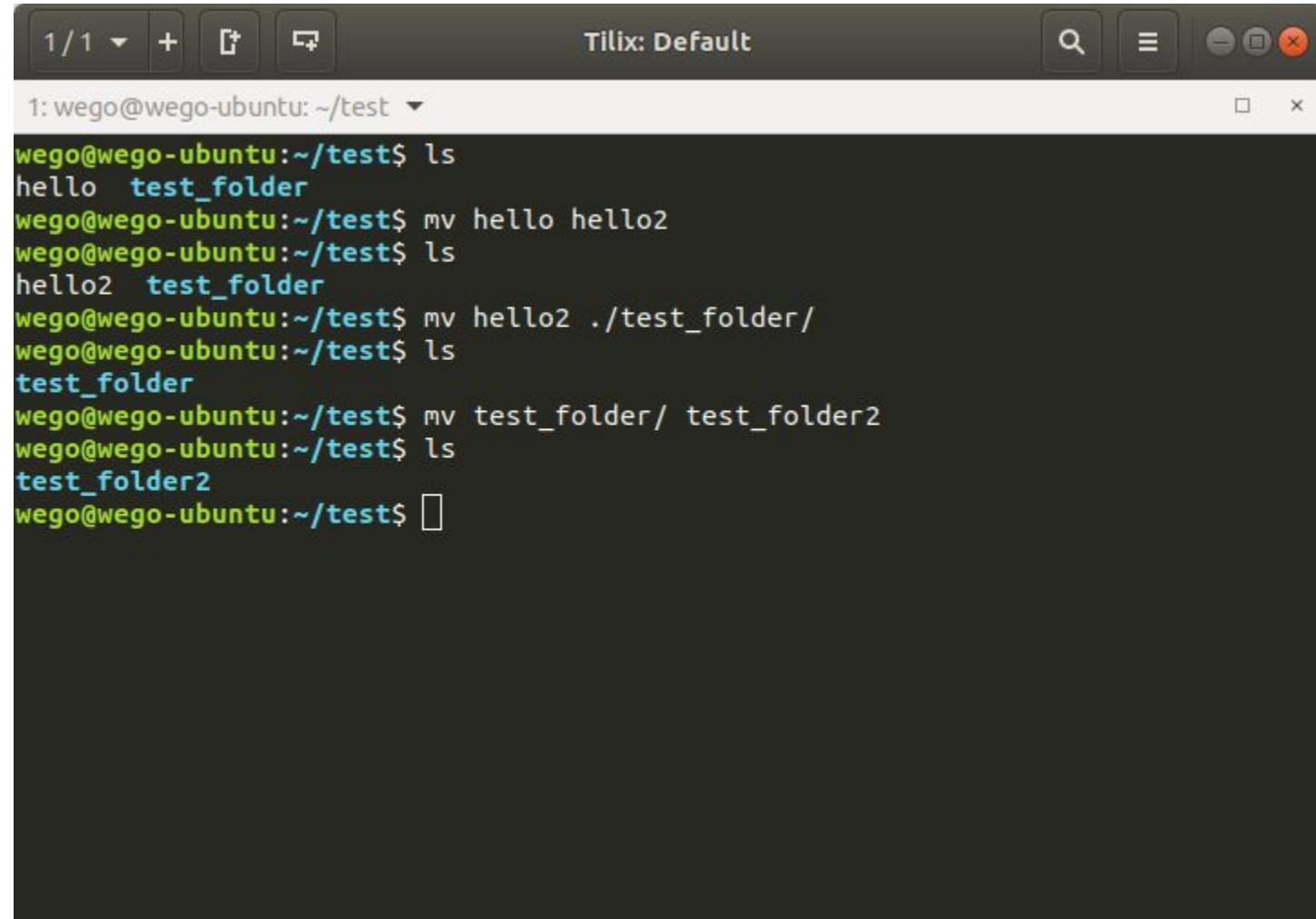
```
1: wego@wego-ubuntu:~/test
wego@wego-ubuntu:~/test$ ls
hello test_folder
wego@wego-ubuntu:~/test$ cp hello hello2
wego@wego-ubuntu:~/test$ ls
hello hello2 test_folder
wego@wego-ubuntu:~/test$ cp -r test_folder/ test_folder2
wego@wego-ubuntu:~/test$ ls
hello hello2 test_folder test_folder2
wego@wego-ubuntu:~/test$ cp /home/wego/Desktop/test ./test
wego@wego-ubuntu:~/test$ ls
hello hello2 test test_folder test_folder2
wego@wego-ubuntu:~/test$ 
```

06 Linux Command

- mv (move) - 파일 또는 폴더를 이동하는 명령어 ★★
 - 사용 방법 ([[]는 생략 가능)
 - mv <<ORIGINAL_FILE>> <<DESTINATION_FILE>>
 - ORIGINAL_FILE에 복사하려는 파일의 경로를 입력합니다.
 - DESTINATION_FILE에 붙여넣으려는 위치의 경로를 입력합니다.
 - 해당 경로는 상대 경로 및 절대 경로를 모두 사용할 수 있습니다.
 - mv의 경우, 폴더든 파일이든 상관 없이 옮길 수 있습니다.
 - 동일한 폴더 내부에서 mv 사용 시, 파일 이름 변경으로도 사용할 수 있습니다.
 - 사용 예시
 - mv hello hello2
 - mv hello ./test_folder/
 - mv test_folder test_folder2

06 Linux Command

- mv (move) - 파일 또는 폴더를 이동하는 명령어



```
1:wego@wego-ubuntu:~/test
wego@wego-ubuntu:~/test$ ls
hello test_folder
wego@wego-ubuntu:~/test$ mv hello hello2
wego@wego-ubuntu:~/test$ ls
hello2 test_folder
wego@wego-ubuntu:~/test$ mv hello2 ./test_folder/
wego@wego-ubuntu:~/test$ ls
test_folder
wego@wego-ubuntu:~/test$ mv test_folder/ test_folder2
wego@wego-ubuntu:~/test$ ls
test_folder2
wego@wego-ubuntu:~/test$
```

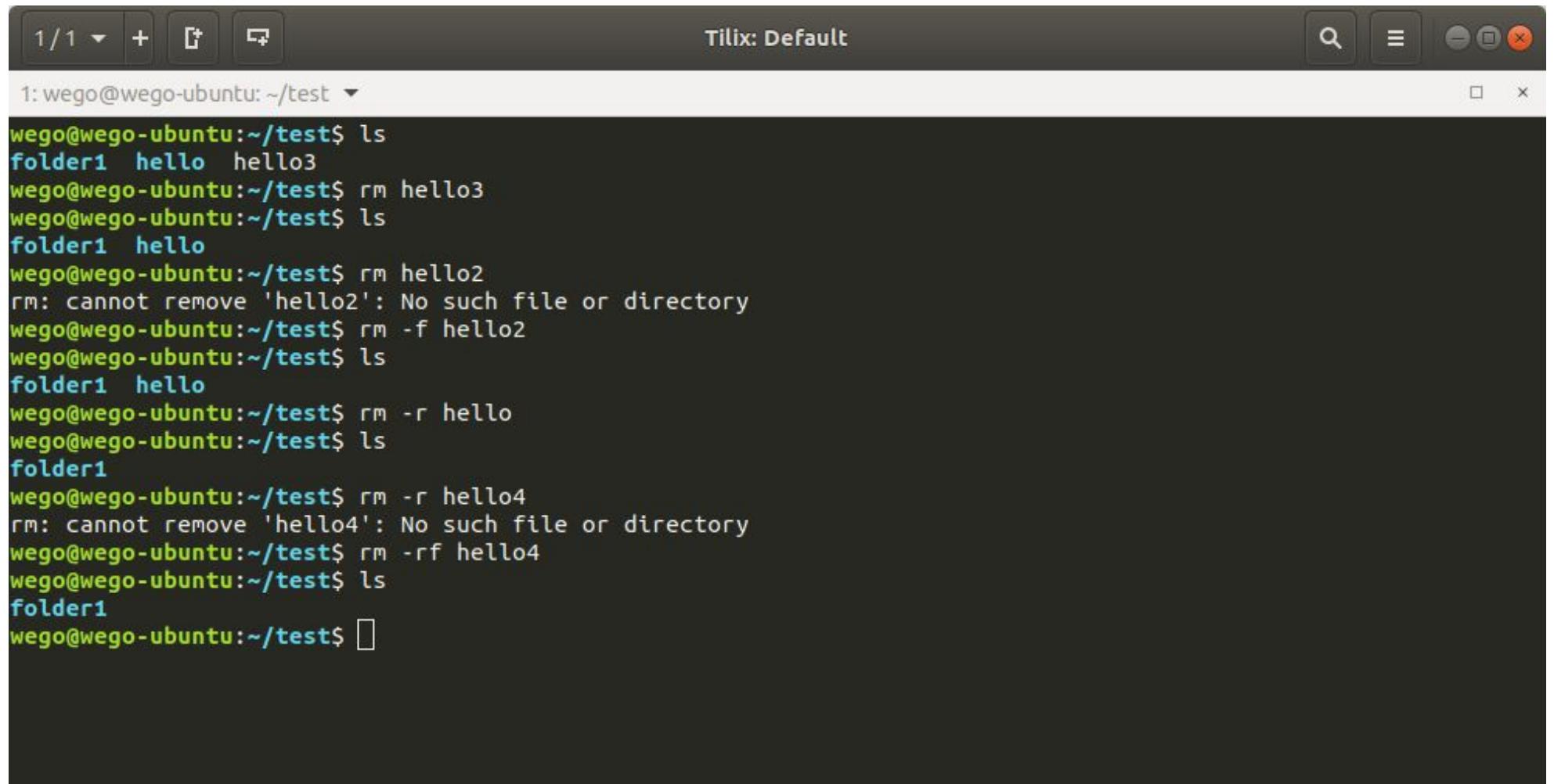
다.

