

# ROS Programming

(file #2 / ?) ver1.0 Kinetic

Yongseok Chi

# Turtlebot3 burger : Standard Specification

## 1. Specification

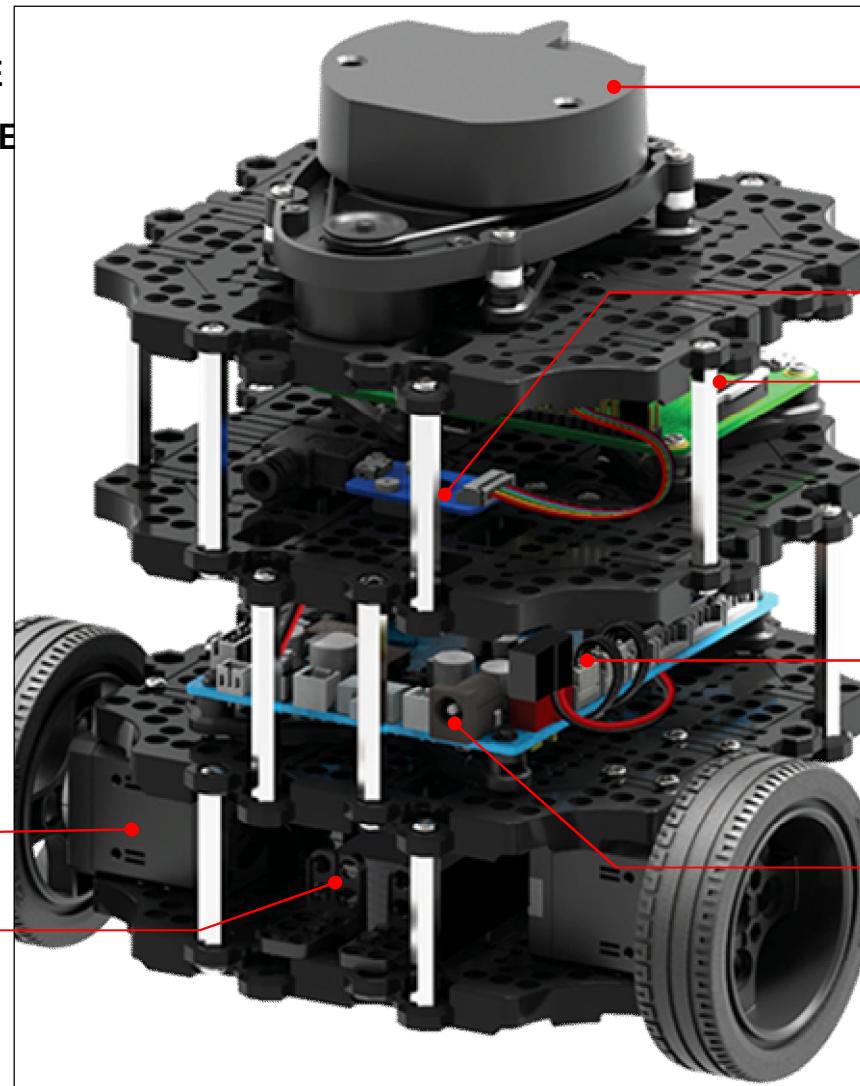
### (1) OPEN SOURCE SOFTWARE

### (2) OPEN SOURCE HARDWARE

#### DYNAMIXEL

- : XL430-W350-T
- : Stall Torque
  - 4.1 [N.m] (at 12.0 [V], 2.3 [A])
- : No Load Speed
  - 46 [rev/min] (at 12.0 [V])

Li-Po battery 11.1V 1800mAh



360° LiDAR

- : HLS-LFCD LDS(Laser Distance Sensor)
- : Detection distance 120mm~3500mm
- : Angular Resolution 1 degree

: USB2LDS (115200 baudrate)

SBC(Single Board Computer)

- : Raspberry PI 3 B+
- Broadcom **BCM2837B0**
- Cortex-A53 64bits Quad-core
- 1.4GHz frequency

OpenCR(Control module for ROS)

- : STM32F746
- ARM Cortex-M7 32bits RISC core
- 216MHz frequency

Input 12V 5A

# Reference books

## 2. Reference books

### (1) 주교재

: ROS 로봇 프로그래밍 : Ruby paper 3-1 ROS 설치

: Robotis e-manual 3장

<https://emanual.robotis.com/docs/en/platform/turtlebot3/quick-start/>

The screenshot shows the 'TurtleBot3' e-manual interface. On the left is a navigation sidebar with the following items:

- 1. Overview
- 2. Features
- 3. Quick Start Guide
  - 3. 1. PC Setup
  - 3. 2. SBC Setup
  - 3. 3. OpenCR Setup
  - 3. 4. Hardware Assembly
  - 3. 5. Bringup
  - 3. 6. Basic Operation

The main content area is titled '3. 1. PC Setup'. It contains a 'WARNING' message: 'WARNING: The contents in this chapter corresponds to the [Remote PC] (your desktop or laptop PC) which will control TurtleBot3. Do not apply this instruction to your TurtleBot3.' Below this is a 'Compatibility WARNING' section with two bullet points:

- Raspberry Pi 4 does not support ROS Kinetic.
- Jetson Nano does not support ROS Kinetic.

A 'NOTE' section at the bottom states: 'NOTE: This instruction was tested on Linux with [Ubuntu 16.04] and [ROS Kinetic Kame].'

**3. 1. 1. Download and Install Ubuntu on PC**

### (2) 실험 재료 : Turtlebot3 burger

## 4. ROS 개발환경 구축

강의실 PC에서



개인PC에서 강의자료 열기

MSI Cubi N100 ADL Win11Home (4GB, M.2 128GB)



Monitor, keyboard, Mouse 연결



수업 마치기 전에, Monitor를 원래 강의실 PC에 연결하기

미 수행시 수업태도 감점

## 4. ROS 개발환경 구축

개인PC(노트북)에서



개인PC에서

- (1) Ubuntu 16.04.7 LTS download
- (2) Boot USB Tool download
- (3) USB에 Ubuntu 16.04.7 LTS 설치

개인PC에서

- (4) RASPBIAN download
- (5) Boot microSD Tool download
- (6) microSD에 RASPBIAN 설치

→ 미니 PC에 Ubuntu와 ROS 설치

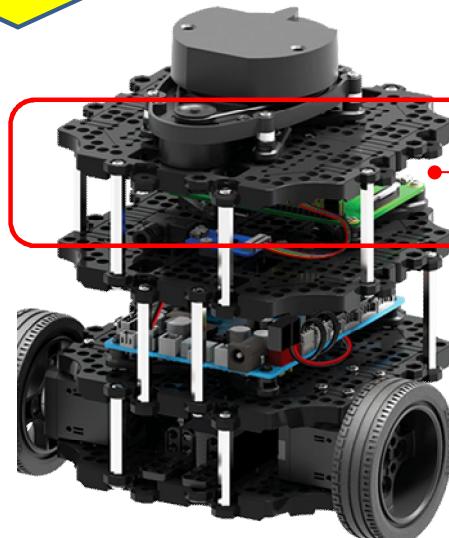


Remote PC

- : 제공되는 미니PC
- : Boot USB로 Ubuntu 16.04.7 LTS 설치
- : ROS 설치



→ Raspberry Pi 3 B+ 에 Raspbian과 ROS 설치



SBC(Single Board Computer)

- : Raspberry Pi 3 B+
- : microSD card로 RASPBIAN 설치
- : ROS 설치

## 4. ROS 개발환경 구축

Ref. <https://emanual.robotis.com/docs/en/platform/turtlebot3/quick-start/>

### 4-1. ROS 개발환경 구축

: Turtlebot 3 개발 환경 구축을 위해 **Remote PC**와 **SBC(Single Board Computer)**를 설정해야 함

- **Remote PC**(제공되는 미니PC)는 ROS를 실행하고 모니터링하기 위한 역할
- **SBC(Raspberry Pi 3 B+)**는 Turtlebot 3 burger(H/W-sensor, openCR)을 직접 구동하기 위한 역할

### 4-2. Remote PC setup

#### : Compatibility WARNING

- **Raspberry Pi 4** does not support ROS Kinetic. ↪ 2022년 9월 state
- **Jetson Nano** does not support ROS Kinetic.

[NVIDIA] Jetson Nano:



#### 4-2-1. Download and Install Ubuntu on PC

##### (1) Ubuntu 16.04.7 LTS (Xenial Xerus)

: Ubuntu 16.04.7 LTS 상위나 하위 버전은 **ROS Kinetic version** 지원 안됨

: PC의 가상 머신(Virtual box)를 사용할 경우 ROS가 정상작동하지 않을 수 있어 **우분투를 단독 설치**

: <https://releases.ubuntu.com/16.04.7/> 를 개인 PC(Remote PC가 아닌 별도 PC)에 download ↪ boot USB 만들기 위함

##### (2) install Ubuntu on Remote PC

: 다운로드 받은 파일은 ISO 디스크 파일이므로 **USB(개인 USB 최소 6GB 필요)**에 바로 옮기면 파일이 손상됨

## 4. ROS 개발환경 구축

: 다운로드 받은 Ubuntu 파일은 ISO 디스크 파일이므로 **USB(개인 USB 최소 6GB 필요)**에 바로 옮기면 파일이 손상됨

( [ubuntu-16.04.7-desktop-amd64.iso](#) )

→ Boot USB 설치하기 위하여 <https://www.balena.io/etcher/>에서 “**balenaEtcher-Setup-1.5.122.exe**” 파일 download

← **balenaEtcher-Setup-1.5.122.exe** 를 관리자 권한으로 개인 PC([Remote PC가 아님](#))에 설치하기

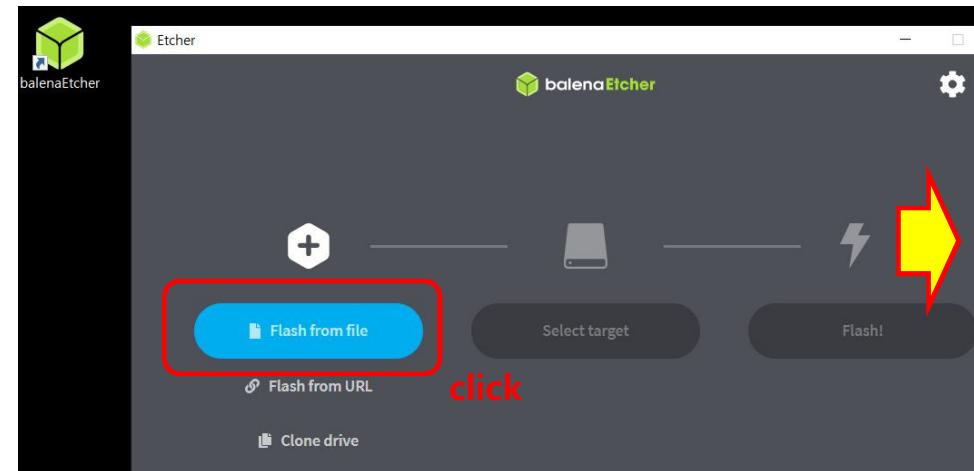
: 위 사이트([balena.io/etcher](#))에서 프로그램을 다운받고



(1) 개인 PC에 관리자권한으로 etcher 프로그램 실행

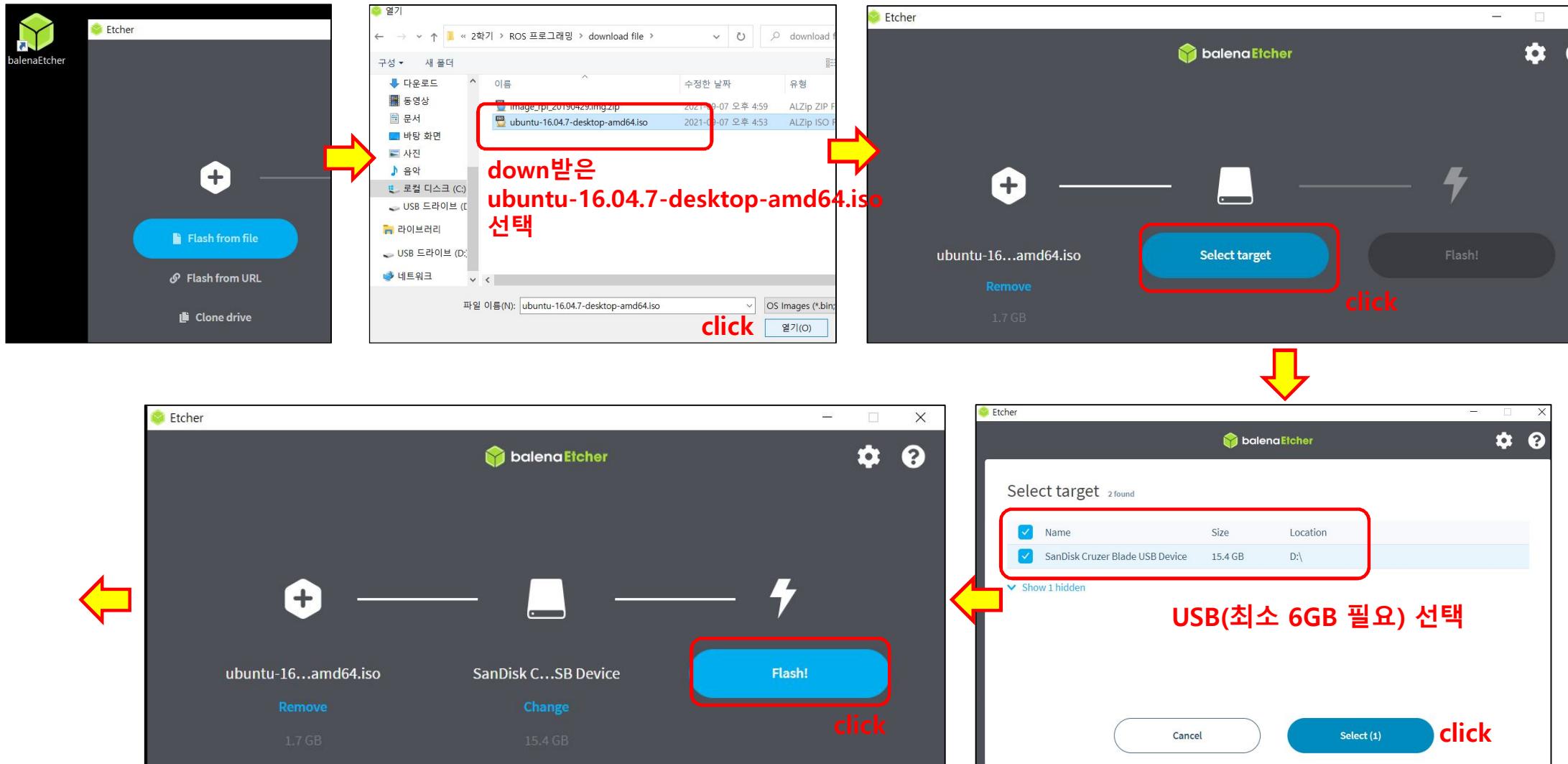
(2) USB(최소 6GB 필요)를 개인 PC에 연결한 후

balena\_etcher에서

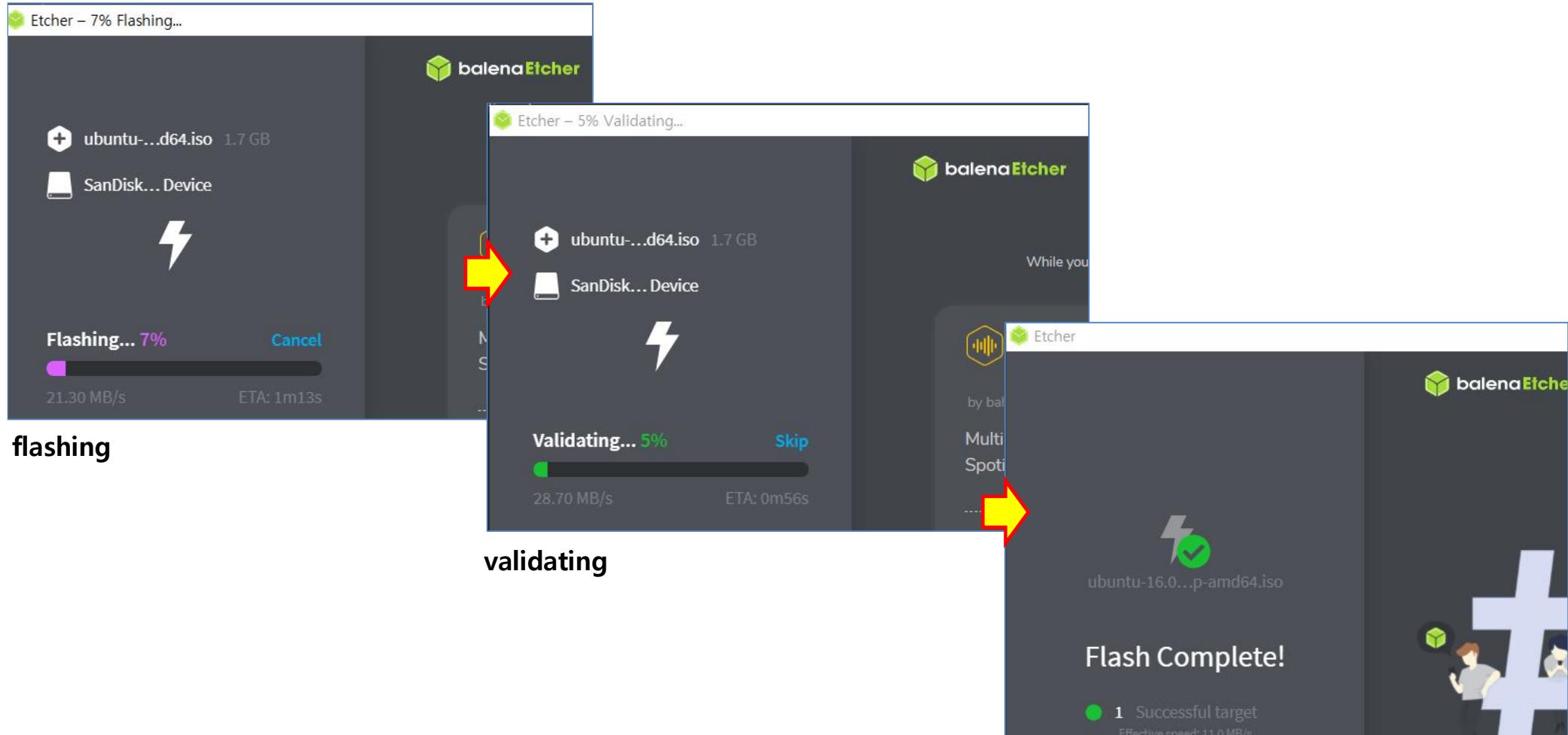


click

## 4. ROS 개발환경 구축



## 4. ROS 개발환경 구축



USB0|| ubuntu-16.04.7-desktop-amd64 설치 완료

## 4. ROS 개발환경 구축

- install Ubuntu on Remote PC(미니PC) 설치 여부 확인

① Remote PC(미니PC)에 Keyboard, Mouse, Monitor 연결 후 전원 ON

: 암호는 1234 (keyboard의 NumLock 해제)

: Ubuntu 설치여부 확인 (아래 Ubuntu 화면)



지난 학기 수업시간에 Ubuntu 설치가 되어 있음



◆ ubuntu version 확인하기

: CTRL + ALT + T 를 사용해 터미널 창 열기

: lsb\_release -a

→ ubuntu version 확인

```
yongseok@yongseok:~$ lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description:    Ubuntu 16.04.7 LTS
Release:        16.04
Codename:       xenial
yongseok@yongseok:~$
```

입력 후 enter

Version 확인

## 4. ROS 개발환경 구축

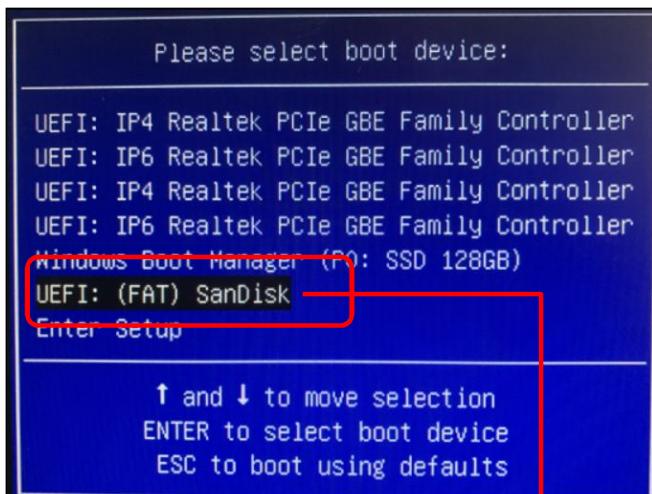
- install Ubuntu on Remote PC(미니PC)

① windows, centos, Ubuntu 등이 설치된 Remote PC(미니PC)에 Ubuntu 설치하기

: 미니 PC에 마우스, 모니터, 키보드 연결하기

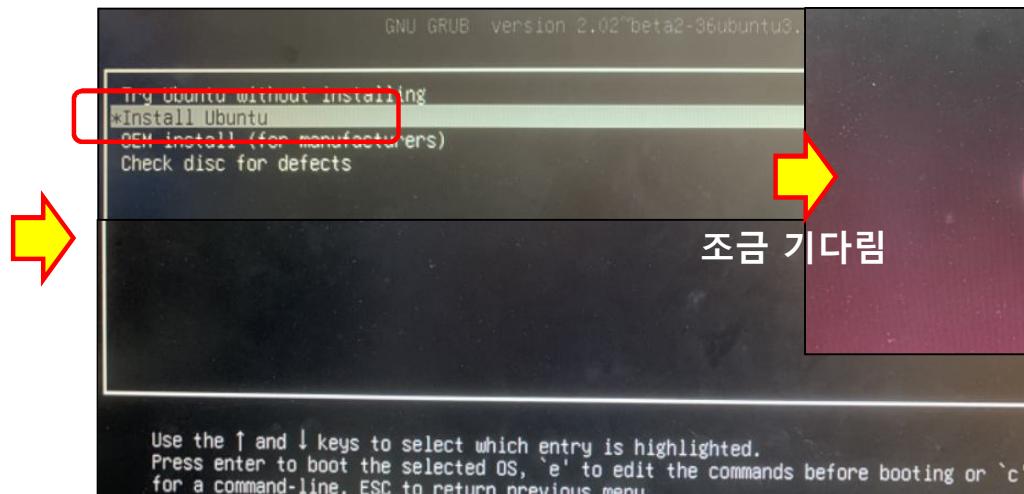
: ubuntu-16.04.7-desktop-amd64.iso가 설치된 USB를 미니PC에 연결

: powering on the PC 이후에 F11 누르기(MSI boot menu)



↑ ↓로 USB 선택 후 enter

(본인 USB 이름을 기억하기)



↑ ↓로 Install Ubuntu 선택 후 enter

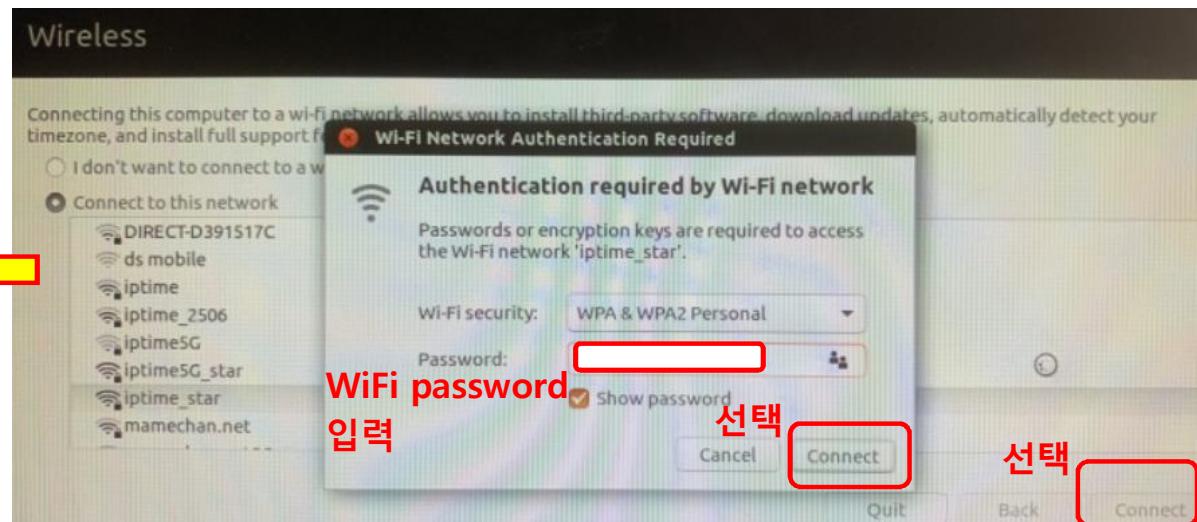
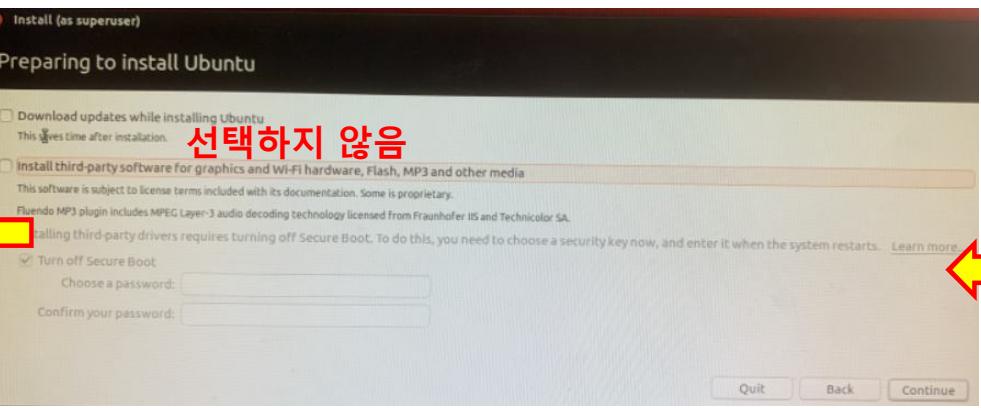
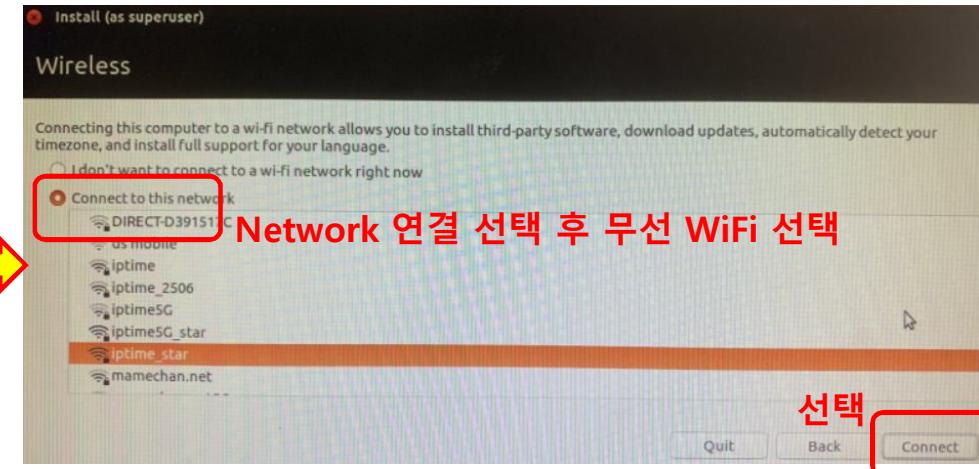
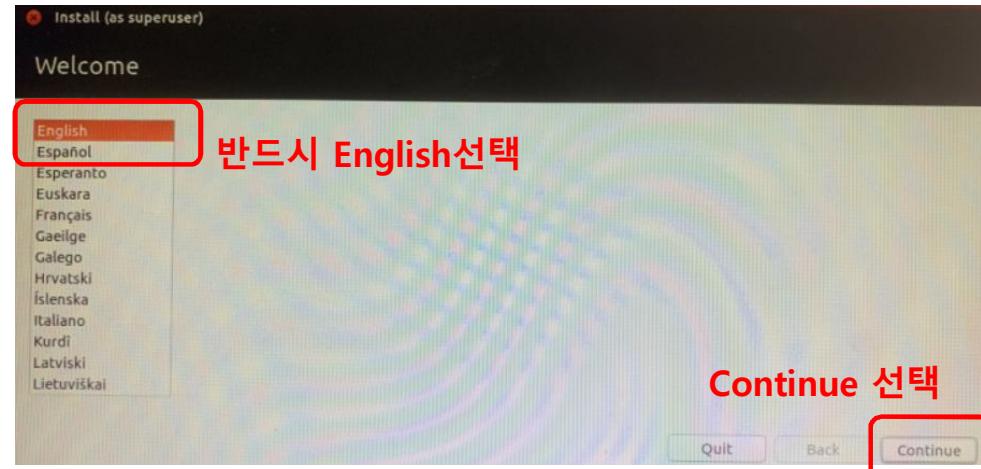
조금 기다림



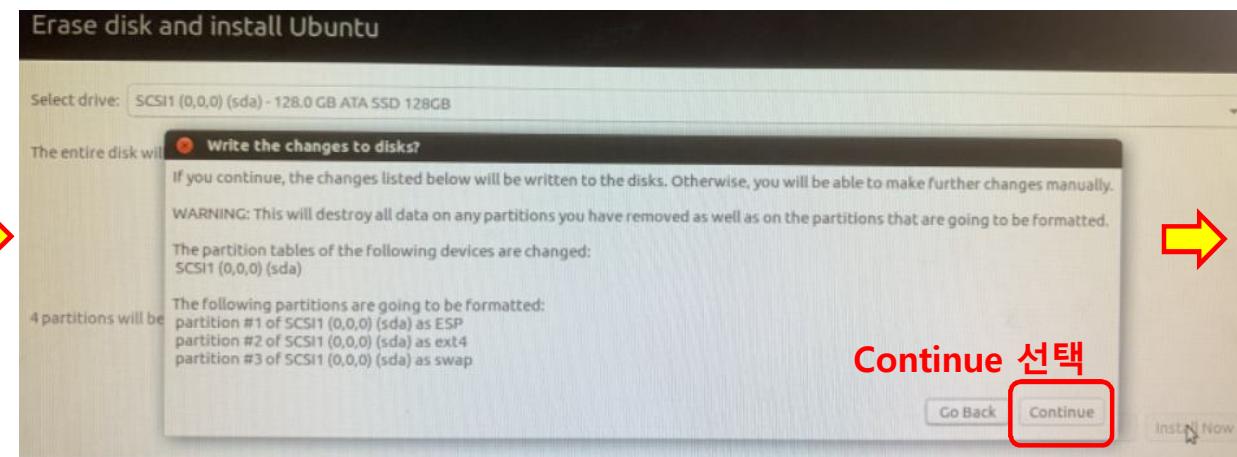
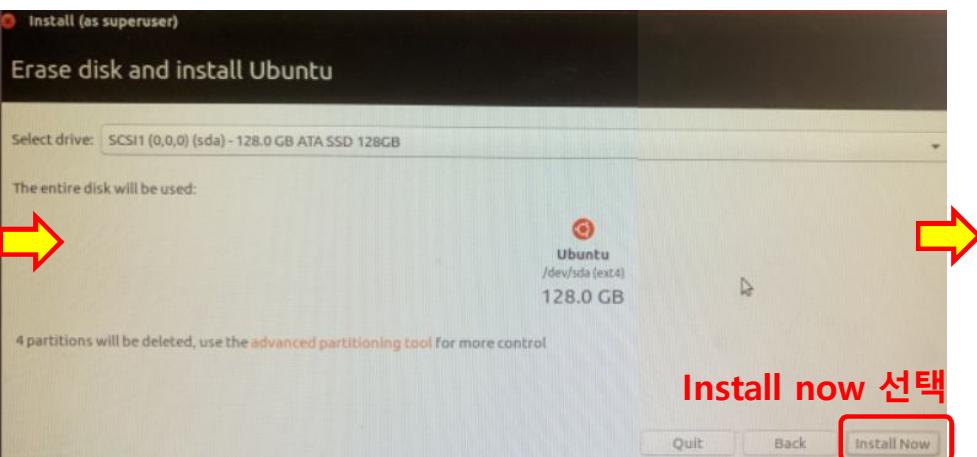
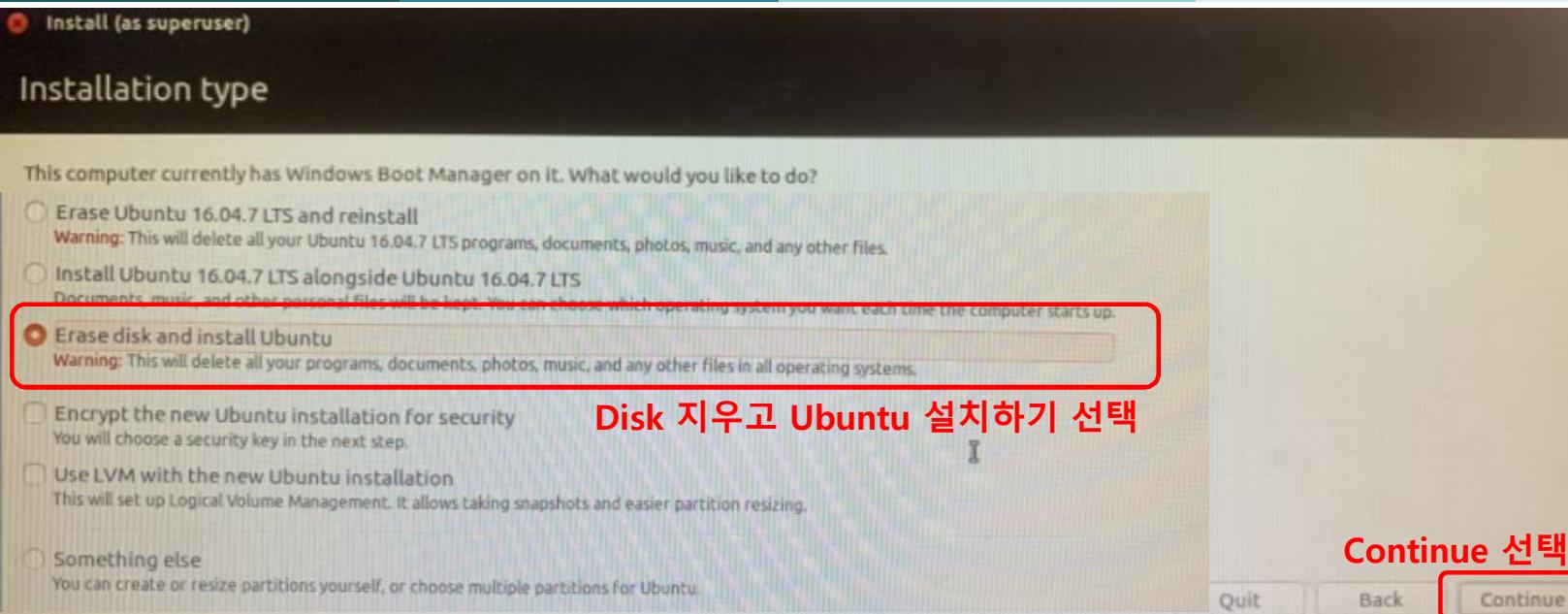
Ubuntu 설치 시작 화면

