

1 서버

1-1 운영체제 및 유틸리티 SW 설치

2개의 USB 플래시 드라이브를 사용하는 걸 가정한다.

- Ubuntu 설치용 USB 플래시 드라이브
- ProjectView 설치용 USB 플래시 드라이브

OS 설치

Ubuntu

Ubuntu는 Server, Desktop, Core, Cloud 버전이 있다. 여기서는 설치 후 웹브라우저도 동작을 확인하는 등 사용성을 고려하여 Desktop 버전을 기준으로 설명한다.

Ubuntu Desktop 22.04.4 LTS (Jammy Jellyfish)

- 인터넷 사용 어려운 환경에서 사용할 것
- 설치 과정에서 한국어를 선택하면 설치 후에 한국어 사용 가능함

Ubuntu Desktop 24.04 LTS (The Noble Numbat)

- 인터넷 사용이 가능한 환경에서 사용할 것
- 설치 과정에서 한국어를 선택해도 설치가 제대로 되지 않음
- 설치 후 settings에서 언어 추가해서 다운로드 받아야 함

이 문서에서는 인터넷 접속이 불가능한 환경을 고려하여 22.04.4 LTS 버전을 기준으로 설명한다.

설치용 ISO 파일을 다운로드 받고 부팅 가능한 USB 플래시 드라이브 만드는 방법은 아래 내용을 참고한다.

- <https://ubuntu.com/download/desktop#how-to-install>

Ubuntu Desktop 24.04.1 LTS

USB 플래시 드라이브로 OS 설치

- 설치할 장비에 Ubuntu 설치용 USB 플래시 드라이브 삽입 후 재부팅
- 재부팅 시 CMOS 설정 진입
 - 장비 벤더별 CMOS 설정 방법 참고 (예: 삼성 PC는 부팅 후 F2 키 눌러서 진입)
 - 부팅 순서를 USB 플래시 드라이브가 우선하도록 설정 (예: 'UEFI: USB' 선택)
 - CMOS 설정 내용 저장 후 종료
- 재부팅 후 OS 설치 진행

설치 과정 중 주요 선택 사항

- Choose your language: 한국어
- 키보드 레이아웃 : 한국어

- Accessibility in Ubuntu: 기본 설정
- 설치 유형: Ubuntu 설치
- How would you like to install Ubuntu?: Interactive installation
- 어떤 앱을 설치하고 시작하시겠습니까?: Extended selection
- Install recommended proprietary software?: 모두 선택
- 사용자 계정 설정
 - 컴퓨터 이름: projectview (프로젝트에서 정의 가능)
 - 이름: projectview (프로젝트에서 정의 가능)
 - 암호: projectview (프로젝트에서 정의 가능)
- Ubuntu Pro 구독: skip for now

설치가 완료되면 Ubuntu 설치용 플래시 드라이브를 제거한 후 재부팅한다.

한글 입력기 설정

- Settings > Keyboard 실행
- Input Sources: Korean 삭제
- Input Sources: 한국어(Hangul) 우측 메뉴 버튼 클릭
- Preferences 메뉴 클릭
- iBusHangul Setup 창 > Hangul 탭 > Hangul Toggle Key 항목에서 기존 항목 삭제
- Add 버튼 클릭, 한영키 누르면 'Alt_R'이 등록됨
- Apply 클릭, OK 클릭

Input Sources에 '한국어', '한국어(Hangul)' 두 개가 보인다면 '한국어'를 삭제하고 '한국어(Hangul)'를 사용한다.

설치 관련 파일 복사

ProjectView USB 플래시 드라이브의 내용을 다운로드 폴더에 복사한다.

```
projectview@projectview:~/다운로드$ ls
certificate  chrome  docker  documents  projectview  utility  'vs code'
```

필수 유틸리티 SW 설치

Chrome

chrome 브라우저를 설치한다.