06 Linux Command

- rm (remove file) - 파일 또는 폴더를 삭제하는 명령어 ★★★
 - 사용 방법 ([[]는 생략 가능)
 - rm [-rf] <<FILE_NAME>>
 - <<FILE_NAME>> 부분을 삭제하려는 파일 또는 폴더를 입력합니다.
 - r 옵션은 폴더를 지울 때 필요한 옵션입니다.
 - f 옵션은 없는 파일 등을 입력했을 때도, 무시하고 진행하도록 하는 명령어입니다.
 - 사용 예시
 - rm hello2
 - rm -f hello3
 - rm -r hello
 - rm -rf hello4

06 Linux Command

- rm (remove file) - 파일 또는 폴더를 삭제하는 명령어



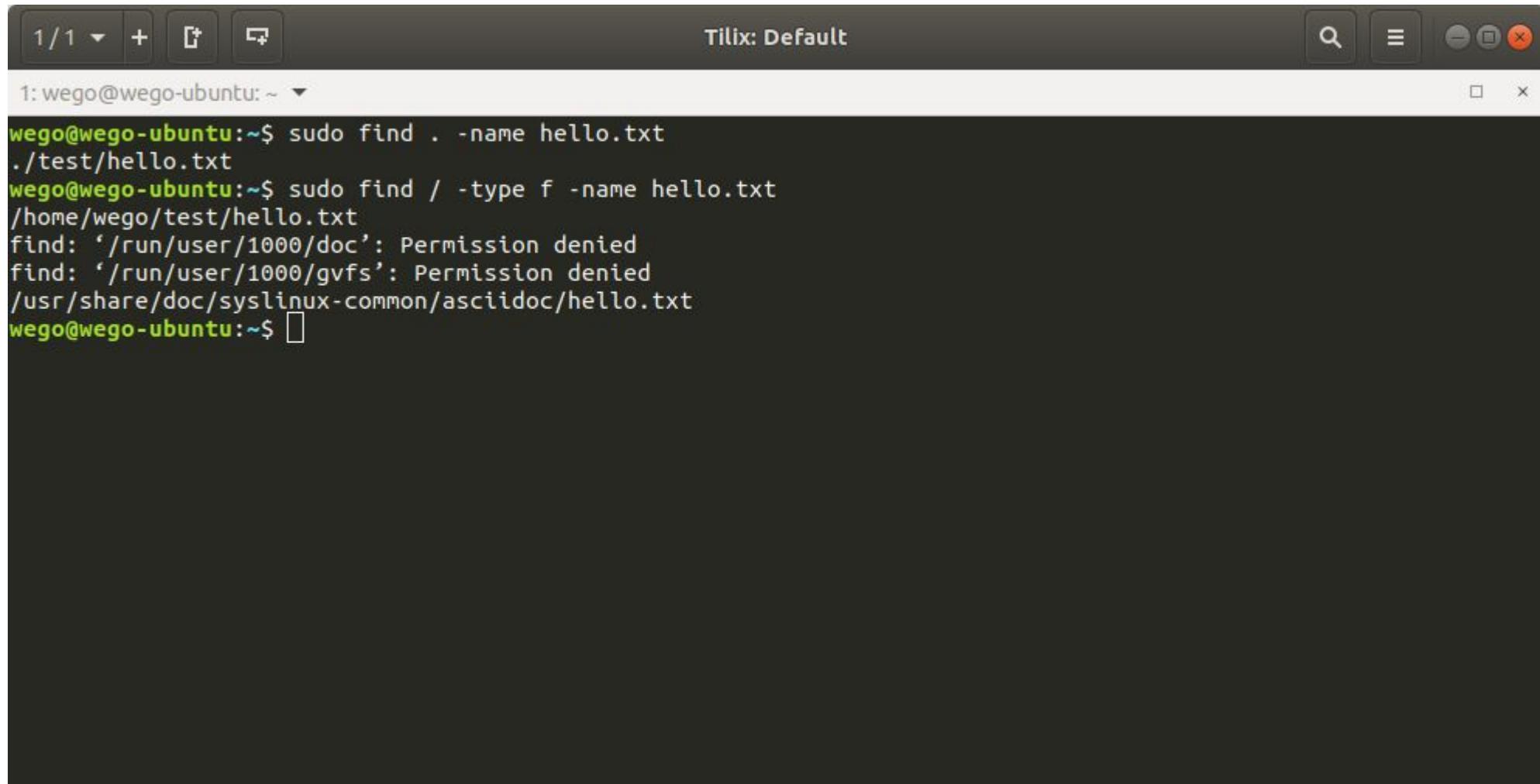
```
1/1 ▾ + 🔍 Tilix: Default
1: wego@wego-ubuntu: ~/test ▾
wego@wego-ubuntu:~/test$ ls
folder1 hello hello3
wego@wego-ubuntu:~/test$ rm hello3
wego@wego-ubuntu:~/test$ ls
folder1 hello
wego@wego-ubuntu:~/test$ rm hello2
rm: cannot remove 'hello2': No such file or directory
wego@wego-ubuntu:~/test$ rm -f hello2
wego@wego-ubuntu:~/test$ ls
folder1 hello
wego@wego-ubuntu:~/test$ rm -r hello
wego@wego-ubuntu:~/test$ ls
folder1
wego@wego-ubuntu:~/test$ rm -r hello4
rm: cannot remove 'hello4': No such file or directory
wego@wego-ubuntu:~/test$ rm -rf hello4
wego@wego-ubuntu:~/test$ ls
folder1
wego@wego-ubuntu:~/test$
```

06 Linux Command

- find - 특정 폴더에서 파일 검색하는 명령어★★★
 - 사용 방법 ([]는 생략 가능)
 - find <<검색할 폴더 위치>> [-name <<FILE_NAME>>] [-type f] [-type d]
 - <<검색할 폴더 위치>>에 폴더 위치를 입력
 - <<FILE_NAME>>에 찾을 파일 또는 폴더 이름을 입력
 - -type f를 입력 시, 파일만 검색
 - -type d를 입력 시, 폴더만 검색
 - 사용 예시
 - find . -name hello.txt
 - find / -type d -name hello.txt

06 Linux Command

- find - 특정 폴더에서 파일 검색하는 명령어



The screenshot shows a terminal window titled "Tilix: Default". The terminal interface includes a header bar with file operations like "New", "Open", "Save", and a search icon. The main area displays a command-line session:

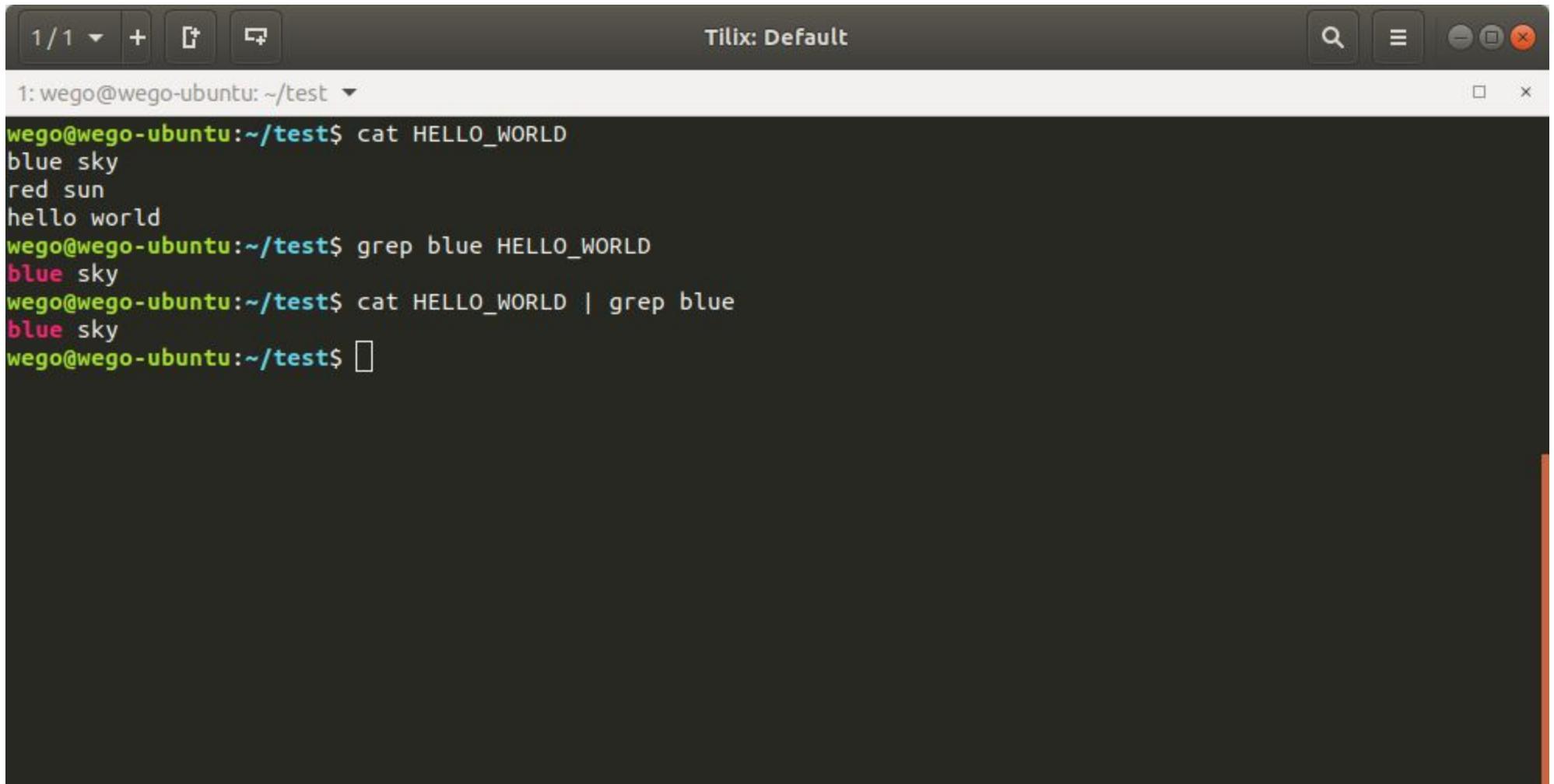
```
1: wego@wego-ubuntu:~ 
wego@wego-ubuntu:~$ sudo find . -name hello.txt
./test/hello.txt
wego@wego-ubuntu:~$ sudo find / -type f -name hello.txt
/home/wego/test/hello.txt
find: '/run/user/1000/doc': Permission denied
find: '/run/user/1000/gvfs': Permission denied
/usr/share/doc/syslinux-common/asciidoc/hello.txt
wego@wego-ubuntu:~$ 
```