이미 Ubuntu가 설치 되어 있어도 F11을 누르면 설치 화면으로 진입

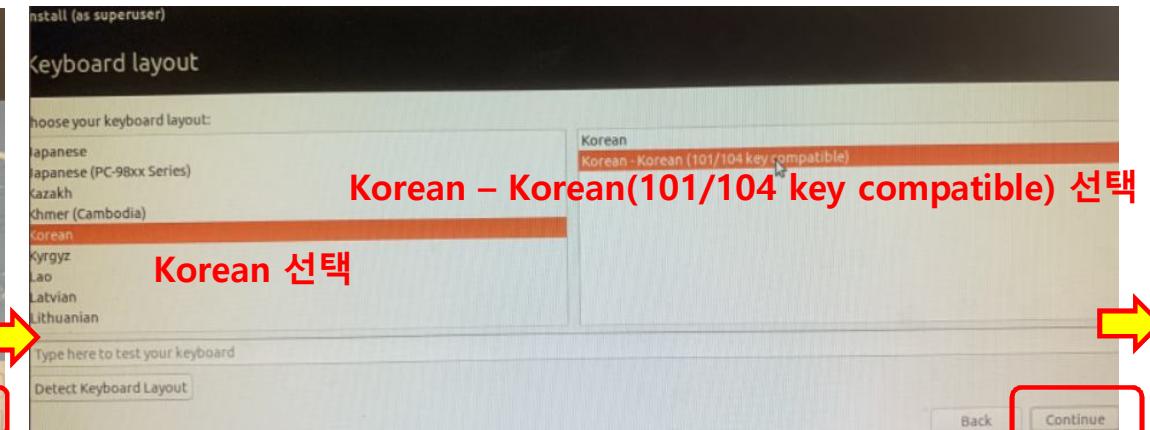
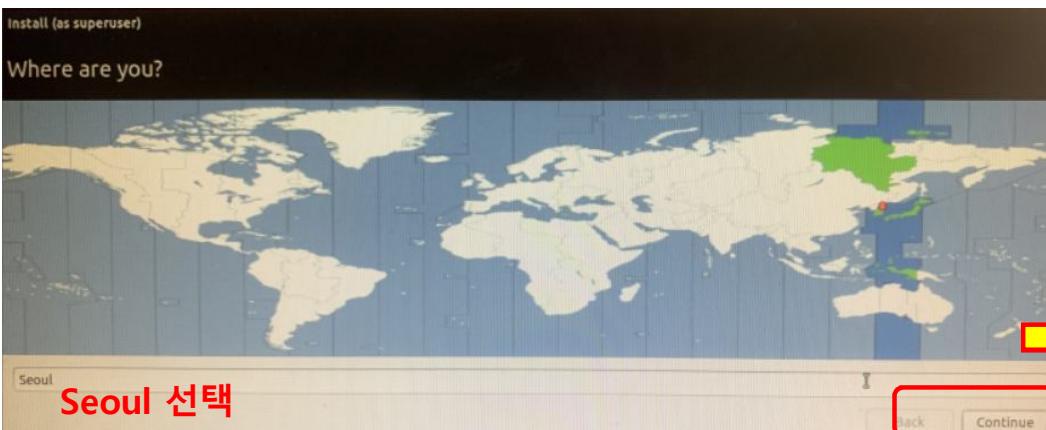
## 4. ROS 개발환경 구축



## 4. ROS 개발환경 구축



## 4. ROS 개발환경 구축

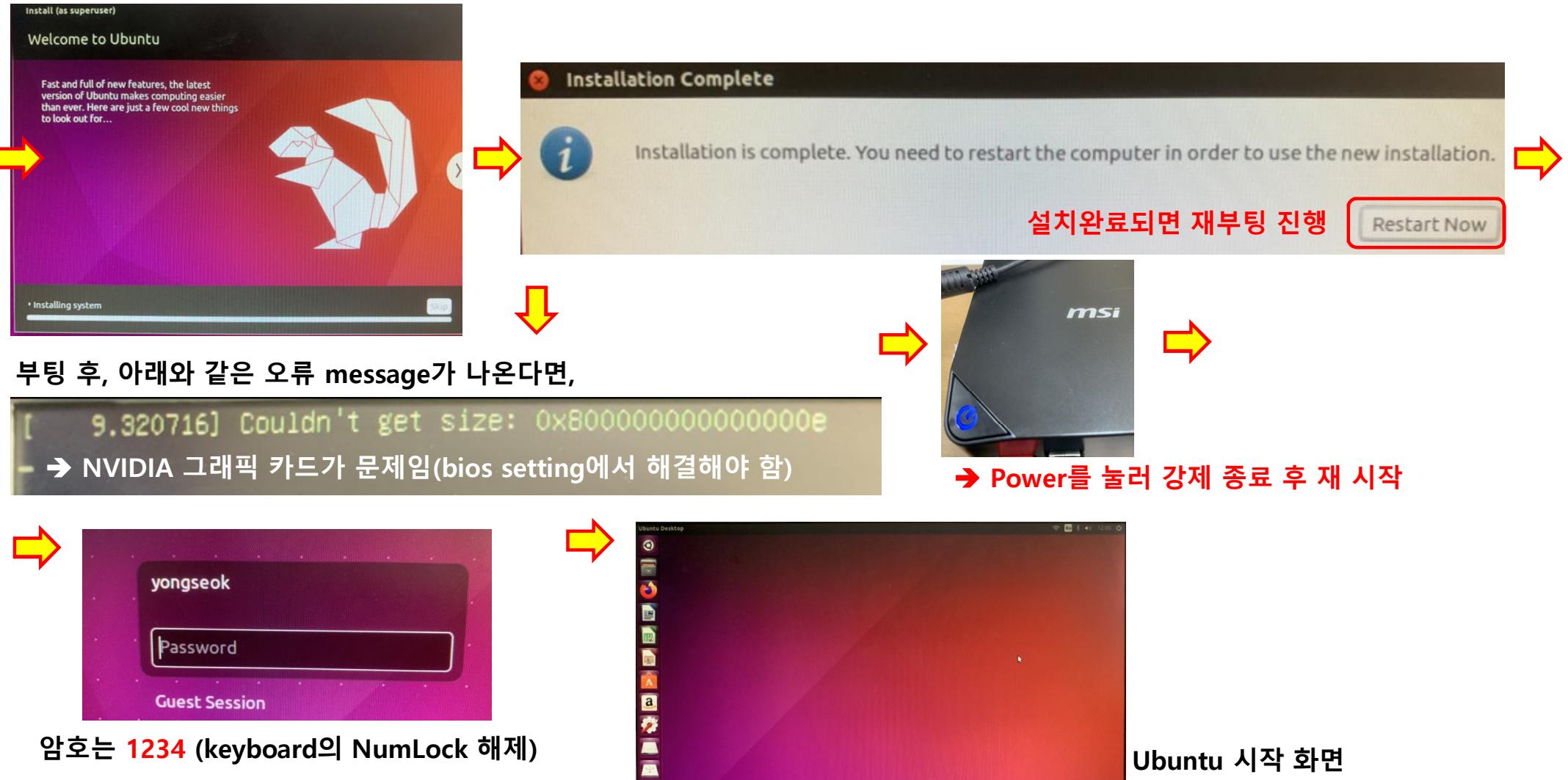


오직 영문 소문자 이름 넣기  
자동으로 생성되는데 지울 것  
숫자 1234 넣기  
check

Your name: yongseok  
Your computer's name: yongseok  
Pick a username: yongseok  
Choose a password: Short password  
Confirm your password:  
Log in automatically  
Require my password to log in  
Encrypt my home folder

Continue

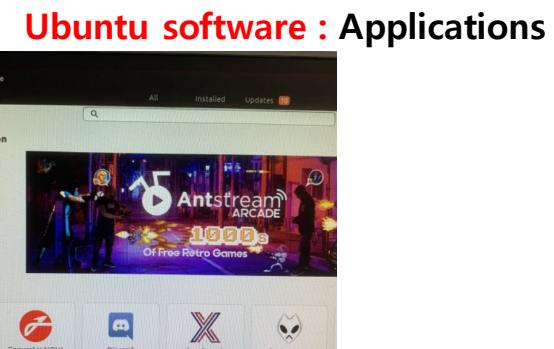
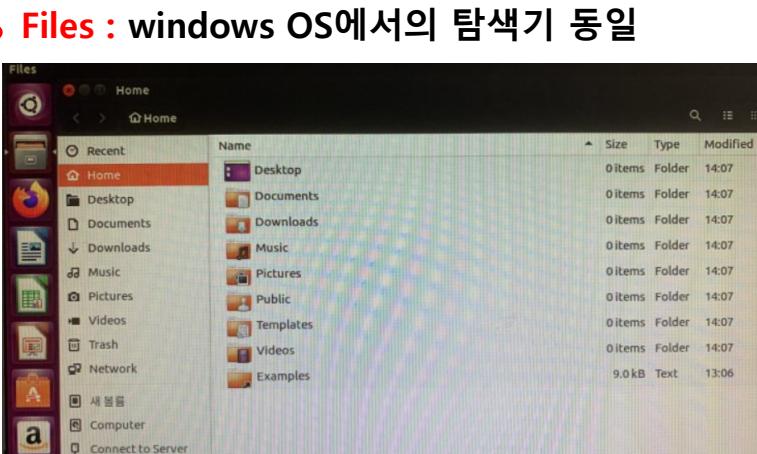
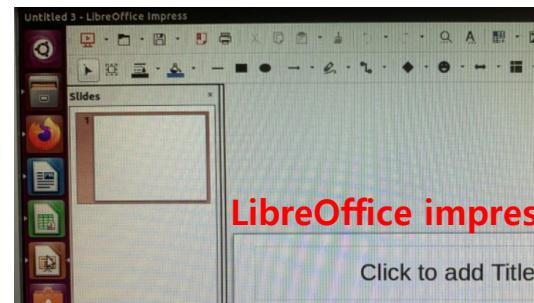
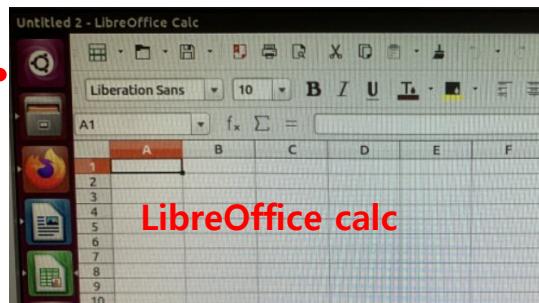
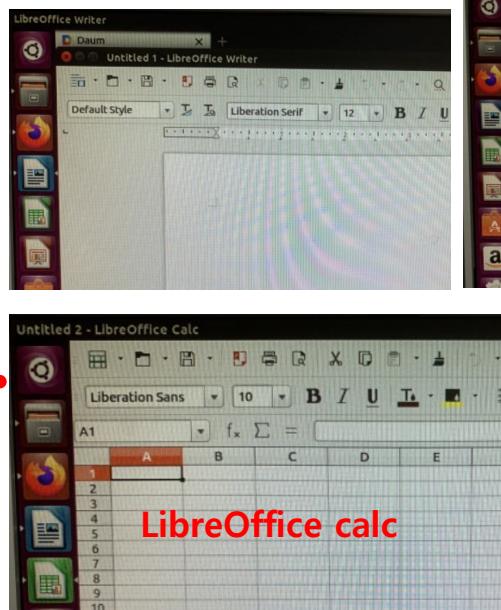
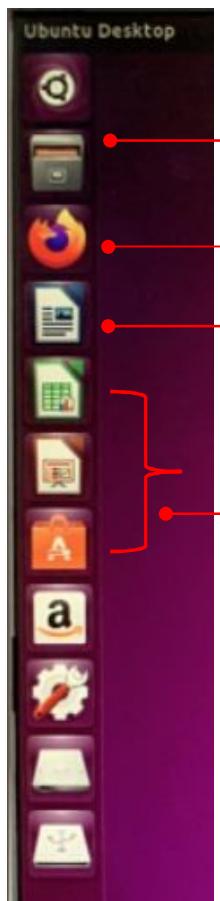
## 4. ROS 개발환경 구축



## 4. ROS 개발환경 구축

### (2) install Ubuntu on Remote PC

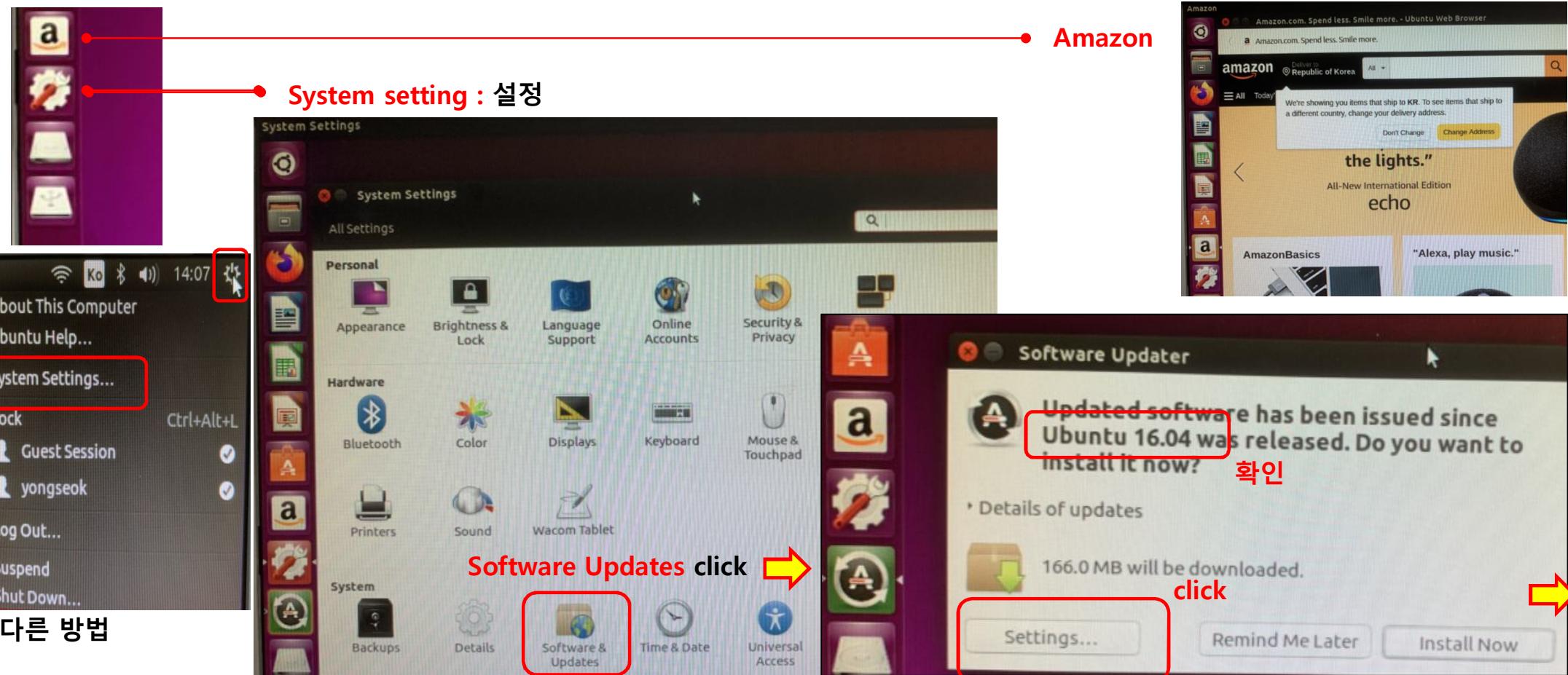
#### ② Ubuntu OS 이해하기



## 4. ROS 개발환경 구축

- install Ubuntu on Remote PC

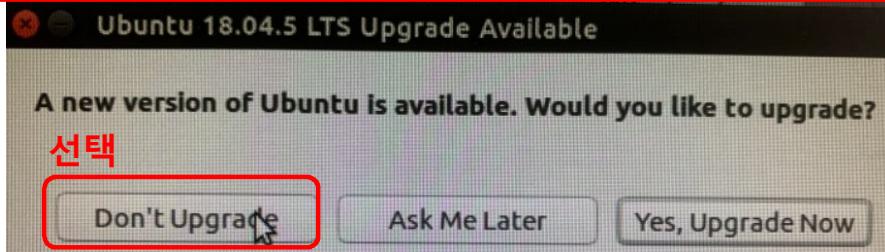
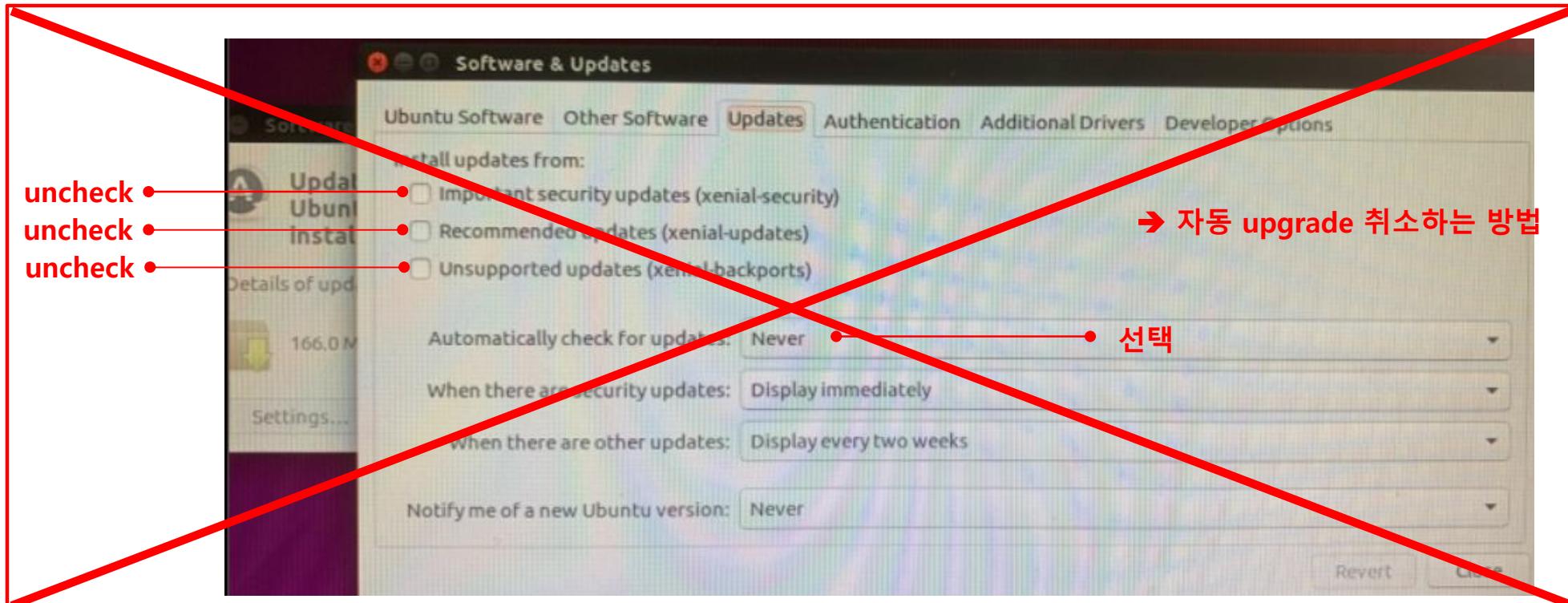
### ② Ubuntu OS 이해하기



## 4. ROS 개발환경 구축

Ubuntu upgrade 금지 : ROS와 TurtleBot에서 지원 안됨 ★

Ubuntu 16.04.7 LTS 유지하기 (ubuntu-18.04.5 LTS upgrade 금지)



## 4. ROS 개발환경 구축

### ③ Terminal 창 열기



: CTRL + ALT + T 를 사용해 터미널 창 열기

→ 반복 실행 시, 반복한 횟수 만큼 터미널 창이 여러 개 열림

```
yongseok@yongseok-MS-B09611: ~
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

yongseok@yongseok-MS-B09611: ~$
```

: sudo 명령어(super\_user do)

→ Linux에서 특정 명령을 실행하거나 파일에 접근하기 위해 root 권한이 필요.

일반 사용자(user)가 root 권한을 사용하기 위해서 sudo 명령어를 사용

→ sudo 다음에 실행할 명령을 입력하면 root 권한으로 명령어를 실행

★: CTRL + Shift + Q 열려있는 터미널 창 닫기

→ 반복 실행 시, 반복한 횟수 만큼 터미널 창이 닫힘

root : 최고 관리자.

: 운영체제의 모든 것을 제어할 권리를 지님

: 시스템 파괴할 권한도 포함(실수든 해킹이든)

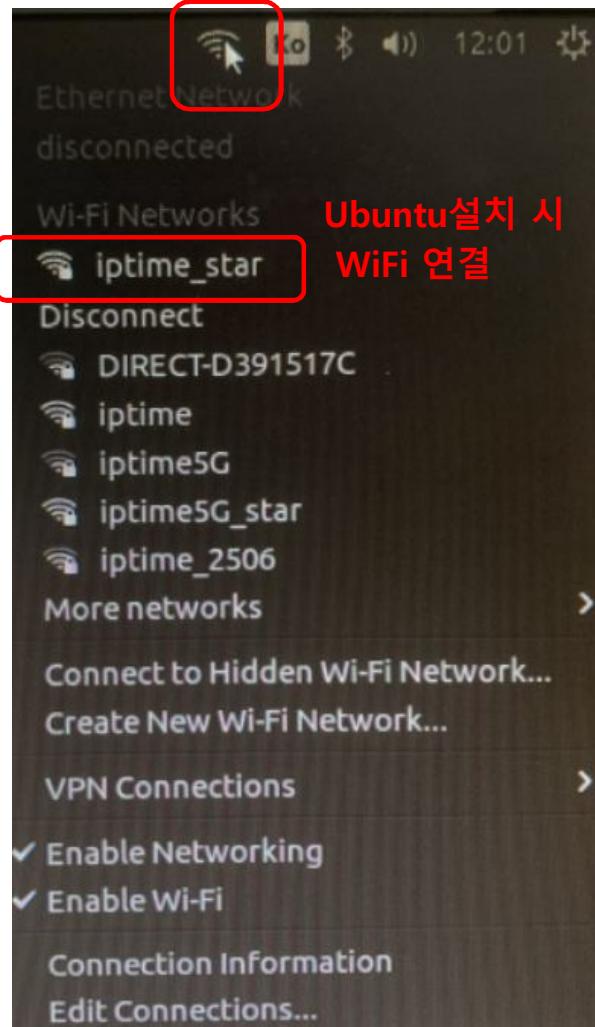
컴퓨터 주인의 계정 :

리눅스 운영체제에서는 기본적으로 컴퓨터 주인의 계정이 root보다 권한이 낮은 일반 사용자로 만들어짐.

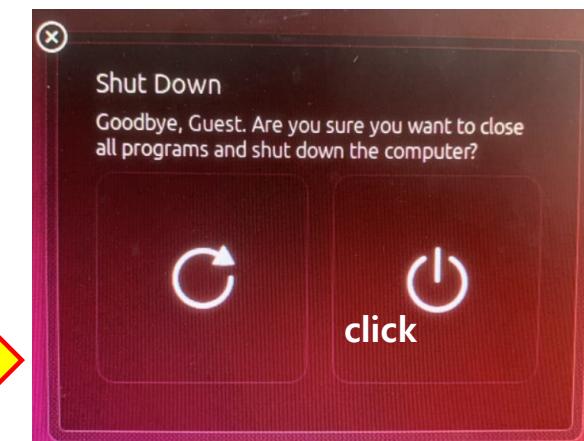
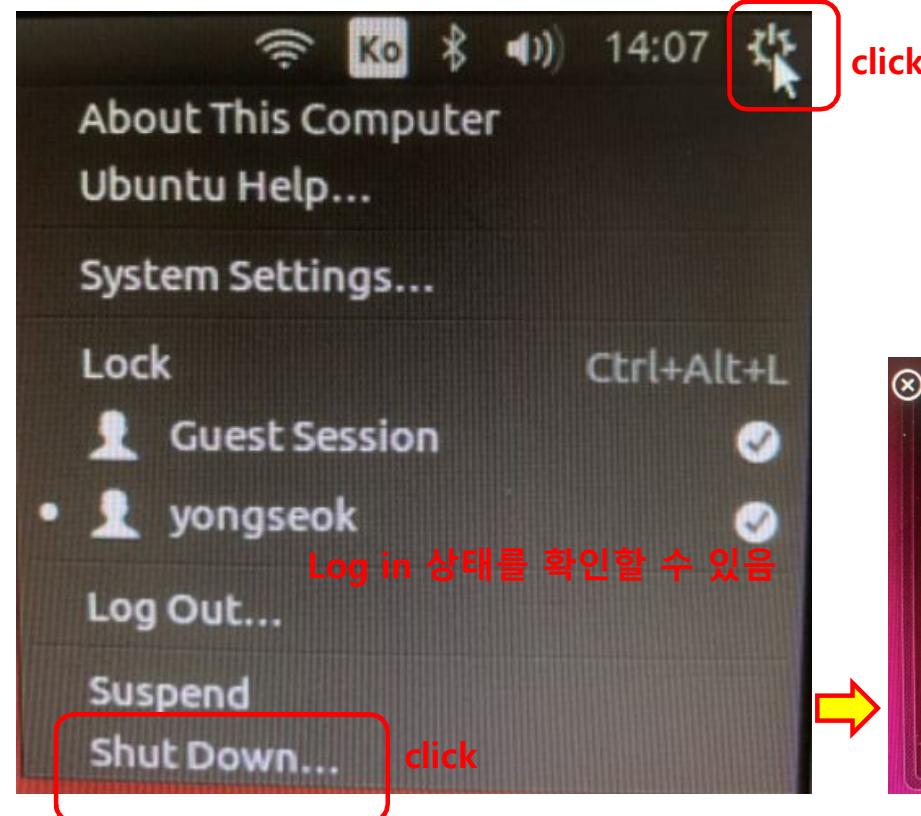
시스템 관리 작업 등을 할 때, sudo 라는 명령을 사용해서 임시 root 권한을 얻음

## 4. ROS 개발환경 구축

### ④ WiFi 확인하기



### ⑤ Shut down



# 강의실 WiFi 설치 (TP-LINK 고객센터 1899 1086) 10대~20대 연결가능

## [ 2407호 TP-Link 0484 ]

: AX10 AX1500 WIFI MODEL

1. 고정 IP 확인
2. TP-Link 전원만 연결하기
3. 뒷면에 RESET Button 볼펜으로 10초 누르기
4. LED 등이 왼쪽부터 초록색 3개 들어옴
5. 핸드폰 설정에서 WIFI TP-Link 0484 선택(암호 뒷면 [31542418](#))  
(못 찾으면 핸드폰 설정 나갔다. 들어갔다 반복)
6. 핸드폰에 주소 입력  
192.168.0.1 (Password 임의 설정 a123456)
7. 선택
8. 선택
9. Advanced – Network – Internet – Static IP 선택
10. 고정 IP입력 후 저장
11. TP-Link 0484에 인터넷 케이블 연결. 왼쪽부터 LED 초록 4개 켜짐
12. 핸드폰에서 인터넷 되는지 확인. 완료

## [ 2408호 TP-Link 1030 ]

: AX10 AX1500 WIFI MODEL

1. 고정 IP 확인
2. TP-Link 전원만 연결하기
3. 뒷면에 RESET Button 볼펜으로 10초 누르기
4. LED 등이 왼쪽부터 초록색 3개 들어옴
5. 핸드폰 설정에서 WIFI TP-Link 1030 선택(암호 뒷면 [56882230](#))  
(못 찾으면 핸드폰 설정 나갔다. 들어갔다 반복)
6. 핸드폰에 주소 입력  
192.168.0.1 (Password 임의 설정 a123456)
7. 선택
8. 선택
9. Advanced – Network – Internet – Static IP 선택
10. 고정 IP입력 후 저장
11. TP-Link 1030 [초록색\(WAN\)](#)에 인터넷 케이블 연결.  
왼쪽부터 LED 초록 4개 켜짐
12. 핸드폰에서 인터넷 되는지 확인. 완료



## 4. ROS 개발환경 구축

### ◆ ubuntu version 확인하기

: CTRL + ALT + T 를 사용해 터미널 창 열기

: lsb\_release -a

#### → ubuntu version 확인

yongseok@yongseok:~\$ lsb\_release -a  
No LSB modules are available.  
Distributor ID: Ubuntu  
Description: Ubuntu 16.04.7 LTS  
Release: 16.04  
Codename: xenial  
yongseok@yongseok:~\$

입력 후 enter

Version 확인

[명령어]

apt-get : 또는 apt(Advanced Packaging Tool)은 Debian 계열의 Ubuntu에서 쓰이는 package 관리 명령어

sudo : super-user 권한으로 실행하기 위함

wget : web get. 웹에서 파일을 다운로드할 때 사용되는 명령어

chmod 755: change mode. 대상 파일과 디렉토리의 사용권한을 변경할 때 사용. 755 소유자 권한(읽고, 쓰기, 실행)

bash : shell은 운영 체제 상에서 다양한 운영 체제 기능과 서비스를 구현하는 인터페이스를 제공하는 프로그램

: bash는 unix의 shell. / bash: 파일을 읽고 실행 bashrc 를 호출

## 4. ROS 개발환경 구축

: sudo apt-get update      가장 중요(ROS 환경을 위해)

: sudo apt-get upgrade

A terminal window showing the execution of two system update commands. The window title bar says 'yongseok@yongseok-MS-B09611:~'. The terminal content is as follows:

```
yongseok@yongseok-MS-B09611:~$ sudo apt-get update
[sudo] password for yongseok:
Hit:1 http://kr.archive.ubuntu.com/ubuntu xenial InRelease
Reading package lists... Done
yongseok@yongseok-MS-B09611:~$ sudo apt-get upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
yongseok@yongseok-MS-B09611:~$
```

Annotations with red lines and arrows explain the process:

- Line 1: '입력 후 enter' (Input then enter) points to the password prompt after 'sudo apt-get update'.
- Line 2: '1234 enter(입력 안보임)' (Input not visible) points to the password entry field.
- Line 3: 'update 화면(완료 확인할 것)' (Update screen (check completion)) points to the output of 'apt-get update'.
- Line 4: '입력 후 enter' (Input then enter) points to the password prompt after 'sudo apt-get upgrade'.
- Line 5: 'upgrade 화면(완료 확인할 것)' (Upgrade screen (check completion)) points to the output of 'apt-get upgrade'.

### ◆ Pdf viewer 설치하기

: sudo apt-get install xpdf

A terminal window showing the execution of the 'xpdf' package installation command. The window title bar says 'yongseok@yongseok:~\$ sudo apt-get install xpdf'. The terminal content is as follows:

```
yongseok@yongseok:~$ sudo apt-get install xpdf
[sudo] password for yongseok:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
```

Annotations with red lines and arrows explain the process:

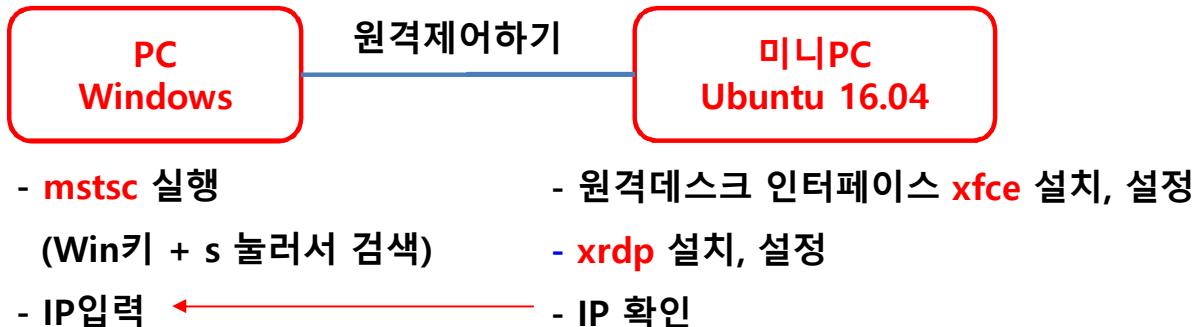
- Line 1: '입력 후 enter' (Input then enter) points to the password prompt after 'sudo apt-get install xpdf'.
- Line 2: '1234 enter(입력 안보임)' (Input not visible) points to the password entry field.
- Line 3: A yellow arrow points to the right with the text '이후, pdf 파일을 열어보기' (Afterwards, open a pdf file) pointing towards the end of the command output.

