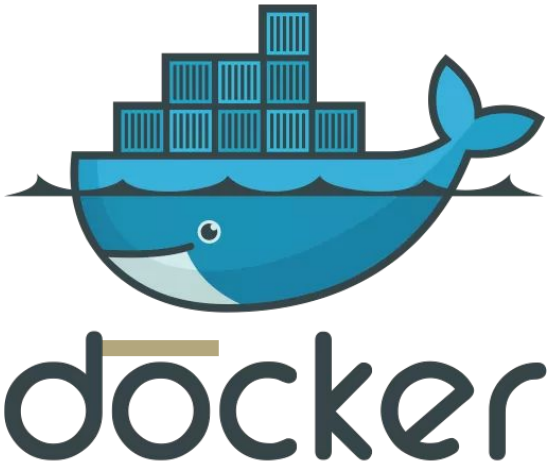


Docker

기본 개념

IndiFlex 시니어코딩

Background

Software(Application) Engineer
- S/W Application

System(Infra) Engineer
- H/W, Network, OS, Middleware

On-Premises

Cloud Platform
- AWS, GCP, KT, etc

Orchestration

Docker ?

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

PyCon 2013 Solomon Kykes - 'The Future of Linux Container'

After 10 years from LXC

Made by Google's Go Language

Immutable, Stateless, Scalable

Community Edition and Enterprise Edition ($1000 per year)

Linux Base*

64-bit OS only

Docker Terms

Docker Image and Container

Docker Engine (Docker Daemon, Core)

Docker Client

Docker Host OS

Docker Machine (Runtime Environment)