06 Linux Command

- grep - 특정한 내용을 검색하는 명령어 ★★★★
 - 사용 방법 ([[]는 생략 가능)
 - grep <<검색할 내용>> <<검색할 파일>>
 - <<검색할 내용>>에 어떤 문구를 검색할 지 입력
 - <<검색할 파일>>에 어떤 파일에서 검색할지 입력
 - <<명령어>> | grep <<검색할 내용>> 과 같은 형태로 입력하면, <<명령어>> 입력시 나오는 내용에서 <<검색할 내용>> 부분을 찾아서 보여줌
 - 사용 예시
 - grep blue HELLO_WORLD
 - cat HELLO_WORLD | grep blue

06 Linux Command

- grep - 특정한 내용을 검색하는 명령어



The screenshot shows a terminal window titled "Tilix: Default". The terminal output is as follows:

```
1: wego@wego-ubuntu:~/test ~
wego@wego-ubuntu:~/test$ cat HELLO_WORLD
blue sky
red sun
hello world
wego@wego-ubuntu:~/test$ grep blue HELLO_WORLD
blue sky
wego@wego-ubuntu:~/test$ cat HELLO_WORLD | grep blue
blue sky
wego@wego-ubuntu:~/test$
```

06 Linux Command

- df - 파일 시스템의 디스크 사용량을 확인할 수 있는 명령어★
 - 사용 방법 ([]는 생략 가능)
 - df -m
 - 사용 예시
 - df -m

06 Linux Command

- df - 파일 시스템의 디스크 사용량을 확인할 수 있는 명령어

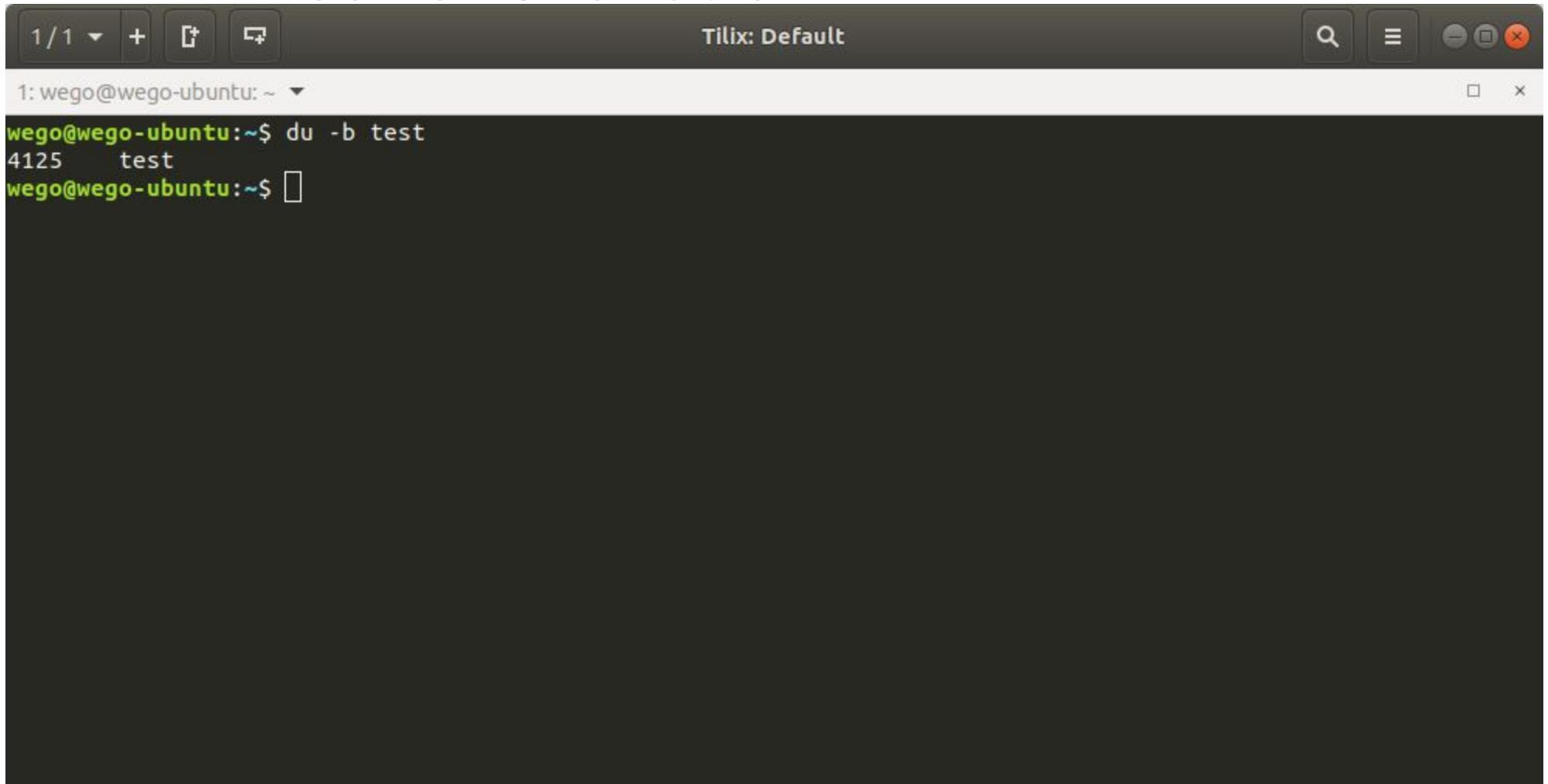
```
1/1 + 🔍 Tilix: Default
1: wego@wego-ubuntu: ~
wego@wego-ubuntu:~$ df -m
Filesystem      1M-blocks  Used Available Use% Mounted on
udev              11875     0    11875   0% /dev
tmpfs             2387     2    2385   1% /run
/dev/nvme0n1p4  240419 207113    21025  91% /
tmpfs             11933   126    11808   2% /dev/shm
tmpfs                 5     1        5   1% /run/lock
tmpfs             11933     0    11933   0% /sys/fs/cgroup
/dev/loop0            3     3        0 100% /snap/gnome-system-monitor/169
/dev/loop1            1     1        0 100% /snap/gnome-characters/741
/dev/loop2            248   248        0 100% /snap/gnome-3-38-2004/87
/dev/loop4            1     1        0 100% /snap/gnome-logs/103
/dev/loop5            163   163        0 100% /snap/gnome-3-28-1804/145
/dev/nvme0n1p1         96   31       66 32% /boot/efi
/dev/loop6            111   111        0 100% /snap/core/12821
/dev/loop3            3     3        0 100% /snap/gnome-calculator/920
/dev/loop7            219   219        0 100% /snap/gnome-3-34-1804/77
/dev/loop8            165   165        0 100% /snap/gnome-3-28-1804/161
/dev/loop9            197   197        0 100% /snap/rpi-imager/184
/dev/loop10           66    66        0 100% /snap/gtk-common-themes/1515
/dev/loop11           56    56        0 100% /snap/core18/2344
/dev/loop12           3     3        0 100% /snap/gnome-calculator/884
/dev/loop13           62    62        0 100% /snap/core20/1405
/dev/loop14           219   219        0 100% /snap/gnome-3-34-1804/72
```

06 Linux Command

- du - 폴더의 크기를 확인하는 명령어★
 - 사용 방법 ([]는 생략 가능)
 - du -b <<FILE_NAME>>]
 - 사용 예시
 - df -b test