## 4. ROS 개발환경 구축

Ref. <https://jimnong.tistory.com/695>

## 4-2-2. Linux on Remote PC

## (1) Windows PC에서 ubuntu를 원격으로 제어하기



mstsc

: windows 환경의  
원격데스크톱 연결 제어 프로그램

# xrdp

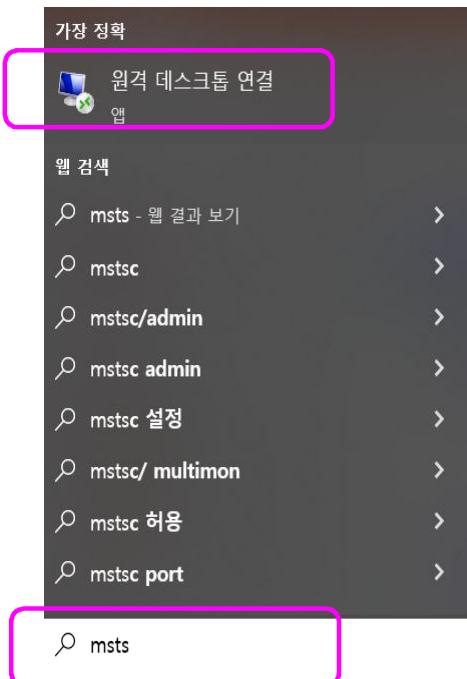
... 원격 접속 프로그램으로, 원도우 환경에서 GUI 환경으로 Linux 를 사용하기 위함

xfce

: 리눅스 플랫폼을 위한 자유 소프트웨어 데스크톱 환경의 GUI 환경

```
: sudo apt-get install xfce4
```

#### → 원격데스크 인터페이스 설치



## 4. ROS 개발환경 구축

### 4-2-2. Linux on Remote PC

#### (1) Windows PC에서 ubuntu를 원격으로 제어하기

: systemctl status xrdp

→ ubuntu에서 xrdp 패키지 설치 여부 확인

: sudo apt-get install xrdp

→ ubuntu에서 xrdp 패키지 설치

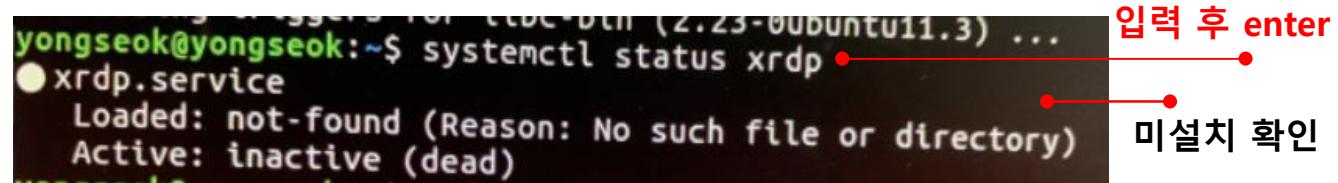
※

.sh (shell script)

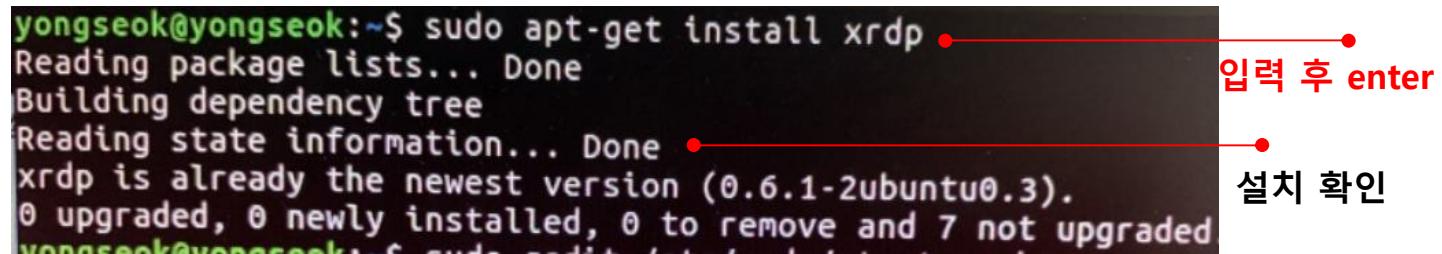
: 예, startwm.sh / install\_ros\_kinetic.sh 등 일반적으로 shell script는 확장자 지니지 않음

: shell script는 명령어를 실행하거나 프로그램을 실행하는 구문으로 이해

(여러 명령어나 프로그램 실행을 파일 형식으로 갖고 있어, 한번에 실행해 줌)



```
yongseok@yongseok:~$ systemctl status xrdp
● xrdp.service
  Loaded: not-found (Reason: No such file or directory)
  Active: inactive (dead)
```



```
yongseok@yongseok:~$ sudo apt-get install xrdp
Reading package lists... Done
Building dependency tree
Reading state information... Done
xrdp is already the newest version (0.6.1-2ubuntu0.3).
0 upgraded, 0 newly installed, 0 to remove and 7 not upgraded
yongseok@yongseok:~$
```

## 4. ROS 개발환경 구축

: sudo gedit /etc/xrdp/startwm.sh

→ xrdp 설정파일중에 startwm.sh 수정(edit) → 저장

```
yongseok@yongseok:~$ sudo gedit /etc/xrdp/startwm.sh
```



Linux 프로그램 편집 tool  
: vi edit , gedit

```
#!/bin/sh
```

```
if [ -r /etc/default/locale ]; then
    . /etc/default/locale
    export LANG LANGUAGE
fi
```

```
#xrdp multiple users configuration
xfce4-session
```

```
#. /etc/X11/Xsession
./usr/bin/startxfce4
```

: 초기화면

```
#!/bin/sh

if [ -r /etc/default/locale ]; then
    . /etc/default/locale
    export LANG LANGUAGE
fi

. /etc/X11/Xsession
```



: 수정화면

```
startwm.sh (/etc/xrdp) - yongseok@yongseok: ~
File Edit View Search Tools Documents Help
startwm.sh
/etc/xrdp

#!/bin/sh

if [ -r /etc/default/locale ]; then
    . /etc/default/locale
    export LANG LANGUAGE
fi

#xrdp multiple users configuration
xfce4-session

#. /etc/X11/Xsession
./usr/bin/startxfce4
```



## 4. ROS 개발환경 구축

: sudo systemctl enable --now xrdp

→ 부팅 후 자동 실행되도록 설정

: sudo ufw allow from any to any port 3389 proto tcp

→ 방화벽 설정 추가. 3389 포트 열기

: sudo /etc/init.d/xrdp restart

→ xrdp를 재 시작

: ifconfig

(네트워크 설정 확인. 유무선)

→ Ubuntu (Remote PC)의 IP 검색하기

ip 기억하기

```
yongseok@yongseok:~$ sudo systemctl enable --now xrdp
[sudo] password for yongseok:
xrdp.service is not a native service, redirecting to systemd-sysv-install
Executing /lib/systemd/systemd-sysv-install enable xrdp
yongseok@yongseok:~$ sudo ufw allow from any to any port 3389 proto tcp
Rules updated
Rules updated (v6)
yongseok@yongseok:~$
```

```
yongseok@yongseok:~$ ifconfig
enp2s0      Link encap:Ethernet HWaddr 4c:cc:6a:9d:47:16
             UP BROADCAST MULTICAST MTU:1500 Metric:1
             RX packets:0 errors:0 dropped:0 overruns:0 frame:0
             TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
             collisions:0 txqueuelen:1000
             RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

lo         Link encap:Local Loopback
           inet addr:127.0.0.1 Mask:255.0.0.0
           inet6 addr: ::1/128 Scope:Host
             UP LOOPBACK RUNNING MTU:65536 Metric:1
             RX packets:3638 errors:0 dropped:0 overruns:0 frame:0
             TX packets:3638 errors:0 dropped:0 overruns:0 carrier:0
             collisions:0 txqueuelen:1000
             RX bytes:713589 (713.5 KB)  TX bytes:713589 (713.5 KB)

wlp3s0      Link encap:Ethernet HWaddr f4:06:69:f0:af:37
             inet addr:192.168.0.15 Bcast:192.168.0.255 Mask:255.255.255.0
             inet6 addr: fe80::5dc8:f153:79d5:78cf/64 Scope:Link
               UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
               RX packets:553870 errors:0 dropped:0 overruns:0 frame:0
               TX packets:224046 errors:0 dropped:0 overruns:0 carrier:0
               collisions:0 txqueuelen:1000
               RX bytes:810194447 (810.1 MB)  TX bytes:23074618 (23.0 MB)
```

## 4. ROS 개발환경 구축

: netstat -antp

→ 원격데스크톱의 기본 포트 3389 확인하기

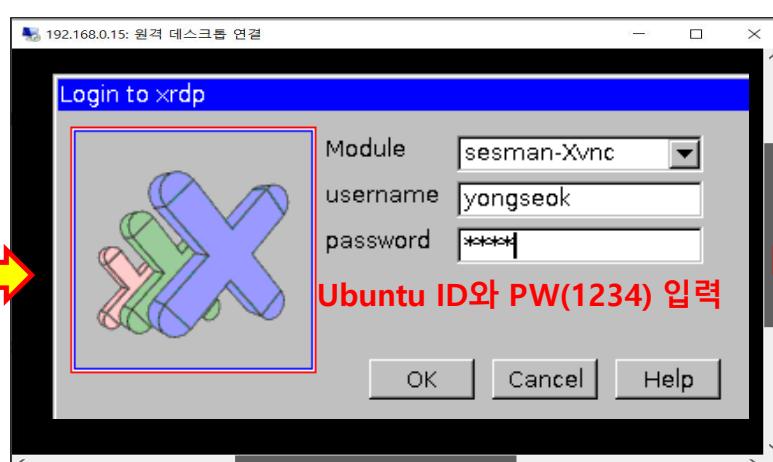
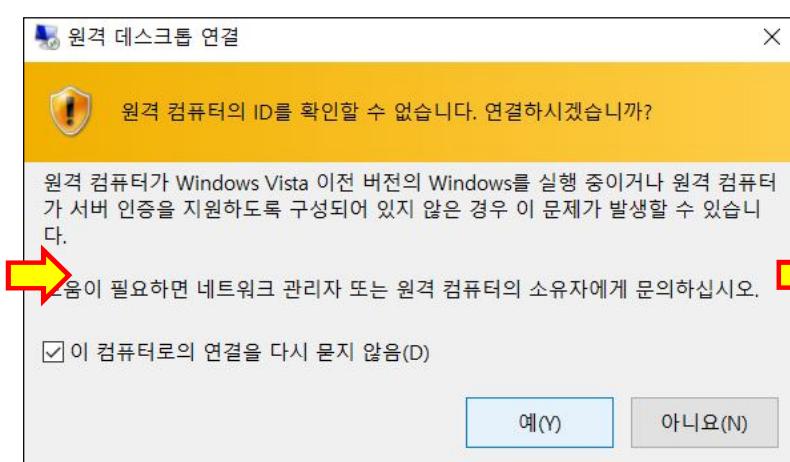
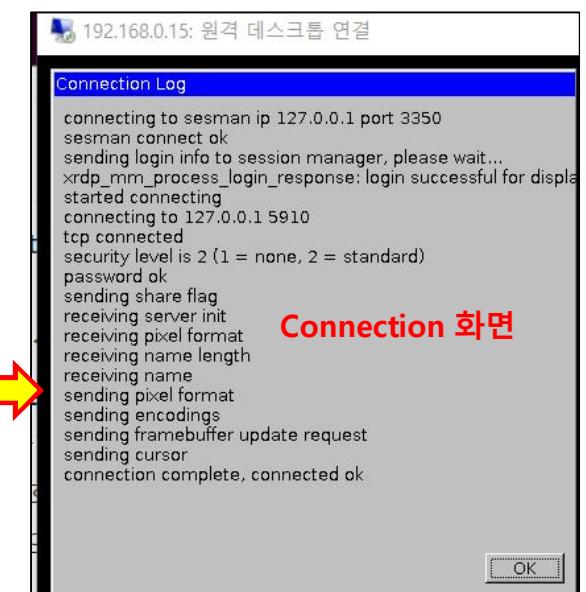
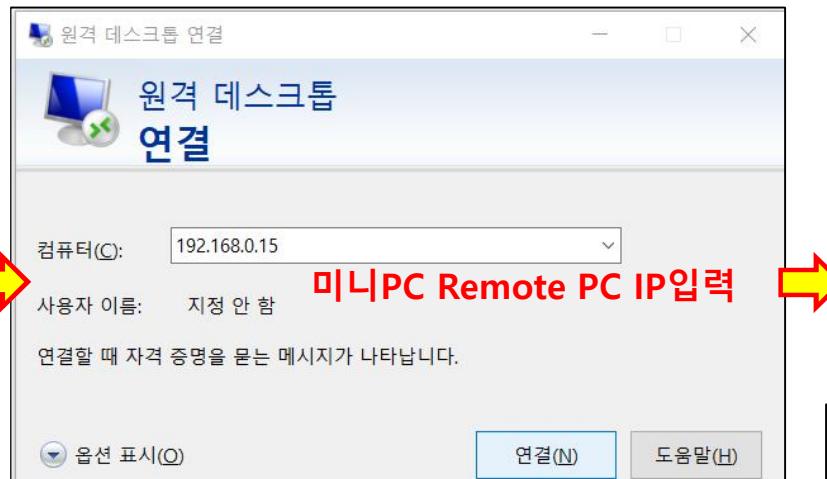
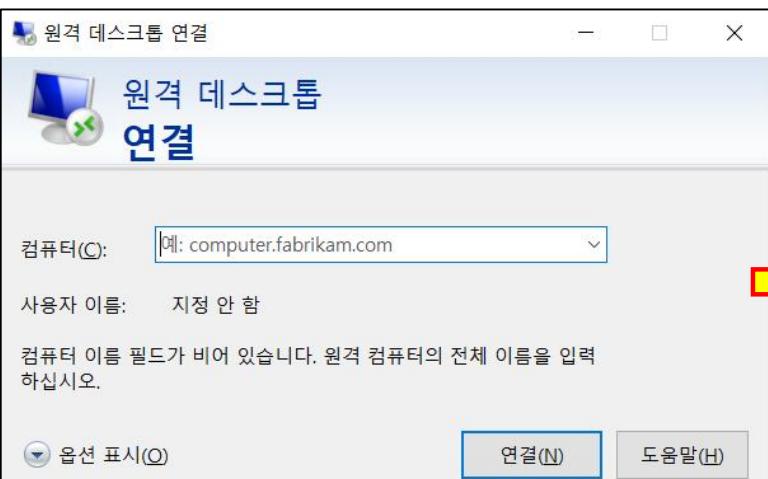
Active Internet connections (servers and established)					
Proto	Recv-Q	Send-Q	Local Address	Foreign Address	State
PID/Program name					
tcp	0	0	127.0.1.1:53	0.0.0.0:*	LISTEN
-	-	-	-	-	-
tcp	0	0	127.0.0.1:3350	0.0.0.0:*	LISTEN
-	-	-	-	-	-
tcp	0	0	127.0.0.1:631	0.0.0.0:*	LISTEN
-	-	-	-	-	-
tcp	0	0	0.0.0.0:3389	0.0.0.0:*	LISTEN
2927/firefox	0	0	192.168.0.15:50036	172.217.26.34:443	ESTABLISHED
2927/firefox	0	0	192.168.0.15:50548	52.39.178.28:443	ESTABLISHED
2927/firefox	0	0	192.168.0.15:60674	142.250.196.130:443	ESTABLISHED
2927/firefox	0	0	192.168.0.15:49770	172.217.25.226:443	ESTABLISHED
2927/firefox	0	1	192.168.0.15:49670	172.217.161.36:443	SYN_SENT
tcp6	0	0	::1:631	:::*	LISTEN

확인하기

확인하기

## 4. ROS 개발환경 구축

: Windows PC에서 **mstsc** 실행(Win키 + s 놀러서 검색)

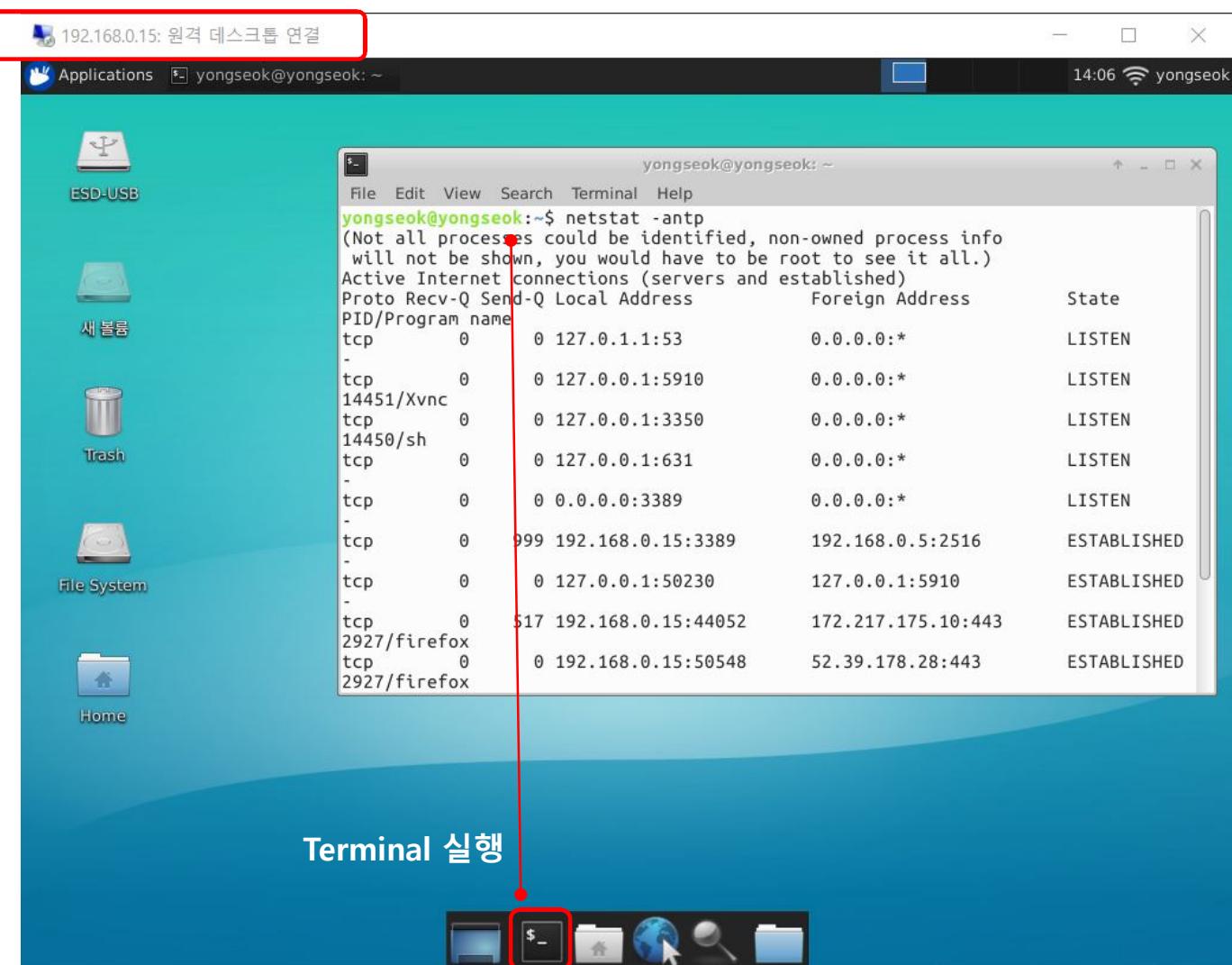


## 4. ROS 개발환경 구축

: Windows PC에서

ubuntu (Remote PC) 원격제어 화면

ubuntu (Remote PC) 확인

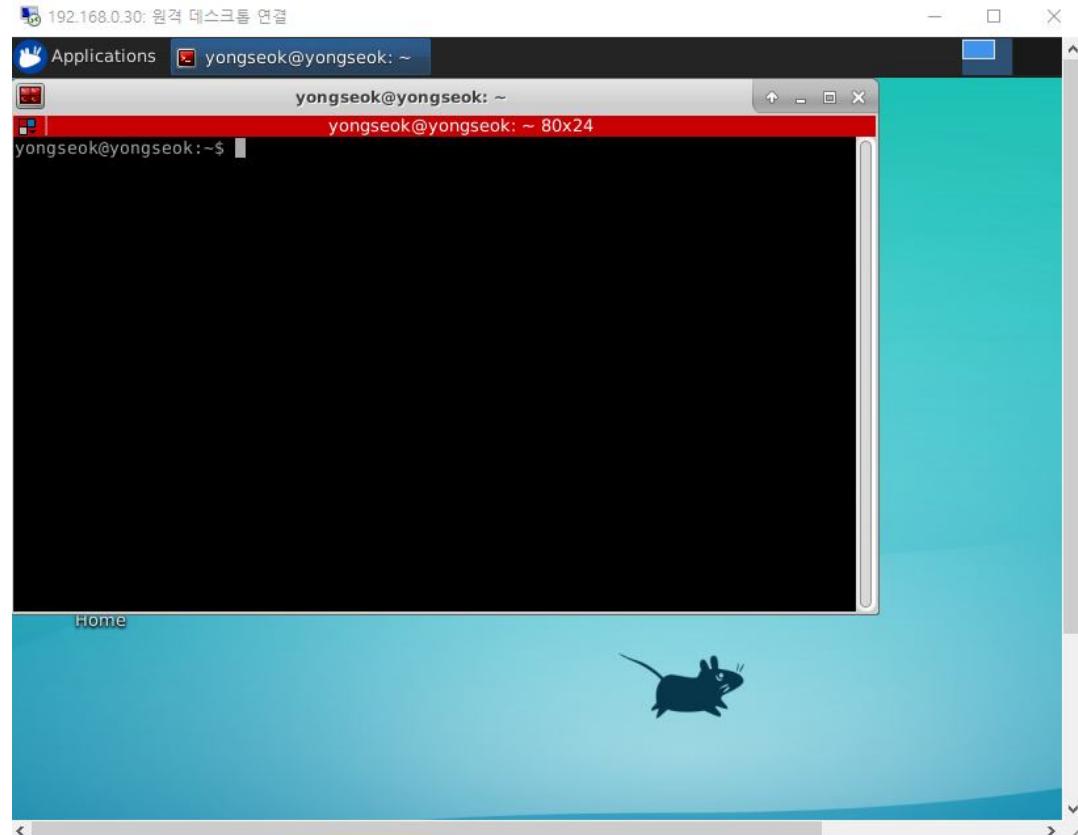


Terminal 실행

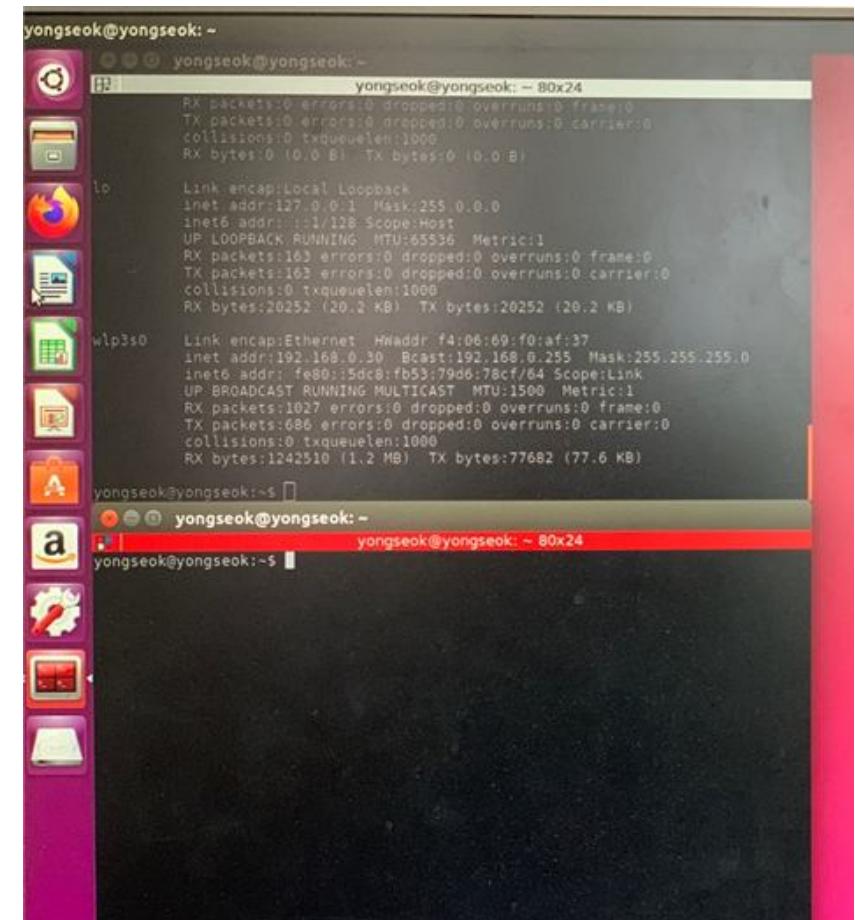
## 4. ROS 개발환경 구축

: Windows PC에서

ubuntu (Remote PC) 원격제어 화면



: Ubuntu 화면



## 4. ROS 개발환경 구축

### 4-2-2. Linux on Remote PC

#### (2) Linux 실습

: 무선 네트워크 on, off

→ **ifconfig** 네트워크 설정 확인 (유무선)

→ **iwconfig** 무선랜 설정들을 확인

{ : Windows PC에서 Remote PC) 원격제어

한다면 명령어 앞에 sudo 필요

```
yongseok@yongseok:~$ sudo iwconfig  
enp2s0    no wireless extensions.
```

```
wlp3s0    IEEE 802.11 ESSID:"iptime_star"  
          Mode:Managed Frequency:2.467 GHz Access Point: 88:36:6C:89:8A:28  
          Bit Rate=150 Mb/s Tx-Power=22 dBm  
          Retry short limit:7 RTS thr:off Fragment thr:off  
          Encryption key:off  
          Power Management:on  
          Link Quality=70/70 Signal level=-36 dBm  
          Rx invalid nwid:0 Rx invalid crypt:0 Rx invalid frag:0  
          Tx excessive retries:0 Invalid misc:801 Missed beacon:0
```

```
lo      no wireless extensions.
```

```
yongseok@yongseok:~$ clear Terminal 창 갈무리
```

```
yongseok@yongseok:~$ ifconfig
```

```
Command 'ifconfig' is available in '/sbin/ifconfig'  
The command could not be located because '/sbin' is not included in the PATH environment variable.  
This is most likely caused by the lack of administrative privileges associated with your user account.
```

```
ifconfig: command not found
```

```
yongseok@yongseok:~$ sudo ifconfig
```

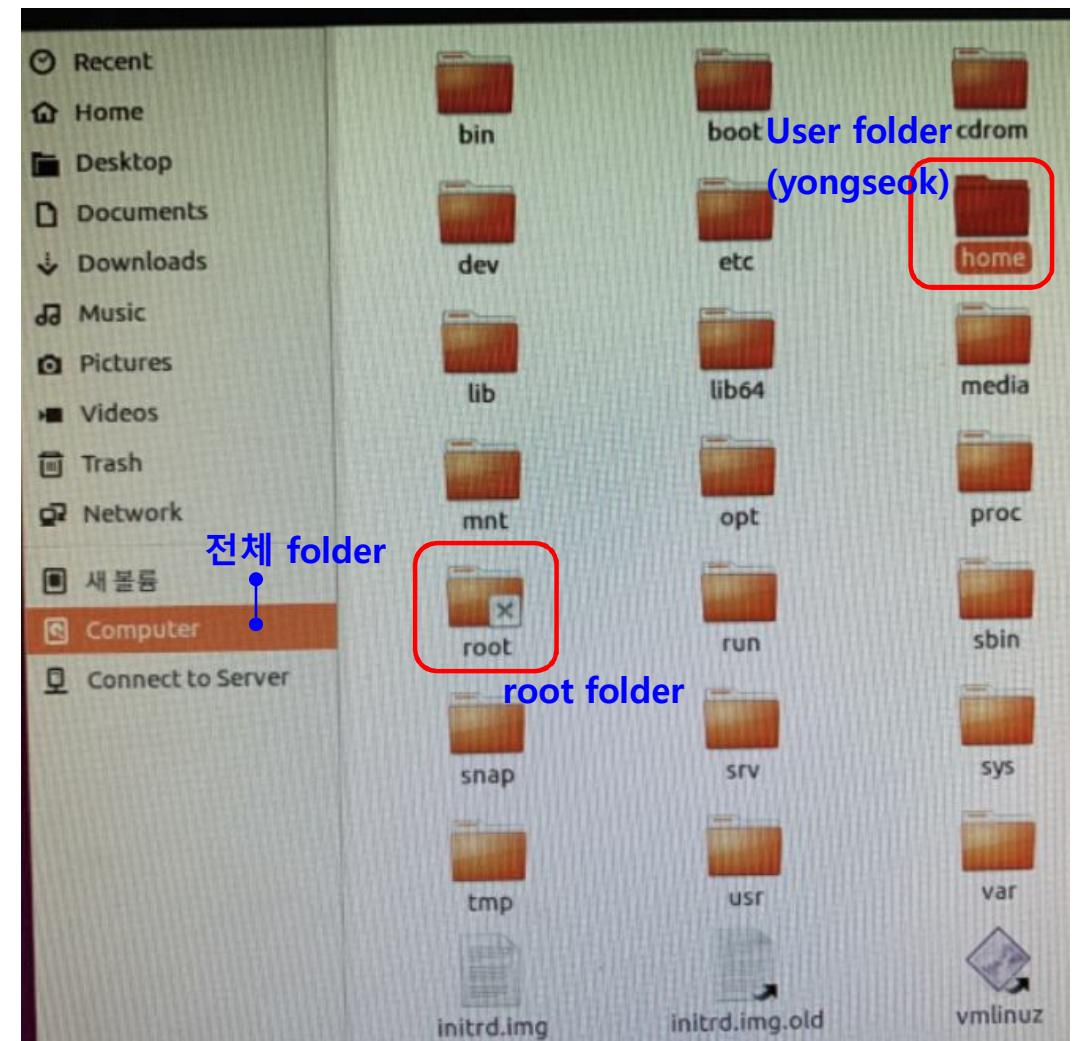
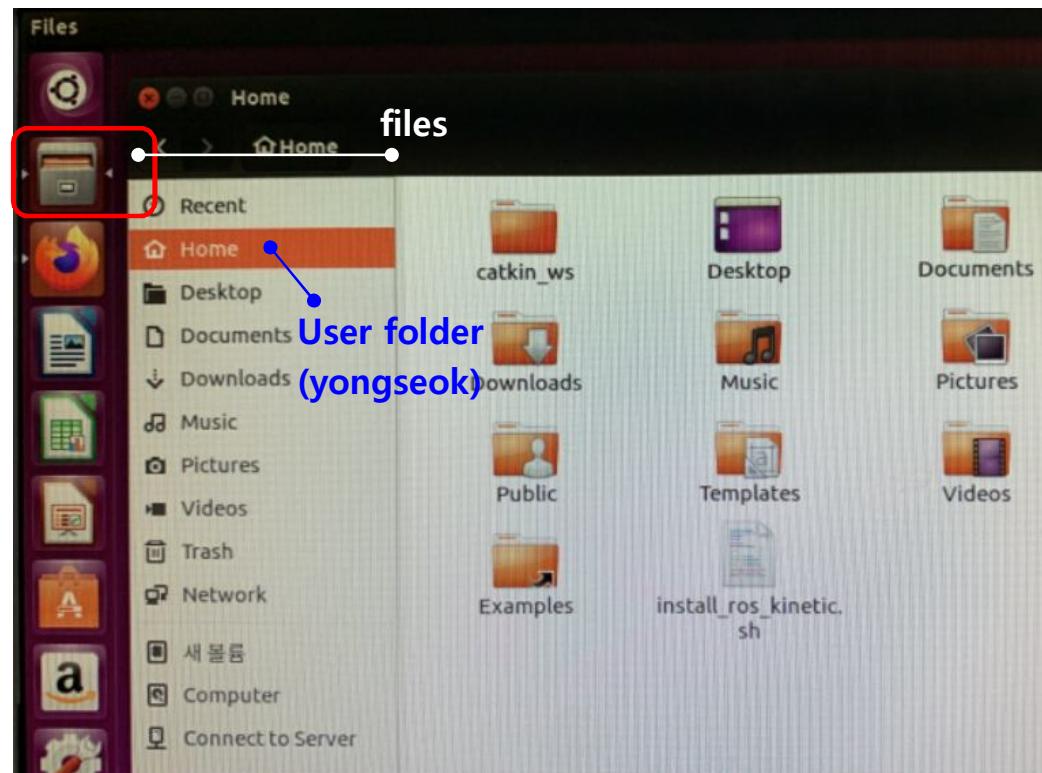
```
[sudo] password for yongseok: _____ 1234 enter  
enp2s0      Link encap:Ethernet HWaddr 4c:cc:6a:9d:47:16  
            UP BROADCAST MULTICAST MTU:1500 Metric:1  
            RX packets:0 errors:0 dropped:0 overruns:0 frame:0  
            TX packets:0 errors:0 dropped:0 overruns:0 carrier:0  
            collisions:0 txqueuelen:1000  
                  .0 B) TX bytes:0 (0.0 B)
```

```
cal Loopback  
.0.0.1 Mask:255.0.0.0  
.1/128 Scope:Host  
UNNING MTU:65536 Metric:1  
740 errors:0 dropped:0 overruns:0 frame:0  
740 errors:0 dropped:0 overruns:0 carrier:0  
txqueuelen:1000  
6715 (74.7 MB) TX bytes:74756715 (74.7 MB)
```

```
hernet HWaddr f4:06:69:f0:af:37  
.168.0.15 Bcast:192.168.0.255 Mask:255.255.255.0  
e80::5dc8:fb53:79d6:78cf/64 Scope:Link  
RUNNING MULTICAST MTU:1500 Metric:1  
RX packets:67312 errors:0 dropped:0 overruns:0 frame:0
```

## 4. ROS 개발환경 구축

: files 경로 확인



## 4. ROS 개발환경 구축

: cd ..

