



Docker 기반으로 스트레스없이 PaaS 인프라에 배포하기

개발머신에 Docker 설치하기

여러분의 파이썬/장고 페이스메이커가 되겠습니다.

Docker 딱 하나. 설치

각 OS에 맞게 Docker를 설치하시고, 버전을 확인해보세요.

“Error response from daemon: Bad response from Docker engine” 와 같은 메시지가 출력될 경우, docker daemon이 실행 중이 아니거나, root 권한이 필요하실 수 있습니다. linux/mac 이라면 `sudo docker --version` 명령을 입력해보세요.

다운받을 때 docker.com 사이트에서 인증을 요구하니, 회원가입/로그인을 미리 해주세요.

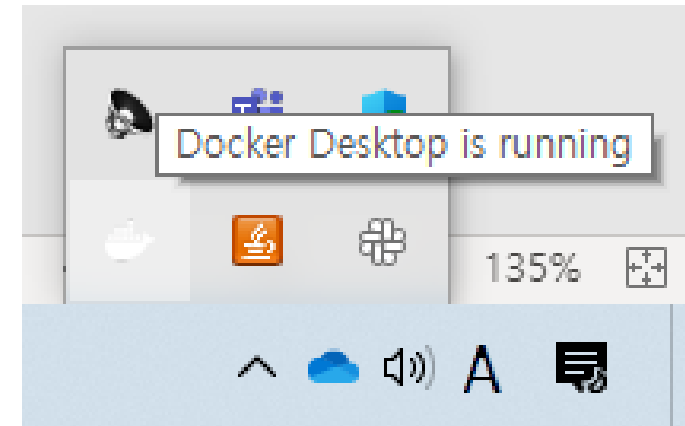
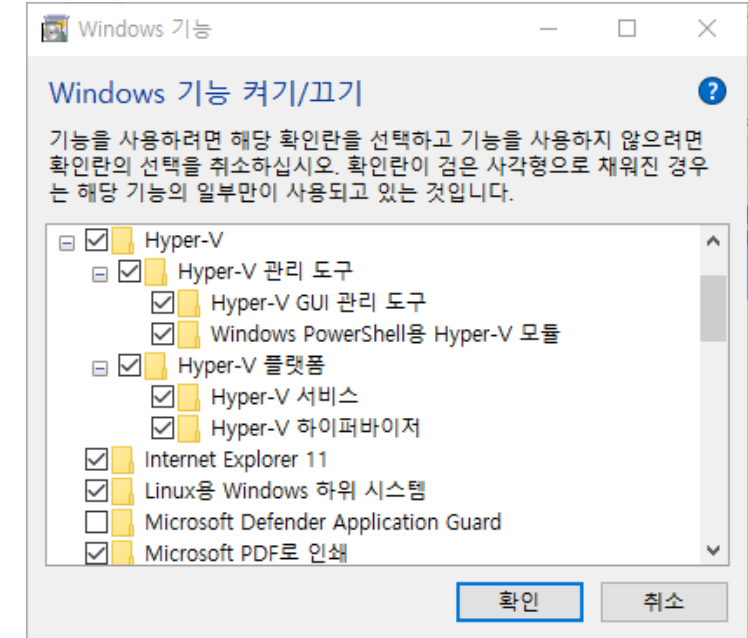
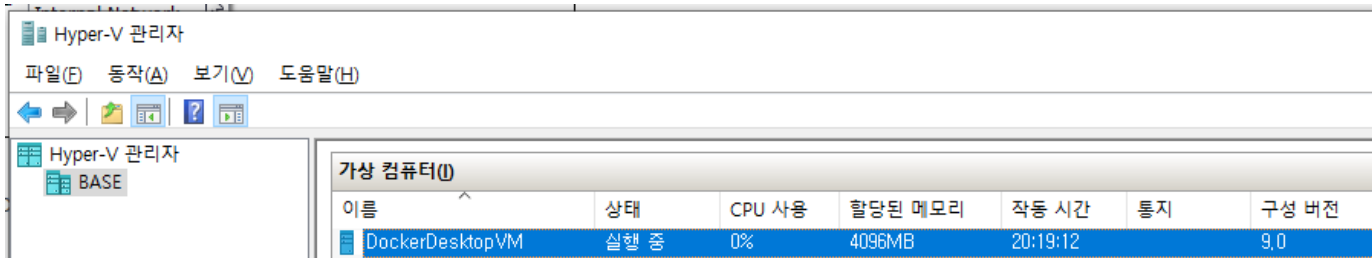