06 Linux Command

- du - 폴더의 크기를 확인하는 명령어



The screenshot shows a terminal window titled "Tilix: Default". The window has a dark theme with light-colored text. At the top, there are several icons: a dropdown menu, a plus sign, a copy/paste icon, a search icon, a list icon, and a close button. The terminal prompt is "1: wego@wego-ubuntu: ~". Below the prompt, the user has run the command "du -b test". The output of the command is "4125 test", indicating that the folder "test" occupies 4125 bytes of disk space. The terminal ends with the prompt "wego@wego-ubuntu: ~\$".

```
1: wego@wego-ubuntu: ~
wego@wego-ubuntu:~$ du -b test
4125 test
wego@wego-ubuntu:~$
```

06 Linux Command

- head - 파일의 가장 앞부분에서 특정 수의 줄만큼 출력★
 - 사용 방법 ([]는 생략 가능)
 - head [-n <<확인할 줄 수>>] <<FILE_NAME>>
 - 확인할 줄 수를 따로 입력하지 않으면, 10줄을 기본으로 출력
 - 사용 예시
 - head ~/.bashrc
 - head -n 5 ~/.bashrc
 - head -n 3 ~/.bashrc

06 Linux Command

- head - 파일의 가장 앞부분에서 특정 수의 줄만큼 출력

```
1/1 + ⌂ ⌂ Tilix: Default □ ×  
1: wego@wego-ubuntu:~   
  
wego@wego-ubuntu:~$ head ~/.bashrc  
# ~/.bashrc: executed by bash(1) for non-login shells.  
# see /usr/share/doc/bash/examples/startup-files (in the package bash-doc)  
# for examples  
  
# If not running interactively, don't do anything  
case $- in  
  *i*) ;;  
  *) return;;  
esac  
  
wego@wego-ubuntu:~$ head -n 5 ~/.bashrc  
# ~/.bashrc: executed by bash(1) for non-login shells.  
# see /usr/share/doc/bash/examples/startup-files (in the package bash-doc)  
# for examples  
  
# If not running interactively, don't do anything  
wego@wego-ubuntu:~$ head -n 3 ~/.bashrc  
# ~/.bashrc: executed by bash(1) for non-login shells.  
# see /usr/share/doc/bash/examples/startup-files (in the package bash-doc)  
# for examples  
wego@wego-ubuntu:~$ █
```

06 Linux Command

- tail - 파일의 가장 뒷부분에서 특정 수의 줄만큼 출력★
 - 사용 방법 ([]는 생략 가능)
 - tail [-n <<확인할 줄 수>>] <<FILE_NAME>>
 - 확인할 줄 수를 입력하지 않으면, 기본으로 10줄 표시
 - 사용 예시
 - tail ~/.bashrc
 - tail -n 5 ~/.bashrc
 - tail -n 3 ~/.bashrc

06 Linux Command

- tail - 파일의 가장 뒷부분에서 특정 수의 줄만큼 출력

```
1/1 + ⌂ ⌂ Tilix: Default □ ×  
1: wego@wego-ubuntu:~   
  
wego@wego-ubuntu:~$ tail ~/.bashrc  
# source ~/sim_ws-devel/setup.bash  
# source ~/workspace/apollo_tse_ws-devel/setup.bash  
  
source /opt/ros/melodic/setup.bash  
source ~/catkin_ws-devel/setup.bash  
  
export EDITOR='code'  
alias sb="source ~/.bashrc"  
alias eb="sudo subl ~/.bashrc"  
  
wego@wego-ubuntu:~$ tail -n 5 ~/.bashrc  
  
export EDITOR='code'  
alias sb="source ~/.bashrc"  
alias eb="sudo subl ~/.bashrc"  
  
wego@wego-ubuntu:~$ tail -n 3 ~/.bashrc  
alias sb="source ~/.bashrc"  
alias eb="sudo subl ~/.bashrc"  
  
wego@wego-ubuntu:~$ █
```

06 Linux Command

- diff - 파일 두 개 사이의 차이를 확인★
 - 사용 방법 ([[]는 생략 가능)
 - diff <<FILE_NAME1>> <<FILE_NAME2>>
 - 사용 예시
 - diff file1 file2

06 Linux Command

- diff - 파일 두 개 사이의 차이를 확인

```
1/1 ▾ + ⌂ ⌂ Tilix: Default □ ×  
1: wego@wego-ubuntu:~/test ▾  
  
wego@wego-ubuntu:~/test$ cat file1  
blue sky  
red sun  
hello world  
wego@wego-ubuntu:~/test$ cat file2  
yellow sky  
blue sun  
hello world  
wego@wego-ubuntu:~/test$ diff file1 file2  
1,2c1,2  
< blue sky  
< red sun  
---  
> yellow sky  
> blue sun  
wego@wego-ubuntu:~/test$
```

06 Linux Command

- chmod - 파일 또는 폴더의 권한을 변경하는 명령어 ★★★★★★
 - 사용 방법 ([]는 생략 가능, {}는 중 하나 선택)
 - chmod {a, u, g, o}{+, -}{r, w, x} <<FILE_NAME>>
 - chmod {000 ~ 777} <<FILE_NAME>>
 - a, u, g, o는 각각 all, user, group, others를 나타내며, 각각 모든 사용자의 권한을 변경, 사용자의 권한 변경, 그룹 사용자의 권한 변경, 이 외 사용자의 권한 변경을 의미
 - +, -는 각각 권한을 추가할지 삭제할지를 지정하는 것
 - r, w, x는 각각 읽기, 쓰기, 실행 권한을 의미
 - 파일 권한 변경은 파일의 소유자 및 관리자만 수행할 수 있음
 - 사용 예시
 - chmod a+x new_file
 - chmod o-x new_file
 - chmod o-r new_file
 - chmod 664 new_file

06 Linux Command

- chmod - 파일 또는 폴더의 권한을 변경하는 명령어

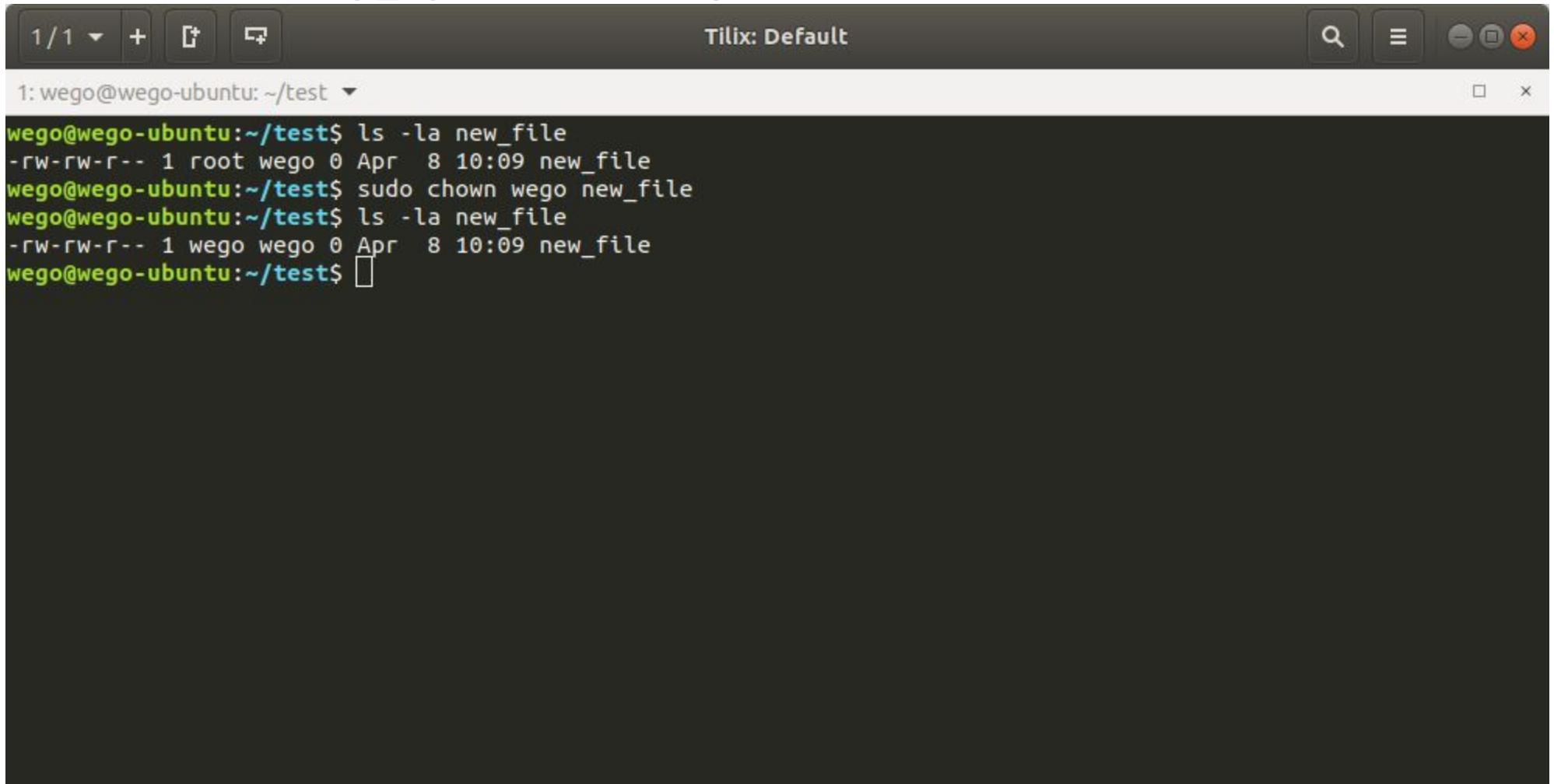
```
1/1 ▾ + ⌂ ⌂ Tilix: Default □ ×  
1: wego@wego-ubuntu:~/test ▾  
  
wego@wego-ubuntu:~/test$ touch new_file  
wego@wego-ubuntu:~/test$ ls -la new_file  
-rw-rw-r-- 1 wego wego 0 Apr  8 10:09 new_file  
wego@wego-ubuntu:~/test$ chmod a+x new_file  
wego@wego-ubuntu:~/test$ ls -la new_file  
-rwxrwxr-x 1 wego wego 0 Apr  8 10:09 new_file  
wego@wego-ubuntu:~/test$ chmod o-x new_file  
wego@wego-ubuntu:~/test$ ls -la new_file  
-rwxrwxr-- 1 wego wego 0 Apr  8 10:09 new_file  
wego@wego-ubuntu:~/test$ chmod o-r new_file  
wego@wego-ubuntu:~/test$ ls -la new_file  
-rwxrwx--- 1 wego wego 0 Apr  8 10:09 new_file  
wego@wego-ubuntu:~/test$ chmod 664 new_file  
wego@wego-ubuntu:~/test$ ls -la new_file  
-rw-rw-r-- 1 wego wego 0 Apr  8 10:09 new_file  
wego@wego-ubuntu:~/test$ █
```