패키지 파일로 설치하는 방법

외부 인터넷 접속이 되지 않는 경우 패키지 파일로 설치한다. 패키지 파일을 확인한다.

```
projectview@projectview:~/다운로드$ cd chrome
projectview@projectview:~/다운로드/Linux/chrome$ ls
google-chrome-stable_current_amd64.deb
```

패키지 파일을 설치한다.

```
projectview@projectview:~/다운로드/Linux/chrome$ sudo dpkg -i *.deb
[sudo] projectview 암호:
Selecting previously unselected package google-chrome-stable.
(데이터베이스 읽는중 ...현재 147586개의 파일과 디렉터리가 설치되어 있습니다.)
Preparing to unpack google-chrome-stable_current_amd64.deb ...
Unpacking google-chrome-stable (125.0.6422.76-1) ...
google-chrome-stable (125.0.6422.76-1) 설정하는 중입니다 ...
update-alternatives: using /usr/bin/google-chrome-stable to provide /usr/bin/x-
www-browser (x-www-browser) in auto mode
update-alternatives: using /usr/bin/google-chrome-stable to provide
/usr/bin/gnome-www-browser (gnome-www-browser) in auto mode
update-alternatives: using /usr/bin/google-chrome-stable to provide
/usr/bin/google-chrome (google-chrome) in auto mode
Processing triggers for gnome-menus (3.36.0-1.1ubuntu3) ...
Processing triggers for desktop-file-utils (0.27-2build1) ...
Processing triggers for man-db (2.12.0-4build2) ...
projectview@projectview:~/다운로드/Linux/chrome$
```

Docker

docker 플랫폼을 설치한다.

패키지 파일로 설치하는 방법

외부 인터넷 접속이 되지 않는 경우 패키지 파일로 설치한다. Ubuntu 버전에 맞는 패키지 파일을 확인한다.

```
projectview@projectview:~/다운로드/Linux/docker$ ls
'22.04.4 LTS' '24.04 LTS'
projectview@projectview:~/다운로드/Linux/docker$ cd '22.04.4 LTS'
projectview@projectview:~/다운로드/Linux/docker/22.04.4 LTS$ ls
containerd.io_1.6.31-1_amd64.deb
docker-buildx-plugin_0.14.0-1~ubuntu.22.04~jammy_amd64.deb
docker-ce-cli_26.1.2-1~ubuntu.22.04~jammy_amd64.deb
docker-ce_26.1.2-1~ubuntu.22.04~jammy_amd64.deb
docker-compose-plugin_2.27.0-1~ubuntu.22.04~jammy_amd64.deb
```

패키지 파일을 설치한다.

```
projectview@projectview:~/다운로드/Linux/docker/22.04.4 LTS$ sudo dpkg -i *.deb
Selecting previously unselected package containerd.io.
(데이터베이스 읽는중 ...현재 163043개의 파일과 디렉터리가 설치되어 있습니다.)
Preparing to unpack containerd.io_1.6.31-1_amd64.deb ...
```

```

Unpacking containerd.io (1.6.31-1) ...
Selecting previously unselected package docker-buildx-plugin.
Preparing to unpack docker-buildx-plugin_0.14.0-1~ubuntu.22.04~jammy_amd64.deb ...
Unpacking docker-buildx-plugin (0.14.0-1~ubuntu.22.04~jammy) ...
Selecting previously unselected package docker-ce-cli.
Preparing to unpack docker-ce-cli_26.1.2-1~ubuntu.22.04~jammy_amd64.deb ...
Unpacking docker-ce-cli (5:26.1.2-1~ubuntu.22.04~jammy) ...
Selecting previously unselected package docker-ce.
Preparing to unpack docker-ce_26.1.2-1~ubuntu.22.04~jammy_amd64.deb ...
Unpacking docker-ce (5:26.1.2-1~ubuntu.22.04~jammy) ...
Selecting previously unselected package docker-compose-plugin.
Preparing to unpack docker-compose-plugin_2.27.0-1~ubuntu.22.04~jammy_amd64.deb
...
Unpacking docker-compose-plugin (2.27.0-1~ubuntu.22.04~jammy) ...
containerd.io (1.6.31-1) 설정하는 중입니다 ...
Created symlink /etc/systemd/system/multi-user.target.wants/containerd.service →
/lib/systemd/system/containerd.service.
docker-buildx-plugin (0.14.0-1~ubuntu.22.04~jammy) 설정하는 중입니다 ...
docker-ce-cli (5:26.1.2-1~ubuntu.22.04~jammy) 설정하는 중입니다 ...
docker-ce (5:26.1.2-1~ubuntu.22.04~jammy) 설정하는 중입니다 ...
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service →
/lib/systemd/system/docker.service.
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket →
/lib/systemd/system/docker.socket.
docker-compose-plugin (2.27.0-1~ubuntu.22.04~jammy) 설정하는 중입니다 ...
Processing triggers for man-db (2.10.2-1) ...
projectview@projectview:~/다운로드/Linux/docker/22.04.4 LTS$

```

apt repository에서 설치하는 방법

외부 인터넷 접속이 되는는 경우 apt로 설치한다.