윈도우에서의 설치 - 윈도우 10 Pro 이상

1. Bios에서 가상화 지원 필요/켜기
2. Hyper-V 설치
3. Docker Desktop for Windows CE 설치

다운로드 위해, Docker Hub 가입 필요

참고) Chocolatey에서 docker-desktop 패키지 지원



<https://hub.docker.com/editions/community/docker-ce-desktop-windows>

윈도우에서의 설치 - Windows Subsystem for Linux v2

윈도우 10, 버전 2004부터 지원 (2020년 04월)

WSL을 통해 Docker, k8s 환경을 직접 구동 가능

```
allieus@Base: /mnt/c/Users/allieus
C:\Users\allieus> wsl
allieus@Base:/mnt/c/Users/allieus$ echo $SHELL
/bin/bash
allieus@Base:/mnt/c/Users/allieus$
```

<https://devblogs.microsoft.com/commandline/wsl2-will-be-generally-available-in-windows-10-version-2004/>



윈도우에서의 설치 - 윈도우 10 Home 이하

윈도우7 이상, 윈도우 10 Home 이하 버전 : Docker ToolBox

https://docs.docker.com/toolbox/toolbox_install_windows/

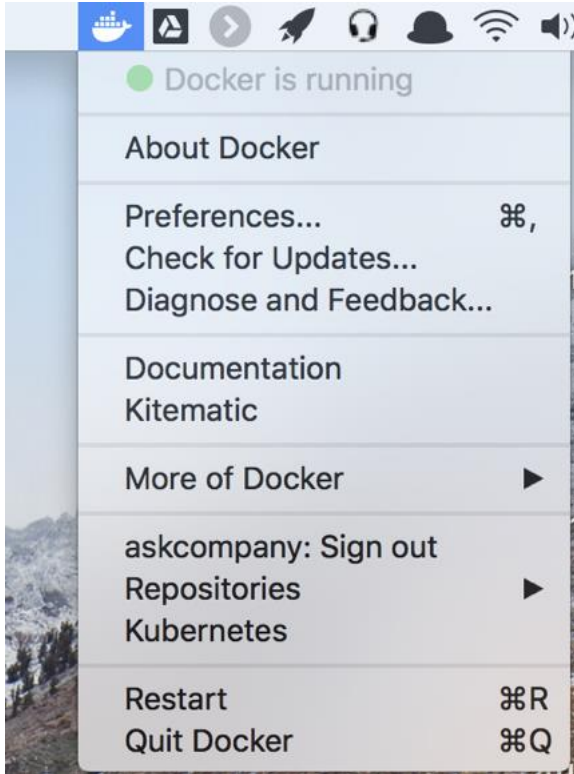
VirtualBox와 Git이 같이 설치되기에, VirtualBox가 이미 설치되어있으시면 미리 제거하시는 것을 추천.

Docker 명령은 VirtualBox VM 상에서 수행이 됩니다.

윈도우 Host 측에서 실행되는 것이 아닙니다.

Docker 서버를 띄운다면, VirtualBox VM에서 포트포워딩이 필요합니다.

맥에서의 설치



<https://docs.docker.com/docker-for-mac/>

리눅스에서의 설치

각 리눅스 배포판에서 지원하는 명령으로 손쉬운 설치가 가능합니다.

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

리눅스에서는 OS에 맞춰, 실행여부를 확인해주세요.

Tip) 기본적으로 도커 컨테이너 안에서 docker 실행은 지원되지 않습니다.