06 Linux Command

- chown - 파일의 소유 권한을 변경★
 - 사용 방법 ([]는 생략 가능, {}는 중 하나 선택)
 - chown <<USER_NAME>> <<FILE_NAME>>
 - 사용 예시
 - chown wego2 new_file

06 Linux Command

- chown - 파일의 소유 권한을 변경



The screenshot shows a terminal window titled "Tilix: Default". The terminal output is as follows:

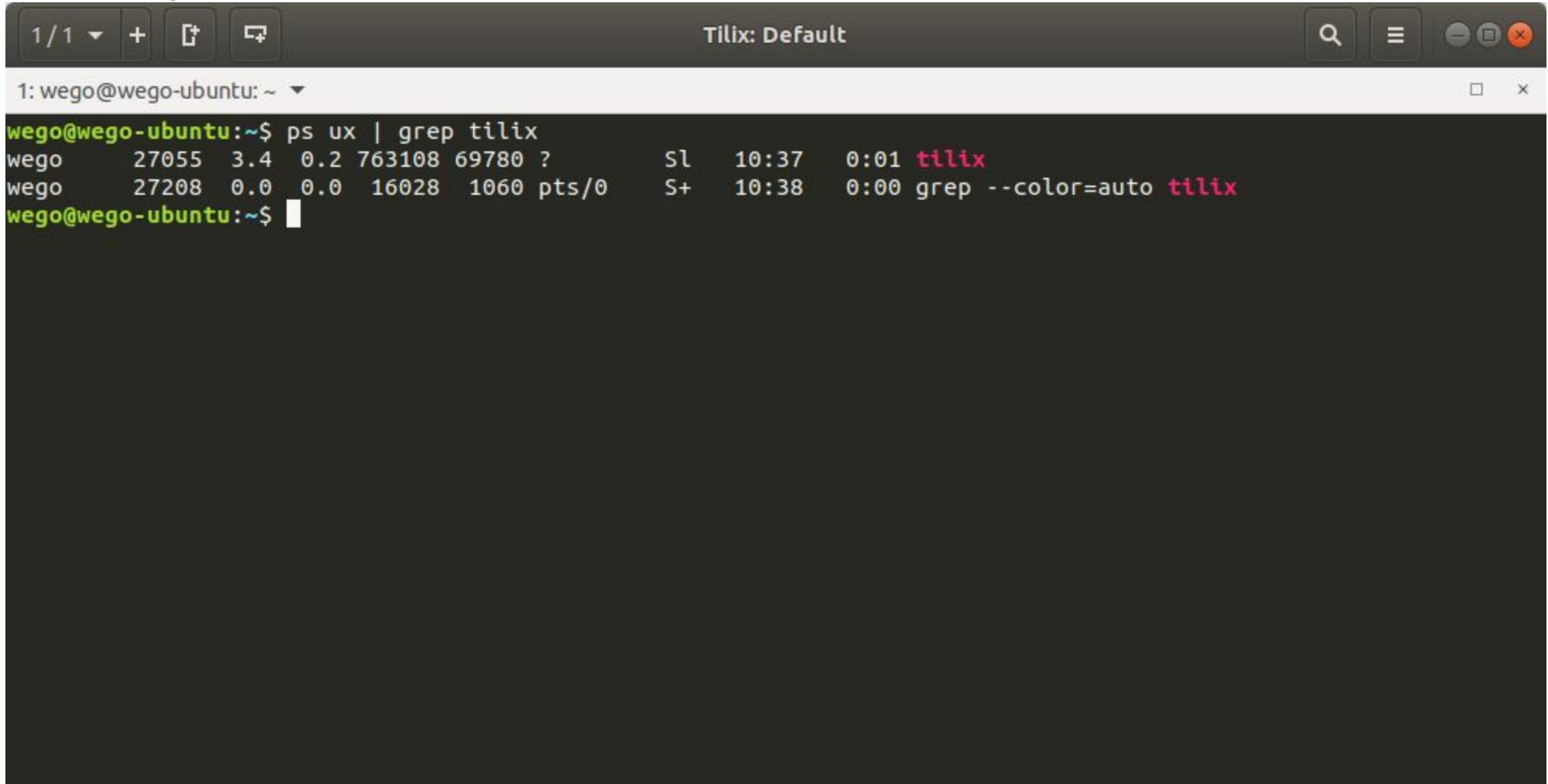
```
1: wego@wego-ubuntu:~/test
wego@wego-ubuntu:~/test$ ls -la new_file
-rw-rw-r-- 1 root wego 0 Apr  8 10:09 new_file
wego@wego-ubuntu:~/test$ sudo chown wego new_file
wego@wego-ubuntu:~/test$ ls -la new_file
-rw-rw-r-- 1 wego wego 0 Apr  8 10:09 new_file
wego@wego-ubuntu:~/test$ 
```

06 Linux Command

- ps - 현재 실행 중인 프로세스 확인 ★★
 - 사용 방법 ([]는 생략 가능, {}는 중 하나 선택)
 - ps ux
 - ps ux 입력 시, 사용 중인 프로세스를 출력해주므로, 이중 필요한 내용만을 검색하기 위해 보통 grep과 함께 사용하는 경우가 많음
 - 가장 왼쪽 두 번째에 출력되는 숫자가 PID (PROCESS IDENTIFICATION NUMBER)입니다.
 - 사용 예시
 - ps ux | grep ros

06 Linux Command

- ps - 현재 실행 중인 프로세스 확인



```
1: wego@wego-ubuntu:~ 
wego@wego-ubuntu:~$ ps ux | grep tilix
wego    27055  3.4  0.2 763108 69780 ?          Sl   10:37   0:01 tilix
wego    27208  0.0  0.0 16028  1060 pts/0        S+   10:38   0:00 grep --color=auto tilix
wego@wego-ubuntu:~$ █
```

06 Linux Command

- kill - 프로세스를 종료하는 명령어 ★★★★
 - 사용 방법 ([]는 생략 가능, {}는 중 하나 선택)
 - kill {-15, -9} <<PROCESS_IDENTIFICATION_NUMBER>>
 - -15는 프로그램 종료 요청 명령 전달
 - -9는 강제 종료
 - 사용 예시
 - kill -9 25343
 - kill -15 25076

06 Linux Command

- kill - 프로세스를 종료하는 명령어

```
1: wego@wego-ubuntu:~ 
wego@wego-ubuntu:~$ ps ux | grep ros
wego    26367  9.5  0.2 765564 66704 pts/5      Sl+  10:34   0:00 /usr/bin/python /opt/ros/melodic/bin/roscore
wego    26388 22.0  0.2 987968 63964 ?          Ssl  10:34   0:01 /usr/bin/python /opt/ros/melodic/bin/rosmaster
r --core -p 11311 -w 3 __log:=/home/wego/.ros/log/0461fde8-b6dc-11ec-bda9-d83bbf196793/master.log
wego    26410  0.3  0.0 342588 18712 ?          Ssl  10:34   0:00 /opt/ros/melodic/lib/rosout/rosout __name:=rosout
__log:=/home/wego/.ros/log/0461fde8-b6dc-11ec-bda9-d83bbf196793/rosout-1.log
wego    26578  0.0  0.0 16028  1060 pts/6      S+   10:34   0:00 grep --color=auto ros
wego@wego-ubuntu:~$ kill -15 26367
wego@wego-ubuntu:~$ ps ux | grep ros
wego    26609  0.0  0.0 16028  1088 pts/6      S+   10:34   0:00 grep --color=auto ros
wego@wego-ubuntu:~$ 
```

06 Linux Command

- ping - 네트워크 연결 상태를 확인하는 명령어 ★★★★
 - 사용 방법 ([]는 생략 가능, {}는 중 하나 선택)
 - ping {{확인할 IP주소}}, {{확인할 URL}} [-c <횟수>]
 - 사용 예시
 - ping 8.8.8.8
 - ping google.com
 - ping 8.8.8.8 -c 5