[현재 나의 위치 확인하기]

: cd ~

: ls

```
yongseok@yongseok:~$ ls
catkin_ws  Downloads          Music    Templates   Videos
Desktop    examples.desktop    Pictures  test
Documents  install_ros_kinetic.sh Public   test_linux
yongseok@yongseok:~$ cd ..
yongseok@yongseok:/home$ ls
yongseok
yongseok@yongseok:/home$ cd ..
yongseok@yongseok:/$ ls
bin      dev  initrd.img     lib64      mnt      root     snap   tmp  vmlinuz
boot    etc  initrd.img.old lost+found  opt      run      srv   usr
cdrom   home lib           내 부모 폴더가 속해 있는 폴더 확인  media   proc   sbin  sys   var
yongseok@yongseok:/$ cd ~
yongseok@yongseok:~$ ls
catkin_ws  Downloads          Music    Templates   Videos
Desktop    examples.desktop    Pictures  test
Documents  install_ros_kinetic.sh Public   test_linux
yongseok@yongseok:~$
```

## 4. ROS 개발환경 구축

: uname

```
yongseok@yongseok:~$ uname • 현재 사용하는 운영체제
```

: date

```
yongseok@yongseok:~$ date  
Wed Sep 15 16:22:01 KST 2021
```

: who

```
yongseok@yongseok:~$ who • 현재 로그인 사용자  
yongseok tty7 2021-09-14 11:08 (:0)
```

: ls

```
yongseok@yongseok:~$ ls • 파일 목록  
catkin_ws Documents examples.desktop Music Public Videos  
Desktop Downloads install_ros_kinetic.sh Pictures Templates
```

: hostname

```
yongseok@yongseok:~$ hostname  
yongseok  
yongseok@yongseok:~$ █
```

: pwd

```
yongseok@yongseok:~$ pwd • 현재 작업 디렉토리를 표시  
/home/yongseok
```

: cd

```
yongseok@yongseok:~$ cd • Home 디렉토리로 이동
```

: ls -s

```
yongseok@yongseok:~$ ls -s • 현재 디렉토리, size 표시  
total 56  
4 catkin_ws 4 Downloads 4 Music 4 Templates  
4 Desktop 12 examples.desktop 4 Pictures 4 test  
4 Documents 4 install_ros_kinetic.sh 4 Public 4 Videos
```

: ls -a

```
yongseok@yongseok:~$ ls -a • 현재 디렉토리, all 표시  
. Documents Public  
.. Downloads .ros  
.bash_history examples.desktop .sudo_as_admin_successful  
.bash_logout .gconf Templates  
.bashrc .gnupg test  
.cache .ICEAuthority Videos  
catkin_ws install_ros_kinetic.sh .vnc  
.compiz .local .wget-hsts  
.config .mozilla .Xauthority  
.dbus Music .xsession-errors  
Desktop Pictures .xsession-errors.old  
.dmrc .profile  
yongseok@yongseok:~$ █
```

## 4. ROS 개발환경 구축

: mkdir 폴더명

```
yongseok@yongseok:~$ cd ~ • Home 디렉토리로 이동(내 고향)
yongseok@yongseok:~$ ls
```

: cd ~

```
catkin_ws  Documents  examples.desktop  Music  Public  test
Desktop   Downloads  install_ros_kinetic.sh  Pictures  Templates  Videos
```

: cd ..

```
yongseok@yongseok:~$ mkdir test_linux • 새로운 디렉토리 만들기
```

: cd 폴더명

```
catkin_ws  Downloads  Music  Templates  Videos
Desktop   examples.desktop  Pictures  test
Documents  install_ros_kinetic.sh  Public  test_linux 생성
```

```
yongseok@yongseok:~$ cd test_linux • 폴더명 디렉토리로 이동하기
```

```
yongseok@yongseok:~/test_linux$ ls
yongseok@yongseok:~/test_linux$ mkdir test_linux2 • 새로운 디렉토리 만들기
```

```
yongseok@yongseok:~/test_linux$ ls
test_linux2 생성
```

```
yongseok@yongseok:~/test_linux$ cd ~ • home디렉토리로 이동
```

```
yongseok@yongseok:~/test_linux$ cd test_linux • 폴더명 디렉토리로 이동하기
```

```
yongseok@yongseok:~/test_linux$ cd .. • 부모 디렉토리로 이동
```

```
yongseok@yongseok:~$ █
```

명령어	의미
ls	파일 및 디렉터리 리스트
ls -a	모든 파일과 디렉터리 리스트
ls -asl	모든 파일 자세히 리스트
mkdir	디렉터리 만들기
cd 디렉터리	디렉터리로 이동
cd	홈 디렉터리로 이동
cd ~	홈 디렉터리로 이동
cd ..	부모 디렉터리로 이동
pwd	현재 작업 디렉터리 프린트

## 4. ROS 개발환경 구축

: ls -asl

```
yongseok@yongseok:~$ ls -asl
total 156
4 drwxr-xr-x 23 yongseok yongseok 4096 Sep 15 16:56 .
4 drwxr-xr-x  3 root    root    4096 Sep 13 23:21 ..
4 -rw-----  1 yongseok yongseok 3494 Sep 14 11:06 .bash_history
4 -rw-r--r--  1 yongseok yongseok 220 Sep 13 23:21 .bash_logout
8 -rw-r--r--  1 yongseok yongseok 4858 Sep 14 01:18 .bashrc
4 drwx----- 16 yongseok yongseok 4096 Sep 14 14:00 .cache
4 drwxrwxr-x  5 yongseok yongseok 4096 Sep 14 01:18 catkin_ws
4 drwx-----  3 yongseok yongseok 4096 Sep 14 11:06 .compiz
4 drwx----- 17 yongseok yongseok 4096 Sep 14 14:00 .config
4 drwx-----  3 yongseok yongseok 4096 Sep 14 11:15 .dbus
4 drwxr-xr-x  2 yongseok yongseok 4096 Sep 13 23:29 Desktop
4 -rw-r--r--  1 yongseok yongseok  25 Sep 13 23:29 .dmrc
4 drwxr-xr-x  2 yongseok yongseok 4096 Sep 13 23:29 Documents
4 drwxr-xr-x  2 yongseok yongseok 4096 Sep 13 23:29 Downloads
12 -rw-r--r--  1 yongseok yongseok 8980 Sep 13 23:21 examples.desktop
4 drwx-----  2 yongseok yongseok 4096 Sep 14 00:54 .gconf
4 drwx-----  3 yongseok yongseok 4096 Sep 14 11:08 .gnupg
4 -rw-----  1 yongseok yongseok  982 Sep 14 14:00 .ICEauthority
4 -rwxr-xr-x  1 yongseok yongseok 2981 Sep 13 23:50 install_ros_kinetic.sh
4 drwx-----  3 yongseok yongseok 4096 Sep 13 23:29 .local
4 drwx-----  5 yongseok yongseok 4096 Sep 13 23:44 .mozilla
4 drwxr-xr-x  2 yongseok yongseok 4096 Sep 13 23:29 Music
4 drwxr-xr-x  2 yongseok yongseok 4096 Sep 13 23:29 Pictures
4 -rw-r--r--  1 yongseok yongseok  655 Sep 13 23:21 .profile
4 drwxr-xr-x  2 yongseok yongseok 4096 Sep 13 23:29 Public
4 drwxrwxr-x  3 yongseok yongseok 4096 Sep 14 00:10 .ros
0 -rw-r--r--  1 yongseok yongseok     0 Sep 13 23:46 .sudo_as_admin_successfu
```

- 모든 디렉토리 자세히 표시

ls -l : 내부에 존재하는 파일과 디렉토리에 대한 상세 정보 표시

ls -al : a옵션은 숨겨진 파일과 디렉토리 정보를 보여줌 + l옵션

## 4. ROS 개발환경 구축

: cat > sample1.txt

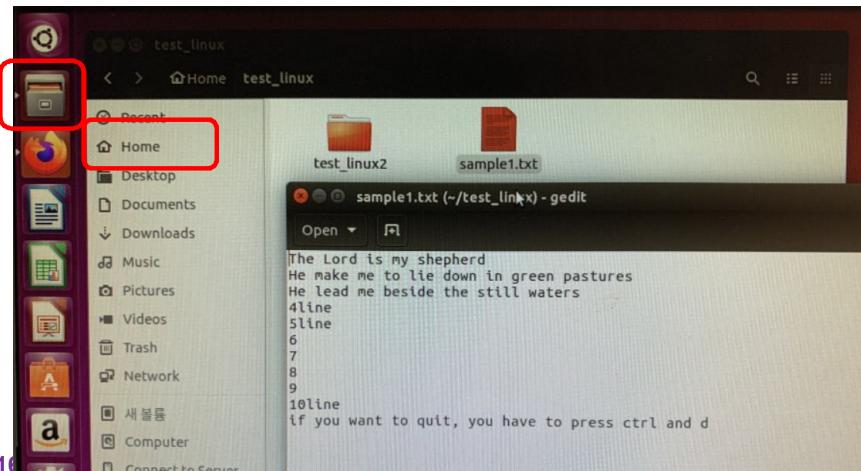
→ cat : concatenate. 여러 개의 텍스트 문자열을 하나의 텍스트 문자열로 연결하는 함수

```
yongseok@yongseok:~$ cd test_linux
yongseok@yongseok:~/test_linux$ cat > sample1.txt
The Lord is my shepherd
He make me to lie down in green pastures
He lead me beside the still waters
4line
5line
6
7
8
9
10line
if you want to quit, you have to press ctrl and d
yongseok@yongseok:~/test_linux$ ls
sample1.txt test_linux2
yongseok@yongseok:~/test_linux$
```

끝내려면, **ctrl + d** 를 누르기

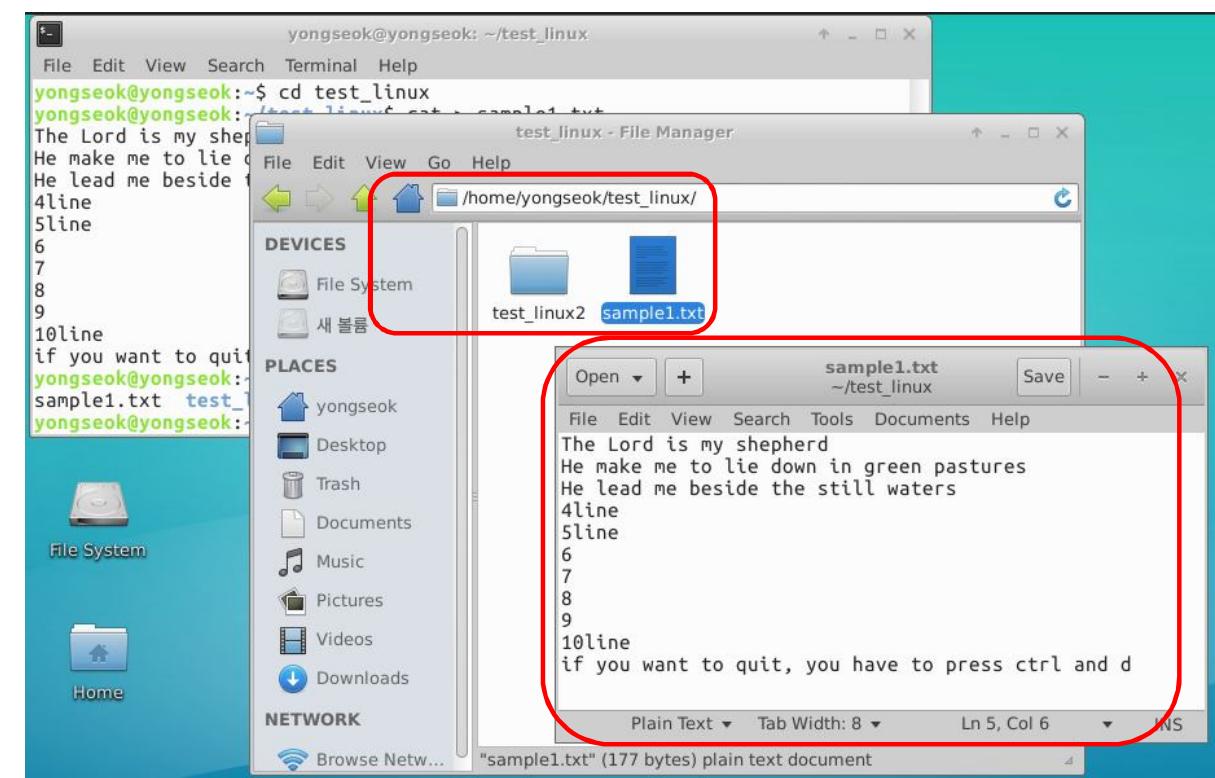


ctrl and d



sample1.txt 파일을 만들어 그 내용을 아래 작성한 내용으로 만듦

• test\_linux 폴더로 이동



→ 원격제어 화면

## 4. ROS 개발환경 구축

: cat sample1.txt

→ terminal 창에서 내용 보기

```
yongseok@yongseok: ~/test_linux
File Edit View Search Terminal Help
yongseok@yongseok:~/test_linux$ cat sample1.txt
The Lord is my shepherd
He make me to lie down in green pastures
He lead me beside the still waters
4line
5line
6
7
8
9
10line
if you want to quit, you have to press ctrl and d
yongseok@yongseok:~/test_linux$
```

내용 보기

: terminal 창을 줄인 후에

: more sample1.txt → terminal 창이 작을 때,

more 명령어를 입력 후 enter마다 내용을 이어서 볼 수 있음

```
yongseok@yongseok: ~/test_linux
File Edit View Search Terminal Help
yongseok@yongseok:~/test_linux$
```

```
yongseok@yongseok: ~/test_linux
File Edit View Search Terminal Help
yongseok@yongseok:~/test_linux$ more sample1.txt
```

```
yongseok@yongseok: ~/tes
File Edit View Search Terminal Help
The Lord is my shepherd
He make me to lie down in green
He lead me beside the still water
4line
5line
6
--More--(64%)
```

enter

enter

```
yongseok@yongseok: ~/test_linux
File Edit View Search Terminal Help
6
7
8
9
10line
if you want to quit, you have to press ctrl and d
yongseok@yongseok:~/test_linux$
```

```
yongseok@yongseok: ~/test_linux
File Edit View Search Terminal Help
He make me to lie down in green pastures
He lead me beside the still waters
4line
5line
6
7
--More--(65%)
```

enter

## 4. ROS 개발환경 구축

: head sample1.txt

→ 파일의 앞부분 10줄을 출력

• 모든 디렉토리 자세히 표시

: tail sample1.txt

→ 파일의 뒷부분 10줄을 출력

```
yongseok@yongseok: ~/test_linux
File Edit View Search Terminal Help
yongseok@yongseok:~/test_linux$ head sample1.txt
The Lord is my shepherd      1 line
He make me to lie down in green 2 pastures
He lead me beside the still waters 3
4line                         4
5line                         5
6                             6
7                             7
8                             8
9                             9
10line                        10
yongseok@yongseok:~/test_linux$
```

```
yongseok@yongseok: ~/test_linux
File Edit View Search Terminal Help
yongseok@yongseok:~/test_linux$ tail sample1.txt
He make me to lie down in green pastures      1 line
He lead me beside the still waters            2
4line                           3
5line                           4
6                             5
7                             6
8                             7
9                             8
10line                          9
if you want to quit, you have to press ctrl and d 10
yongseok@yongseok:~/test_linux$
```

## 4. ROS 개발환경 구축

: wc sample1.txt

→ wc : word count 파일 내용의 저장된 줄 수, 단어 수, 문자 수를 출력

```
yongseok@yongseok: ~/test_linux
File Edit View Search Terminal Help
yongseok@yongseok:~/test_linux$ wc sample1.txt
11 40 177 sample1.txt
yongseok@yongseok:~/test_linux$
```

줄 수  
단어 수  
문자 수  
파일 이름

입력

: sudo rm 파일이름

→ 파일을 삭제할 경우

```
yongseok@yongseok:~$ ls
catkin_ws examples.desktop Public
Desktop install_ros_kinetic.sh Templates
Documents Music test
Downloads Pictures test_linux
yongseok@yongseok:~$ cd test_linux
yongseok@yongseok:~/test_linux$ cat > sample2.txt
yongseok@yongseok:~/test_linux$ ls
sample1.txt sample2.txt test_linux2
yongseok@yongseok:~/test_linux$ sudo rm sample2.txt
yongseok@yongseok:~/test_linux$ ls
sample1.txt test_linux2
yongseok@yongseok:~/test_linux$
```

Sample2.txt의 내용을 yongseok 입력 후 ctl + d로 입력을 마침

생성

입력

Sample2.txt 삭제되어 안보임

## 4. ROS 개발환경 구축

: cat > list1.txt concatenate.

여러 개의 텍스트 문자열을 하나의 텍스트 문자열로 연결하는 함수

cat > list2.txt

```
yongseok@yongseok: ~/test_linux
File Edit View Search Terminal Help
yongseok@yongseok:~/test_linux$ clear
yongseok@yongseok:~/test_linux$ ls
sample1.txt test_linux2
yongseok@yongseok:~/test_linux$ cat > list1.txt
hi my name is yongseok
if you want to quit, press the ctrl + d
yongseok@yongseok:~/test_linux$ ls
list1.txt sample1.txt test_linux2
yongseok@yongseok:~/test_linux$ cat > list2.txt
the eagle does not catch filies
press thr ctrl + d
yongseok@yongseok:~/test_linux$ ls
list1.txt list2.txt sample1.txt test_linux2
yongseok@yongseok:~/test_linux$ ls -sl
total 16
4 -rw-rw-r-- 1 yongseok yongseok 64 Sep 16 11:59 list1.txt
4 -rw-rw-r-- 1 yongseok yongseok 51 Sep 16 12:00 list2.txt
4 -rw-rw-r-- 1 yongseok yongseok 177 Sep 15 17:16 sample1.txt
4 drwxrwxr-x 2 yongseok yongseok 4096 Sep 15 16:56 test_linux2
yongseok@yongseok:~/test_linux$
```

파일 크기

: cat > list1.txt list2.txt > list3.txt

두개의 파일을 하나로 만들어 새로운 파일 만들기

```
yongseok@yongseok: ~/test_linux
File Edit View Search Terminal Help
yongseok@yongseok:~/test_linux$ cat list1.txt list2.txt > list3.txt
yongseok@yongseok:~/test_linux$ ls
list1.txt list2.txt list3.txt sample1.txt test_linux2
yongseok@yongseok:~/test_linux$ ls -sl
total 20
4 -rw-rw-r-- 1 yongseok yongseok 39 Sep 16 12:12 list1.txt
4 -rw-rw-r-- 1 yongseok yongseok 51 Sep 16 12:00 list2.txt
4 -rw-rw-r-- 1 yongseok yongseok 90 Sep 16 12:12 list3.txt
4 -rw-rw-r-- 1 yongseok yongseok 177 Sep 15 17:16 sample1.txt
4 drwxrwxr-x 2 yongseok yongseok 4096 Sep 15 16:56 test_linux2
yongseok@yongseok:~/test_linux$ cat list3.txt
my name is yongseok
press the ctrl + d
the eagle does not catch filies
press thr ctrl + d
yongseok@yongseok:~/test_linux$ cat list1.txt
my name is yongseok
press the ctrl + d
yongseok@yongseok:~/test_linux$ cat list2.txt
the eagle does not catch filies
press thr ctrl + d
yongseok@yongseok:~/test_linux$
```

생성

39+51 = 90 크기

## 4. ROS 개발환경 구축

: cat >> list1.txt

→ list1.txt에 내용 추가 하기

```
yongseok@yongseok:~/test_linux$ cat list1.txt  
my name is yongseok  
press the ctrl + d  
yongseok@yongseok:~/test_linux$ cat >> list1.txt  
add the contents  
press thr ctrl and d  
yongseok@yongseok:~/test_linux$ cat list1.txt  
my name is yongseok  
press the ctrl + d  
add the contents  
press thr ctrl and d  
yongseok@yongseok:~/test_linux$
```

내용 추가됨

: cp sample1.txt sample2.txt

→ sample1.txt의 내용을 sample2.txt 만들어 복사하기

```
yongseok@yongseok:~/test_linux$ ls  
list1.txt list2.txt list3.txt sample1.txt test_linux2  
yongseok@yongseok:~/test_linux$ cat sample1.txt  
The Lord is my shepherd  
He make me to lie down in green pastures  
He lead me beside the still waters  
4line  
5line  
6  
7  
8  
9  
10line  
if you want to quit, you have to press ctrl and d  
yongseok@yongseok:~/test_linux$ cp sample1.txt sample2.txt 입력  
yongseok@yongseok:~/test_linux$ ls  
list1.txt list2.txt list3.txt sample1.txt sample2.txt test_linux2  
yongseok@yongseok:~/test_linux$ cat sample2.txt  
The Lord is my shepherd  
He make me to lie down in green pastures  
He lead me beside the still waters  
4line  
5line  
6  
7  
8  
9  
10line  
if you want to quit, you have to press ctrl and d  
yongseok@yongseok:~/test_linux$
```

Sample1.txt

내용 확인

입력

생성 확인

Sample2.txt

내용 확인

## 4. ROS 개발환경 구축

: mkdir 폴더 이름

→ 폴더 만들기

: rmdir 폴더 이름

→ 폴더 삭제하기

: sudo rm 파일이름

→ 파일을 삭제할 경우

: mv sample2.txt sample3.txt

→ sample2 이름의 파일을 sample3로 변경

```
yongseok@yongseok:~/test_linux$ ls
list1.txt list2.txt list3.txt sample1.txt sample2.txt test_linux2
yongseok@yongseok:~/test_linux$ mkdir making1
yongseok@yongseok:~/test_linux$ ls
list1.txt list3.txt sample1.txt test_linux2
list2.txt making1 sample2.txt
yongseok@yongseok:~/test_linux$ rmdir making1
yongseok@yongseok:~/test_linux$ ls
list1.txt list2.txt list3.txt sample1.txt sample2.txt test_linux2
yongseok@yongseok:~/test_linux$ sudo rm list3.txt
[sudo] password for yongseok:
yongseok@yongseok:~/test_linux$ ls
list1.txt list2.txt sample1.txt sample2.txt test_linux2
yongseok@yongseok:~/test_linux$ mv sample2.txt sample3.txt
yongseok@yongseok:~/test_linux$ ls
list1.txt list2.txt sample1.txt sample3.txt test_linux2
yongseok@yongseok:~/test_linux$ █
```

## 4. ROS 개발환경 구축

사용권한(permission mode) : 읽기(r), 쓰기(w), 실행(x) 권한

권한	파일	디렉터리
r	파일에 대한 읽기 권한	디렉터리 내에 있는 파일명을 읽을 수 있는 권한
w	파일에 대한 쓰기 권한	디렉터리 내에 파일을 생성하거나 삭제할 수 있는 권한
x	파일에 대한 실행 권한	디렉터리 내로 탐색을 위해 이동할 수 있는 권한

사용 권한은

소유자(owner)/그룹(group)/기타(others)로 구분하여 관리.

d	r	w	x	r	-	x	r	-	x
A: 파일유형	B: 소유자 권한			C: 그룹 소유자 권한			D: 일반 사용자 권한		

d 디렉토리

ls -l : 내부에 존재하는 파일과 디렉토리에 대한 상세 정보 표시

ls -al : a옵션은 숨겨진 파일과 디렉토리 정보를 보여줌 + l옵션

```
yongseok@yongseok:~/test_linux$ ls -al
total 32
drwxrwxr-x  3 yongseok yongseok 4096 Sep 16 14:57 .
drwxr-xr-x 23 yongseok yongseok 4096 Sep 15 16:56 ..
-rw-rw-r--  1 yongseok yongseok    8 Sep 16 12:21 list1.txt
-rw-rw-r--  1 yongseok yongseok   51 Sep 16 12:00 list2.txt
-rw-rw-r--  1 yongseok yongseok   90 Sep 16 12:12 list3.txt
-rw-rw-r--  1 yongseok yongseok  177 Sep 15 17:16 sample1.txt
-rw-rw-r--  1 yongseok yongseok  177 Sep 16 14:57 sample2.txt
drwxrwxr-x  2 yongseok yongseok 4096 Sep 15 16:56 test_linux2
yongseok@yongseok:~/test_linux$
```

owner	group	others	
rw-	rw-	r--	list1.txt
rw-	rw-	r--	list2.txt
rw-	rw-	r--	list3.txt
rw-	rw-	r--	sample1.txt
rw-	rw-	r--	sample2.txt
6	6	4	

## 4. ROS 개발환경 구축

ls -l : 내부에 존재하는 파일과 디렉토리에 대한 상세 정보 표시

ls -al : a옵션은 숨겨진 파일과 디렉토리 정보를 보여줌 + l옵션

```
yongseok@yongseok:~/test_linux$ ls -sl
total 20
4 -rw-rw-r-- 1 yongseok yongseok 78 Sep 16 12:21 list1.txt
4 -rw-rw-r-- 1 yongseok yongseok 51 Sep 16 12:00 list2.txt
4 -rw-rw-r-- 1 yongseok yongseok 177 Sep 15 17:16 sample1.txt
4 -rw-rw-r-- 1 yongseok yongseok 177 Sep 16 14:57 sample3.txt
4 drwxrwxr-x 2 yongseok yongseok 4096 Sep 15 16:56 test_linux2
yongseok@yongseok:~/test_linux$ ls -al
total 28
drwxrwxr-x 3 yongseok yongseok 4096 Sep 16 16:39 .
drwxr-xr-x 23 yongseok yongseok 4096 Sep 15 16:56 ..
-rw-rw-r-- 1 yongseok yongseok 78 Sep 16 12:21 list1.txt
-rw-rw-r-- 1 yongseok yongseok 51 Sep 16 12:00 list2.txt
-rw-rw-r-- 1 yongseok yongseok 177 Sep 15 17:16 sample1.txt
-rw-rw-r-- 1 yongseok yongseok 177 Sep 16 14:57 sample3.txt
drwxrwxr-x 2 yongseok yongseok 4096 Sep 15 16:56 test_linux2
yongseok@yongseok:~/test_linux$ ls -asl
total 28
4 drwxrwxr-x 3 yongseok yongseok 4096 Sep 16 16:39 .
4 drwxr-xr-x 23 yongseok yongseok 4096 Sep 15 16:56 ..
4 -rw-rw-r-- 1 yongseok yongseok 78 Sep 16 12:21 list1.txt
4 -rw-rw-r-- 1 yongseok yongseok 51 Sep 16 12:00 list2.txt
4 -rw-rw-r-- 1 yongseok yongseok 177 Sep 15 17:16 sample1.txt
4 -rw-rw-r-- 1 yongseok yongseok 177 Sep 16 14:57 sample3.txt
4 drwxrwxr-x 2 yongseok yongseok 4096 Sep 15 16:56 test_linux2
yongseok@yongseok:~/test_linux$ █
```

## 4. ROS 개발환경 구축

: chmod 8bit값 파일 이름



→ chmod : change mode. 파일의 속성 변경

사용권한 rw- rw- r--  
2진수: 110 110 100  
8진수: 6 6 4  
\$ chmod 664 sample1.txt

```
yongseok@yongseok:~/test_linux$ ls -asl sample1.txt
4 -rw-rw-r-- 1 yongseok yongseok 177 Sep 15 17:16 sample1.txt
yongseok@yongseok:~/test_linux$ chmod 777 sample1.txt
yongseok@yongseok:~/test_linux$ ls -asl sample1.txt
4 -rwxrwxrwx 1 yongseok yongseok 177 Sep 15 17:16 sample1.txt
yongseok@yongseok:~/test_linux$
```

사용권한 rwx rwx rwx  
2진수: 111 111 111  
8진수: 7 7 7  
\$ chmod 777 sample1.txt

: 내가 속한 폴더가 아닌 경우, sudo 명령어와 함께 사용

→ sudo chmod

## 4. ROS 개발환경 구축

df : disk driver에 있는 사용 가능한 공간을 표시(disk free) Kbyte 단위 df -k Mbyte 단위 df -m

free : 메모리의 양. 시스템의 현재 남은 공간을 표시

```
yongseok@yongseok:~$ df  
Filesystem 1K-blocks Used Available Use% Mounted on  
udev 1928424 0 1928424 0% /dev  
tmpfs 391840 11256 380584 3% /run  
/dev/sda2 121438316 9065028 106181468 8% /  
tmpfs 1959188 612 1958576 1% /dev/shm  
tmpfs 5120 4 5116 1% /run/lock  
tmpfs 1959188 0 1959188 0% /sys/fs/cgroup  
/dev/sda1 523248 5212 518036 1% /boot/efi  
tmpfs 391840 116 391724 1% /run/user/1000
```

```
yongseok@yongseok:~$ free  
total used free shared buff/cache available  
Mem: 3918376 1498028 425788 230524 1994560 1863840  
Swap: 1000444 0 1000444
```

```
yongseok@yongseok:~$ df -k  
Filesystem 1K-blocks Used Available Use% Mounted on  
udev 1928424 0 1928424 0% /dev  
tmpfs 391840 11256 380584 3% /run  
/dev/sda2 121438316 9065040 106181456 8% /  
tmpfs 1959188 612 1958576 1% /dev/shm  
tmpfs 5120 4 5116 1% /run/lock  
tmpfs 1959188 0 1959188 0% /sys/fs/cgroup  
/dev/sda1 523248 5212 518036 1% /boot/efi  
tmpfs 391840 116 391724 1% /run/user/1000
```

```
yongseok@yongseok:~$ df -m  
Filesystem 1M-blocks Used Available Use% Mounted on  
udev 1884 0 1884 0% /dev  
tmpfs 383 11 372 3% /run  
/dev/sda2 118593 8853 103693 8% /  
tmpfs 1914 1 1913 1% /dev/shm  
tmpfs 5 1 5 1% /run/lock  
tmpfs 1914 0 1914 0% /sys/fs/cgroup  
/dev/sda1 511 6 506 1% /boot/efi  
tmpfs 383 1 383 1% /run/user/1000
```

```
yongseok@yongseok:~$
```

## 4. ROS 개발환경 구축

: Linux

명령어

<code>ls</code>	디렉터리 내용의 목록을 표시한다.
<code>clear</code>	화면을 지운다.
<code>echo</code>	인수로 지정한 문자열을 출력한다.
<code>cp</code>	파일을 복사한다.
<code>rm</code>	파일을 삭제한다.
<code>mv</code>	파일을 이동하거나 이름을 바꾼다.
<code>ln</code>	다른 파일을 가리키는 연결을 생성한다.
<code>cd</code>	현재 디렉터리를 이동한다.
<code>mkdir</code>	디렉터리를 생성한다.
<code>rmdir</code>	디렉터리를 삭제한다.
<code>pwd</code>	현재 작업 중인 디렉터리의 경로를 표시한다.
<code>date</code>	시스템에 설정된 현재의 시간과 날짜를 출력한다.
<code>dirs</code>	디렉터리 스택에 저장된 내용을 표시한다.
<code>pushd</code>	디렉터리 스택에 (현재의) 디렉터리를 저장한다.
<code>popd</code>	디렉터리 스택에서 마지막에 저장한 디렉터리를 삭제하고 그 디렉터리로 이동한다.
<code>cat</code>	파일의 내용을 표시한다.
<code>more</code>	파일의 내용을 페이지 단위로 표시한다.
<code>wc</code>	문서 내에서의 줄(단어, 글자)의 수를 알려준다.
<code>man</code>	유ти리티나 API에 대한 도움말 매뉴얼을 보여준다.
<code>apropos</code>	해당 주제와 관련된 명령어들을 표시한다.
<code>which</code>	명령어의 위치하고 있는 경로를 표시한다.

<code>find</code>	시스템에서 파일이나 명령어를 찾는다.
<code>grep</code>	파일 내의 내용을 패턴을 이용해서 검색한다.
<code>adduser</code>	시스템에 사용자를 추가한다.
<code>sudo</code>	다른 사용자의 보안 권한으로 명령어를 수행한다.
<code>su</code>	root나 다른 사용자로 변경한다.
<code>who</code>	사용자에 대한 정보를 표시한다.
<code>wall</code>	모든 사용자에게 메시지를 표시한다.
<code>logout</code>	현재의 시스템에서 계정을 로그아웃한다.
<code>dmesg</code>	커널의 부팅 로그 및 커널 로그를 출력한다.
<code>chmod</code>	파일의 접근 권한을 변경한다.
<code>chown</code>	파일의 소유자를 변경한다.
<code>chgrp</code>	파일과 관련된 그룹을 변경한다.
<code>passwd</code>	사용자의 비밀번호를 변경한다.
<code>du</code>	디스크 사용량을 검사한다.
<code>df</code>	디스크 남은 공간을 표시한다.
<code>free</code>	시스템의 현재 남은 공간을 표시한다.
<code>mount</code>	디바이스를 시스템에 연결(Mount)한다.
<code>ps</code>	프로세스의 상태를 표시한다.
<code>kill</code>	프로세스에 시그널을 보내며, 주로 프로세스를 죽이는 데 사용한다.
<code>top</code>	시스템에서의 현재 프로세스의 상태를 표시한다.
<code>fg</code>	프로세스를 포그라운드 모드로 실행한다.

## 4. ROS 개발환경 구축

: Linux

명령어

<b>bg</b>	프로세스를 백그라운드 모드로 실행한다.
<b>sync</b>	현재 캐시된 내용을 저장한다.
<b>cal</b>	달력을 보여준다.
<b>tar</b>	여러 개의 파일을 하나의 파일로 묶는다.
<b>compress</b>	파일을 압축하거나 파일의 압축을 해제한다.
<b>halt</b>	시스템을 종료한다.
<b>reboot</b>	시스템을 재시작한다.
<b>poweroff</b>	시스템 전원을 내린다.
<b>startx</b>	x 윈도우를 시작한다.
<b>telnet</b>	원격 서버 터미널로 접속한다.
<b>ftp</b>	ftp 서버에 접속한다.
<b>apt-get</b>	우분투(Ubuntu)를 포함한 데비안(Debian) 계열에서 소프트웨어 패키지를 설치 및 관리한다.
<b>apt-cache</b>	우분투(Ubuntu)를 포함한 데비안(Debian) 계열에서 소프트웨어 패키지를 검색하여 표시한다.