다음 명령으로 도커 GPG 키를 등록한다.

```

sudo apt-get update
sudo apt-get install ca-certificates curl
sudo install -m 0755 -d /etc/apt/keyrings
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o
/etc/apt/keyrings/docker.asc
sudo chmod a+r /etc/apt/keyrings/docker.asc

```

등록 결과는 다음과 같다.

```

projectview@projectview:~$ sudo apt-get update
sudo apt-get install ca-certificates curl
sudo install -m 0755 -d /etc/apt/keyrings
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o
/etc/apt/keyrings/docker.asc
sudo chmod a+r /etc/apt/keyrings/docker.asc

```

```

받기:1 https://download.docker.com/linux/ubuntu noble InRelease [48.8 kB]
오류:1 https://download.docker.com/linux/ubuntu noble InRelease
  다음 서명들은 공개키가 없기 때문에 인증할 수 없습니다: NO_PUBKEY 7EA0A9C3F273FCD8
기존:2 https://dl.google.com/linux/chrome/deb stable InRelease
기존:3 http://security.ubuntu.com/ubuntu noble-security InRelease
기존:4 http://kr.archive.ubuntu.com/ubuntu noble InRelease
기존:5 http://kr.archive.ubuntu.com/ubuntu noble-updates InRelease
기존:6 http://kr.archive.ubuntu.com/ubuntu noble-backports InRelease
패키지 목록을 읽는 중입니다... 완료
W: GPG 오류: https://download.docker.com/linux/ubuntu noble InRelease: 다음 서명들
은 공개키가 없기 때문에 인증할 수 없습니다: NO_PUBKEY 7EA0A9C3F273FCD8
E: The repository 'https://download.docker.com/linux/ubuntu noble InRelease' is
not signed.
N: Updating from such a repository can't be done securely, and is therefore
disabled by default.
N: See apt-secure(8) manpage for repository creation and user configuration
details.
... 생략 ...
Unpacking curl (8.5.0-2ubuntu10.1) ...
curl (8.5.0-2ubuntu10.1) 설정하는 중입니다 ...
Processing triggers for man-db (2.12.0-4build2) ...
projectview@projectview:~$

```

다음 명령으로 repository를 등록한다.

```

echo \
  "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc]
https://download.docker.com/linux/ubuntu \
  $(. /etc/os-release && echo "$VERSION_CODENAME") stable" | \
  sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
sudo apt-get update

```

등록 결과는 아래와 같다.

```

projectview@projectview:~$ echo \
  "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc]
https://download.docker.com/linux/ubuntu \
  $(. /etc/os-release && echo "$VERSION_CODENAME") stable" | \
  sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
sudo apt-get update
받기:1 https://download.docker.com/linux/ubuntu noble InRelease [48.8 kB]
받기:2 https://download.docker.com/linux/ubuntu noble/stable amd64 Packages [6,952
B]
기존:3 https://dl.google.com/linux/chrome/deb stable InRelease
기존:4 http://security.ubuntu.com/ubuntu noble-security InRelease
기존:5 http://kr.archive.ubuntu.com/ubuntu noble InRelease
기존:6 http://kr.archive.ubuntu.com/ubuntu noble-updates InRelease
기존:7 http://kr.archive.ubuntu.com/ubuntu noble-backports InRelease
내려받기 55.8 k바이트, 소요시간 2초 (32.2 k바이트/초)

```

```
패키지 목록을 읽는 중입니다... 완료
projectview@projectview:~$
```

Mint 리눅스를 사용하는 경우 위 내용 중 아래 내용을 교체한다.

- VERSION_CODENAME를 UBUNTU_CODENAME로 교체

다음 명령으로 도커를 설치한다.

```
sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin
docker-compose-plugin
```