Docker Compose

Docker Registry, Hub, Swarm

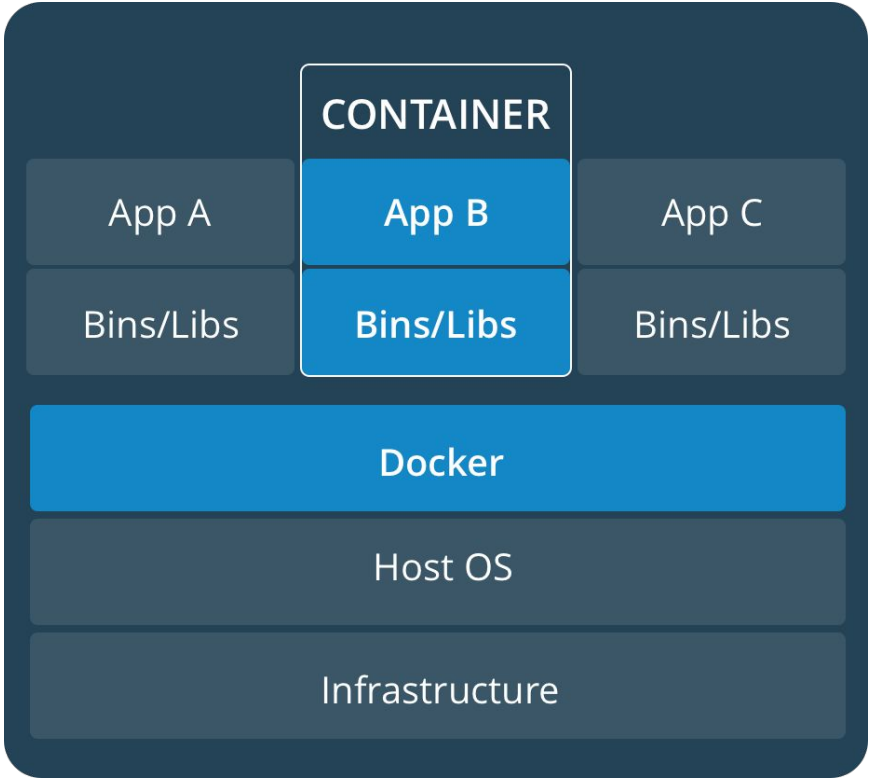
Docker Container

Logical Area in OS(Docker Host OS)
 - Process, Network, FS

Like Each Server

Directory, Libraries and IP Shared

cf. VM(Virtual Machine)은 별도의 OS



Docker Image

Container based on Docker Image

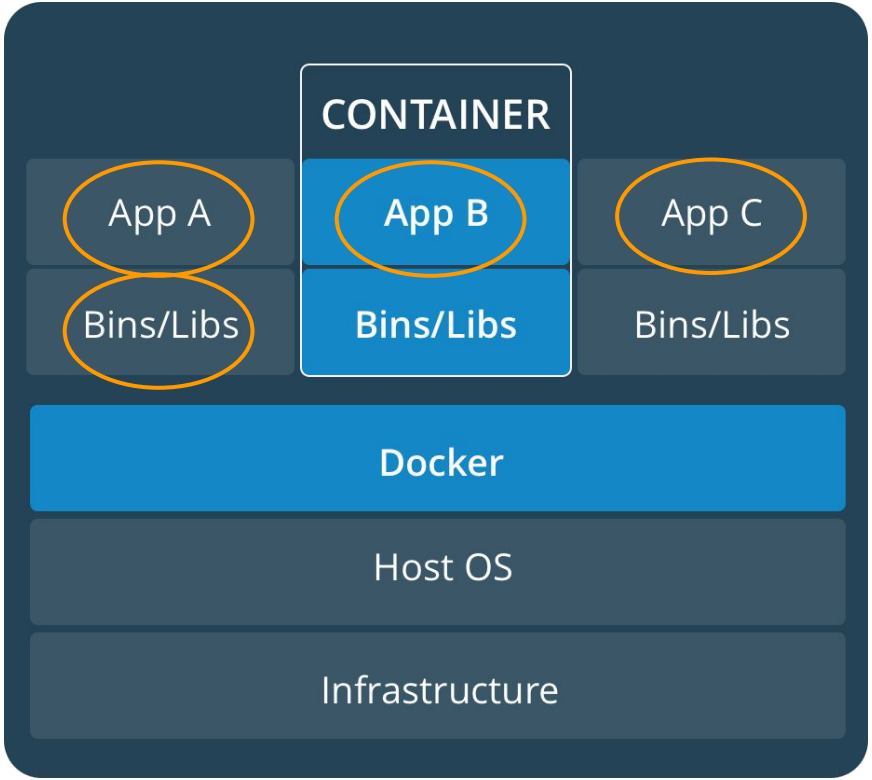
App, Libs, Middleware, OS, NW, etc

ex. MySQL image, Ubuntu image, etc

Image **Build** (Make) : Dockerfile

Image **Ship** (Share)

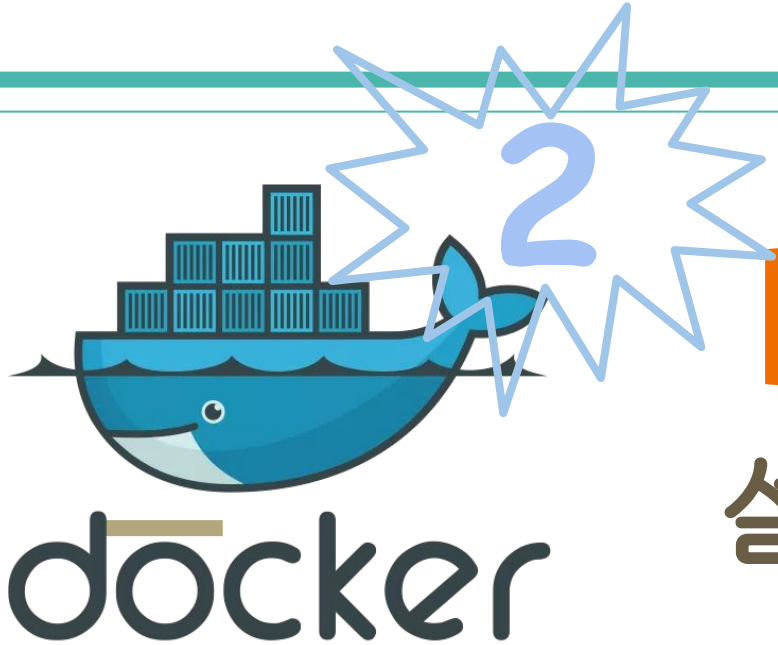
Docker Hub



Docker Priciple

Namespace (Linux Kernel's Isolation Technology)
- PID namespace
- Network namespace
- And UID, MOUNT, UTS, IPC namespace

docker0 NIC (Network Interface Controller)
- Auto **eth0** Private IP (172.17.0.0/16 subnet mask)
- NAT, NAPT (Network Address & Port Translation) by Linux IPTable