06 Linux Command

- ping - 네트워크 연결 상태를 확인하는 명령어

```
1/1 + Tilix: Default □ ×
1:wego@wego-ubuntu:~ 
wego@wego-ubuntu:~$ ping 8.8.8.8 -c 5
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=115 time=33.5 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=115 time=32.4 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=115 time=33.7 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=115 time=32.3 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=115 time=33.6 ms

--- 8.8.8.8 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4006ms
rtt min/avg/max/mdev = 32.399/33.164/33.779/0.644 ms
wego@wego-ubuntu:~$ ping google.com -c 5
PING google.com (216.58.220.142) 56(84) bytes of data.
64 bytes from nrt20s17-in-f14.1e100.net (216.58.220.142): icmp_seq=1 ttl=114 time=32.3 ms
64 bytes from nrt20s17-in-f14.1e100.net (216.58.220.142): icmp_seq=2 ttl=114 time=34.1 ms
64 bytes from nrt20s17-in-f14.1e100.net (216.58.220.142): icmp_seq=3 ttl=114 time=33.6 ms
64 bytes from nrt20s17-in-f14.1e100.net (216.58.220.142): icmp_seq=4 ttl=114 time=34.0 ms
64 bytes from nrt20s17-in-f14.1e100.net (216.58.220.142): icmp_seq=5 ttl=114 time=33.7 ms

--- google.com ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4004ms
rtt min/avg/max/mdev = 32.344/33.584/34.105/0.683 ms
wego@wego-ubuntu:~$ █
```

06 Linux Command

- wget - 온라인 파일을 다운로드하는 명령어★★★
 - 사용 방법 ([[]는 생략 가능, {}는 중 하나 선택)
 - wget <<다운로드 링크>>
 - 다운로드 링크는 구글에서 이미지 검색 후, 마우스 우클릭 이미지 주소 복사 선택
 - 사용 예시
 - wget <<이미지 주소>>

06 Linux Command

- wget - 온라인 파일을 다운로드하는 명령어

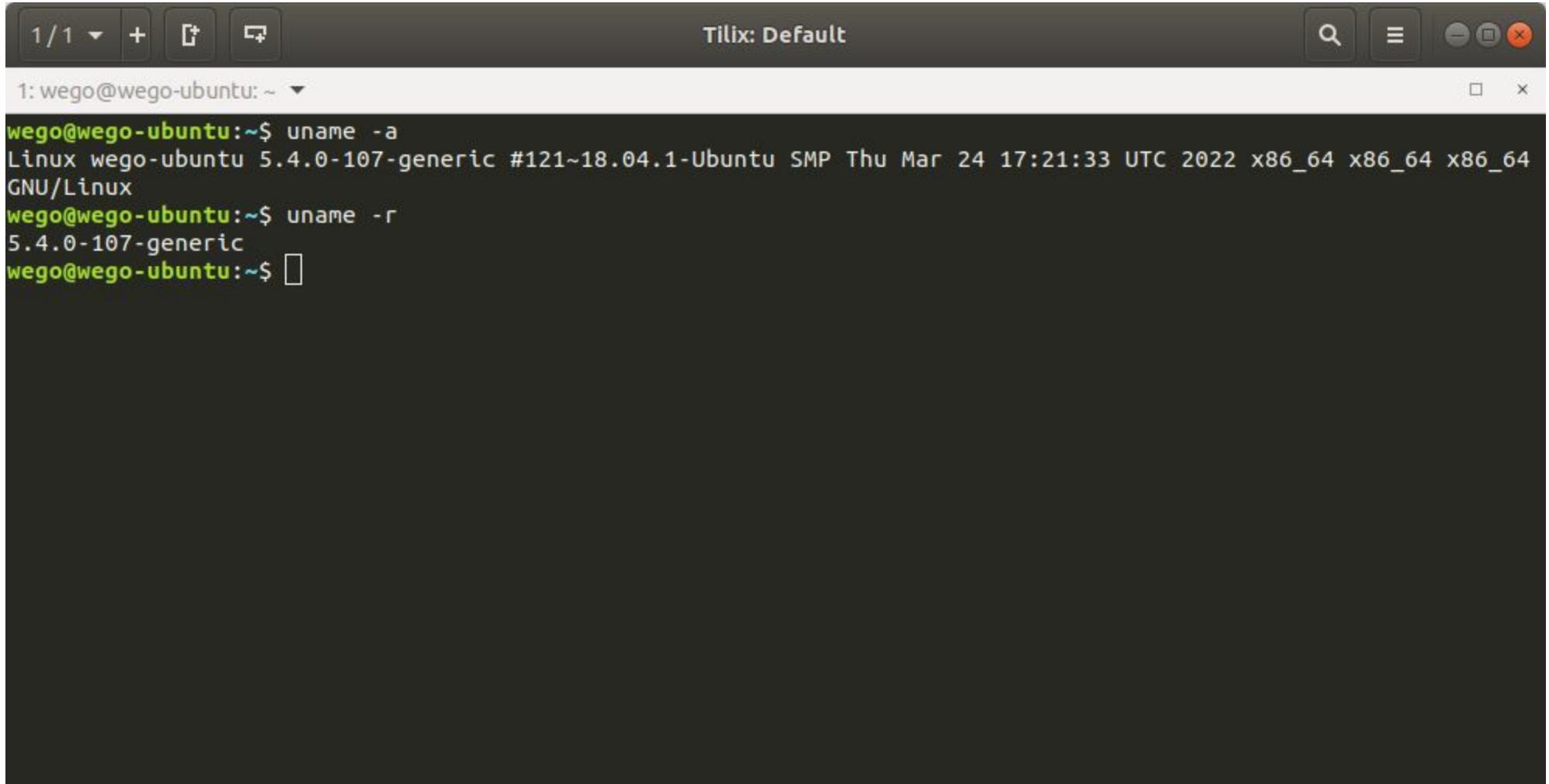
```
1 / 1 + 🔍 Tilix: Default 1: wego@wego-ubuntu:~ ~  
wego@wego-ubuntu:~$ wget https://mblogthumb-phinf.pstatic.net/MjAxNzAzMTJfMTE4/MDAxNDg5MzA3MzQ1MTgw.KqmJui4wG7CVfymQFTxx5mj9G3pnCA0-VYLrnxRawC8g.b7EPdikTxrHtZhdCPUZhjZZa44zuXArJw9lLEXSPqwkg.PNG.wlsdml1103/%EB%A6%AC%EB%88%85%EC%8A%A4.png?type=w800  
--2022-04-08 12:46:48-- https://mblogthumb-phinf.pstatic.net/MjAxNzAzMTJfMTE4/MDAxNDg5MzA3MzQ1MTgw.KqmJui4wG7CVfymQFTxx5mj9G3pnCA0-VYLrnxRawC8g.b7EPdikTxrHtZhdCPUZhjZZa44zuXArJw9lLEXSPqwkg.PNG.wlsdml1103/%EB%A6%AC%EB%88%85%EC%8A%A4.png?type=w800  
Resolving mblogthumb-phinf.pstatic.net (mblogthumb-phinf.pstatic.net)... 210.205.5.155, 220.85.12.19, 210.205.5.159  
Connecting to mblogthumb-phinf.pstatic.net (mblogthumb-phinf.pstatic.net)|210.205.5.155|:443... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 28876 (28K) [image/png]  
Saving to: ‘리눅스.png?type=w800’  
  
리눅스.png?type=w800 100%[=====>] 28.20K --- KB/s in 0.006s  
  
2022-04-08 12:46:48 (4.29 MB/s) - ‘리눅스.png?type=w800’ saved [28876/28876]  
wego@wego-ubuntu:~$
```

06 Linux Command

- uname - Unix Name의 줄임말로, 리눅스 시스템 관련 정보 확인 명령어★
 - 사용 방법 ([]는 생략 가능, {}는 중 하나 선택)
 - uname {-a, -r}
 - -a 명령어 입력 시, 모든 리눅스 시스템 정보 출력
 - -r 명령어 입력 시, 리눅스 커널 확인 가능
 - 사용 예시
 - uname -a
 - uname -r

06 Linux Command

- uname - Unix Name의 줄임말로, 리눅스 시스템 관련 정보 확인 명령어



```
1: wego@wego-ubuntu: ~ 
wego@wego-ubuntu:~$ uname -a
Linux wego-ubuntu 5.4.0-107-generic #121~18.04.1-Ubuntu SMP Thu Mar 24 17:21:33 UTC 2022 x86_64 x86_64 x86_64
GNU/Linux
wego@wego-ubuntu:~$ uname -r
5.4.0-107-generic
wego@wego-ubuntu:~$ 
```

06 Linux Command

- top - 각 프로세스의 CPU 사용량을 출력★
 - 사용 방법 ([]는 생략 가능, {}는 중 하나 선택)
 - top
 - 사용 예시
 - top

