

Docker 실습 1 - 설치

1. Docker 설치

- 1) Set up the repository
- 2) Install Docker Engine
- 3) 정상 설치 확인

2. Docker 권한 설정

1. Docker 설치

- 공식 문서
 - <https://docs.docker.com/engine/install/ubuntu/>
 - 여러가지 install 방법 중, **Install using the repository** 방법으로 진행하겠습니다.
- 앞으로 어떤 오픈소스를 사용하더라도, 블로그가 아닌 공식 문서를 참고하시는 습관을 들이는 것을 추천 드리겠습니다.

1) Set up the repository

- 먼저 apt 라는 패키지 매니저를 업데이트합니다.

```
sudo apt-get update
```

- 그리고, docker 의 prerequisite package 들을 설치합니다.

```
sudo apt-get install \
    apt-transport-https \
    ca-certificates \
    curl \
    gnupg \
    lsb-release
```

- Docker 의 GPG key 를 추가합니다.

```
curl -fsSL https://download.docker.com/linux/ubuntu/gpg
| sudo gpg --dearmor -o /usr/share/keyrings/docker-archi
ve-keyring.gpg
```

- 그 다음 stable 버전의 repository 를 바라보도록 설정합니다.

```
echo \
  "deb [arch=amd64 signed-by=/usr/share/keyrings/docker-
archive-keyring.gpg] https://download.docker.com/linux/u
buntu \
  $(lsb_release -cs) stable" | sudo tee /etc/apt/source
s.list.d/docker.list > /dev/null
```

2) Install Docker Engine

- Docker 엔진의 최신 버전을 설치합니다.

```
sudo apt-get update
sudo apt-get install docker-ce docker-ce-cli containerd.
io
```

3) 정상 설치 확인

- 서비스 상태 확인

```
sudo systemctl status docker.service
sudo systemctl status containerd.service
```

초록색으로 active (running) 이라고 표시되는지 확인

- docker container 를 실행시켜, 정상적으로 설치되었는지 확인합니다.

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

- 다음과 같이 출력 된다면, 정상적으로 설치가 된 것을 확인할 수 있습니다

```
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
b8dfde127a29: Pull complete
Digest: sha256:0fe98d7debd9049c50b597ef1f85b7c1e8cc81f59
c8d623fcb2250e8bec85b38
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/>

2. Docker 권한 설정

- 현재는 모든 docker 관련 작업이 root 유저에게만 권한이 있기 때문에, docker 관련 명령을 수행하려면 `sudo` 를 앞에 붙여주어야만 가능합니다.
- 예를 들면,
 - `docker ps` 를 수행하면 다음과 같이 Permission denied 라는 메시지가 출력됩니다.

```
Got permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Get http://%2Fvar%2Frun%2Fdocker.sock/v1.40/containers/json?all=1: dial unix /var/run/docker.sock: connect: permission denied
```

- 따라서, root 유저가 아닌 host 의 기본 유저에게도 권한을 주기 위해 다음과 같은 명령을

새로 띄운 터미널에서 수행해주셔야 합니다.

```
sudo usermod -a -G docker $USER  
sudo service docker restart
```

- 그 다음, **VM 을 로그아웃 한 다음에 다시 로그인**하면 다음과 같이 정상적으로 출력되는 것을 확인하실 수 있습니다.

| CONTAINER ID | IMAGE | COMMAND | CREATED | STATUS | P |
|--------------|-------|---------|---------|--------|---|
| ORTS | NAMES | | | | |