docker

# reference:

https://docs.docker.com/

# docker best practise:

## stack방식 선호:

/d/DEV/workspace\_docker/00\_db:

my\_db

docker stack deploy -c oradb.yml oradb

docker stack deploy -c w-mysql.yml w-mysql

docker stack ls

docker stack ps w-mysql

docker stack rm w-mysql

docker ps

## docker run 방식:

parameter이 있으면 docker run으로 하자.

좋기는 shell만들어서 실행.

------------------------------------------------------------------------------

## container로부터 image 만들기:

docker commit c3f279d17e0a mr838wind/targetImage:tagname

## docker 자주 쓰는 명령어

docker images

docker images mr8\*/\*

docker ps

docker run -it imageId:tag /bin/bash

docker exec -it containerId /bin/bash

docker exec -it containerId /bin/sh

alias dc\_stop=docker stop $(docker ps -aq)

alias dc\_rm=docker rm $(docker ps -aq)

--name containerName

--rm (stop할 때 제거)

# @problems;

## 여러 개 ip일 때 :

docker swarm init --advertise-addr 192.168.99.100

## oracle 시간이 다름:

docker run -d --rm --name "oradb" -e TZ=Asia/Seoul --shm-size=1g -p 1521:1521 -p 18080:8080 -v /c/Users/windfall/00\_docker\_ws/oracle/init:/etc/entrypoint-initdb.d alexeiled/docker-oracle-xe-11g

아직도 약간 차이 있음( 20분좌우)

------------------------------------------------------------------------------

# ubuntu 환경:

## >> run

docker run -it mr838wind/ubuntu /bin/bash

mr838wind/ubuntu --- 기본 tool이 깔려있음

-----------기본 tool [s] -----------

ubuntu:latest

apt-get update

apt-get install -y build-essential libgmp3-dev golang git tree vim

----------- 기본 tool [e] -----------

------------------------------------------------------------------------------

# ethereum:

## >> run

cd /c/Users/windfall/00\_docker\_ws/eth\_data

docker run -v /c/Users/windfall/00\_docker\_ws/eth\_data:/root/.ethereum ethereum/client-go init /root/.ethereum/genesis.json

docker run -it -p 30303:30303 -p 8545:8545 -v /c/Users/windfall/00\_docker\_ws/eth\_data:/root/.ethereum \

-d --name myeth mr838wind/myeth --networkid 4649 \

--nodiscover --maxpeers 0 --rpc --rpcaddr "0.0.0.0" --rpccorsdomain "\*" \

--ipcpath "/Users/user/Library/Ethereum/geth.ipc" \

--rpcapi "admin,db,eth,debug,miner,net,shh,txpool,personal,web3"

## >> check

docker exec -it myeth /bin/sh

geth attach http://192.168.99.100:8545

miner.start(1)

miner.stop()

web3.fromWei(eth.getBalance(eth.accounts[0]), "ether")

personal.unlockAccount(eth.accounts[0], "pass0", 0)

personal.unlockAccount(eth.accounts[1], "pass1", 0)

personal.unlockAccount(eth.accounts[2], "pass2", 0)

--------------------------------------old [s] --------------------

cd /c/Users/windfall/00\_docker\_ws/eth\_data

docker run -v /c/Users/windfall/00\_docker\_ws/eth\_data:/root/.ethereum ethereum/client-go init /root/.ethereum/genesis.json

docker run -it -p 30303:30303 -p 8545:8545 -v /c/Users/windfall/00\_docker\_ws/eth\_data:/root/.ethereum \

-d --name myeth ethereum/client-go --networkid 4649 \

--nodiscover --maxpeers 0 --rpc --rpcaddr "0.0.0.0" --rpccorsdomain "\*" \

--ipcpath "/Users/user/Library/Ethereum/geth.ipc" \

--rpcapi "admin,db,eth,debug,miner,net,shh,txpool,personal,web3"

## 주의:

--ipcpath "/Users/user/Library/Ethereum/geth.ipc" 가 없으면 error

## mining

--mine --minerthreads=1 --etherbase=0x0000000000000000000000000000000000000000 \

## remove

--rm

docker exec -it myeth /bin/sh

geth attach http://192.168.99.100:8545

## accounts

personal.newAccount("pass0")

personal.newAccount("pass1")

personal.newAccount("pass2")

personal.unlockAccount(eth.accounts[0], "pass0", 0)

personal.unlockAccount(eth.accounts[1], "pass1", 0)

personal.unlockAccount(eth.accounts[2], "pass2", 0)

--------------------------------------old [e] --------------------

------------------------------------------------------------------------------

# oracle:

## >> info:

192.168.99.100

system

oracle

## >> run stack

my\_db

docker stack deploy -c oradb.yml oradb

## >> run

>#(-v에 들어가는 이름 꼭 /c/Users 밑에것을 사용하자)

>#(-v 설정이 문제가 해결안됨)

docker run -d --rm --name "oradb" --shm-size=1g -p 1521:1521 -p 18080:8080 alexeiled/docker-oracle-xe-11g

docker run -d --rm --name "oradb" --shm-size=1g -p 1521:1521 -p 18080:8080 -v /c/Users/windfall/00\_docker\_ws/oracle/init:/etc/entrypoint-initdb.d alexeiled/docker-oracle-xe-11g

## >> check

docker exec -it oradb /bin/bash

su - oracle

sqlplus / as sysdba

## >> volume

C:\Users\windfall\00\_docker\_ws\oracle\init\db\_init\_devprmowner.sql

>>> 참고:

00\_db\_addon.sql

00\_db\_init\_00.sql

0000\_docker\_oracle\_patch.sql

## >> example

$ my\_shell

$ ./run\_oradb2.sh 혹은 $ ./run\_oradb.sh

db 접속후:

C:\Users\windfall\00\_docker\_ws\oracle\back\_data 에 있는

0000\_docker\_oracle\_patch.sql

00\_my\_proc\_func.sql

실행 해줌

주의: 시간이 20분좌우 차이있음.

------------------------------------------------------------------------------

# mysql:

## >> run stack

my\_db

docker stack deploy -c w-mysql.yml w-mysql

## >> run

docker run --rm --name w-mysql -p 3306:3306 -e MYSQL\_ROOT\_PASSWORD=root -d mysql:5.7

docker run --rm --name w-mysql -p 3306:3306 -v /c/Users/windfall/00\_docker\_ws/mysql/init:/docker-entrypoint-initdb.d -e MYSQL\_ROOT\_PASSWORD=root -d mysql:5.7

## >> check

docker exec -it w-mysql sh -c 'exec mysql -uroot -proot'

docker exec -it w-mysql /bin/bash

mysql -uroot -proot

show databases;

show tables;

------------------------------------------------------------------------------

@problems:

여러개 internet 환경일때 하나 지정해야 함.

docker swarm init --advertise-addr 192.168.99.100