## 4. ROS 개발환경 구축

### 4-2-3. Install ROS on Remote PC 에서 (즉 우분투 환경에서 작업하기, Remote PC에서 작업하기 )

#### (1) Install ROS Kinetic 삭제

: Remote PC 성능에 따라서 ROS Kinetic 설치가 안정화 되지 않는 현상이 있음

→ ROS Kinetic 설치가 안되면 down받은 ROS등을 제거하고 재설치 해야 함

: ROS Kinetic 삭제하는 방법

→ 기존에 설치된 ros 삭제

\$ sudo apt-get remove ros-\* (패키지만 삭제, 설정 파일은 남겨둠)

\$ sudo apt-get purge ros-\* (패키지와 설정파일 함께 삭제)



\$ sudo apt autoremove

→ 삭제 확인

\$ rosversion -d (여기에서 <unknown>이라고 뜨면 삭제가 된 것임)

(만약 기존 ros version이 확인되면, 재부팅 후 다시 확인해 보면, 삭제된 것을 확인할 수 있음)

## 4. ROS 개발환경 구축

Ref. ROS 로봇 프로그래밍(Ruby paper) 3-1 ROS 설치

### 4-2-3. Install ROS on Remote PC에서 (즉 우분투 환경에서 작업하기, Remote PC에서 작업하기 )

#### (2) Install ROS Kinetic

: 방법 1(수동설치방법, 방법2 자동설치. 방법2로 하세요) 교재 26page ROS 로봇 프로그래밍(Ruby paper) 3-1 ROS 설치

ⓐ 다른 PC간 통신(server, client)에서 ROS Time 오차를 줄일 수 있도록 NTP(Network Time Protocol)를 설정(시간 동기화)

- 설정 : **chrony** (Linux time synchronizing)를 설치한 후에 **ntpdate** 명령어로 NTP서버를 지정
- **CTRL + ALT + T** 를 사용해 터미널 창 열기

```
$ sudo apt-get install -y chrony ntpdate
```

```
$ sudo ntpdate -q ntp.ubuntu.com
```

The terminal window shows the following command being run:

```
yongseok@yongseok:~$ sudo apt-get install -y chrony ntpdate
```

Followed by the output of the command:

```
[sudo] password for yongseok:  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
chrony is already the newest version (2.1.1-1ubuntu0.1).  
ntpdate is already the newest version (1:4.2.8p4+dfsg-3ubuntu5.10).  
0 upgraded, 0 newly installed, 0 to remove and 9 not upgraded.
```

Then, the command:

```
yongseok@yongseok:~$ sudo ntpdate -q ntp.ubuntu.com
```

Followed by the output:

```
server 91.189.89.199, stratum 2, offset 0.000245, delay 0.28320  
server 91.189.94.4, stratum 2, offset 0.003806, delay 0.28764  
server 91.189.91.157, stratum 2, offset -0.008543, delay 0.22823  
server 91.189.89.198, stratum 2, offset -0.017860, delay 0.29843  
26 Sep 16:15:52 ntpdate[5117]: adjust time server 91.189.91.157 offset -0.008543  
sec
```

Finally, the command:

```
yongseok@yongseok:~$
```

Annotations on the right side of the terminal window:

- y library 설치 시 yes
- q : ntpdate 설정 옵션
- Time synchronizing

#### 1. 제일먼저

무선 연결 반드시 확인

(실제 YOUTUBE 를 실행하여 연결되는지 확인)

## 4. ROS 개발환경 구축

Ref. ROS 로봇 프로그래밍(Ruby paper) 3-1 ROS 설치

### ⓑ ros-latest.list에 ROS 저장 주소 추가

→ \$ sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu \$(lsb\_release -sc) main" > /etc/apt/sources.list.d/ros-latest.list'

: \$(lsb\_release -sc)는 리눅스 배포 판의 정보 중 코드 네임을 가져오는 것

```
sec
yongseok@yongseok:~$ sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu $(lsb_release -sc) main" > /etc/apt/sources.list.d/ros-latest.list'
sh: 1: cannot create /etc/apt/: Is a directory
sh: 2: sources.list.d/ros-latest.list: not found
yongseok@yongseok:~$
```

### ⓒ ROS 저장소로부터 패키지를 받기 위해 공개 키 추가 (아래 old key) ⭐

→ \$ sudo apt-key adv --keyserver hkp://ha.pool.sks-keyservers.net:80 --recv-key

421C365BD9FF1F717815A3895523BAEEB01FA116

```
yongseok@yongseok:~$ sudo apt-key adv --keyserver hkp://ha.pool.sks-keyservers.net:80 --recv-key 421C365BD9FF1F717815A3895523BAEEB01FA116
Executing: /tmp/tmp.oK6zdGGNCd/gpg.i.sh --keyserver hkp://ha.pool.sks-keyservers.net:80 --recv-key 421C365BD9FF1F717815A3895523BAEEB01FA116
usage: gpg [options] [filename]
yongseok@yongseok:~$
```

• 한 칸 띄기

## 4. ROS 개발환경 구축

Ref. ROS 로봇 프로그래밍(Ruby paper) 3-1 ROS 설치

### ④ 현재 설치된 Ubuntu package index 업데이트

→ \$ sudo apt-get update && sudo apt-get upgrade -y

```
yongseok@yongseok:~$ sudo apt-get update && sudo apt-get upgrade -y
[youtube] http://kr.archive.ubuntu.com/ubuntu xenial InRelease
[youtube] http://packages.ros.org/ubuntu xenial InRelease
[mysql] http://kr.archive.ubuntu.com/ubuntu xenial-updates InRelease [109 kB]
[mysql] http://security.ubuntu.com/ubuntu xenial-security InRelease [109 kB]
[youtube] http://archive.ubuntu.com/ubuntu xenial-backports InRelease [107 kB]
```

-Y : YES

-설치 내용 생략

```
Unpacking libgnutls30:amd64 (3.4.10-4ubuntu1.9) over (3.4.10-4ubuntu1.8) ...
Processing triggers for libc-bin (2.23-9ubuntu1.3) ...
Setting up libgnutls30:amd64 (3.4.10-4ubuntu1.9) ...
Setting up libgnutls-openssl27:amd64 (3.4.10-4ubuntu1.9) ...
Processing triggers for libc-bin (2.23-9ubuntu1.3) ...
yongseok@yongseok:~$ █
```

## 4. ROS 개발환경 구축

Ref. ROS 로봇 프로그래밍(Ruby paper) 3-1 ROS 설치

④ ROS Kinetic Kame 설치

: ROS 패키지(ROS, rqt, RViz, 로봇 관련 라이브러리, 시뮬레이션, 내비게이션 등)

→ \$ sudo apt-get install ros-kinetic-desktop-full

```
yongseok@yongseok:~$ sudo apt-get install ros-kinetic-desktop-full
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed: 시간 소요
  cmake cmake-data fltk1.3-doc fluid fonts-lato freeglut3 freeglut3-dev
  gazebo7 gazebo7-common gazebo7-plugins
```

#### -설치 내용 생략

시간 소요

: rqt 관련 모든 패키지를 설치(rqt 플러그인 사용 위함)

→ \$ sudo apt-get install ros-kinetic-rqt\*

```
yongseok@yongseok:~$ sudo apt-get install ros-kinetic-rqt*
Reading package lists... Done
Building dependency tree
Reading state information... Done
Note, selecting 'ros-kinetic-rqt-drone-teleop' for glob 'ros-kinetic-rqt*'
Note, selecting 'ros-kinetic-rqt-gui-py' for glob 'ros-kinetic-rqt*'
Note, selecting 'ros-kinetic-rqt-tf-tree' for glob 'ros-kinetic-rqt*'
```

## -설치 내용 생략

## 4. ROS 개발환경 구축

Ref. ROS 로봇 프로그래밍(Ruby paper) 3-1 ROS 설치

### ① rosdep 초기화

: rosdep은 ros의 핵심 컴포넌트들을 사용하거나

컴파일할 때 의존성 패키지를 설치하기 위해(초기화하고, 업데이트를 진행) 사용자 편의성을 높여주는 기능

→ \$ sudo rosdep init  
\$ rosdep update

```
Processing triggers for libc-bin (2.23-0ubuntu11.3) ...
yongseok@yongseok:~$ sudo rosdep init 초기화
ERROR: default sources list file already exists:
/etc/ros/rosdep/sources.list.d/20-default.list
Please delete if you wish to re-initialize
yongseok@yongseok:~$ rosdep update 업데이트
reading in sources list data from /etc/ros/rosdep/sources.list.d
Hit https://raw.githubusercontent.com/ros/rosdistro/master/rosdep/osx-homebrew.yaml
Hit https://raw.githubusercontent.com/ros/rosdistro/master/rosdep/base.yaml
Hit https://raw.githubusercontent.com/ros/rosdistro/master/rosdep/python.yaml
Hit https://raw.githubusercontent.com/ros/rosdistro/master/rosdep/ruby.yaml
```

-설치 내용 생략

### ② rosinstall 설치

: ROS의 다양한 패키지를 인스톨 프로그램

→ \$ sudo apt-get install python-rosinstall

```
updated cache in /home/yongseok/.ros/rosdep/sources.list.d
yongseok@yongseok:~$ sudo apt-get install python-rosinstall
Reading package lists... Done
Building dependency tree
Reading state information... Done
python-rosinstall is already the newest version (0.7.8-1).
0 upgraded, 0 newly installed, 0 to remove and 7 not upgrad
```

## 4. ROS 개발환경 구축

방법 2에서도 동일

Ref. ROS 로봇 프로그래밍(Ruby paper) 3-1 ROS 설치

### ④ 환경설정 파일 불러오기

: ROS\_ROOT, ROS\_PACKAGE\_PATH 등의 환경변수 정의된 환경설정 파일 불러오기

→ \$ source /opt/ros/kinetic/setup.bash ★

```
yongseok@yongseok:~$ source /opt/ros/kinetic/setup.bash
```

bash : 파일을 읽고 실행 명령어

### ⑤ 작업 폴더 생성 및 초기화

: ROS에서는 catkin이라는 ROS 전용 빌드 시스템 사용

: catkin 사용하려면 catkin 작업 폴더를 생성하고 초기화해야 함

→ \$ mkdir -p ~/catkin\_ws/src

\$ cd ~/catkin\_ws/src

```
yongseok@yongseok:~$ mkdir -p ~/catkin_ws/src
```

```
yongseok@yongseok:~$ cd ~/catkin_ws/src
```

```
yongseok@yongseok:~/catkin_ws/src$
```

: mkdir -p : -parent 필요한 경우 상위경로까지 생성

\$ catkin\_init\_workspace ★

● CMakeLists.txt 파일 생성

-directory 확인

```
yongseok@yongseok:~/catkin_ws/src$ catkin_init_workspace
```

```
File "/home/yongseok/catkin_ws/src/CMakeLists.txt" already existsyongseok@yongseok:~/catkin_ws/src$
```

-CMakeLists.txt 파일 확인

## 4. ROS 개발환경 구축

Ref. ROS 로봇 프로그래밍(Ruby paper) 3-1 ROS 설치

### ① build

: 경로 확인 (build 경로)

→ \$ cd ~/catkin\_ws/

\$ catkin\_make



\$ ls

```
[100%] Built target turtlebot3_example_generate_messages
yongseok@yongseok:~/catkin_ws$ ls
build  devel  install_ros_kinetic.sh  src
yongseok@yongseok:~/catkin_ws$
```



우리가 만든 src 폴더 확인

그리고 build 와 devel 폴더 생성

: catkin 빌드 파일은 build 폴더에 저장

: 빌드 후 실행 파일은 devel 폴더에 저장

```
yongseok@yongseok:~/catkin_ws/src$ catkin_init_workspace
File "/home/yongseok/catkin_ws/src/CMakeLists.txt" already existsyongseok@yongseok:~/catkin_ws$ cd ~ /catkin_ws/
yongseok@yongseok:~/catkin_ws$ catkin_make
Base path: /home/yongseok/catkin_ws
Source space: /home/yongseok/catkin_ws/src
Build space: /home/yongseok/catkin_ws/build
Devel space: /home/yongseok/catkin_ws/devel
Install space: /home/yongseok/catkin_ws/install
#####
##### Running command: "make cmake_check_build_system" in "/home/yongseok/catkin_ws/build"
d: "make -j2 -l2" in "/home/yongseok/catkin_ws/build"
```

만일 이 내용이 없다면 catkin\_make가 불안전  
(source code와 환경설정 build 안 된 것임)  
→ ubuntu 재설치  
(roscore는 설치되었어도 문제가 내재)

```
0%] Built target std_msgs_generate_messages_py
0%] Built target _turtlebot3_msgs_generate_messages_check_deps_SensorState
0%] Built target _turtlebot3_msgs_generate_messages_check_deps_Sound
0%] Built target std_msgs_generate_messages_cpp
0%] Built target turtlebot3_msgs_generate_messages_check_deps_VersionInfo
```

-실행 내용 생략

```
[ 84%] Built target turtlebot3_example_generate_messages_nodejs
[ 96%] Built target turtlebot3_example_generate_messages_lisp
[ 96%] Built target turtlebot3_msgs_generate_messages
[100%] Built target turtlebot3_diagnostics
[100%] Built target turtlebot3_example_generate_messages 100% build 확인
yongseok@yongseok:~/catkin_ws$
```



## 4. ROS 개발환경 구축

Ref. ROS 로봇 프로그래밍(Ruby paper) 3-1 ROS 설치

### ⑤ catkin 빌드 시스템과 관련된 환경 파일 적용하기

→ \$ source ~/catkin\_ws/devel/setup.bash ★

```
yongseok@yongseok:~/catkin_ws$ ls  
build  devel  install_ros_kinetic.sh  src  
yongseok@yongseok:~/catkin_ws$ source ~/catkin_ws/devel/setup.bash  
yongseok@yongseok:~/catkin_ws$ █
```

테스트를 위해

CTRL + Shift +Q 열려있는 Terminal 창 닫기

다시 새로운 창 열기

CTRL + ALT + T 를 사용해 터미널 창 열기

build과정으로 생성된 devel파일 안의 setup.bash를 실행하여  
그 작업공간이 ros 환경의 최상위에 overlay되도록 쉘에 등록  
→ setup.bash를 실행하면 environment variables가 setting되고  
이것이 시스템이 필요한 모든 package와 library 등을 찾게 도와줌

## 4. ROS 개발환경 구축

Ref. ROS 로봇 프로그래밍(Ruby paper) 3-1 ROS 설치

### ★ 최종 테스트

→ \$ roscore

: ROS master 구동하는 명령어



오류가 발생한다면



ROS 설치에 문제 있음

재설치 필요



문제가 지속 발생 경우

UBUNTU 재설치

```
yongseok@yongseok:~$ roscore
... logging to /home/yongseok/.ros/log/83cce24c-1e9c-11ec-b5be-f40669f0af37/roslaunch-yongseok-21558.log
Checking log directory for disk usage. This may take awhile.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://localhost:36691/
ros_comm version 1.12.17

SUMMARY
=====

PARAMETERS
* /rosdistro: kinetic
* /rosversion: 1.12.17 → Version Kinetic 확인

NODES

auto-starting new master
process[master]: started with pid [21569]
ROS_MASTER_URI=http://localhost:11311/

setting /run_id to 83cce24c-1e9c-11ec-b5be-f40669f0af37
process[rosout-1]: started with pid [21582]
started core service [/rosout] → 이 상태에서 끝내려면 Ctrl+c ★
```

## 4. ROS 개발환경 구축

방법 2 참고

Ref. <https://emanual.robotis.com/docs/en/platform/turtlebot3/quick-start/>

### 4-2-3. Install ROS on Remote PC에서 (즉 우분투 환경에서 작업하기, Remote PC에서 작업하기 )

#### (3) Install ROS Kinetic

: 방법 2(자동설치) <https://emanual.robotis.com/docs/en/platform/turtlebot3/quick-start> 1.1.4까지 진행

→ Firefox web browser에서 위 주소 열기

drag 하여 터미널 창  
에 붙여 넣기 실행

drag 하여 실행하기

drag 하고  
TXT 파일을 만들어 붙여 넣고  
다시 COPY후 터미널 창에 붙이기

The screenshot shows the 'TurtleBot3' page on the Robotis E-Manual website. A red arrow points from the text 'drag 하여 터미널 창에 붙여 넣기 실행' to the 'install\_ros\_kinetic.sh' command in the terminal window. Another red arrow points from the text 'drag 하여 실행하기' to the 'install\_ubuntu.sh' command. A third red arrow points from the text 'drag 하고 TXT 파일을 만들어 붙여 넣고 다시 COPY후 터미널 창에 붙이기' to the 'install\_turtlebot3.sh' command. The terminal window contains the following commands:

```
$ sudo apt-get update
$ sudo apt-get upgrade
$ wget https://raw.githubusercontent.com/ROBOTIS-GIT/robotis_tools/master/install_ros_kinetic.sh
$ chmod 755 ./install_ros_kinetic.sh
$ bash ./install_ros_kinetic.sh
```

If the above installation fails, please refer to the official ROS Kinetic installation guide.

**3. 1. 2. Install ROS on Remote PC**

Open the terminal with **Ctrl**+**Alt**+**T** and enter below commands one at a time. In order to check the details of the easy installation script, please refer to the [script file](#).

```
$ sudo apt-get update
$ sudo apt-get upgrade
$ wget https://raw.githubusercontent.com/ROBOTIS-GIT/robotis_tools/master/install_ros_kinetic.sh
$ chmod 755 ./install_ros_kinetic.sh
$ bash ./install_ros_kinetic.sh
```

**3. 1. 3. Install Dependent ROS Packages**

```
$ sudo apt-get install ros-kinetic-joy ros-kinetic-teleop-twist-joy
$ sudo apt-get install ros-kinetic-teleop-twist-keyboard ros-kinetic-laser-proc
$ sudo apt-get install ros-kinetic-rgbd-launch ros-kinetic-depthimage-to-laserscan
$ sudo apt-get install ros-kinetic-rosserial-arduino ros-kinetic-rosserial-python
$ sudo apt-get install ros-kinetic-rosserial-server ros-kinetic-rosserial-client
$ sudo apt-get install ros-kinetic-rosserial-msgs ros-kinetic-amcl ros-kinetic-map-server
$ sudo apt-get install ros-kinetic-move-base ros-kinetic-urdf ros-kinetic-xacro
$ sudo apt-get install ros-kinetic-compressed-image-transport ros-kinetic-rqt*
$ sudo apt-get install ros-kinetic-gmapping ros-kinetic-navigation ros-kinetic-interactive-markers
```

**3. 1. 4. Install TurtleBot3 Packages**

Install TurtleBot3 via Debian Packages.

```
$ sudo apt-get install ros-kinetic-dynamixel-sdk
$ sudo apt-get install ros-kinetic-turtlebot3-msgs
$ sudo apt-get install ros-kinetic-turtlebot3
```

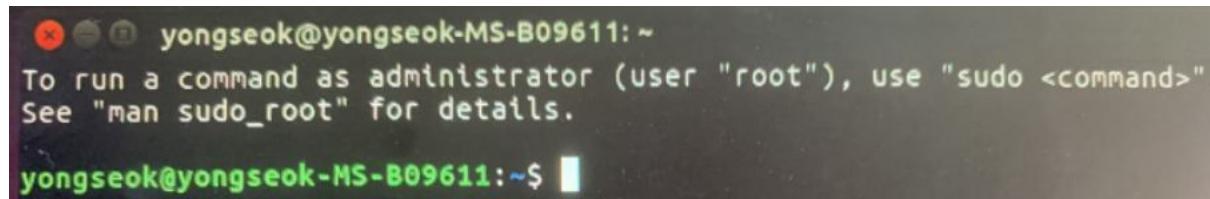
## 4. ROS 개발환경 구축

방법 2 참고

Ref. <https://emanual.robotis.com/docs/en/platform/turtlebot3/quick-start/>

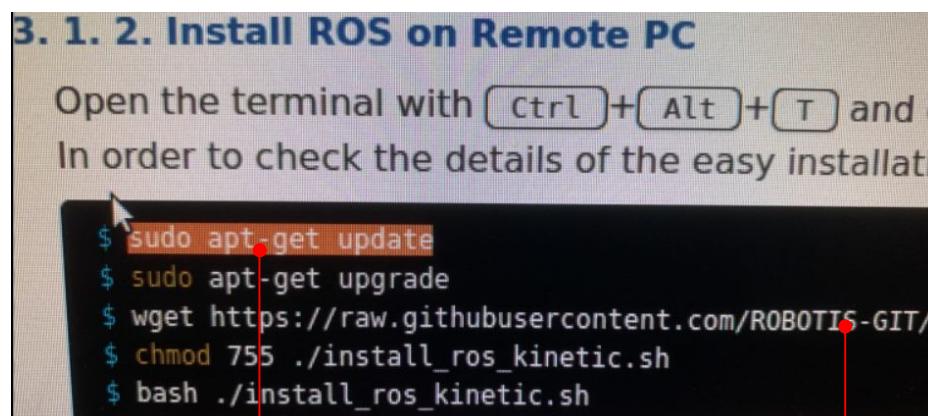
방법 2 참고: Install ROS Kinetic

: Open the terminal with **CTRL + ALT + T**



```
yongseok@yongseok-MS-B09611:~  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
yongseok@yongseok-MS-B09611:~$
```

: <https://emanual.robotis.com/docs/en/platform/turtlebot3/quick-start>에서



3. 1. 2. Install ROS on Remote PC

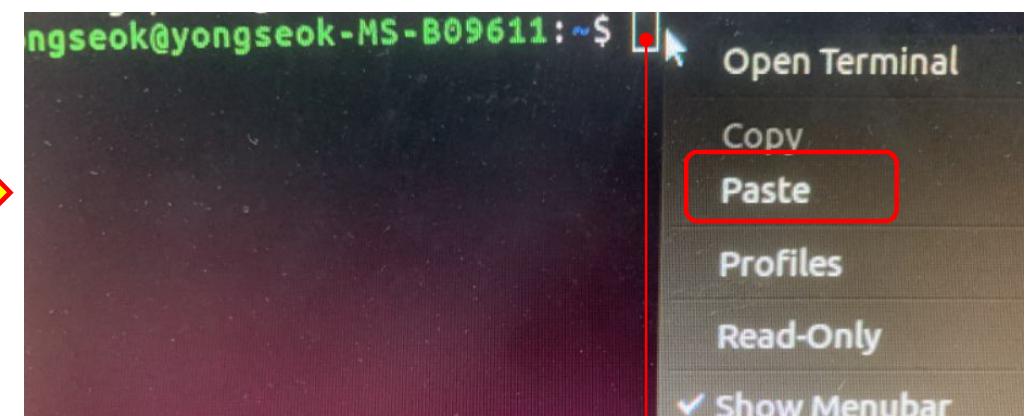
Open the terminal with **ctrl + Alt + T** and enter the following command.

In order to check the details of the easy installation, click the link below.

```
$ sudo apt-get update  
$ sudo apt-get upgrade  
$ wget https://raw.githubusercontent.com/ROBOTIS-GIT/  
$ chmod 755 ./install_ros_kinetic.sh  
$ bash ./install_ros_kinetic.sh
```

drag 하고 copy

Copy시 주의 : 전체 copy할것



우 click → paste click

또는 Ctl + Shift + V

## 4. ROS 개발환경 구축

### 방법 2 참고

Ref. <https://emanual.robotis.com/docs/en/platform/turtlebot3/quick-start/>

: sudo apt-get update

```
yongseok@yongseok-MS-B09611:~  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
yongseok@yongseok-MS-B09611:~$ sudo apt-get update  
[sudo] password for yongseok:  
Hit:1 http://kr.archive.ubuntu.com/ubuntu xenial InRelease  
Reading package lists... Done  
yongseok@yongseok-MS-B09611:~$
```

- 입력 후 enter
- 1234 enter(입력 안보임)
- update 화면(완료 확인할 것)

: sudo apt-get upgrade

```
yongseok@yongseok-MS-B09611:~  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
yongseok@yongseok-MS-B09611:~$ sudo apt-get update  
[sudo] password for yongseok:  
Hit:1 http://kr.archive.ubuntu.com/ubuntu xenial InRelease  
Reading package lists... Done  
yongseok@yongseok-MS-B09611:~$ sudo apt-get upgrade  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
Calculating upgrade... Done  
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.  
yongseok@yongseok-MS-B09611:~$
```

- 입력 후 enter
- upgrade 화면(완료 확인할 것)

## 4. ROS 개발환경 구축

방법 2 참고

Ref. <https://emanual.robotis.com/docs/en/platform/turtlebot3/quick-start/>

: 위에 동일한 방법으로

: wget [https://raw.githubusercontent.com/ROBOTIS-GIT/robotis\\_tools/master/install\\_ros\\_kinetic.sh](https://raw.githubusercontent.com/ROBOTIS-GIT/robotis_tools/master/install_ros_kinetic.sh)

```
yongseok@yongseok-MS-B09611:~$ wget https://raw.githubusercontent.com/ROBOTIS-GIT/robotis_tools/master/install_ros_kinetic.sh
--2021-09-13 19:25:06-- https://raw.githubusercontent.com/ROBOTIS-GIT/robotis_tools/master/install_ros_kinetic.sh
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 185.199.108.133, 185.199.110.133, 185.199.111.133, ...
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|185.199.108.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2981 (2.9K) [text/plain]
Saving to: 'install_ros_kinetic.sh'

install_ros_kinetic 100%[=====] 2.91K --.-KB/s in 0s

2021-09-13 19:25:06 (15.2 MB/s) - 'install_ros_kinetic.sh' saved [2981/2981]
```

입력 후  
enter

완료 확인

→ 주의: robotis emanual에서 제대로 copy 하기

```
$ apt-get upgrade
$ wget https://raw.githubusercontent.com/ROBOTIS-GIT/robotis_tools/master/install_ros_kinetic.sh
$ chmod 755 ./install_ros_kinetic.sh
```

화면에 보이는 부분 외에 우측까지  
주소가 있음

: chmod 755 ./install\_ros\_kinetic.sh

```
yongseok@yongseok-MS-B09611:~$ chmod 755 ./install_ros_kinetic.sh
```

입력 후  
enter

## 4. ROS 개발환경 구축

방법 2 참고

Ref. <https://emanual.robotis.com/docs/en/platform/turtlebot3/quick-start/>

: 위에 동일한 방법으로

: bash ./install\_ros\_kinetic.sh

```
yongseok@yongseok-MS-B09611:~$ bash ./install_ros_kinetic.sh
[Note] Target OS version >> Ubuntu 16.04.x (xenial) or Linux Mint 18.x
[Note] Target ROS version >> ROS Kinetic Kame
[Note] Catkin workspace >> /home/yongseok/catkin_ws
PRESS [ENTER] TO CONTINUE THE INSTALLATION
IF YOU WANT TO CANCEL, PRESS [CTRL] + [C]

[Set the target OS, ROS version and name of catkin workspace]
[Update the package lists and upgrade them]
Hit:1 http://kr.archive.ubuntu.com/ubuntu xenial InRelease
Reading package lists... Done
Reading package lists... Done
Building dependency tree
Reading state information... Done
```

~ 화면 생략

```
[Add the ROS repository]
[Download the ROS keys]
./install_ros_kinetic.sh: line 35: curl: command not found
gpg: no valid OpenPGP data found.
[Check the ROS keys]
[Failed to receive the ROS key, aborts the installation]
yongseok@yongseok-MS-B09611:~$
```

→ ROS Key(new version)를 설치하지 못했다면, 아래 실행

```
sudo apt-key adv --keyserver 'hkp://keyserver.ubuntu.com:80' --recv-key
C1CF6E31E6BADE8868B172B4F42ED6FBAB17C654
```

• 입력 후 enter

version 확인

• enter

우분투 재설치 하기

ros 설치 완료\_20211007\_debugging.TXT 처  
럼 하기



C:\2021\_lecture\W  
ROS 프로그래밍

ROS Key 성공 화면

```
-- Generating done
-- Build files have been written to: /home/dabin/catkin_ws/build
#####
##### Running command: "make -j2 -l2" in "/home/dabin/catkin_ws/build"
#####
[Set the ROS environment]
[Complete!!!]
dabin@dabin:~$
```

## 4. ROS 개발환경 구축

Ref. <https://emanual.robotis.com/docs/en/platform/turtlebot3/quick-start/>

### (4) Install Dependent ROS Packages (종속 package)

: <https://emanual.robotis.com/docs/en/platform/turtlebot3/quick-start>에서

#### 3. 1. 3. Install Dependent ROS Packages

```
$ sudo apt-get install ros-kinetic-joy ros-kinetic-teleop-twist-joy  
ros-kinetic-teleop-twist-keyboard ros-kinetic-laser-proc  
ros-kinetic-rgbd-launch ros-kinetic-depthimage-to-laserscan  
ros-kinetic-rosserial-arduino ros-kinetic-rosserial-python  
ros-kinetic-rosserial-server ros-kinetic-rosserial-client  
ros-kinetic-rosserial-msgs ros-kinetic-amcl ros-kinetic-map-server  
ros-kinetic-move-base ros-kinetic-urdf ros-kinetic-xacro  
ros-kinetic-compressed-image-transport ros-kinetic-rqt*  
ros-kinetic-gmapping ros-kinetic-navigation ros-kinetic-interactive-markers
```

Copy후 terminal창에서 실행

한 칸 띄기

```
$ sudo apt-get install ros-kinetic-joy ros-kinetic-teleop-twist-joy ros-kinetic-teleop-twist-keyboard ros-kinetic-laser-proc  
ros-kinetic-rgbd-launch ros-kinetic-depthimage-to-laserscan ros-kinetic-rosserial-arduino ros-kinetic-rosserial-python ros-  
kinetic-rosserial-server ros-kinetic-rosserial-client ros-kinetic-rosserial-msg ros-kinetic-amcl ros-kinetic-map-server ros-  
kinetic-move-base ros-kinetic-urdf ros-kinetic-xacro ros-kinetic-compressed-image-transport ros-kinetic-rqt-image-view  
ros-kinetic-gmapping ros-kinetic-navigation ros-kinetic-interactive-markers
```

## 4. ROS 개발환경 구축

Ref. <https://emanual.robotis.com/docs/en/platform/turtlebot3/quick-start/>



```
yongseok@yongseok:~$ gedit ~/.bashrc
yongseok@yongseok:~$ sudo apt-get install ros-kinetic-joy ros-kinetic-teleop-twist-joy ros-kinetic-teleop-twist-keyboard ros-kinetic-laser-proc ros-kinetic-rgbd-launch ros-kinetic-depthimage-to-laserscan ros-kinetic-rosserial-arduino ros-kinetic-rosserial-python ros-kinetic-rosserial-server ros-kinetic-rosserial-client ros-kinetic-rosserial-msg ros-kinetic-amcl ros-kinetic-map-server ros-kinetic-move-base ros-kinetic-urdf ros-kinetic-xacro ros-kinetic-compressed-image-transport ros-kinetic-rqt-image-view ros-kinetic-gmapping ros-kinetic-navigation ros-kinetic-interactive-markers
[sudo] password for yongseok:
Reading package lists... Done
Building dependency tree
Reading state information... Done
E: Unable to locate package ros-kinetic-rosserial-msg
yongseok@yongseok:~$
```

} 입력 후 실행

## 4. ROS 개발환경 구축

Ref. <https://emanual.robotis.com/docs/en/platform/turtlebot3/quick-start/>

### (5) Install TurtleBot3 Packages

: Install TurtleBot3 via Debian Package

: <https://emanual.robotis.com/docs/en/platform/turtlebot3/quick-start>에서

#### 3. 1. 4. Install TurtleBot3 Packages

Install TurtleBot3 via Debian Packages.

```
$ sudo apt-get install ros-kinetic-dynamixel-sdk  
$ sudo apt-get install ros-kinetic-turtlebot3-msgs  
$ sudo apt-get install ros-kinetic-turtlebot3
```

\$ sudo apt-get install ros-kinetic-dynamixel-sdk

```
yongseok@yongseok:~$ sudo apt-get install ros-kinetic-dynamixel-sdk  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
The following NEW packages will be installed:  
  ros-kinetic-dynamixel-sdk  
0 upgraded, 1 newly installed, 0 to remove and 7 not upgraded.  
Need to get 60.3 kB of archives.  
After this operation, 535 kB of additional disk space will be used.  
Get:1 http://packages.ros.org/ros/ubuntu xenial/main amd64 ros-kinetic-dynamixel-  
-sdk amd64 3.7.51-2xenial-20210503-104902-0800 [60.3 kB]  
Fetched 60.3 kB in 1s (32.2 kB/s)  
Selecting previously unselected package ros-kinetic-dynamixel-sdk.  
(Reading database ... 308039 files and directories currently installed.)  
Preparing to unpack .../ros-kinetic-dynamixel-sdk_3.7.51-2xenial-20210503-104902-  
-0800_amd64.deb ...  
Unpacking ros-kinetic-dynamixel-sdk (3.7.51-2xenial-20210503-104902-0800) ...  
Setting up ros-kinetic-dynamixel-sdk (3.7.51-2xenial-20210503-104902-0800) ...  
yongseok@yongseok:~$
```

● kinetic-dynamixel 확인

## 4. ROS 개발환경 구축

Ref. <https://emanual.robotis.com/docs/en/platform/turtlebot3/quick-start/>

```
$ sudo apt-get install ros-kinetic-turtlebot3-msgs
```

```
Setting up ros_kinetic_dynamixel-srv (3.7.51-2xenial-20210503-104902-0800) ...
yongseok@yongseok:~$ sudo apt-get install ros-kinetic-turtlebot3-msgs
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  ros-kinetic-turtlebot3-msgs
0 upgraded, 1 newly installed, 0 to remove and 7 not upgraded.
Need to get 22.3 kB of archives.
After this operation, 200 kB of additional disk space will be used.
Get:1 http://packages.ros.org/ros/ubuntu xenial/main amd64 ros-kinetic-turtlebot
3-msgs amd64 1.0.1-1xenial-20210503-100300-0800 [22.3 kB]
Fetched 22.3 kB in 3s (6,939 B/s)
Selecting previously unselected package ros-kinetic-turtlebot3-msgs.
(Reading database ... 308094 files and directories currently installed.)
Preparing to unpack .../ros-kinetic-turtlebot3-msgs_1.0.1-1xenial-20210503-10030
0-0800_amd64.deb ...
Unpacking ros-kinetic-turtlebot3-msgs (1.0.1-1xenial-20210503-100300-0800) ...
Setting up ros-kinetic-turtlebot3-msgs (1.0.1-1xenial-20210503-100300-0800) ...
yongseok@yongseok:~$ █
```

● kinetic-turtlebot3-msgs

확인

## 4. ROS 개발환경 구축

Ref. <https://emanual.robotis.com/docs/en/platform/turtlebot3/quick-start/>

```
$ sudo apt-get install ros-kinetic-turtlebot3
```

~ 화면 생략

```
) ...  
Setting up ros-kinetic-turtlebot3-slam (1.2.5-1xenial-20210503-155605-0800) ...  
Setting up ros-kinetic-turtlebot3 (1.2.5-1xenial-20210503-160735-0800) ... I  
Processing triggers for libc-bin (2.23-0ubuntu11.3) ...  
yongseok@yongseok:~$ █
```

[emanual.robotis.com/docs/en/platform/turtlebot3/quick-start/](https://emanual.robotis.com/docs/en/platform/turtlebot3/quick-start/) 에서 1.1.4 까지 진행

# 4. ROS 개발환경 구축

Ref. <https://emanual.robotis.com/docs/en/platform/turtlebot3/quick-start/>

## (6) ROS 환경변수 설정 :

```
$ source /opt/ros/kinetic/setup.bash
```

```
$ source ~/catkin_ws/devel/setup.bash
```

: 위와 같은 ROS 설치 과정에서 사용된 환경설정 파일을

매번 불러오기 번거롭기에 bashrc 파일에 환경변수 설정

```
$ gedit ~/.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
    *t* );
    *) return;;
esac

# don't put duplicate lines or lines starting with space in the history.
# See bash(1) for more options
HISTCONTROL=ignoreboth

# append to the history file, don't overwrite it
shopt -s histappend

# for setting history length see HISTSIZE and HISTFILESIZE in bash(1)
HISTSIZE=1000
HISTFILESIZE=2000

# check the window size after each command and, if necessary,
# update the values of LINES and COLUMNS.
shopt -s checkwinsize

# If set, the pattern "*" used in a pathname expansion context will
# match all files and zero or more directories and subdirectories.
#shopt -s globstar

# make less more friendly for non-text input files, see lesspipe(1)
[ -x /usr/bin/lesspipe ] && eval "$(SHELL=/bin/sh lesspipe)"

# set variable identifying the chroot you work in (used in the prompt below)
if [ -z "$debian_chroot" ] && [ -r /etc/debian_chroot ]; then
    debian_chroot=$(cat /etc/debian_chroot)
fi

# set a fancy prompt (non-color, unless we know we "want" color)
case "$TERM" in
    xterm-color|*-256color) color_prompt=yes;;
esac

# uncomment for a colored prompt, if the terminal has the capability; turned
# off by default to not distract the user: the focus is in a terminal window
# should be on the output of commands, not on the prompt
#force_color_prompt=yes
```

```
# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
    if [ -f /usr/share/bash-completion/bash_completion ]; then
        . /usr/share/bash-completion/bash_completion
    elif [ -f /etc/bash_completion ]; then
        . /etc/bash_completion
    fi
fi
alias eb='nano ~/.bashrc'
alias sb='source ~/.bashrc'
alias gs='git status'
alias gp='git pull'
alias cw='cd ~/catkin_ws'
alias cs='cd ~/catkin_ws/src'
alias cm='cd ~/catkin_ws && catkin_make'
source /opt/ros/kinetic/setup.bash
source ~/catkin_ws/devel/setup.bash
export ROS_MASTER_URI=http://localhost:11311
export ROS_HOSTNAME=localhost
source /opt/ros/kinetic/setup.bash
source /opt/ros/kinetic/setup.bash
alias eb='nano ~/.bashrc'
alias sb='source ~/.bashrc'
alias gs='git status'
alias gp='git pull'
alias cw='cd ~/catkin_ws'
alias cs='cd ~/catkin_ws/src'
alias cm='cd ~/catkin_ws && catkin_make'
source /opt/ros/kinetic/setup.bash
source ~/catkin_ws/devel/setup.bash
export ROS_MASTER_URI=http://localhost:11311
export ROS_HOSTNAME=localhost
alias eb='nano ~/.bashrc'
alias sb='source ~/.bashrc'
alias gs='git status'
alias gp='git pull'
alias cw='cd ~/catkin_ws'
alias cs='cd ~/catkin_ws/src'
alias cm='cd ~/catkin_ws && catkin_make'
source /opt/ros/kinetic/setup.bash
source ~/catkin_ws/devel/setup.bash
export ROS_MASTER_URI=http://localhost:11311
export ROS_HOSTNAME=localhost
alias eb='nano ~/.bashrc'
alias sb='source ~/.bashrc'
alias gs='git status'
alias gp='git pull'
alias cw='cd ~/catkin_ws'
alias cs='cd ~/catkin_ws/src'
alias cm='cd ~/catkin_ws && catkin_make'
source /opt/ros/kinetic/setup.bash
source ~/catkin_ws/devel/setup.bash
export ROS_MASTER_URI=http://localhost:11311
export ROS_HOSTNAME=localhost
```

확인

## 4. ROS 개발환경 구축

방법 2에서도 동일

Ref. ROS 로봇 프로그래밍(Ruby paper) 3-1 ROS 설치

### 환경설정 파일 불러오기

: ROS\_ROOT, ROS\_PACKAGE\_PATH 등의 환경변수 정의된 환경설정 파일 불러오기

→ \$ source /opt/ros/kinetic/setup.bash

```
yongseok@yongseok:~$ source /opt/ros/kinetic/setup.bash
```

bash : 파일을 읽고 실행 명령어

### 작업 폴더 생성 및 초기화

: ROS에서는 catkin이라는 ROS 전용 빌드 시스템 사용

: catkin 사용하려면 catkin 작업 폴더를 **생성하고 초기화해야 함**

→ \$ mkdir -p ~/catkin\_ws/src

\$ cd ~/catkin\_ws/src

\$ catkin\_init\_workspace

```
yongseok@yongseok:~$ mkdir -p ~/catkin_ws/src  
yongseok@yongseok:~$ cd ~/catkin_ws/src  
yongseok@yongseok:~/catkin_ws/src$
```

• CMakeLists.txt 파일 생성

• directory 확인

```
yongseok@yongseok:~/catkin_ws/src$ catkin_init_workspace  
File "/home/yongseok/catkin_ws/src/CMakeLists.txt" already exists  
yongseok@yongseok:~/catkin_ws/src$
```

-CMakeLists.txt 파일 확인

catkin\_init\_workspace : 작업공간 초기화를 의미.