06 Linux Command

- top - 각 프로세스의 CPU 사용량을 출력

```
1/1 + 🔍  Tilix: Default  🔍  ⌂  ✎  ×  
1: wego@wego-ubuntu: ~ ▾  
  
top - 12:50:59 up 4:06, 1 user, load average: 0.64, 0.92, 0.75  
Tasks: 422 total, 1 running, 328 sleeping, 0 stopped, 0 zombie  
%Cpu0 : 2.2 us, 0.6 sy, 0.0 ni, 95.4 id, 0.1 wa, 0.0 hi, 1.7 si, 0.0 st  
%Cpu1 : 2.2 us, 0.6 sy, 0.0 ni, 96.6 id, 0.1 wa, 0.0 hi, 0.5 si, 0.0 st  
%Cpu2 : 2.2 us, 0.6 sy, 0.0 ni, 97.0 id, 0.1 wa, 0.0 hi, 0.1 si, 0.0 st  
%Cpu3 : 2.3 us, 0.6 sy, 0.2 ni, 96.8 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st  
%Cpu4 : 2.2 us, 0.7 sy, 0.1 ni, 97.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st  
%Cpu5 : 2.2 us, 0.6 sy, 0.1 ni, 96.9 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st  
%Cpu6 : 2.1 us, 0.6 sy, 0.2 ni, 97.0 id, 0.1 wa, 0.0 hi, 0.0 si, 0.0 st  
%Cpu7 : 2.3 us, 0.6 sy, 0.0 ni, 97.1 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st  
%Cpu8 : 2.2 us, 0.6 sy, 0.0 ni, 97.2 id, 0.1 wa, 0.0 hi, 0.0 si, 0.0 st  
%Cpu9 : 2.2 us, 0.6 sy, 0.2 ni, 97.0 id, 0.1 wa, 0.0 hi, 0.0 si, 0.0 st  
%Cpu10 : 1.7 us, 1.1 sy, 0.0 ni, 97.1 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st  
%Cpu11 : 2.2 us, 0.7 sy, 0.0 ni, 96.9 id, 0.1 wa, 0.0 hi, 0.0 si, 0.0 st  
KiB Mem : 24437356 total, 13458620 free, 4055304 used, 6923432 buff/cache  
KiB Swap: 2097148 total, 2097148 free, 0 used. 19799548 avail Mem  
  


| PID   | USER | PR | NI | VIRT    | RES    | SHR    | S | %CPU | %MEM | TIME+ COMMAND                                      |
|-------|------|----|----|---------|--------|--------|---|------|------|----------------------------------------------------|
| 11586 | wego | 20 | 0  | 46124   | 4468   | 3472   | R | 12.5 | 0.0  | 0:00.02 top                                        |
| 1461  | root | 20 | 0  | 24.865g | 177960 | 80840  | S | 6.2  | 0.7  | 3:34.08 /usr/lib/xorg/Xorg -core :0 -seat seat0 +  |
| 4839  | wego | 20 | 0  | 40.984g | 238708 | 124528 | S | 6.2  | 1.0  | 13:13.04 /usr/share/discord/Discord --type=render+ |
| 5074  | wego | 20 | 0  | 16.516g | 92556  | 65716  | S | 6.2  | 0.4  | 1:15.36 /opt/naver/whale/whale --type=utility --+  |
| 17292 | wego | 20 | 0  | 24.745g | 258356 | 103704 | S | 6.2  | 1.1  | 0:17.56 /opt/naver/whale/whale --type=renderer --+ |
| 17319 | wego | 20 | 0  | 24.723g | 182556 | 99716  | S | 6.2  | 0.7  | 3:11.62 /opt/naver/whale/whale --type=renderer --+ |

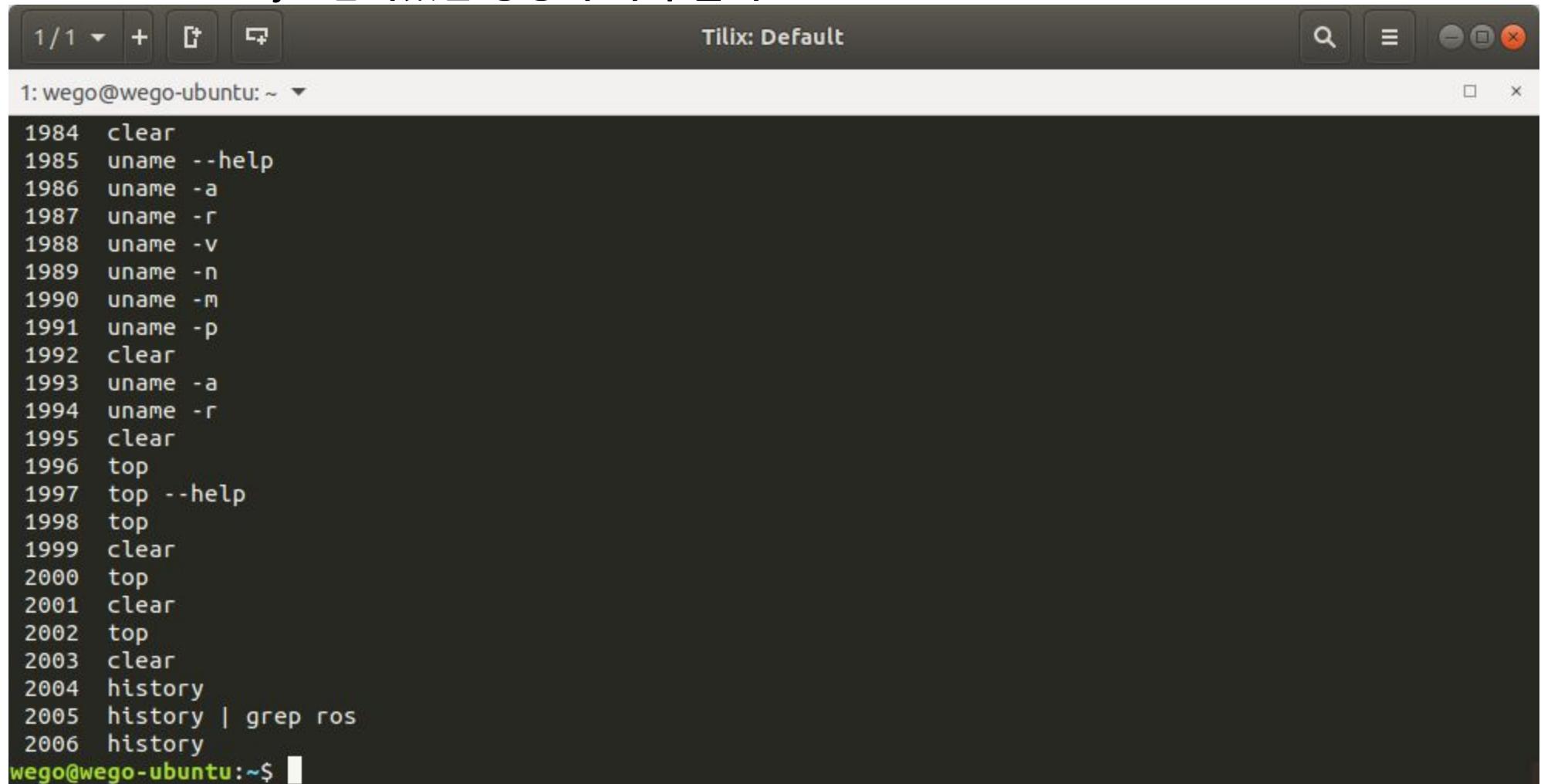

```

06 Linux Command

- history - 입력했던 명령어 기록 출력★
 - 사용 방법 ([]는 생략 가능, {}는 중 하나 선택)
 - history
 - 사용 예시
 - history | grep ros

06 Linux Command

- history - 입력했던 명령어 기록 출력



The screenshot shows a terminal window titled "Tilix: Default". The window has a dark theme with light-colored text. At the top, there are several icons: a dropdown menu, a plus sign, a search icon, and window control buttons. The title bar displays the window name. The main area of the terminal shows a list of command entries:

```
1: wego@wego-ubuntu: ~
1984 clear
1985 uname --help
1986 uname -a
1987 uname -r
1988 uname -v
1989 uname -n
1990 uname -m
1991 uname -p
1992 clear
1993 uname -a
1994 uname -r
1995 clear
1996 top
1997 top --help
1998 top
1999 clear
2000 top
2001 clear
2002 top
2003 clear
2004 history
2005 history | grep ros
2006 history
wego@wego-ubuntu:~$
```

06 Linux Command

- man - 특정 명령어의 manual 출력 ★★
 - 사용 방법 ([]는 생략 가능, {}는 중 하나 선택)
 - man <<COMMAND_NAME>>
 - 명령어에 따라 manual이 없는 경우도 있음
 - 사용 예시
 - man ls
 - man mv
 - man mkdir