<https://docs.docker.com/install/linux/docker-ce/ubuntu/>

설치 확인

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.19582.1001]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\allieus> docker --version
Docker version 19.03.8, build afacb8b

C:\Users\allieus> docker run --rm hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
1b930d010525: Pull complete
Digest: sha256:f9dfddf63636d84ef479d645ab5885156ae030f611a56f3a7ac7f2fdd86d7e4e
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/

C:\Users\allieus>
```


주요 명령어

`docker pull nginx:latest`

`docker run --rm --detach --publish 8000:80 nginx`

--name 옵션으로 컨테이너 이름을 지정할 수 있습니다. 지정하지 않으면 임의의 이름이 지정됩니다.

--rm 옵션을 주면, stop과 동시에 컨테이너가 제거됩니다.

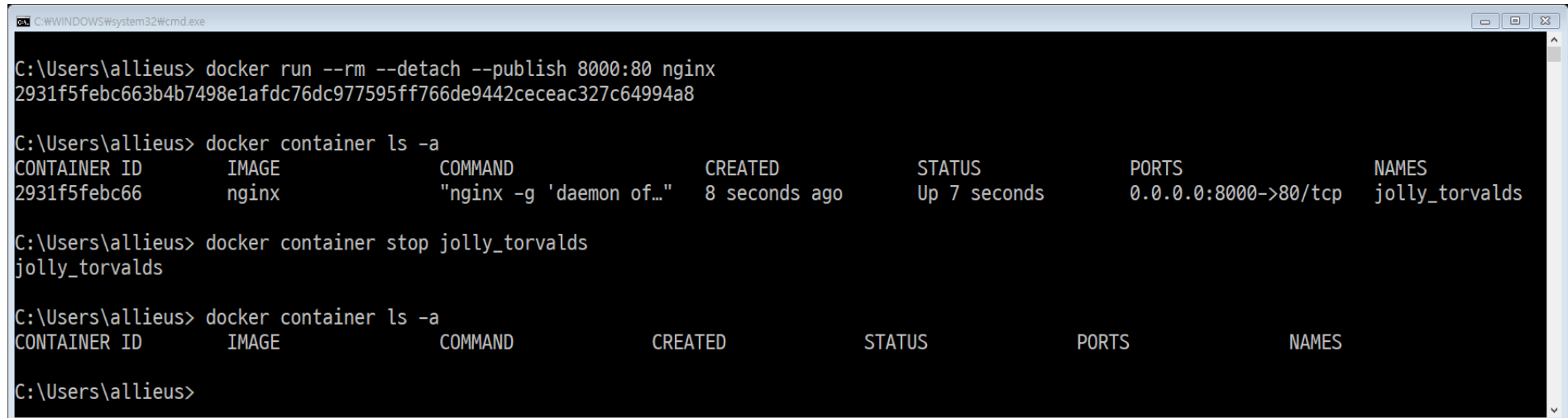
`docker container ls`

`docker <container> ps`

`docker <container> stop zen_lumiere`

`docker <container> rm zen_lumiere`

`docker images ls`

A screenshot of a Windows command prompt window titled "C:\WINDOWS\system32\cmd.exe". The window shows a series of Docker commands and their outputs. The first command is `docker run --rm --detach --publish 8000:80 nginx`, which returns a long container ID. The second command is `docker container ls -a`, which displays a table of containers. The table has columns for CONTAINER ID, IMAGE, COMMAND, CREATED, STATUS, PORTS, and NAMES. It shows one container named "jolly_torvalds" based on the "nginx" image. The third command is `docker container stop jolly_torvalds`, which returns the container name. The fourth command is `docker container ls -a`, which shows an empty table. The prompt ends with `C:\Users\allieus>`.

```
C:\Users\allieus> docker run --rm --detach --publish 8000:80 nginx
2931f5febc663b4b7498e1afdc76dc977595ff766de9442ceceac327c64994a8

C:\Users\allieus> docker container ls -a
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                NAMES
2931f5febc66   nginx     "nginx -g 'daemon of..." 8 seconds ago  Up 7 seconds  0.0.0.0:8000->80/tcp  jolly_torvalds

C:\Users\allieus> docker container stop jolly_torvalds
jolly_torvalds

C:\Users\allieus> docker container ls -a
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                NAMES
C:\Users\allieus>
```

Life is short.
You need Python and Django.

I will be your pacemaker.