Docker 설치하기

IndiFlex 시니어코딩

Install Docker in Windows

Windows 10

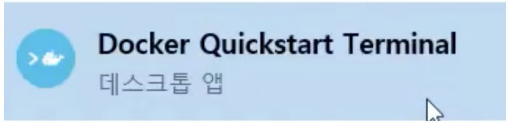
- Need Docker for Windows Only
- 작업관리자 > 성능 > 가상화

Under Windows 10 (Windows 7)

- Need Docker ToolBox (based on Oracle VirtualBox - Linux VM)
- Check Virtualization Tool

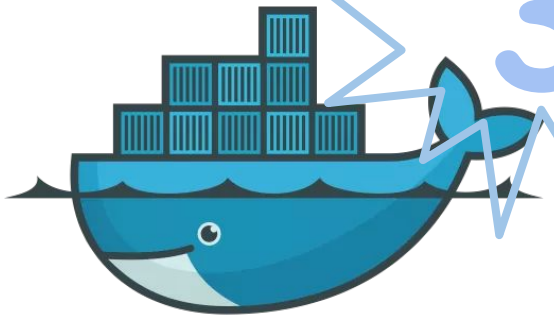
cf. Docker for MAC (By docker account)

Install Docker in Windows (Cont'd)




```
# Run Docker Quick Starter

$ docker version          # cf. docker --version
$ docker info             # docker system info
$ docker --help
$ docker run hello-world
$ docker images
$ docker ps                       # cf. docker ps -a
$ docker-machine ls       # Oracle VM VirtualBox
                          # Homedir : 사용자/.docker
```

https://docs.docker.com/docker-for-windows/#getting-started-with-windows-containers



3

Docker

기본 명령어

IndiFlex 시니어코딩

Run Docker Container (Create Container)

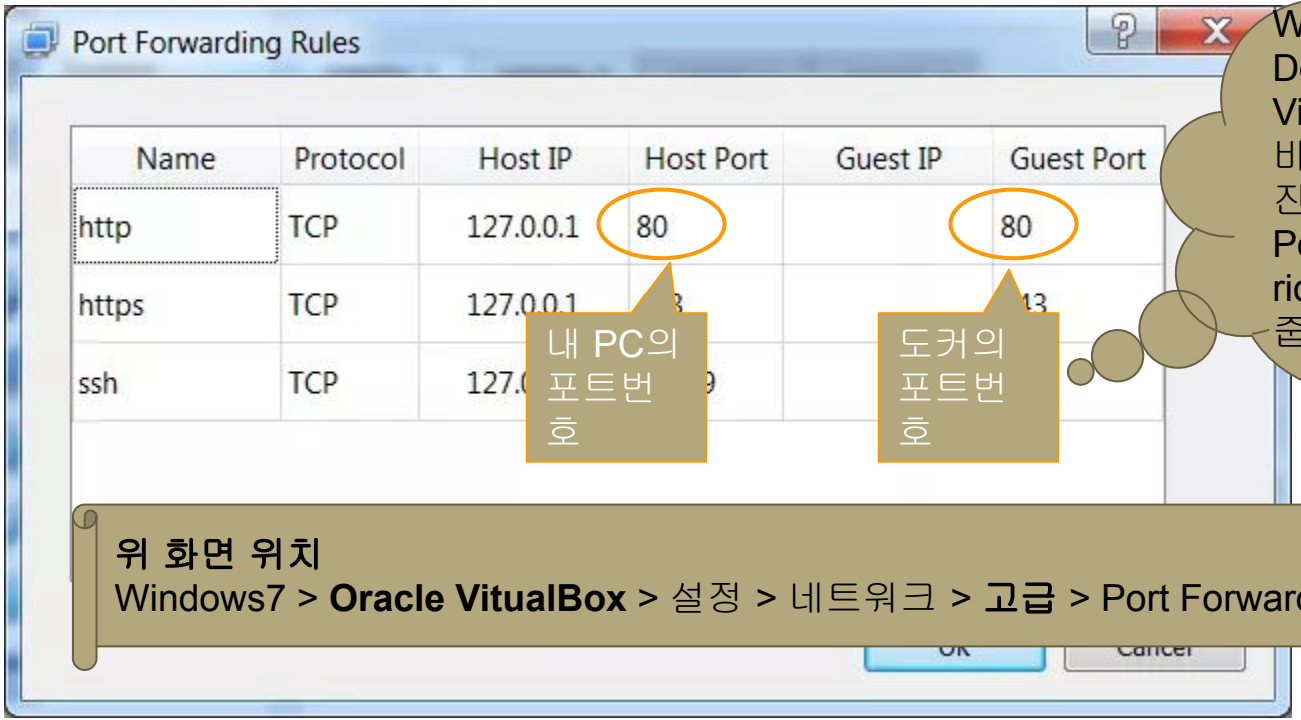
```
# run docker container for ubuntu image

docker container run <docker-image-name> <command>

$> docker container run ubuntu:latest /bin/echo 'Hello world'

$> docker ps -a
$> docker container ps -a

$> docker system df

$> docker image ls
```