06 Linux Command

- man - 특정 명령어의 manual 출력

```
1/1 + 🔍 1: wego@wego-ubuntu: ~ Tilix: Default
LS(1) User Commands LS(1)

NAME
ls - list directory contents

SYNOPSIS
ls [OPTION]... [FILE]...

DESCRIPTION
List information about the FILEs (the current directory by default). Sort entries alphabetically if none of -cftuvSUX nor --sort is specified.

Mandatory arguments to long options are mandatory for short options too.

-a, --all
      do not ignore entries starting with .

-A, --almost-all
      do not list implied . and ..

--author
      with -l, print the author of each file

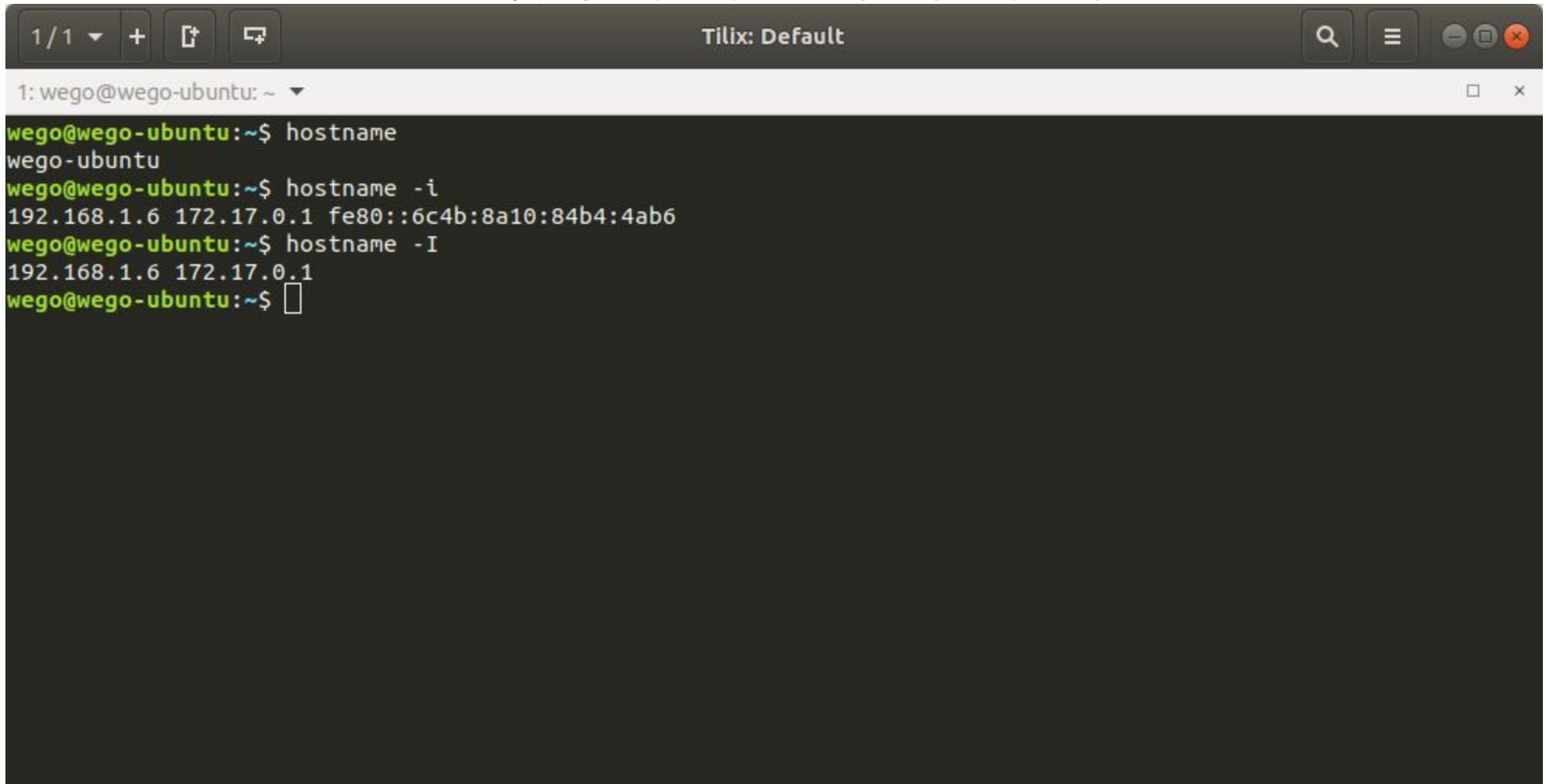
Manual page ls(1) line 1 (press h for help or q to quit)
```

06 Linux Command

- **hostname** - 호스트 및 네트워크 정보를 확인하는 명령어 ★★
 - 사용 방법 ([]는 생략 가능, {}는 중 하나 선택)
 - `hostname [-l]`
 - 사용 예시
 - `hostname`
 - `hostname -i`
 - `hostname -l`

06 Linux Command

- hostname - 호스트 및 네트워크 정보를 확인하는 명령어



```
1: wego@wego-ubuntu: ~
wego@wego-ubuntu:~$ hostname
wego-ubuntu
wego@wego-ubuntu:~$ hostname -i
192.168.1.6 172.17.0.1 fe80::6c4b:8a10:84b4:4ab6
wego@wego-ubuntu:~$ hostname -I
192.168.1.6 172.17.0.1
wego@wego-ubuntu:~$ 
```

06 Linux Command

- 터미널 사용 중에, 입력한 첫번째 글자로 이동하려면 Ctrl + A 를 입력하면 됩니다.
- 반대로 마지막 글자로 이동하려면, Ctrl + E 를 입력하면 됩니다.
- 여러 개의 명령어를 순서대로 입력하려면 중간에 ;을 입력하면 됩니다.
 - cd test; mkdir hello; touch hello2

06 Etc

- Nano 편집기
- 터미널에서 사용 가능한 편집기 중 사용 방법이 쉬운 편에 속하는 편집기
- \$ nano <<FILE_NAME>> 으로 편집을 시작할 수 있음
- 실행 중 아래에 사용 방법이 표시되며, 저장을 위해서는 Ctrl + X 입력하여 저장할 파일의 이름을 정하면 저장 가능

The screenshot shows the GNU nano 2.9.3 text editor interface. The title bar reads "GNU nano 2.9.3" and "hello_world.c". The main area contains the following C code:

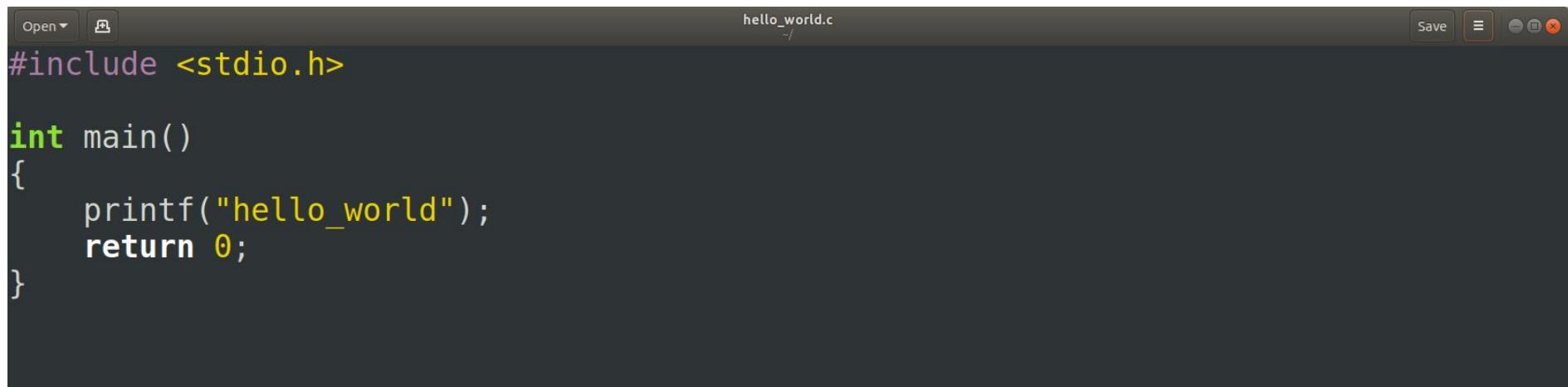
```
#include <stdio.h>

int main()
{
    printf("hello_world");
    return 0;
}
```

At the bottom of the screen, a menu of keyboard shortcuts is displayed. The "Read 7 lines" option is highlighted. The menu items are:

^G Get Help	^O Write Out	^W Where Is	^K Cut Text	^J Justify	^C Cur Pos
^X Exit	^R Read File	^\\ Replace	^U Uncut Text	^T To Spell	^_ Go To Line

- Gedit 편집기
 - GUI환경에서만 사용 가능
 - 일반적으로 사용 가능한 단축키를 거의 다 사용 가능
 - Python 코드 작성 시, Tab으로 공백 생성과 Space로 공백 생성이 다른 문제가 있음



A screenshot of the Gedit text editor interface. The title bar shows "hello_world.c" and the file path "~/". The editor window displays the following C code:

```
#include <stdio.h>

int main()
{
    printf("hello_world");
    return 0;
}
```

- Visual Studio Code
 - GUI환경에서만 사용 가능
 - 일반적으로 사용 가능한 단축키를 거의 다 사용 가능
 - 일반적으로 사용하기에 좋은 Editor
- 설치 방법
 - \$ sudo apt-get install curl
 - \$ sudo sh -c 'curl https://packages.microsoft.com/keys/microsoft.asc | gpg --dearmor > /etc/apt/trusted.gpg.d/microsoft.gpg'
 - \$ sudo sh -c 'echo "deb [arch=amd64]
https://packages.microsoft.com/repos/vscode stable main" >
/etc/apt/sources.list.d/vscode.list'
 - \$ sudo apt update
 - \$ sudo apt install code

- SSH
 - Secure Shell Protocol : 서로 통신 할 때 보안적으로 안전하게 통신을 위한 프로토콜
 - UI 없이 터미널로만 원격 제어
 - 데이터 전송, 원격 제어가 대표적인 사용 예
- 설치 방법
 - \$ sudo apt install ssh
- 접속 방법
 - \$ ssh <<USER_NAME>>@<<IP_ADDRESS>>
 - USER_NAME과 IP_ADDRESS를 설정하여 접속하여 사용 가능 (UI는 없음)



WeGo Robotics

Tel. 031 – 229 – 3553



Fax. 031 – 229 – 3554



제품 문의: go.sales@wego-robotics.com

기술 문의: go.support@wego-robotics.com