위의 명령들을 실행하고 난 뒤에 작업공간이 비어 있음(src 폴더 안에 CMakeLists.txt만 존재)

## 4. ROS 개발환경 구축

Ref. ROS 로봇 프로그래밍(Ruby paper) 3-1 ROS 설치

### build

: 경로 확인 (build 경로)

→ \$ cd ~/catkin\_ws/

\$ catkin\_make



\$ ls

```
[100%] Built target turtlebot3_example_generate_messages
yongseok@yongseok:~/catkin_ws$ ls
build  devel  install_ros_kinetic.sh  src
yongseok@yongseok:~/catkin_ws$
```



우리가 만든 src 폴더 확인

그리고 build 와 devel 폴더 생성

→ catkin 빌드 파일은 build 폴더에 저장

→ 빌드 후 실행 파일은 devel 폴더에 저장

```
yongseok@yongseok:~/catkin_ws/src$ catkin_init_workspace
File "/home/yongseok/catkin_ws/src/CMakeLists.txt" already exists
yongseok@yongseok:~/catkin_ws$ catkin_make
Base path: /home/yongseok/catkin_ws
Source space: /home/yongseok/catkin_ws/src
Build space: /home/yongseok/catkin_ws/build
Devel space: /home/yongseok/catkin_ws/devel
Install space: /home/yongseok/catkin_ws/install
#####
##### Running command: "make cmake_check_build_system" in "/home/yongseok/catkin_ws/build"
d: "make -j2 -l2" in "/home/yongseok/catkin_ws/build"
```

만일 이 내용이 없다면 catkin\_make가 불안전  
(source code와 환경설정 build 안 된 것임)  
→ ubuntu 재설치  
(roscore는 설치되었어도 문제가 내재)

```
0%] Built target std_msgs_generate_messages_py
0%] Built target _turtlebot3_msgs_generate_messages_check_deps_SensorState
0%] Built target _turtlebot3_msgs_generate_messages_check_deps_Sound
0%] Built target std_msgs_generate_messages_cpp
0%] Built target turtlebot3_msgs_generate_messages_check_deps_VersionInfo
```

-실행 내용 생략

```
[ 84%] Built target turtlebot3_example_generate_messages_nodejs
[ 96%] Built target turtlebot3_example_generate_messages_lisp
[ 96%] Built target turtlebot3_msgs_generate_messages
[100%] Built target turtlebot3_diagnostics
[100%] Built target turtlebot3_example_generate_messages
yongseok@yongseok:~/catkin_ws$
```



100% build 확인

## 4. ROS 개발환경 구축

Ref. ROS 로봇 프로그래밍(Ruby paper) 3-1 ROS 설치

catkin 빌드 시스템과 관련된 환경 파일 적용하기

→ \$ source ~/catkin\_ws/devel/setup.bash ★

```
yongseok@yongseok:~/catkin_ws$ ls  
build  devel  install_ros_kinetic.sh  src  
yongseok@yongseok:~/catkin_ws$ source ~/catkin_ws/devel/setup.bash  
yongseok@yongseok:~/catkin_ws$ █
```

테스트를 위해

CTRL + Shift +Q 열려있는 Terminal 창 닫기

다시 새로운 창 열기

CTRL + ALT + T 를 사용해 터미널 창 열기

build과정으로 생성된 devel파일 안의 setup.bash를 실행하여  
→ 그 작업공간이 ros 환경의 최상위에 overlay되도록 쉘에 등록  
→ setup.bash를 실행하면 environment variables가 setting되고  
이것이 시스템이 필요한 모든 package와 library 등을 찾게 도와줌

## 4. ROS 개발환경 구축

Ref. ROS 로봇 프로그래밍(Ruby paper) 3-1 ROS 설치

### ★ 최종 테스트

→ \$ roscore

: ROS master 구동하는 명령어



오류가 발생한다면



ROS 설치에 문제 있음

재설치 필요



문제가 지속 발생 경우

UBUNTU 재설치

```
yongseok@yongseok:~$ roscore
... logging to /home/yongseok/.ros/log/83cce24c-1e9c-11ec-b5be-f40669f0af37/roslaunch-yongseok-21558.log
Checking log directory for disk usage. This may take awhile.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://localhost:36691/
ros_comm version 1.12.17

SUMMARY
=====

PARAMETERS
* /rosdistro: kinetic
* /rosversion: 1.12.17 → Version Kinetic 확인

NODES

auto-starting new master
process[master]: started with pid [21569]
ROS_MASTER_URI=http://localhost:11311/

setting /run_id to 83cce24c-1e9c-11ec-b5be-f40669f0af37
process[rosout-1]: started with pid [21582]
started core service [/rosout] → 이 상태에서 끝내려면 Ctrl+c ★
```

## 4. ROS 개발환경 구축

Ref. <https://emanual.robotis.com/docs/en/platform/turtlebot3/quick-start/>



### 최종 정리 ROS 설치

ros 설치 완료\_20211007\_debugging.txt

설치시 와이파이 상태 반드시 지속적으로 확인하기  
(youtube등을 들어가면서)



### build

```
$ cd ~/catkin_ws/  
$ catkin_make
```



CTRL + Shift +Q 열려있는 Terminal 창 닫기  
다시 새로운 창 열기  
CTRL + ALT + T 를 사용해 터미널 창 열기



### ROS master 구동

```
$ roscore  
: kinetic version 확인하기  
: 끝내려면 Ctrl+c
```



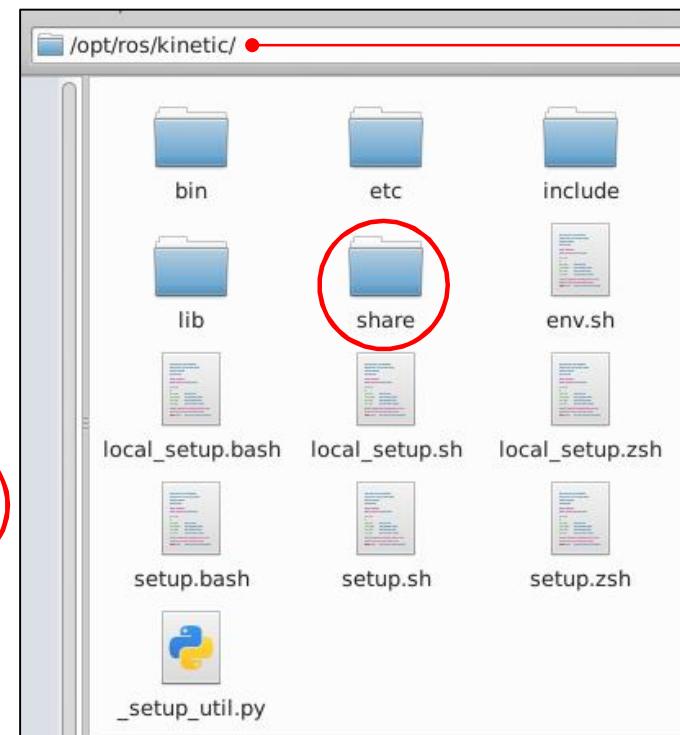
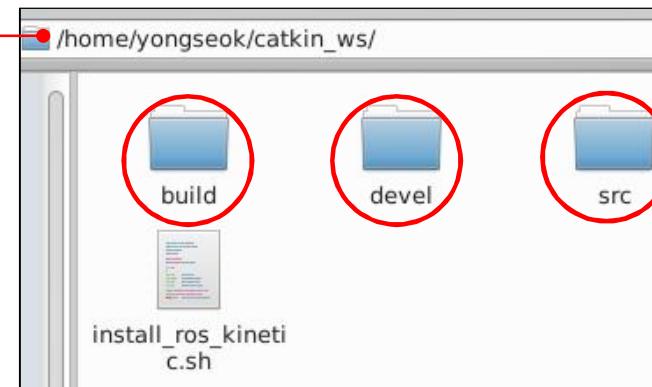
오류가 발생한다면  
ROS 설치에 문제 있음  
재설치 필요

## 4. ROS 개발환경 구축

### [파일 구성]

- : ROS 파일 시스템은 설치 폴더와 사용자 작업 폴더로 구분
  - ROS 설치 폴더는 ROS를 설치하면 /opt 폴더에 ros 폴더 생성  
→ ROS 설치 폴더 경로 /opt/ros/kinetic
  - ros 폴더에 roscore를 포함한 핵심 유ти리티와 rqt, Rviz, 로봇 관련 라이브러리, 시뮬레이션, 내비게이션 등 설치
- 사용자 작업 폴더는 사용자 폴더인 ~/catkin\_ws 사용

사용자 작업 폴더



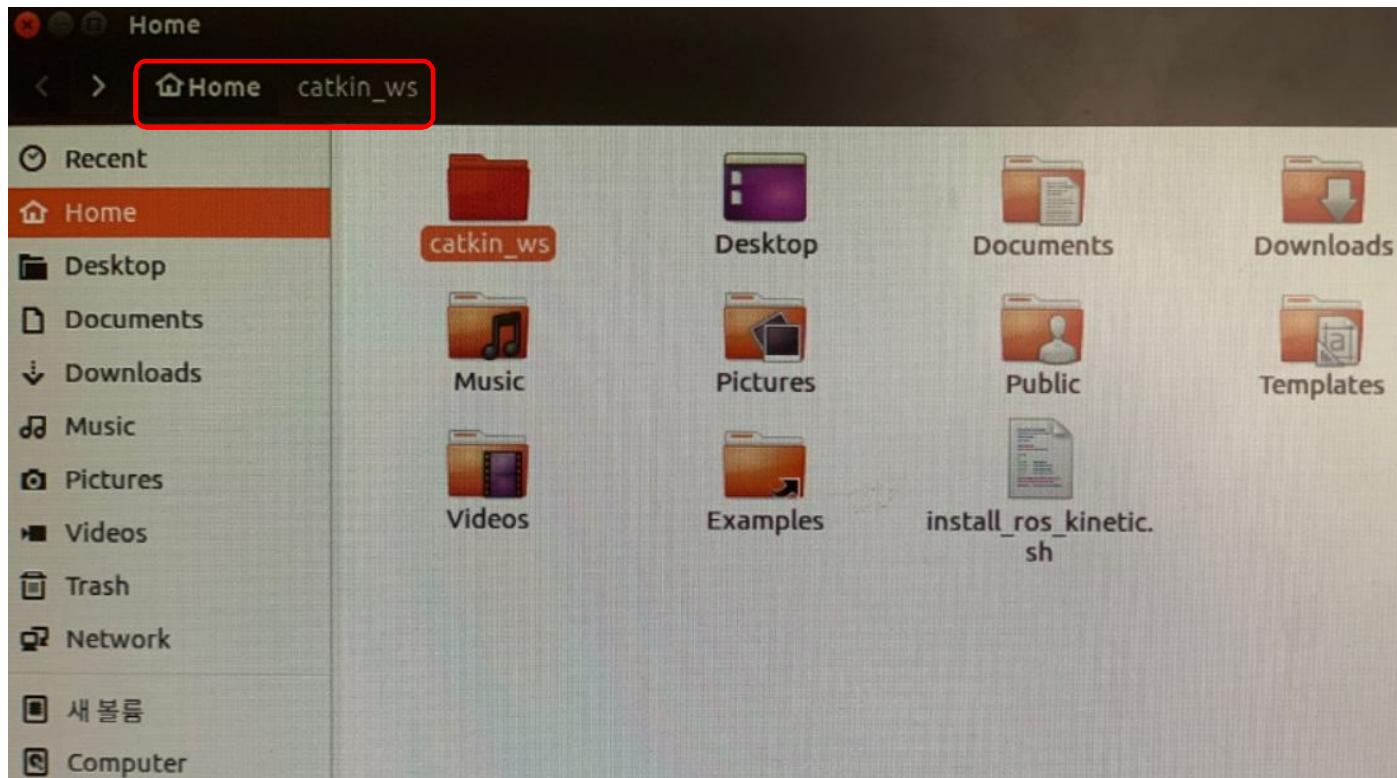
ROS 설치 폴더

## 4. ROS 개발환경 구축

### [파일 구성]

: 사용자 작업 폴더는 **사용자 폴더인**

: **~/catkin\_ws == Home/catkin\_ws**

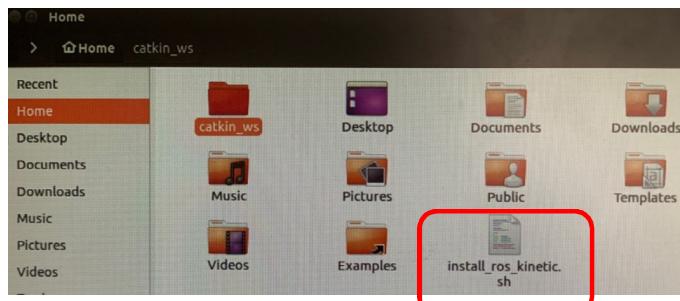


## 4. ROS 개발환경 구축

### [파일 구성]

: 사용자 작업 폴더는 사용자 폴더인

: ~/catkin\_ws == Home/catkin\_ws



Install\_ros\_kinetic.sh

```
install_ros_kinetic.sh (~) - gedit
Open ▾

#!/bin/bash
# Apache License 2.0
# Copyright (c) 2017, ROBOTIS CO., LTD.

echo ""
echo "[Note] Target OS version >>> Ubuntu 16.04.x (xenial) or Linux Mint 18.x"
echo "[Note] Target ROS version >>> ROS Kinetic Kame"
echo "[Note] Catkin workspace >>> $HOME/catkin_ws"
echo ""
echo "PRESS [ENTER] TO CONTINUE THE INSTALLATION"
echo "IF YOU WANT TO CANCEL, PRESS [CTRL] + [C]"
read

echo "[Set the target OS, ROS version and name of catkin workspace]"
name_os_version=${name_os_version:="xenial"}
name_ros_version=${name_ros_version:="kinetic"}
name_catkin_workspace=${name_catkin_workspace:="catkin_ws"}

echo "[Update the package lists]"
sudo apt-get update -y

echo "[Install build environment, the chrony, ntpdate and set the ntpdate]"
sudo apt-get install -y chrony ntpdate curl build-essential git
sudo ntpdate ntp.ubuntu.com

echo "[Add the ROS repository]"
if [ ! -e /etc/apt/sources.list.d/ros-latest.list ]; then
    sudo sh -c "echo \`deb http://packages.ros.org/ros/ubuntu ${name_os_version} main\` > sources.list.d/ros-latest.list"
fi

echo "[Download the ROS keys]"
roskey=apt-key list | grep "Open Robotics"
if [ -z "$roskey" ]; then
    curl -s https://raw.githubusercontent.com/ros/rosdistro/master/ros.asc | sudo apt-key add -
fi

echo "[Check the ROS keys]"
roskey=apt-key list | grep "Open Robotics"
if [ -n "$roskey" ]; then
    echo "[ROS key exists in the list]"
else
    echo "[Failed to receive the ROS key, aborts the installation]"
    exit 0
fi

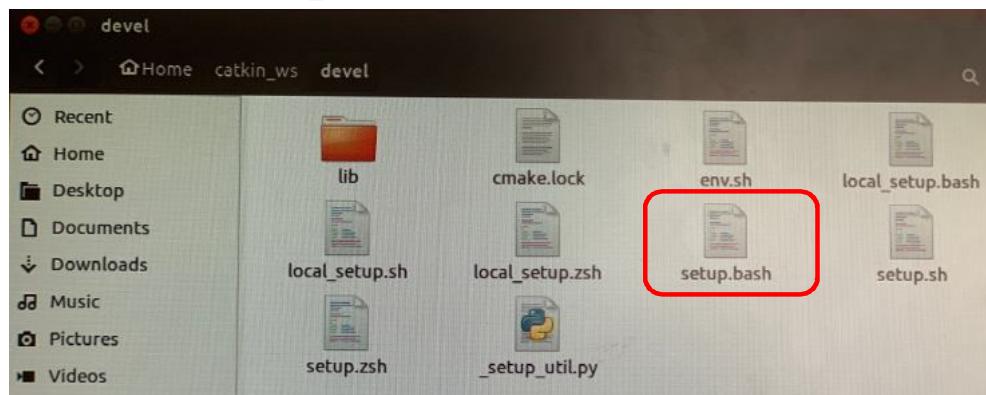
echo "[Update the package lists]"
sudo apt-get update

echo "[Install the ros-desktop-full and all rqt plugins]"
sudo apt-get install ros-$name_catkin_workspace-desktop-full ros-$name_catkin_workspace-ros-base ros-$name_catkin_workspace-ros-visualization
```

## 4. ROS 개발환경 구축

### [파일 구성]

: Home/catkin\_ws-devel



Setup.bash

The terminal window shows the content of the 'setup.bash' file:

```
#!/usr/bin/env bash
# generated from catkin/cmake/templates/setup.bash.in

CATKIN_SHELL=bash

# source setup.sh from same directory as this file
_CATKIN_SETUP_DIR=$(builtin cd "$(dirname "${BASH_SOURCE[0]}")" > /dev/null || pwd)
. "$_CATKIN_SETUP_DIR/setup.sh"
```

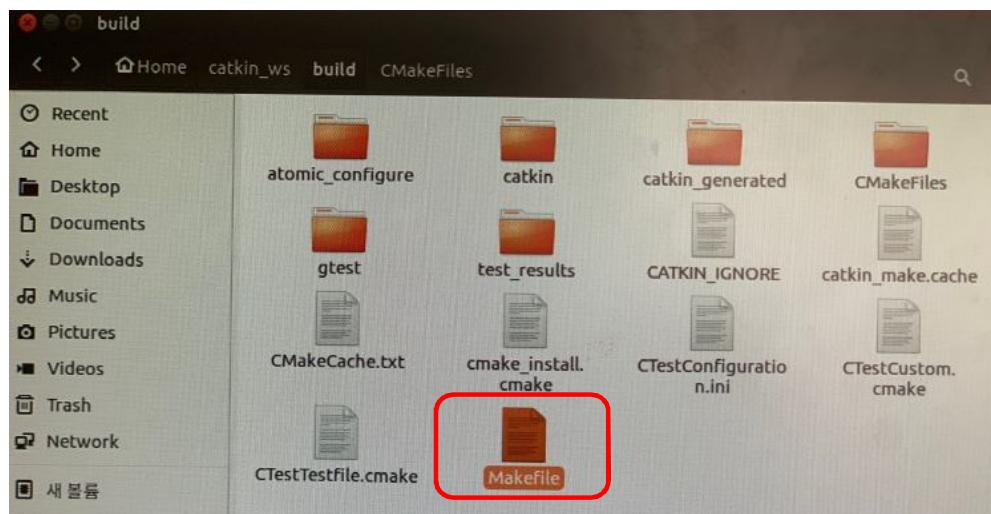
Below the terminal is a screenshot of an Ubuntu 18.04.6 LTS upgrade dialog box:

Ubuntu 18.04.6 LTS Upgrade Available  
A new version of Ubuntu is available. Would you like to upgrade?  
Don't Upgrade Ask Me Later Yes, Upgrade Now

## 4. ROS 개발환경 구축

## [파일 구성]

: Home/catkin\_ws/build

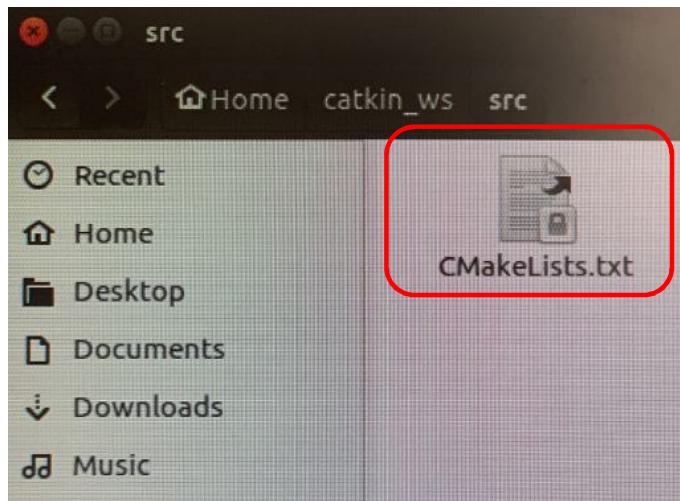


## Makefile

## 4. ROS 개발환경 구축

### [파일 구성]

: Home/catkin\_ws/src



CMakeLists.txt

```
# toplevel CMakeLists.txt for a catkin workspace
# catkin/cmake/toplevel.cmake

cmake_minimum_required(VERSION 3.0.2)
project(Project)

set(CATKIN_TOPLEVEL TRUE)

# search for catkin within the workspace
set(_cmd "catkin_find_pkg" "catkin" "${CMAKE_SOURCE_DIR}")
execute_process(COMMAND ${_cmd}
    RESULT_VARIABLE _res
    OUTPUT_VARIABLE _out
    ERROR_VARIABLE _err
    OUTPUT_STRIP_TRAILING_WHITESPACE
    ERROR_STRIP_TRAILING_WHITESPACE
)
if(NOT _res EQUAL 0 AND NOT _res EQUAL 2)
    # searching for catkin resulted in an error
    string(REPLACE ";" " " _cmd_str "${_cmd}")
    message(FATAL_ERROR "Search for 'catkin' in workspace failed")
endif()

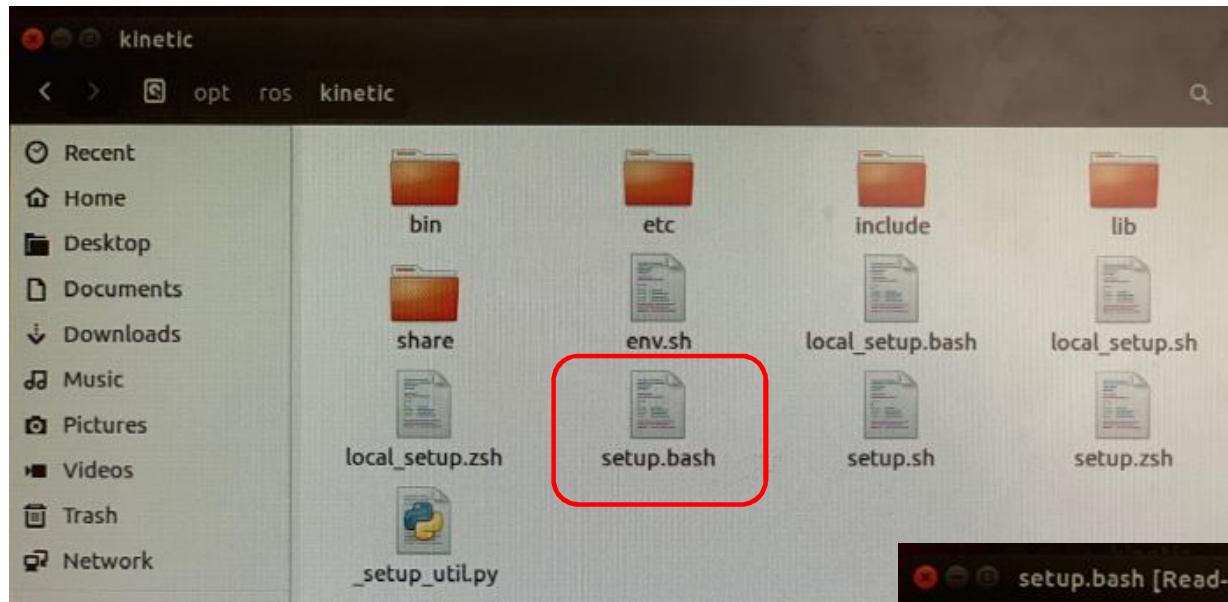
# include catkin from workspace or via find_package()
if(_res EQUAL 0)
    set(catkin_EXTRAS_DIR "${CMAKE_SOURCE_DIR}/${_out}/cmake")
    # include all.cmake without add_subdirectory to let it operate
    include(${catkin_EXTRAS_DIR}/all.cmake NO_POLICY_SCOPE)
    add_subdirectory("${_out}")

else()
    # use either CMAKE_PREFIX_PATH explicitly passed to CMake as
    # or CMAKE_PREFIX_PATH from the environment
    if(NOT DEFINED CMAKE_PREFIX_PATH)
        if(NOT "$ENV{CMAKE_PREFIX_PATH}" STREQUAL "")
            if(NOT WIN32)
                string(REPLACE ":" ";" CMAKE_PREFIX_PATH $ENV{CMAKE_PREFIX_PATH})
            else()
                set(CMAKE_PREFIX_PATH $ENV{CMAKE_PREFIX_PATH})
            endif()
        endif()
    endif()
endif()
```

## 4. ROS 개발환경 구축

### [파일 구성]

: ROS 설치 폴더 경로 /opt/ros/kinetic



```
#!/usr/bin/env bash
# generated from catkin/cmake/templates/setup.bash.in

CATKIN_SHELL=bash

# source setup.sh from same directory as this file
_CATKIN_SETUP_DIR=$(builtin cd `dirname "${BASH_SOURCE[0]}` > /dev/null && pwd)
. "$_CATKIN_SETUP_DIR/setup.sh"
```

## 4. ROS 개발환경 구축

### (7) 네트워크 구성 ☆

: turtlebot은 Remote PC(미니PC)와 SBC(Single Board Computer : Raspberry Pi 3 B+)가 원격으로 연결되어야 함

: 원격으로 연결을 위해 Wifi를 사용하여 IP 주소를 맞춰 주어야 함

: Remote PC의 IP 주소 확인

\$ ifconfig

```
yongseok@yongseok:~$ ifconfig
enp2s0    Link encap:Ethernet HWaddr 4c:cc:6a:9d:47:16
          UP BROADCAST MULTICAST MTU:1500 Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
            UP LOOPBACK RUNNING MTU:65536 Metric:1
            RX packets:3638 errors:0 dropped:0 overruns:0 frame:0
            TX packets:3638 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:1000
            RX bytes:713589 (713.5 KB) TX bytes:713589 (713.5 KB)

wlp3s0    Link encap:Ethernet HWaddr f4:06:69:f0:af:37
          inet addr:192.168.0.15 Bcast:192.168.0.255 Mask:255.255.255.0
          inet6 addr: fe80::f406:69ff:fe0af:37/64 Scope:Link
            UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
            RX packets:553870 errors:0 dropped:0 overruns:0 frame:0
            TX packets:224046 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:1000
            RX bytes:810194447 (810.1 MB) TX bytes:23074618 (23.0 MB)
```

Remote PC (미니PC)  
: ROS Master



SBC(Single Board Computer)  
: Raspberry Pi 3 B+



WiFi

실습 환경이 변경될 때마다,  
네트워크 구성은 지속적으로 변경해야 함

## 4. ROS 개발환경 구축

### (7) 네트워크 구성

: 환경설정 파일에서 Remote PC IP 입력하기

\$ gedit ~/.bashrc



모든 ROS\_MASTER\_URI과 ROS\_HOSTNAME에서

localhost를 본인 Remote PC IP로 변경하기

Export path = 새로 등록할 환경변수 주소

export ROS\_MASTER\_URI=http://localhost:11311

export ROS\_HOSTNAME= localhost



export ROS\_MASTER\_URI=http://192.168.0.15:11311

export ROS\_HOSTNAME=192.168.0.15



: 저장하기

: 설정 반영하기



\$ source ~/.bashrc

yongseok@yongseok:~\$ source ~/.bashrc  
yongseok@yongseok:~\$

```
alias eb='nano ~/.bashrc'
alias sb='source ~/.bashrc'
alias gs='git status'
alias gp='git pull'
alias cw='cd ~/catkin_ws'
alias cs='cd ~/catkin_ws/src'
alias cm='cd ~/catkin_ws && catkin_make'
source /opt/ros/kinetic/setup.bash
source ~/catkin_ws/devel/setup.bash
export ROS_MASTER_URI=http://localhost:11311
export ROS_HOSTNAME=localhost
source /opt/ros/kinetic/setup.bash
source /opt/ros/kinetic/setup.bash
alias eb='nano ~/.bashrc'
alias sb='source ~/.bashrc'
alias gs='git status'
alias gp='git pull'
alias cw='cd ~/catkin_ws'
alias cs='cd ~/catkin_ws/src'
alias cm='cd ~/catkin_ws && catkin_make'
source /opt/ros/kinetic/setup.bash
source ~/catkin_ws/devel/setup.bash
export ROS_MASTER_URI=http://localhost:11311
export ROS_HOSTNAME=localhost
alias eb='nano ~/.bashrc'
alias sb='source ~/.bashrc'
alias gs='git status'
alias gp='git pull'
alias cw='cd ~/catkin_ws'
alias cs='cd ~/catkin_ws/src'
alias cm='cd ~/catkin_ws && catkin_make'
source /opt/ros/kinetic/setup.bash
source ~/catkin_ws/devel/setup.bash
export ROS_MASTER_URI=http://localhost:11311
export ROS_HOSTNAME=localhost
```

## 4. ROS 개발환경 구축

### (8) Set TurtleBot3 Model Name

```
$ echo "export TURTLEBOT3_MODEL=burger" >> ~/.bashrc
```

\$ source ~/.bashrc : 환경변수 설정 반영하기

```
yongseok@yongseok:~$ echo "export TURTLEBOT3_MODEL=burger" >> ~/.bashrc
yongseok@yongseok:~$ gedit ~/.bashrc
yongseok@yongseok:~$ source ~/.bashrc
```

gedit ~/.bashrc를 실행하여

echo 명령어로 입력된 내용을 확인하기



```
source /opt/ros/kinetic/setup.bash
source ~/catkin_ws/devel/setup.bash
export ROS_MASTER_URI=http://192.168.0.15:11311
export ROS_HOSTNAME=192.168.0.15
export TURTLEBOT3_MODEL=burger
```

## 4. ROS 개발환경 구축

```
alias eb='nano ~/.bashrc'  
alias sb='source ~/.bashrc'  
alias gs='git status'  
alias gp='git pull'  
alias cw='cd ~/catkin_ws'  
alias cs='cd ~/catkin_ws/src'  
alias cm='cd ~/catkin_ws && catkin_make'
```

### (9) alias 단축 명령어

```
alias eb='nano ~/.bashrc'  
alias sb='source ~/.bashrc'  
alias gs='git status'  
alias gp='git pull'  
alias cw='cd ~/catkin_ws'  
alias cs='cd ~/catkin_ws/src'  
alias cm='cd ~/catkin_ws && catkin_make'
```

### (9) alias 단축 명령어

- eb nano 편집기로 bashrc 파일을 열기 nano ~/.bashrc
- sb 환경변수 설정을 반영 source ~/.bashrc
- gs git status 현재상태를 확인
- gp git pull fetch하고 + merge하는 명령어
- cw cd ~/catkin\_ws build 디렉토리 이동
- cs cd ~/catkin\_ws/src 이동
- cm cd ~/catkin\_ws && catkin\_make

## (10) ROS 등작 테스트 (turtlesim package)

: 새 터미널에서

\$ roscore

: 새 터미널에서

\$ rosrun turtlesim

turtlesim\_node

→ turtlesim 열림

: 새 터미널에서

\$ rosrun turtlesim

turtle\_teleop\_key

→ → 우측으로 회전

←좌측으로 회전

↑ 전진 ↓ 후진



[tab] key 사용

: 명령어 자동완성

③ turtlesim package의 turtlesim\_node 실행결과



⑤ 실행결과  
→ 우측으로 회전 ←좌측으로 회전  
↑ 전진 ↓ 후진

```
roscore http://192.168.0.15:11311
① roscore 실행
yongseok@yongseok:~$ roscore
... logging to /home/yongseok/.ros/log/65949316-1f81-11ec-b5be-f40669f0af37/r
oslaunch-yongseok-6033.log
Checking log directory for disk usage. This may take awhile.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://192.168.0.15:39841/
ros_comm version 1.12.17

SUMMARY
=====

PARAMETERS
* /rosdistro: kinetic
* /rosversion: 1.12.17

NODES
auto-starting new master
process[master]: started with pid [6043]
ROS_MASTER_URI=http://192.168.0.15:11311

setting /run_id to 65949316-1f81-11ec-b5be-f40669f0af37
process[rosout-1]: started with pid [6056]
started core service [/rosout]
```

② 새 터미널에서 rosrun turtlesim turtlesim\_node 실행
yongseok@yongseok:~\$ rosrun turtlesim turtlesim\_node
[ INFO] [1632740216.631981920]: Starting turtlesim with node name /turtlesim
[ INFO] [1632740216.654538701]: Spawning turtle [turtle1] at x=[5.544445], y=[5.544445], theta=[0.000000]

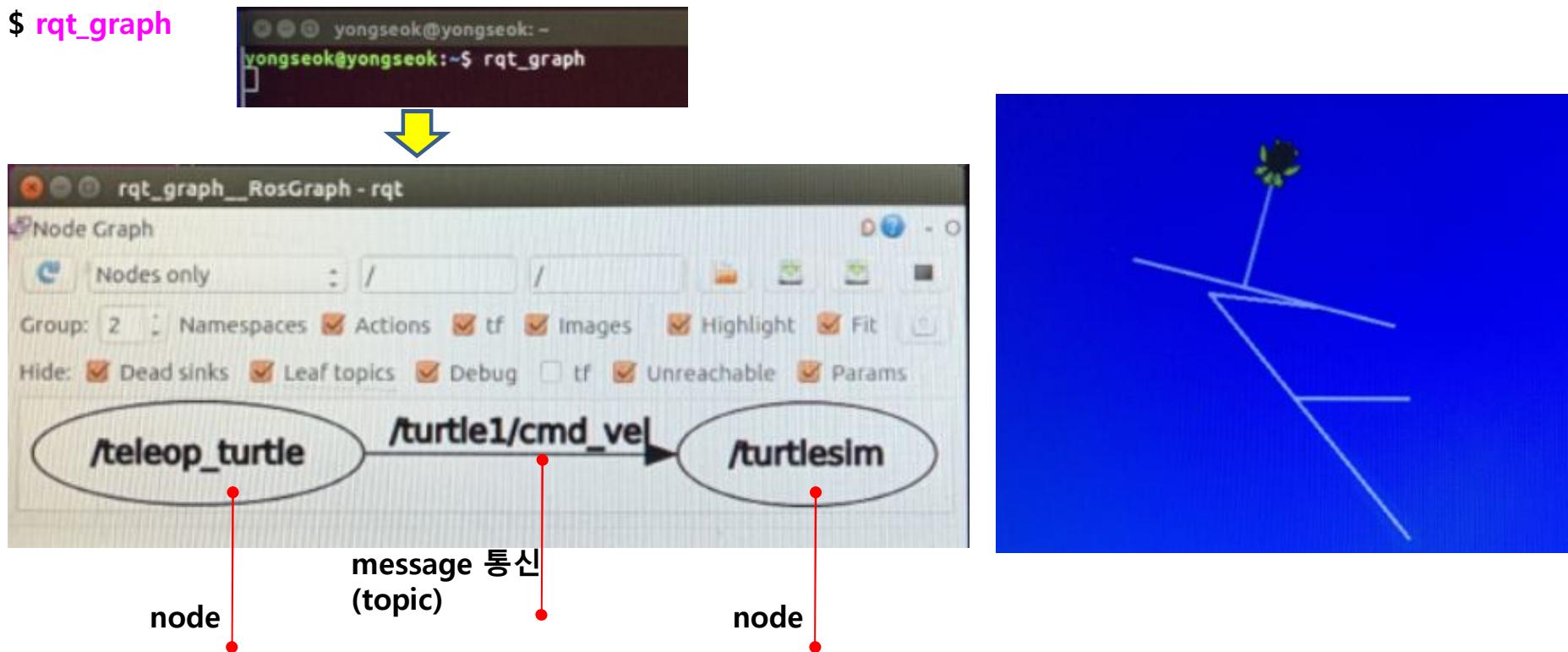
④ 새 터미널에서
rosrun turtlesim turtle\_teleop\_key 실행
yongseok@yongseok:~\$ rosrun turtlesim turtle\_teleop\_key
Reading from keyboard
-----
Use arrow keys to move the turtle.