https://hub.docker.com

Download Image & Start Docker Container

```
$> docker pull <docker-image-name[:tag-name]>
$> docker container run --name <container> -d -p 80:80 <image>
$> docker [start | stop | restart] <container-name>
$> docker container [pause | unpause] <container-name>

$> docker pull nginx
$> docker image ls
$> docker container run --name webserver -d -p 80:80 nginx
$> docker container ps -a
$> docker container [top | stats] webserver

$> docker container rm webserver      # cf. prune
```

Windows7 Oracle VitualBox Port Forwarding



Windows7과 Docker사이에 Oracle VitualBox가 있으므로 바로 도커 컨테이너로 진입을 못하기 때문에 Port Forward로 ridge(다리)를 놓아 줍니다~

위 화면 위치
Windows7 > **Oracle VitualBox** > 설정 > 네트워크 > **고급** > Port Forwarding

https://hub.docker.com

Docker Image Commands

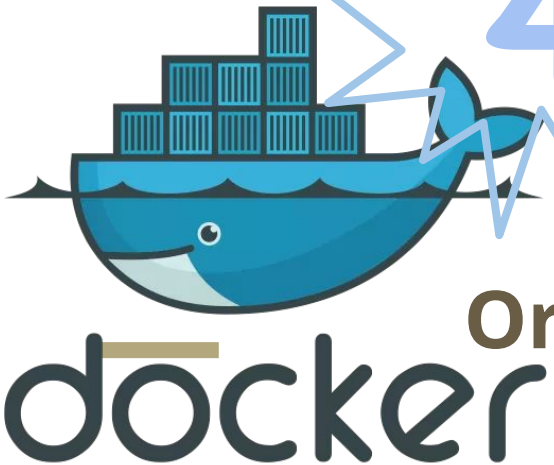
```
docker image <command> [image-name]

$> docker image [pull | push]
$> docker image ls
$> docker image inspect nginx
$> docker search python
$> docker image rm nginx      # nginx image 삭제
$> docker image prune         # 사용하지 않는 모든 image 삭제!
$> docker login               # login to docker hub
$> docker logout
```

Enter the Docker Container

```
$> docker container run -it --name "test1" centos /bin/cal

$> docker container run -it --name "cosh" centos /bin/bash
[root@b5152134ff15 /]# cat /etc/hosts
[root@b5152134ff15 /]# exit          # Ctrl+P, Ctrl+Q

$> docker container run -it --restart=always --name "centsh"
centos /bin/bash

$> docker container run -itd --name ubsh ubuntu bash

$> docker container attach ubsh          # Ctrl+P, Ctrl+Q
$> docker container [stop | start] <container-name>
```