설치 결과는 다음과 같다..

```
projectview@projectview:~$ sudo apt-get install docker-ce docker-ce-cli
containerd.io docker-buildx-plugin docker-compose-plugin
패키지 목록을 읽는 중입니다... 완료
의존성 트리를 만드는 중입니다... 완료
상태 정보를 읽는 중입니다... 완료
다음의 추가 패키지가 설치될 것입니다 :
  docker-ce-rootless-extras git git-man liberror-perl libslirp0 pigz
  slirp4netns
제안하는 패키지:
  aufs-tools cgroupfs-mount | cgroup-lite git-daemon-run | git-daemon-sysvinit
  git-doc git-email git-gui gitk gitweb git-cvs git-mediawiki git-svn
다음 새 패키지를 설치할 것입니다:
  containerd.io docker-buildx-plugin docker-ce docker-ce-cli
  docker-ce-rootless-extras docker-compose-plugin git git-man liberror-perl
  libslirp0 pigz slirp4netns
0개 업그레이드, 12개 새로 설치, 0개 제거 및 14개 업그레이드 안 함.
126 M바이트 아카이브를 받아야 합니다.
이 작업 후 458 M바이트의 디스크 공간을 더 사용하게 됩니다.
계속 하시겠습니까? [Y/n] y
받기:1 https://download.docker.com/linux/ubuntu noble/stable amd64 containerd.io
amd64 1.6.32-1 [30.0 MB]
... 생략 ...
Processing triggers for libc-bin (2.39-0ubuntu8.1) ...
projectview@projectview:~$
```

인터넷 접속이 가능한 환경에서는 다음 명령으로 정상적으로 설치가 되었는지 확인할 수 있다.

```
sudo docker run hello-world
```

확인 결과는 다음과 같다.

```
projectview@projectview:~$ sudo docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
c1ec31eb5944: Pull complete
Digest: sha256:266b191e926f65542fa8daaec01a192c4d292bff79426f47300a046e1bc576fd
Status: Downloaded newer image for hello-world:latest
```

Hello from Docker!

This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:

1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
(amd64)
3. The Docker daemon created a new container from that image which runs the executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it to your terminal.

To try something more ambitious, you can run an Ubuntu container with:

```
$ docker run -it ubuntu bash
```

Share images, automate workflows, and more with a free Docker ID:

<https://hub.docker.com/>

For more examples and ideas, visit:

<https://docs.docker.com/get-started/>

```
projectview@projectview:~$
```

버전 확인

다음 명령으로 버전을 확인한다.

```
sudo docker version
```

확인 결과는 다음과 같다.

```
projectview@projectview:~$ sudo docker version
Client: Docker Engine - Community
Version:           26.1.3
API version:       1.45
Go version:        go1.21.10
Git commit:        b72abbb
Built:             Thu May 16 08:33:35 2024
OS/Arch:           linux/amd64
Context:           default
```

```

Server: Docker Engine - Community
Engine:
  Version:      26.1.3
  API version:  1.45 (minimum version 1.24)
  Go version:   go1.21.10
  Git commit:   8e96db1
  Built:        Thu May 16 08:33:35 2024
  OS/Arch:      linux/amd64
  Experimental: false
containerd:
  Version:      1.6.32
  GitCommit:    8b3b7ca2e5ce38e8f31a34f35b2b68ceb8470d89
runc:
  Version:      1.1.12
  GitCommit:    v1.1.12-0-g51d5e94
docker-init:
  Version:      0.19.0
  GitCommit:    de40ad0
projectview@projectview:~$

```

사용자 그룹 추가

docker 구동, 종료 시 sudo 명령 사용을 생략하기 위해 다음 명령으로 docker 그룹에 projectview 계정을 추가한다.

```
sudo usermod -aG docker $USER
```

실행 결과는 다음과 같다.

```
projectview@projectview:~$ sudo usermod -aG docker $USER
```

변경 사항을 반영하기 위해 다음 명령을 실행한다. 가상 머신을 사용하는 경우 가상 머신 재구동이 필요할 수 있다.

```
newgrp docker
```

실행 결과는 다음과 같다.

```
projectview@projectview:~$ newgrp docker
```

docker 그룹 추가 전에 docker 명령을 실행한 적이 있다면 아래와 같은 에러를 볼 수 있다.


```
WARNING: Error loading config file: /home/user/.docker/config.json -  
stat /home/user/.docker/config.json: permission denied
```

이때는 사용자 계정 아래의 .docker 디렉토리 전체를 삭제하거나 (재생성됨) 아래 명령으로 소유권과 권한을 변경한다.

```
projectview@projectview:~$ sudo chown "$USER":"$USER" /home/"$USER"/.docker -R  
projectview@projectview:~$ sudo chmod g+rwX "$HOME/.docker" -R
```

sudo 명령을 쓰지 않고 도커 구동이 가능한지 다음 명령으로 확인한다.

```
docker run hello-world
```

실행 결과는 다음과 같다.

```
projectview@projectview:~$ docker run hello-world
```

필수 유틸리티 설치

net-tools

IP 주소를 확인하기 위해 net-tools를 설치한다.