## (10) ROS 동작 테스트 (rqt\_graph 노드)

: rqt\_graph 노드는 현재 실행 중인 노드들의 정보를 GUI 형태로 시각화

: 새 터미널에서

\$ rqt\_graph



: 실행된 roscore 및 노드는 터미널 창에서 [Ctrl+c]를 눌러 종료 후에, 그리고 CTRL + Shift +Q 열려있는 Terminal 창 닫기

## 4. ROS 개발환경 구축

실험 환경 1 : 만일 TurtleBot3 burger 조립되어 있지 않다면, 92 페이지로

실험 환경 2 : 만일 TurtleBot3 burger 조립되어 있다면,

: <https://youtu.be/rvm-m2ogrLA> 조립 동영상 보고 조립 과정 이해하기

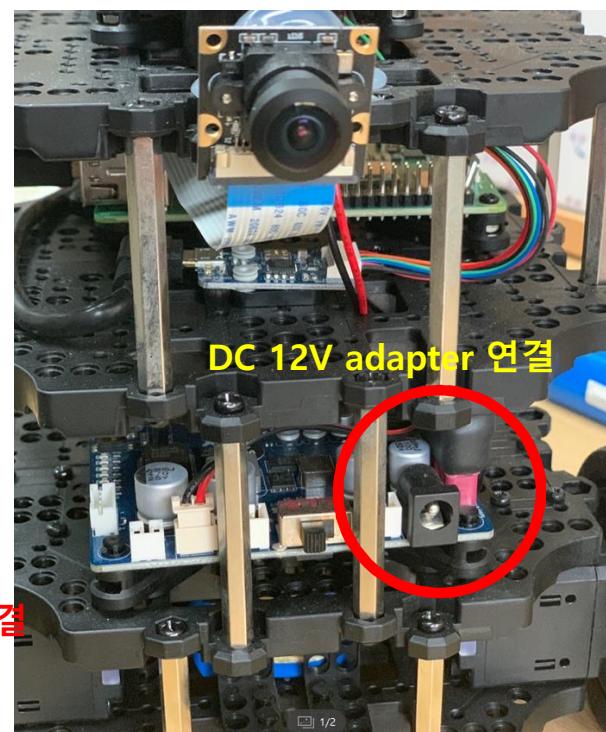
: [https://emanual.robotis.com/docs/en/platform/turtlebot3/hardware\\_setup/#hardware-assembly](https://emanual.robotis.com/docs/en/platform/turtlebot3/hardware_setup/#hardware-assembly) (조립 pdf)



전원 인가 및 충전 방법



90/116



Battery 사전에 충전하기  
(미충전 경우 Buzzer 소리 발생)  
→ 방치 시 완전 방전됨



## 4. ROS 개발환경 구축

실험 환경 2 : 만일 TurtleBot3 burger 조립되어 있다면,

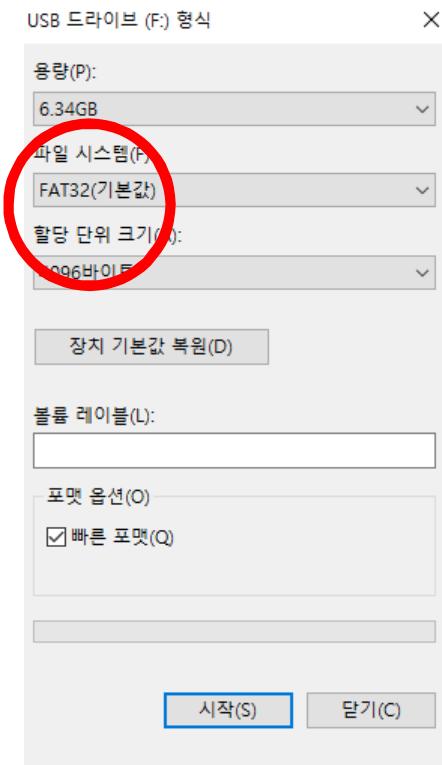
- ① TurtleBot3 burger의 SBC에서 MicroSD 카드 빼기



- ② MicroSD를 USB socket에 넣기



- ③ 강의용 PC에서 Format 진행하기



# 4. ROS 개발환경 구축

Ref. [https://emanual.robotis.com/docs/en/platform/turtlebot3/sbc\\_setup/#sbc-setup](https://emanual.robotis.com/docs/en/platform/turtlebot3/sbc_setup/#sbc-setup)

## 4-2-4. Install ROS SBC(Single Board Computer) - Raspberry Pi 3 B+에서 작업하기

### (1) Raspbian & ROS Kinetic 다운받기

: TurtleBot3 burger에 조립된 Raspberry Pi 3 B+에 Linux 계열의 Raspbian OS와 ROS 를 설치

: Raspbian 파일을 [www.raspberrypi.org/software](http://www.raspberrypi.org/software) 에서 받을 수 있지만

→ robotis에서 제공한 raspbian은 ROS package 포함되어 효과적이며 호환성으로 인해

→ [https://emanual.robotis.com/docs/en/platform/turtlebot3/sbc\\_setup/#sbc-setup](https://emanual.robotis.com/docs/en/platform/turtlebot3/sbc_setup/#sbc-setup)

SBC(Single Board Computer)  
: Raspberry Pi 3 B+



3. Quick Start Guide

3. 1. PC Setup

**3. 2. SBC Setup**

3. 3. OpenCR Setup

3. 4. Hardware Assembly

3. 5. Bringup

3. 6. Basic Operation

4. SLAM

5. Navigation

6. Simulation

7. Manipulation

8. Autonomous Driving

9. Machine Learning

10. Examples

11. Friends(Locomotion)

12. Learn

13. More Info

14. FAQ

**3. 2. 2. Download TurtleBot3 SBC Image**

Download the correct image file for your hardware and ROS version.  
Kinetic version images use Raspberry Pi OS(Raspbian OS).

**Download** **Raspberry Pi 3B+** ROS Kinetic image

SHA256 : eb8173f3727db08087990b2c4e2bb211e70bd54644644834771fc8b971856b97

The recovery image files can be modified without a prior notice.

Raspberry Pi 4 does not support Ubuntu 16.04 nor Debian Jessie, therefore, ROS Kinetic is not supported.

**3. 2. 3. Unzip the downloaded image file**

Extract the **.img** file and save it in the local disk.

**3. 2. 4. Burn the image file**

You can use various image burning tools.  
For example, **Raspberry Pi Imager** or Linux **Disk Utility** can be used.  
Choose your preferred tool to burn the image to microSD.

**3. 2. 4. 1. Raspberry Pi Imager**

Please refer to [this article](#) to find more information about Raspberry Pi Imager.

**Download** Raspberry Pi Imager from [raspberrypi.org](http://raspberrypi.org)

① windows 기반의 개인 PC에서 Download  
→ **(tb3\_rpi3b+\_kinetic\_20220513.zip)**

② Download 파일을 압축을 풀면 .img 생성  
→ **(tb3\_rpi3b+\_kinetic\_20220513.img)**

③ microSD에 img 굽기 위해 tool 준비



개인 PC에  
삽입하기

## 4. ROS 개발환경 구축

3. Quick Start Guide
3.1. PC Setup
<b>3.2. SBC Setup</b>
3.3. OpenCR Setup
3.4. Hardware Assembly
3.5. Bringup
3.6. Basic Operation
4. SLAM
5. Navigation
6. Simulation
7. Manipulation
8. Autonomous Driving
9. Machine Learning
10. Examples
11. Friends(Locomotion)
12. Learn
13. More Info

3. 2. 4. Burn the image file  
You can use various image burning tools.  
For example, [Raspberry Pi Imager](#) or Linux [Disk Utility](#) can be used.  
Choose your preferred tool to burn the image to microSD.

3. 2. 4. 1. Raspberry Pi Imager  
Please refer to this article to find more information about Raspberry Pi Imager.



microSD에 img 굽기 위해 tool 준비

④ click



⑤ imager download  
(imager\_1.7.3.exe)

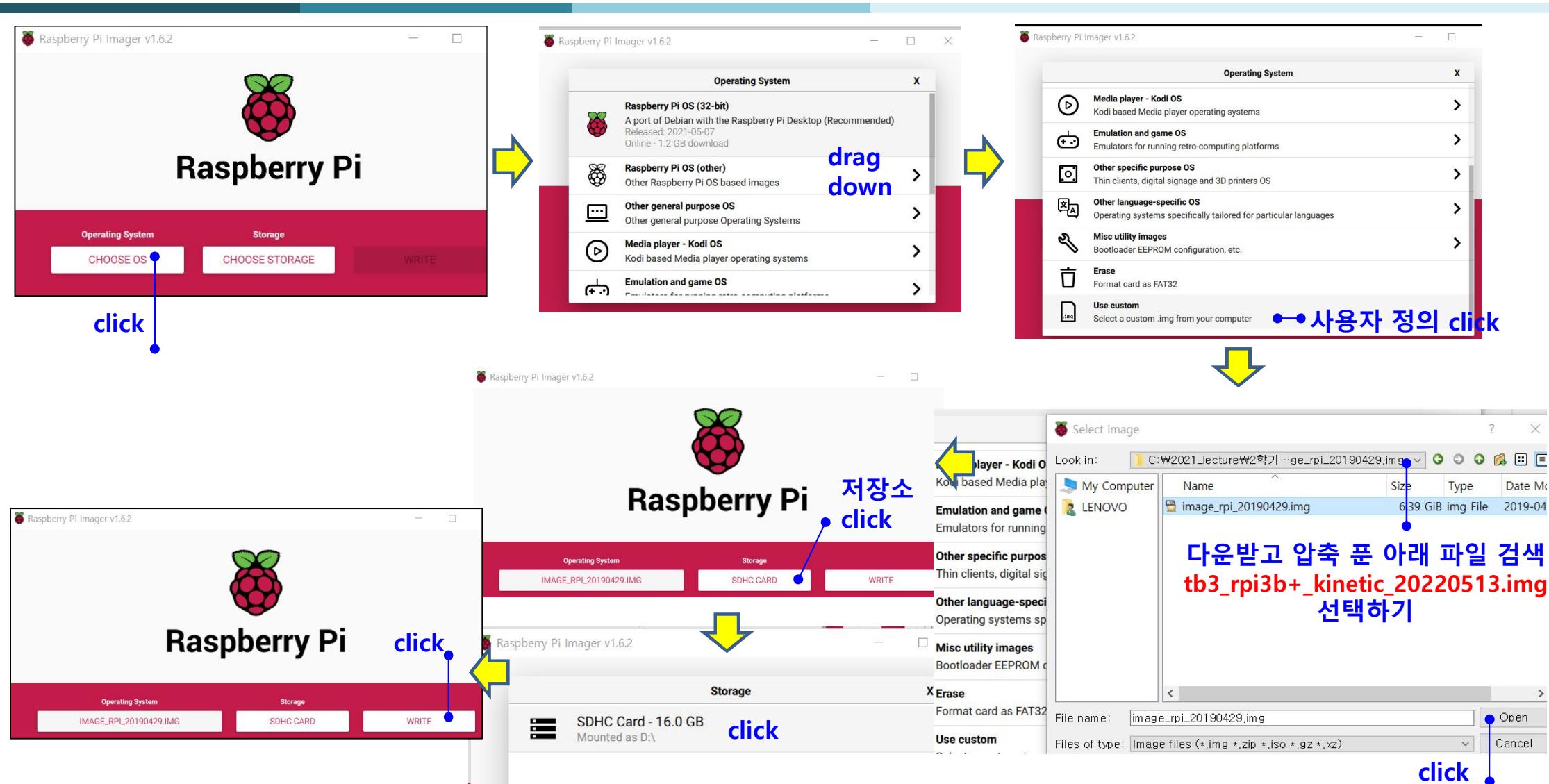
⑥ 안내 youtube 동영상을 보면서  
microSD에 img 굽기



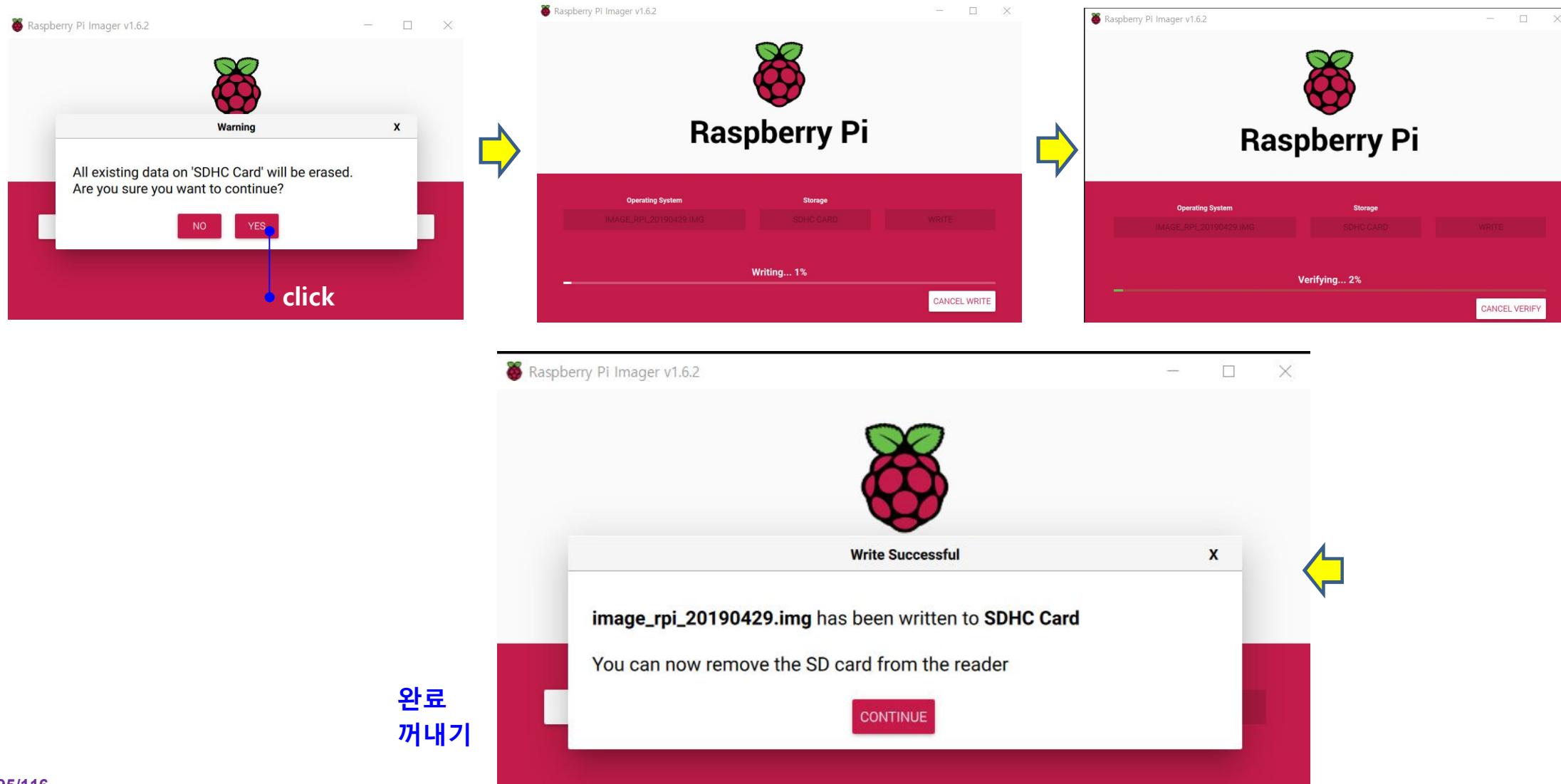
→ imager\_1.7.3.exe 관리자 권한으로 실행



## 4. ROS 개발환경 구축



## 4. ROS 개발환경 구축



## 4. ROS 개발환경 구축

실험 환경 1 : 만일 TurtleBot3 burger 조립되어 있지 않다면, 97 페이지(다음 페이지)로  
실험 환경 2 : 만일 TurtleBot3 burger 조립되어 있다면,

### 4-2-4. Install ROS SBC(Single Board Computer) - Raspberry Pi 3 B+에서 작업하기

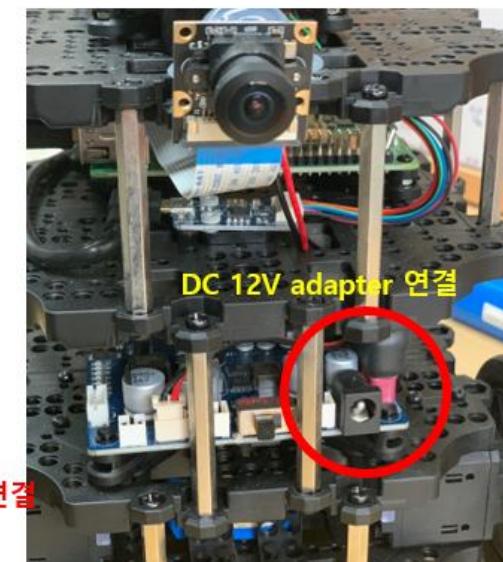
#### (2) Boot Up the Raspberry Pi

: 이미지 파일 저장이 완료되면 MicroSD를 Raspberry Pi 뒷면의 메모리 카드 슬롯에 삽입



: Raspberry Pi에 모니터, 키보드, 마우스 연결(다음 페이지 그림) 후  
아래 그림과 같이 전원 투입

전원 인가 및 충전 방법



## 4. ROS 개발환경 구축

실험 환경 1 : 만일 TurtleBot3 burger 조립되어 있지 않다면,

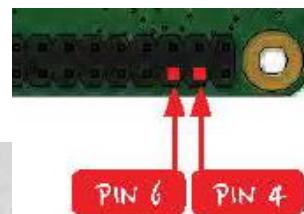
### 4-2-4. Install ROS SBC(Single Board Computer) - Raspberry Pi 3 B+에서 작업하기

#### (2) Boot Up the Raspberry Pi

: 이미지 파일 저장이 완료되면 마이크로 SD를 Raspberry Pi 뒷면의 메모리 카드 슬롯에 삽입



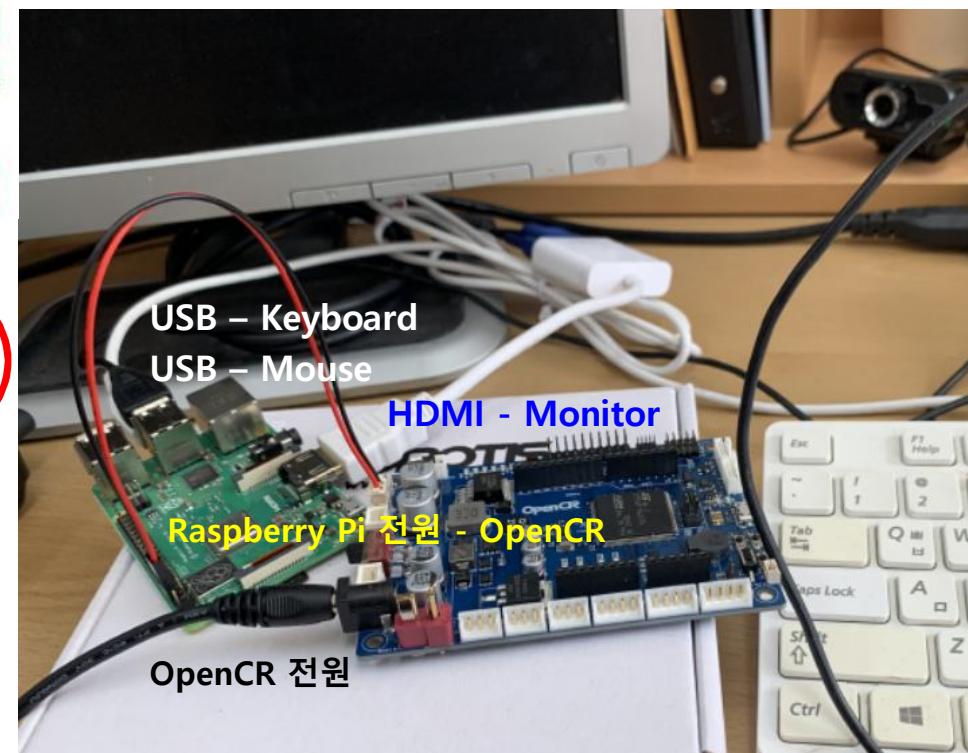
: 아래 그림과 같이 결선 후 전원 투입



Raspberry Pi 전원  
: PIN4 +(RED) ,  
PIN6 -(BLACK)



: CPU속도, RAM 속도, Ethernet 속도

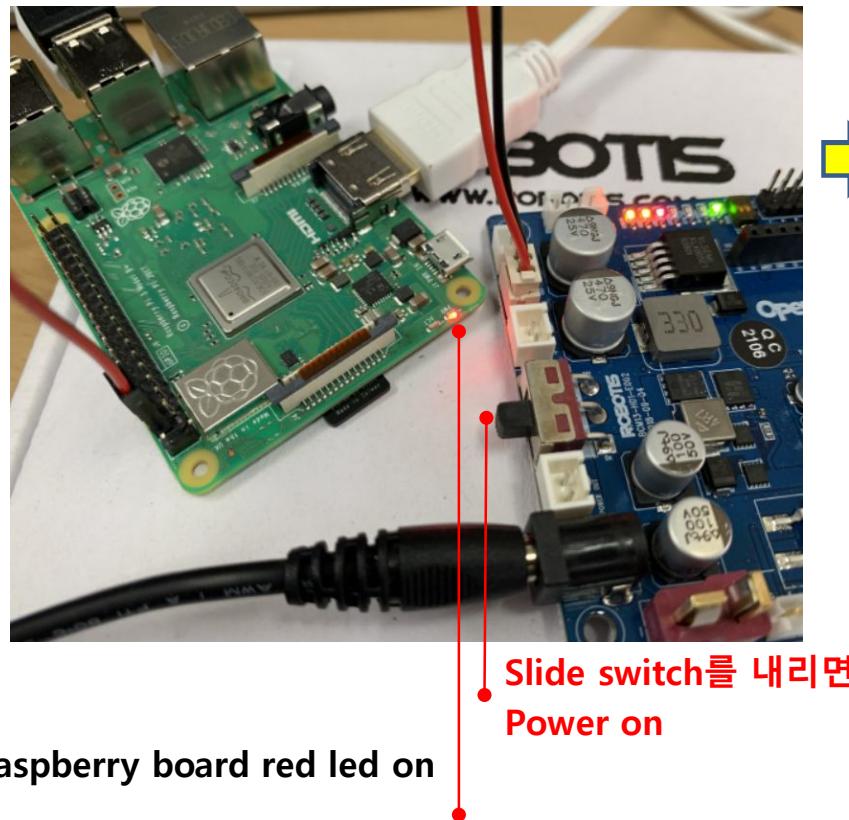


## 4. ROS 개발환경 구축

### 4-2-4. Install ROS SBC(Single Board Computer) - Raspberry Pi 3 B+에서 작업하기

#### (2) Boot Up the Raspberry Pi

: 전원인가 후, Raspbian OS boot up

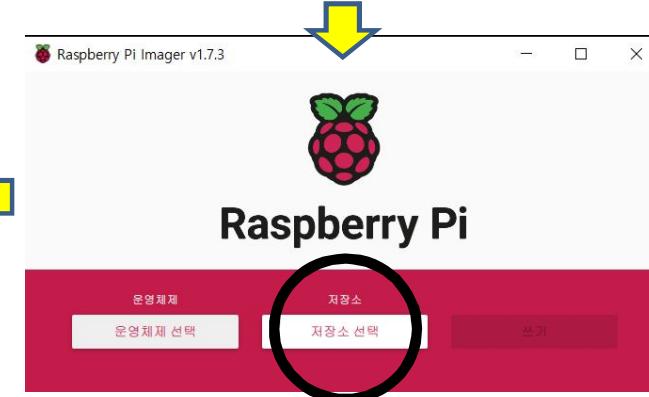
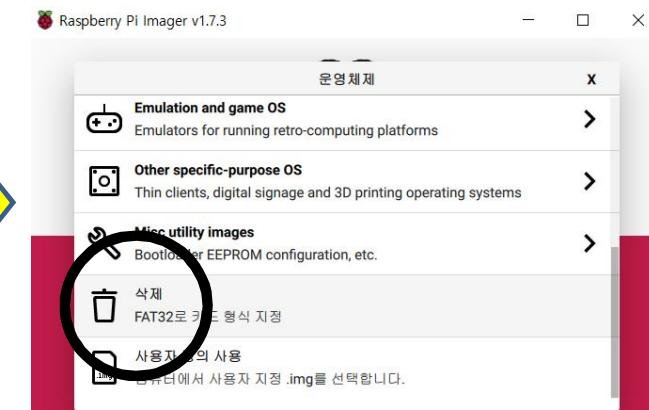
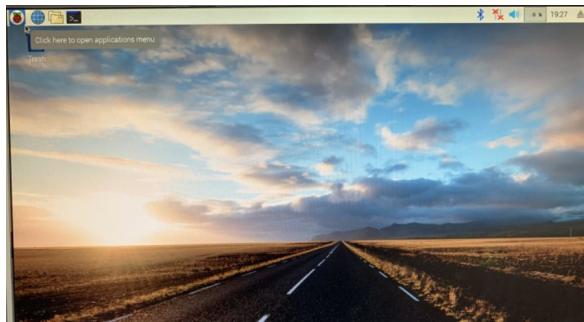


## 4. ROS 개발환경 구축

### 4-2-4. Install ROS SBC(Single Board Computer) - Raspberry Pi 3 B+에서 작업하기

#### (2) 문제 해결

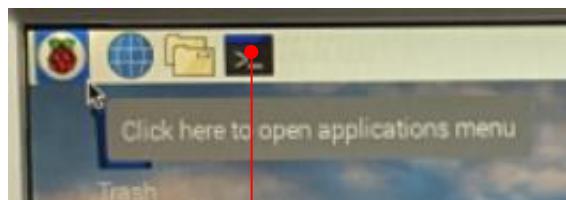
: 만일 Raspbian OS 가 Booting되지 않는다면, MicroSD를 imager\_1.7.3.exe 에서 지우고 다시 baking.



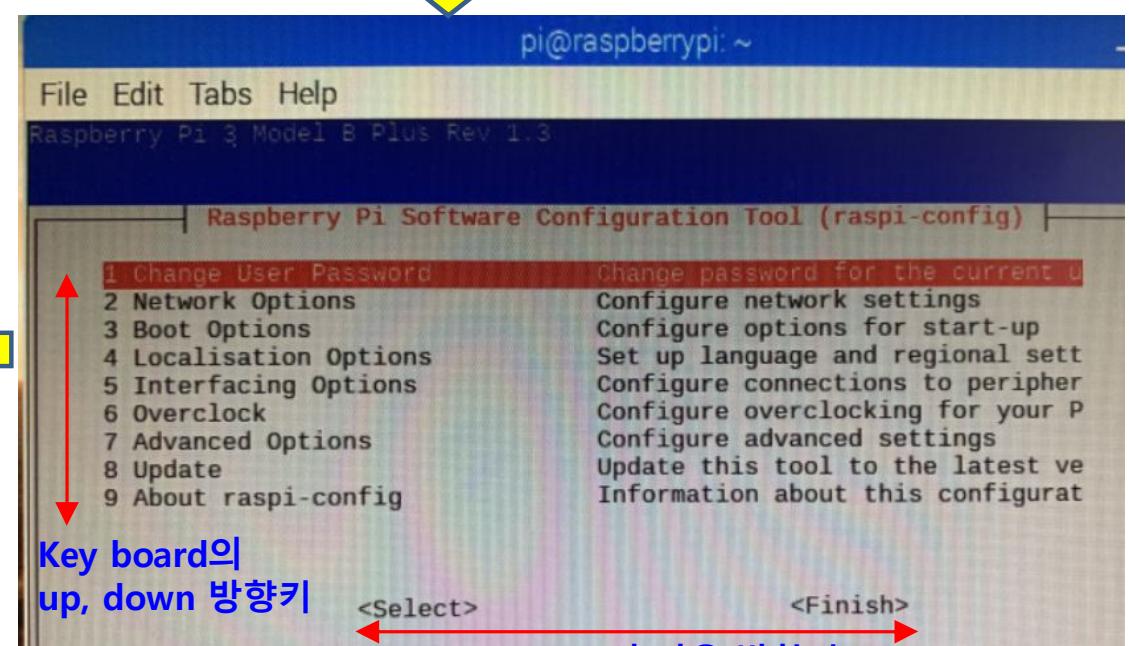
삭제하고  
다시 Baking

## 4. ROS 개발환경 구축

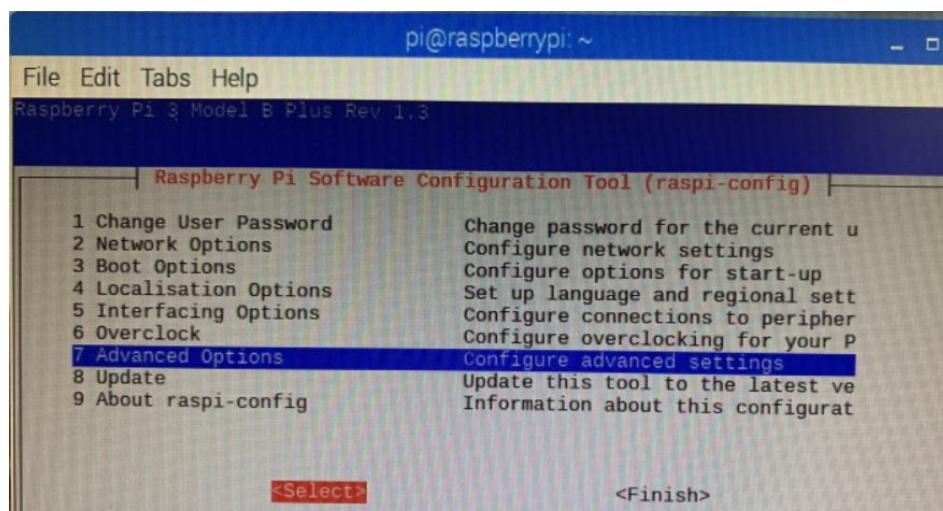
: Mouse를 사용하여 “Terminal” click



: sudo raspi-config 입력 pi@raspberrypi:~\$ sudo raspi-config



: 상하 key 사용하여 Advanced options 선택

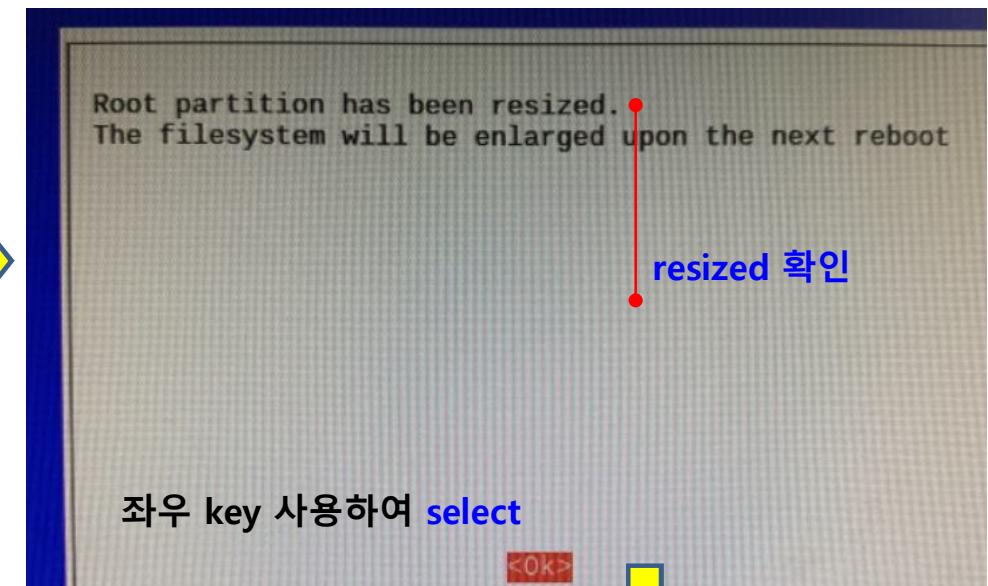
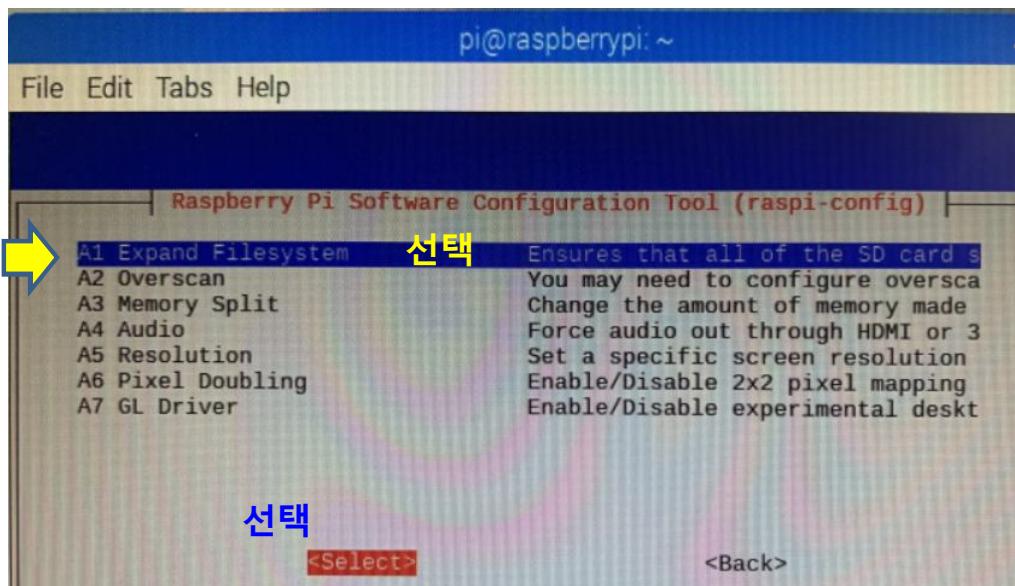