Attach & Exec on Running Container

```
$> docker container attach ub

#> exit(kill) or Ctrl+P, Ctrl+Q (detach)

$> docker container exec -it ub /bin/cat /etc/hosts

$> docker start webserver
$> docker container exec -it webserver /bin/echo "Hello"

$> docker container port oracle

$> docker container rename webserver nginxserver
```



4

Docker

Oracle, MySQL 설치

IndiFlex 시니어코딩

Install Oracle-XE

https://hub.docker.com

```
$> docker search oracle

$> docker pull sath89/oracle-xe-11g    # cf. wnameless/oracle-xe-11g
$> docker images         # same as $> docker image ls

$> docker run -d --name ora -p 8080:8080 -p 1521:1521
sath89/oracle-xe-11g
$> docker run -d -p 8080:8080 -p 1521:1521 -v /my/oracle/data:/u01/app/oracle

$> docker ps

$> docker exec -it oracle bash
```

Install Oracle-XE (after Feb. 14 2019)

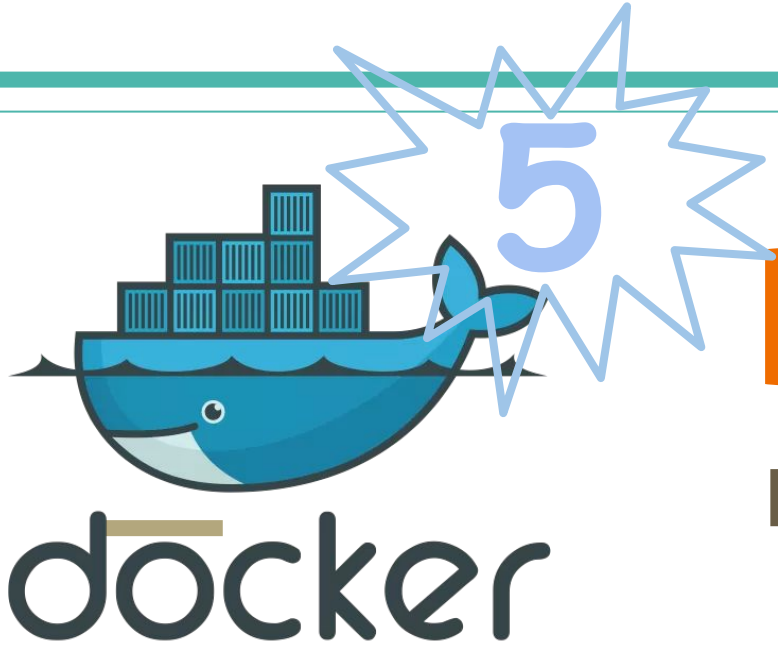
```
# 저작권 문제로 hub.docker.com에서 오라클 이미지 모두 제거됨

# ubuntu나 centos 컨테이너에서 직접 오라클 설치하면 됨
```

Install MySQL

```
$> docker search mysql

$> docker pull mysql:5.7
$> docker images

$> docker run -d -p 3306:3306 -e MYSQL_ROOT_PASSWORD=r!
       --name mysql5 mysql:5.7

$> docker ps

$> docker exec -it mysql5 bash

#> mysql -u root -p
```

Tips

```
$> docker ps -a --format '{{.Names}} by {{.Image}} ---- {{.Status}}'

$> docker logs <container-id>

# backup the container to image
$> docker commit -p <container-name> <image-name to save>
$> docker save -o ~/backup/oracle.tar <backup-image-name>

# load backuped image
$> docker load < ~/backup/oracle.tar
```



Docker

File Sharing

IndiFlex 시니어코딩

File Copy & Share

docker [container] cp <container-name>:<path> <client-path>

docker [container] cp <client-file> <container-name>:<path>

```
$> docker cp mysql5:/backup/dump.sql  .

$> docker cp dump.sql mysql5:/

# Share Directory

$> docker run -v <localpath>:<container-path>
```

cf. $> docker stop `docker ps -q`