apt로 설치하는 방법

외부 인터넷 접속이 가능한 경우 apt로 설치한다.

```
sudo apt install net-tools
```

설치 결과는 아래와 같다.

```
projectview@projectview:~$ sudo apt install net-tools  
패키지 목록을 읽는 중입니다... 완료  
의존성 트리를 만드는 중입니다... 완료  
상태 정보를 읽는 중입니다... 완료  
다음 새 패키지를 설치할 것입니다:  
  net-tools  
0개 업그레이드, 1개 새로 설치, 0개 제거 및 14개 업그레이드 안 함.  
204 k바이트 아카이브를 받아야 합니다.
```

```

이 작업 후 811 k바이트의 디스크 공간을 더 사용하게 됩니다.
받기:1 http://kr.archive.ubuntu.com/ubuntu noble/main amd64 net-tools amd64 2.10-
0.1ubuntu4 [204 kB]
내려받기 204 k바이트, 소요시간 2초 (89.0 k바이트/초)
Selecting previously unselected package net-tools.
(데이터베이스 읽는중 ...현재 149050개의 파일과 디렉터리가 설치되어 있습니다.)
Preparing to unpack .../net-tools_2.10-0.1ubuntu4_amd64.deb ...
Unpacking net-tools (2.10-0.1ubuntu4) ...
net-tools (2.10-0.1ubuntu4) 설정하는 중입니다 ...
Processing triggers for man-db (2.12.0-4build2) ...
projectview@projectview:~$

```

패키지 파일로 설치하는 방법

외부 인터넷 접속이 불가능한 경우 패키지로 설치한다. 패키지 파일을 확인한다.

```

projectview@projectview:~/다운로드$ cd utility
projectview@projectview:~/다운로드/utility$ ls
net-tools_2.10-0.1_amd64.deb

```

패키지 파일을 설치한다.

```

projectview@projectview:~/다운로드/utility$ sudo dpkg -i *.deb
Selecting previously unselected package net-tools.
(데이터베이스 읽는중 ...현재 163275개의 파일과 디렉터리가 설치되어 있습니다.)
Preparing to unpack net-tools_2.10-0.1_amd64.deb ...
Unpacking net-tools (2.10-0.1) ...
net-tools (2.10-0.1) 설정하는 중입니다 ...
Processing triggers for man-db (2.10.2-1) ...
projectview@projectview:~/다운로드/utility$

```

선택 유틸리티 SW 설치

Visual Studio Code

GUI를 쓸 수 있는 Ubuntu Desktop인 경우 편집기로 Visual Studio Code를 사용할 수 있다.

패키지 파일로 설치하는 방법

패키지 파일을 확인한다.

```

projectview@projectview:~$ cd 다운로드/vs\ code/
projectview@projectview:~/다운로드/Linux/vs code$ ls

```

```
code_1.89.1-1715060508_amd64.deb
shd101wyy.markdown-preview-enhanced-0.8.13.vsix
projectview@projectview:~/다운로드/Linux/vs code$
```

다음 명령으로 패키지를 설치한다.

```
sudo dpkg -i *.deb
```

설치 결과는 다음과 같다.

```
projectview@projectview:~/다운로드/Linux/vs code$ sudo dpkg -i *.deb
Selecting previously unselected package code.
(데이터베이스 읽는중 ...현재 149098개의 파일과 디렉터리가 설치되어 있습니다.)
Preparing to unpack code_1.89.1-1715060508_amd64.deb ...
Unpacking code (1.89.1-1715060508) ...
code (1.89.1-1715060508) 설정하는 중입니다 ...
Processing triggers for gnome-menus (3.36.0-1.1ubuntu3) ...
Processing triggers for desktop-file-utils (0.27-2build1) ...
Processing triggers for shared-mime-info (2.4-4) ...
projectview@projectview:~/다운로드/Linux/vs code$
```

Visual Studio Code Extensions

필요에 따라 Visual Studio Code Extensions를 설치한다.

- VS Code 실행
- VS Code 왼쪽 메뉴 '확장(Extensions)' 메뉴 선택
- 확장 팬 우측 '...' 메뉴 선택
- 메뉴에서 'VSIX에서 설치...(Install from VSIX...)' 선택
- 아래 vsix 파일 선택

Markdown Preview

위에서 아래의 마크다운 뷰어를 선택한다.

- hd101wyy.markdown-preview-enhanced-0.8.13.vsix