좌우 key 사용하여 select

Raspberry Pi Software Configuration Tool 화면

## 4. ROS 개발환경 구축

: Advanced options → Expand Filesystem 선택 : SDcard 전체를 사용하기 위한 옵션

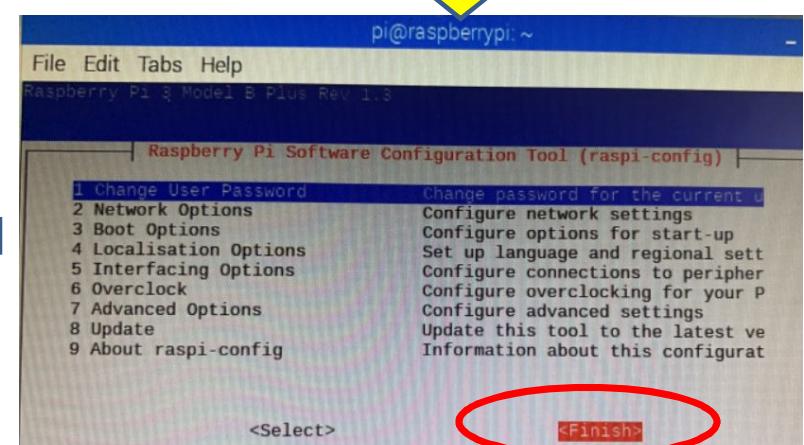
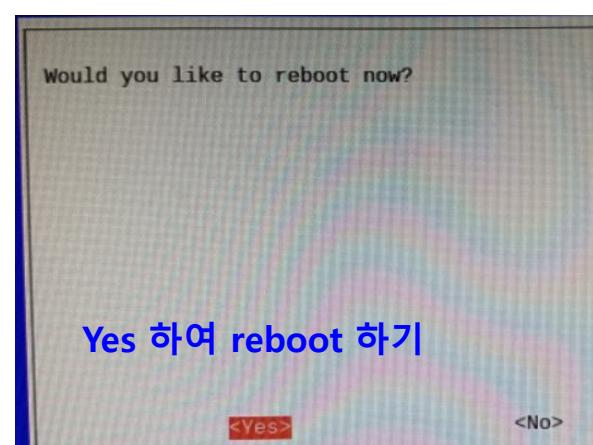


Boot mode에서

→ login ID와 Password 요구하면

login ID pi

Password raspberry



## 4. ROS 개발환경 구축

### 4-2-4. Install ROS SBC(Single Board Computer) - Raspberry Pi 3 B+에서 작업하기

#### (3) 네트워크 설정

[중요] Remote PC를 먼저 전원 on하고 IP 확인하기



- double click하여 무선네트워크 연결하기

(지난 시간 Remote PC와 같은 공유기 사용하기)

: Remote PC의 IP 주소 확인 (terminal 창 여는 방법 Ctrl + Alt + T)

~ \$ ifconfig

```
pi@raspberrypi: ~ $ ifconfig
eth0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
      ether b8:27:eb:d2:44:7e txqueuelen 1000  (Ethernet)
      RX packets 0 bytes 0 (0.0 B)
      RX errors 0 dropped 0 overruns 0 frame 0
      TX packets 0 bytes 0 (0.0 B)
      TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
      inet 127.0.0.1 netmask 255.0.0.0
      inet6 ::1 prefixlen 128 scopeid 0x10<host>
          loop txqueuelen 1000  (Local Loopback)
          RX packets 0 bytes 0 (0.0 B)
          RX errors 0 dropped 0 overruns 0 frame 0
          TX packets 0 bytes 0 (0.0 B)
          TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      inet 192.168.0.22 netmask 255.255.255.0 broadcast 192.168.0.255
      inet6 fe80::8f13:e4ef:2b0b:3f54 prefixlen 64 scopeid 0x20<link>
          ether b8:27:eb:87:11:2b txqueuelen 1000  (Ethernet)
          RX packets 177 bytes 40517 (39.5 KiB)
          RX errors 0 dropped 0 overruns 0 frame 0
          TX packets 131 bytes 18998 (18.5 KiB)
          TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

SBC ID 확인

## 4. ROS 개발환경 구축

### 4-2-4. Install ROS SBC(Single Board Computer) - Raspberry Pi 3 B+에서 작업하기

#### (3) 네트워크 설정

: NTP(Network Time Protocol)를 설정(시간 동기화)

~ \$ sudo apt-get install ntpdate

~ \$ sudo ntpdate ntp.ubuntu.com

```
pi@raspberrypi:~ $ sudo apt-get install ntpdate
Reading package lists... Done
Building dependency tree
Reading state information... Done
ntpdate is already the newest version (1:4.2.8p10+dfsg-3+deb9u2).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
pi@raspberrypi:~ $ sudo ntpdate ntp.ubuntu.com
28 Sep 21:26:20 ntpdate[1822]: adjust time server 91.189.91.157 offset -0.001470
sec
pi@raspberrypi:~ $
```

## 4. ROS 개발환경 구축

### 4-2-4. Install ROS SBC(Single Board Computer) - Raspberry Pi 3 B+에서 작업하기

#### (4) Network configuration for ROS

: SBC IP 확인후 Remote PC와 동일하게 IP를 변경 해야함

: 그러나 설정방법은 위는 Remote PC 주소를, 아래는 SBC IP 주소를 기재 → nano 편집기를 활용



```
~ $ nano ~/.bashrc pi@raspberrypi:~ $ nano ~/.bashrc
```

```
File Edit Tabs Help pi@raspberrypi:~  
GNU nano 2.7.4 File: /home/pi/.bashrc  
#!/bin/bash  
# ~/.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 S- in  
#   'i') ;;  
# esac  
  
# don't put duplicate lines or lines starting with space in the history  
# See bash(1) for more options  
HISTCONTROL=ignoreboth  
  
# append to the history file, don't overwrite it  
shopt -s histappend  
  
# for setting history length see HISTSIZE and HISTFILESIZE in bash(1)  
HISTSIZE=1000  
HISTFILESIZE=2000  
  
# check the window size after each command and, if necessary,  
# update the values of LINES and COLUMNS.  
shopt -s checkwinsize  
  
# If set, the pattern "##" used in a pathname expansion context will  
# match all files and zero or more directories and subdirectories.  
shopt -s globstar  
  
# make less more friendly for non-text input files, see lesspipe(1)  
#[ -x /usr/bin/lesspipe ] && eval "$SHELL=/bin/sh lesspipe"  
  
# set variable identifying the chroot you work in (used in the prompt below)  
if [ -z "${debian_chroot:-}" ] && [ -r /etc/debian_chroot ]; then  
    debian_chroot=$(cat /etc/debian_chroot)  
fi
```

```
alias eb='nano ~/.bashrc'  
alias sb='source ~/.bashrc'  
alias gs='git status'  
alias gp='git pull'  
alias cw='cd ~/catkin_ws'  
alias cs='cd ~/catkin_ws/src'  
alias cm='cd ~/catkin_ws && catkin_make'  
  
source /opt/ros/kinetic/setup.bash  
source ~/catkin_ws/devel/setup.bash  
  
export ROS_MASTER_URI=http://localhost:11311  
export ROS_HOSTNAME=localhost
```

Remote PC ID를  
먼저 확인후에



입력하기 :

- Remote PC ID
- SBC ID

```
export ROS_MASTER_URI=http://192.168.0.15:11311  
export ROS_HOSTNAME=192.168.0.22
```

## 4. ROS 개발환경 구축

### 4-2-4. Install ROS SBC(Single Board Computer) - Raspberry Pi 3 B+에서 작업하기

#### (4) Network configuration for ROS

: LIDAR 설정

- LDS-01로 변경하기



~ \$ nano ~/.bashrc



```
source /opt/ros/kinetic/setup.bash  
source ~/catkin_ws/devel/setup.bash  
  
export ROS_MASTER_URI=http://192.168.0.15:11311  
export ROS_HOSTNAME=192.168.0.16  
  
export TURTLEBOT3_MODEL=burger  
export LDS_MODEL=LDS-01
```

bashrc file의 맨 밑에  
LDS-02를 LDS-01로 변경



: 작업 완료되면 **CTRL + O** (저장) 후

[https://emanual.robotis.com/docs/en/platform/turtlebot3/sbc\\_setup/#sbc-setup](https://emanual.robotis.com/docs/en/platform/turtlebot3/sbc_setup/#sbc-setup)



위 메시지가 나오면 **ENTER** 그리고 **CTRL + X** (현재 페이지 나가기)



: 환경변수 실행(반영)

~ \$ source ~/.bashrc

```
pi@raspberrypi:~ $ source ~/.bashrc
```



★ : LDS -02를 위한 update 작업(2022년) – 먼저 네트워크 상태 반드시 확인

~ \$ sudo apt update

~ \$ sudo apt install libudev-dev

~ \$ cd ~/catkin\_ws/src

~ \$ git clone -b develop https://github.com/ROBOTIS-GIT/lid08\_driver.git

~ \$ cd ~/catkin\_ws/src/turtlebot3 && git pull

~ \$ rm -r turtlebot3\_description/ turtlebot3\_teleop/ turtlebot3\_navigation/ turtlebot3\_slam/ turtlebot3\_example/

~ \$ cd ~/catkin\_ws && catkin\_make

~ \$ source ~/.bashrc

LDS-01                    LDS-02



→ Shutdown후 다시 booting

## 4. ROS 개발환경 구축

### [추후]

: Remote PC에서 SBC를 원격으로 제어할 수 있음

→ Remote PC에 아래와 같이 SBC([Raspberry Pi 3 B+](#)) IP 주소를 입력하면 연결됨

( [ssh](#) : Secure Shell 명령어를 통한 원격 제어함)

→ Connection 비밀번호: [turtlebot](#)

\$ [ssh pi@192.168.xxx.xxx](#)      (The IP 192.168.xxx.xxx is your Raspberry Pi's IP or hostname)

## 4. ROS 개발환경 구축

### 4-2-5. OpenCR setup

: OpenCR(Control module for ROS)

- STM32F746 (ARM Cortex-M7 32bits RISC core)
- 216MHz frequency

: <https://emanual.robotis.com/docs/en/parts/controller/opencr10/>

The screenshot shows the 'OpenCR 1.0' documentation page. The left sidebar contains a navigation menu with the following items:

- Enter Search Terms
- OpenCR 1.0
- 1. Introduction
- 2. Specifications
- 3. Layout / Pin Map
- 4. Arduino IDE
- 5. Examples
- 6. Bootloader
- 7. Downloader
- 8. Reference

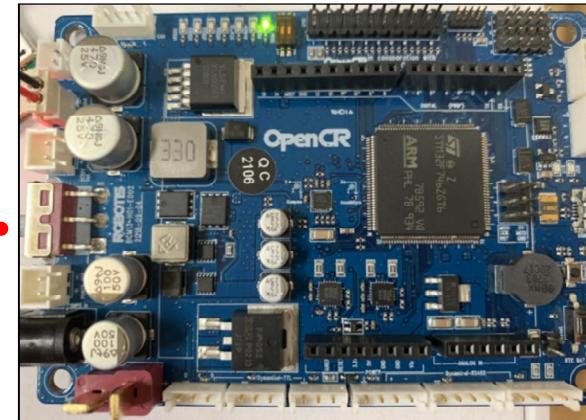
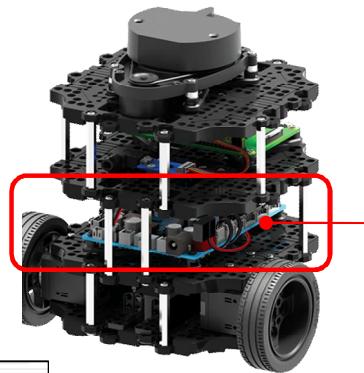
The main content area displays the following sections:

### 1. Introduction

OpenCR1.0 is developed for ROS embedded systems to provide completely open-source hardware. Everything about the board; Schematics, PCB Gerber, BOM and the firmware source code for the OP3 are free to distribute under open-source licenses for users and the ROS community. The STM32F7 series chip inside the OpenCR1.0 board is based on a very powerful ARM Cortex-M7 point unit. The development environment for OpenCR1.0 is wide open from Arduino IDE and Scratch for you traditional firmware development for the expert.

### 2. Specifications

Items	Specifications
Microcontroller	STM32F746ZGT6 / 32-bit ARM Cortex®-M7 with FPU (216MHz, 462DMIPS) Reference Manual, Datasheet
Sensors	(Discontinued) Gyroscope 3Axis, Accelerometer 3Axis, Magnetometer 3Axis (MPU9250) (New) 3-axis Gyroscope, 3-Axis Accelerometer, A Digital Motion Processor™ (ICM-20648)
Programmer	ARM Cortex 10pin JTAG/SWD connector USB Device Firmware Upgrade (DFU) Serial
Digital I/O	32 pins (L 14, R 18) *Arduino connectivity 5Pin OLLO x 4 GPIO x 18 pins PWM x 6 I2C x 1 SPI x 1
Analog INPUT	ADC Channels (Max 12bit) x 6 USB x 1 (Micro-B USB connector/USB 2.0/Host/Peripheral/OTG)



OpenCR에 대해 개인적으로 학습하기

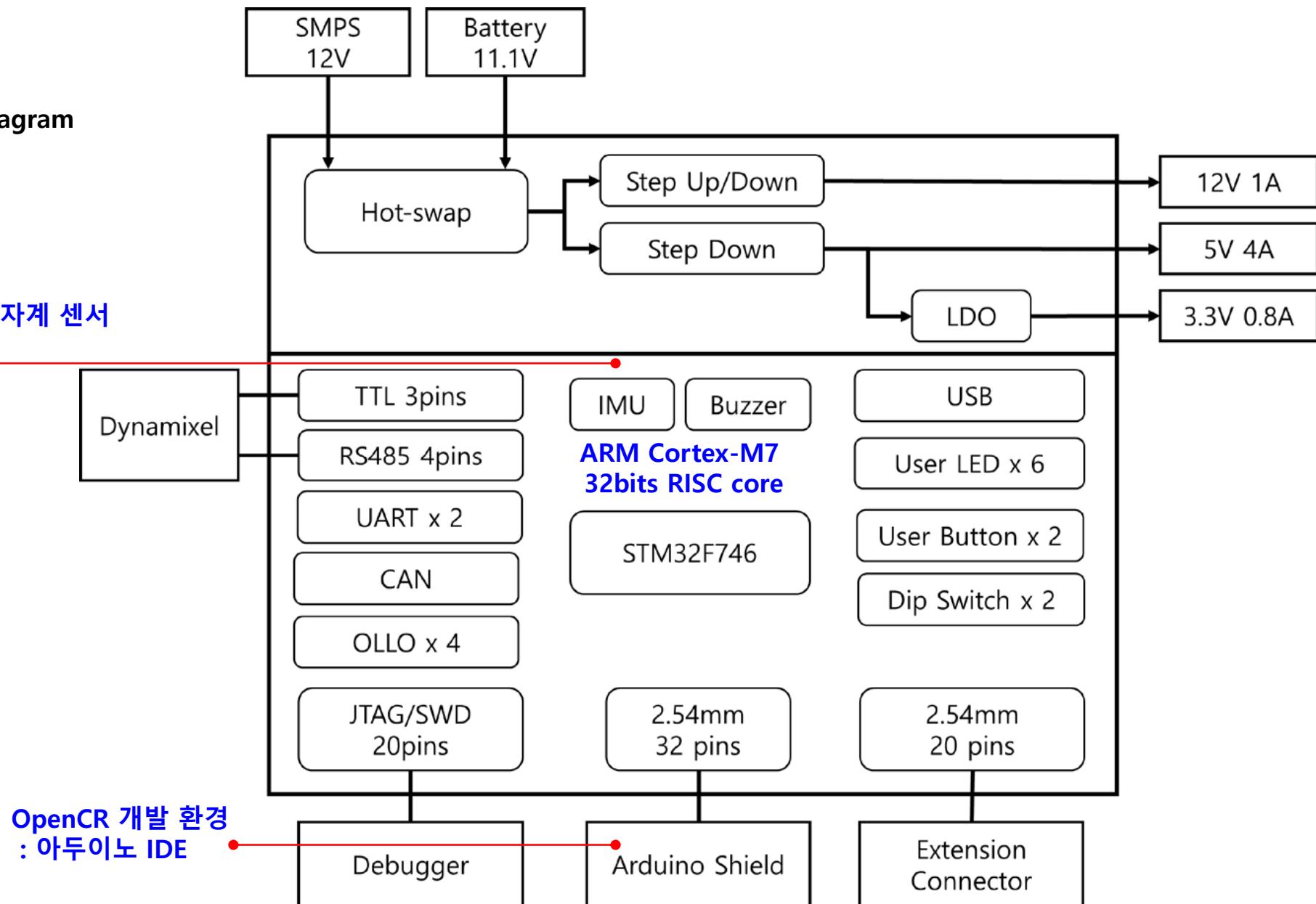
#### 4-2-5. OpenCR setup

: OpenCR block diagram

IMU

: 자이로/가속도/지자계 센서

: MPU925010



OpenCR 개발 환경  
: 아두이노 IDE

## 4. ROS 개발환경 구축

실험 환경 1 : 만일 TurtleBot3 burger 조립되어 있지 않다면,

### 4-2-5. OpenCR setup

: [https://emanual.robotis.com/docs/en/platform/turtlebot3/opencr\\_setup/#opencr-setup](https://emanual.robotis.com/docs/en/platform/turtlebot3/opencr_setup/#opencr-setup)

E-manual 3.3과 같이  
openCR 설정하기

The screenshot shows the '3. Quick Start Guide' section of the E-manual. A red arrow points from the 'E-manual 3.3과 같이 openCR 설정하기' text to the '3. 3. OpenCR Setup' link in the list.

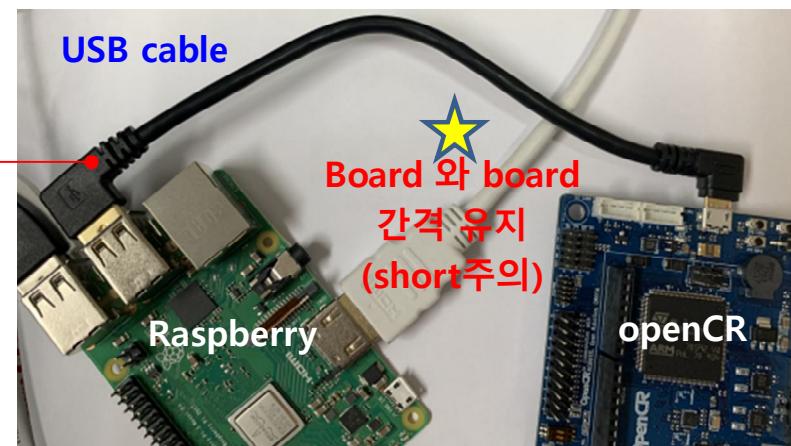
**3. 3. OpenCR Setup**

1. Connect the OpenCR to the Raspberry Pi using the micro USB cable.
2. Install required packages on the Raspberry Pi to upload the OpenCR firmware.  
\$ sudo dpkg --add-architecture armhf  
\$ sudo apt-get update  
\$ sudo apt-get install libcurl4-openssl-dev
3. Depending on the platform, use either burger or waffle for the OPENCR\_MODEL name.  
\$ export OPENCR\_PORT=/dev/ttyACM0

: openCR firmware 설정

① Connect the OpenCR to the Raspberry Pi  
using the USB cable

② Power on  
(openCR의 slide switch)  
Raspbian OS 화면에서...



전원이 인가되면  
USB 초록 LED 동작됨



## 4. ROS 개발환경 구축

: openCR firmware 설정

③ Raspbian에서 CTRL + ALT + T를 사용해 터미널 창 열기

Install packages on the Raspberry Pi to upload the OpenCR firmware

```
$ sudo dpkg --add-architecture armhf
```

```
$ sudo apt-get update
```

```
$ sudo apt-get install libc6:armhf
```

```
pi@raspberrypi:~ $ sudo apt-get install libc6:armhf
Reading package lists... Done
Building dependency tree
Reading state information... Done
libc6 is already the newest version (2.24-11+deb9u4).
0 upgraded, 0 newly installed, 0 to remove and 286 not upgraded.
pi@raspberrypi:~ $
```

dpkg

: debian package 관리 시스템의 소프트웨어. dpkg 명령어

armhf

: arm core Hardware FPU 구조의 debian 명칭

libc6:armhf

: C library compiler

```
pi@raspberrypi:~ $ sudo dpkg --add-architecture armhf
pi@raspberrypi:~ $ sudo apt-get update
Get:1 http://archive.raspberrypi.org/debian stretch InRelease [25.3 kB]
Get:2 http://archive.raspberrypi.org/debian stretch/main armhf Packages [192 kB]
Get:3 http://raspbian.raspberrypi.org/raspbian stretch InRelease [15.0 kB]
Get:4 http://archive.raspberrypi.org/debian stretch/ui armhf Packages [44.6 kB]
Get:5 http://raspbian.raspberrypi.org/raspbian stretch/main armhf Packages [11.7
MB]
Get:6 http://packages.ros.org/ros/ubuntu stretch InRelease [4,687 B]
Err:6 http://packages.ros.org/ros/ubuntu stretch InRelease
      The following signatures couldn't be verified because the public key is not av
ailable: NO_PUBKEY F42ED6FBAB17C654
Get:7 http://raspbian.raspberrypi.org/raspbian stretch/contrib armhf Packages [5
6.9 kB]
Get:8 http://raspbian.raspberrypi.org/raspbian stretch/non-free armhf Packages [98.9 kB]
Fetched 12.1 MB in 12s (987 kB/s)
Reading package lists... Done
W: An error occurred during the signature verification. The repository is not up
dated and the previous index files will be used. GPG error: http://packages.ros.
org/ros/ubuntu stretch InRelease: The following signatures couldn't be verified
because the public key is not available: NO_PUBKEY F42ED6FBAB17C654
W: Failed to fetch http://packages.ros.org/ros/ubuntu/dists/stretch/InRelease  T
he following signatures couldn't be verified because the public key is not avail
able: NO_PUBKEY F42ED6FBAB17C654
W: Some index files failed to download. They have been ignored, or old ones used
instead.
```

## 4. ROS 개발환경 구축

### ④ OPENCR\_MODEL name burger 로 변경하기

```
$ export OPENCR_PORT=/dev/ttyACM0
```

```
$ export OPENCR_MODEL=burger
```

```
$ rm -rf ./opencr_update.tar.bz2
```

```
pi@raspberrypi:~ $ export OPENCR_PORT=/dev/ttyACM0
pi@raspberrypi:~ $ export OPENCR_MODEL=burger
pi@raspberrypi:~ $ rm -rf ./opencr_update.tar.bz2
pi@raspberrypi:~ $
```

### ⑤ firmware와 loader 내려 받고, 압축

```
$ wget https://github.com/ROBOTIS-GIT/OpenCR-Binaries/raw/master/turtlebot3/ROS1/latest/opencr_update.tar.bz2
```

```
$ tar -xvf opencr_update.tar.bz2
```

```
pi@raspberrypi:~ $ wget https://github.com/ROBOTIS-GIT/OpenCR-Binaries/raw/master/turtlebot3/ROS1/latest/opencr_update.tar.bz2
--2021-09-29 02:32:19-- https://github.com/ROBOTIS-GIT/OpenCR-Binaries/raw/master/turtlebot3/ROS1/latest/opencr_update.tar.bz2
Resolving github.com (github.com)... 15.164.81.167
Connecting to github.com (github.com)|15.164.81.167|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://raw.githubusercontent.com/ROBOTIS-GIT/OpenCR-Binaries/master/turtlebot3/ROS1/latest/opencr_update.tar.bz2 [following]
--2021-09-29 02:32:19-- https://raw.githubusercontent.com/ROBOTIS-GIT/OpenCR-Binaries/master/turtlebot3/ROS1/latest/opencr_update.tar.bz2
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 185.199.110.8, 185.199.109.133, 185.199.108.133, ...
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|185.199.110.8|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 806597 (788K) [application/octet-stream]
Saving to: 'opencr_update.tar.bz2'

opencr_update.tar.bz2 100%[=====] 787.69K 3.36MB/s in 0.2s
2021-09-29 02:32:20 (3.36 MB/s) - 'opencr_update.tar.bz2' saved [806597/806597]
```

#### export

: shell 변수를 환경 변수로 저장

: OPENCR\_PORT= 는 이후의 폴더로 저장  
ttyACM0

: 리눅스에서 포트에 장치를 연결할 때,  
특히 Arduino-uno를 연결할 때 부여되는 이름

#### rm -rf

: directory 삭제. -rf 파일 권한에 관계없이 제거  
tar.bz2

: tar 여러 개의 파일을 압축하기.  
: bz2 압축풀기. tar.bz2 압축하거나 풀거나  
: -x, -v, -f 등 옵션

#### wget

: Web Get'의 약어로 웹 상의 파일을 다운로드

```
pi@raspberrypi:~ $ tar -xvf opencr_update.tar.bz2
```

```
opencr_update/
opencr_update/update.sh
opencr_update/released_1.2.6.txt
opencr_update/open_manipulator.opencr
opencr_update/waffle_turtlebot3_core_noetic.ino.bin
opencr_update/burger.opencr
opencr_update/opencr_ld_shell_arm
opencr_update/open_manipulator_turtlebot3_core_noetic.ino.bin
opencr_update/burger_noetic.opencr
opencr_update/burger_turtlebot3_core.ino.bin
opencr_update/open_manipulator_turtlebot3_core.ino.bin
opencr_update/burger_turtlebot3_core_noetic.ino.bin
opencr_update/opencr_ld_shell_x86
opencr_update/om_with_tb3.opencr
opencr_update/waffle.opencr
opencr_update/waffle_noetic.opencr
opencr_update/open_manipulator_noetic.opencr
opencr_update/waffle_turtlebot3_core.ino.bin
pi@raspberrypi:~ $
```

## 4. ROS 개발환경 구축

### ⑥ Upload firmware to the OpenCR

```
$ cd ./opencr_update  
$ ./update.sh $OPENCR_PORT $OPENCR_MODEL.opencr
```

```
pi@raspberrypi:~ $ cd ./opencr_update  
pi@raspberrypi:~/opencr_update $ ./update.sh $OPENCR_PORT $OPENCR_MODEL.opencr  
armv7l  
arm  
[ ]  
OpenCR Update Start..  
opencr_ld_shell ver 1.0.0  
opencr_ld_main  
[ ] file name : burger.opencr • 변경된 이름 확인  
[ ] file size : 178 KB  
[ ] fw_name : burger • 변경된 이름 확인  
[ ] fw_ver : 1.2.2  
[OK] Open port : /dev/ttyACM0  
[ ]  
[ ] Board Name : OpenCR R1.0  
[ ] Board Ver : 0x17020800  
[ ] Board Rev : 0x00000000  
[OK] flash_erase : 1.22s  
[OK] flash_write : 2.04s  
[OK] CRC Check : 12206B2 12206B2 , 0.006000 sec  
[OK] Download  
[OK] jump_to_fw • Firmware 완료가 되면 출력됨  
pi@raspberrypi:~/opencr_update $
```

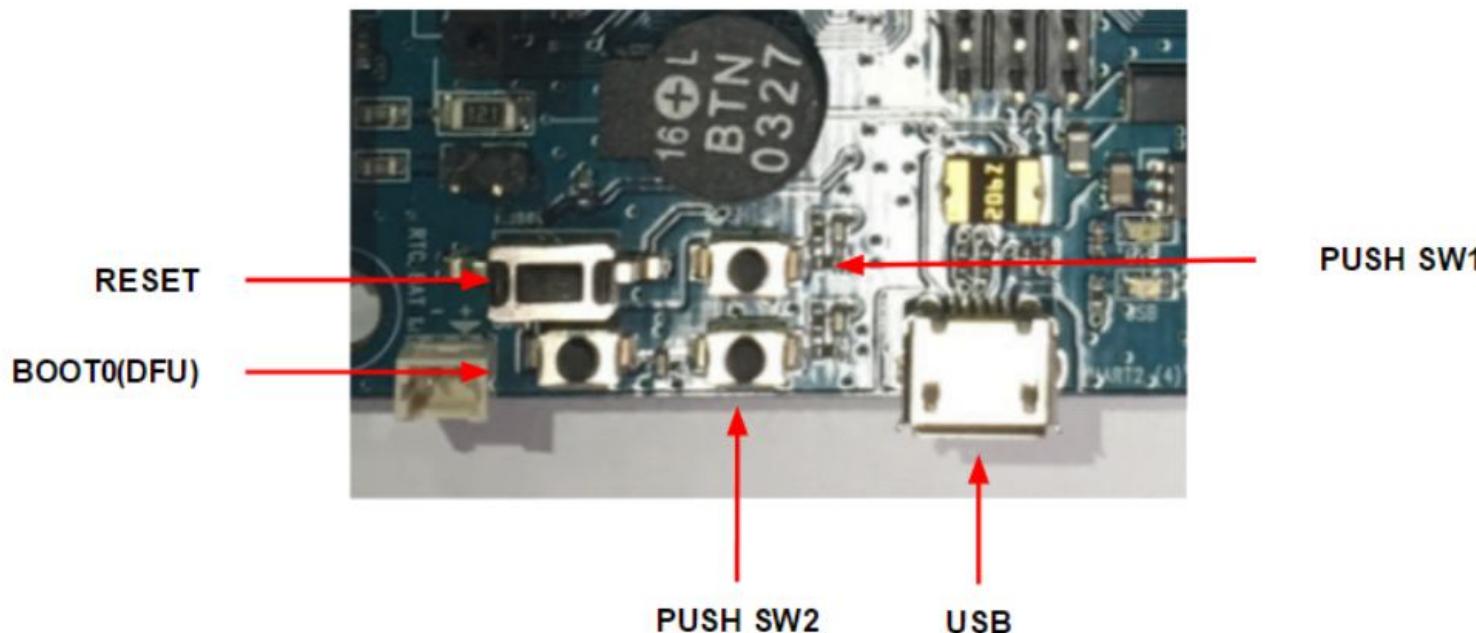
## 4. ROS 개발환경 구축

⑦ 만일 firmware 업로드 실패하면, (jump to fw message 출력 안되면)

: [https://emanual.robotis.com/docs/en/platform/turtlebot3/opencr\\_setup/#opencr-setup](https://emanual.robotis.com/docs/en/platform/turtlebot3/opencr_setup/#opencr-setup) 의 아래 실행하기

If firmware upload fails, try uploading with the recovery mode. Below sequence activates the recovery mode of OpenCR. Under the recovery mode, the **STATUS** led of OpenCR will blink periodically.

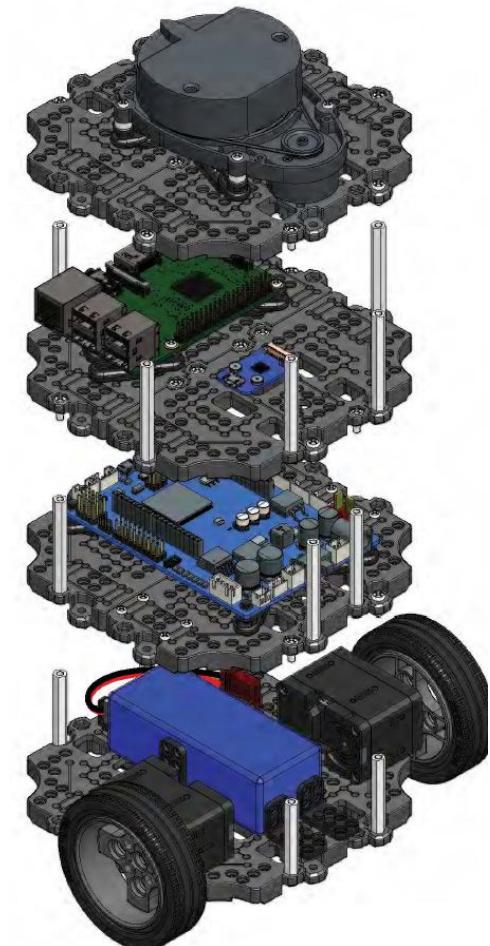
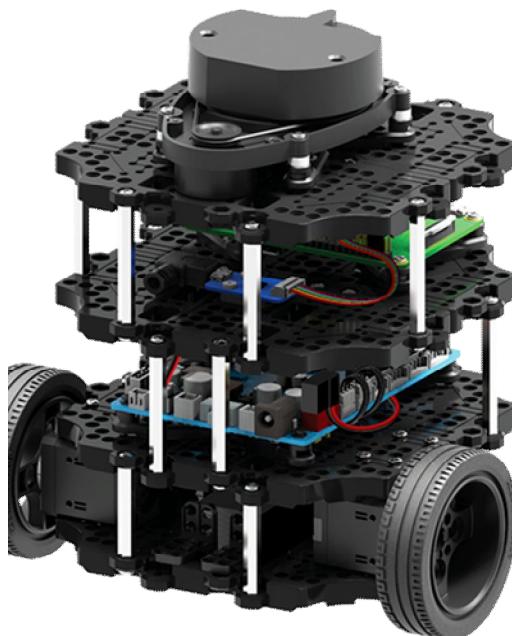
- Hold down the **PUSH SW2** button.
- Press the **Reset** button.
- Release the **Reset** button.
- Release the **PUSH SW2** button.



## 4. ROS 개발환경 구축

### ⑧ TurtleBot3 burger 조립하기

- : <https://youtu.be/rvm-m2ogrLA> 조립 동영상 참고하여 조립하기
- : 조립 pdf 파일 참고하여 조립하기
- : [https://emanual.robotis.com/docs/en/platform/turtlebot3/hardware\\_setup/#hardware-assembly](https://emanual.robotis.com/docs/en/platform/turtlebot3/hardware_setup/#hardware-assembly)



## 4. ROS 개발환경 구축

### 4-2-6. Online Competition on ROS Development Studio (RDS)

: <https://emanual.robotis.com/docs/en/platform/turtlebot3/challenges/#events>

#### (1) Online Competition using TurtleBot3

: are preparing an online competition on **ROS Development Studio (RDS)** with **TurtleBot3 AutoRace and Task Mission** using **TurtleBot3 and OpenManipulator**.  
: can participate free of charge in this online competition and learn about **SLAM, Navigatin, Autonomous driving, Manipulation** in a defined rule.

#### (2) TurtleBot3 AutoRace on RDS

#### (3) Task Mission using TurtleBot3 and OpenManipulator on RDS

➔ 대회 사진 공유



감사합니다