**Docker Tutorial**

**Docker is a centralized platform for packaging, deploying, and running applications.**

1. Provides easy way to deploy an application

**Docker Registry, Docker Repository, Tags**

**Docker Registry:** place where all repositories are there.

**Docker Repository:** place where all images/tags are present.

**Tag:** its kind of war/ear file or image.

**All commands**

docker --version : to see the version

**Following commands are to run specific appliation in seoerae command prompts**

docker run -p 5000:5000 in28min/hello-world-python:0.0.1.RELEASE

docker run -p 5000:5000 in28min/hello-world-java:0.0.1.RELEASE

docker run -p 5000:5000 in28min/hello-world-nodejs:0.0.1.RELEASE

**Following command is used to run application with detached mode(detached from command prompts)**

docker run -d -p 5000:5000 in28min/hello-world-nodejs:0.0.1.RELEASE

docker run -d -p 5001:5000 in28min/hello-world-python:0.0.1.RELEASE

**command to see the specific logs**

docker logs 04e52ff9270f5810eefe1f77222852dc1461c22440d4ecd6228b5c38f09d838e

docker logs c2ba

**See all images**

docker images

**See all the containers in list format**

docker container ls

docker container ls –a

**Stoppng contaners**

docker container stop f708b7ee1a8b

docker run -d -p 5001:8080 in28min/hello-world-rest-api:0.0.1.RELEASE

docker pull mysql

docker search mysql

docker image history in28min/hello-world-java:0.0.1.RELEASE

docker image history 100229ba687e

docker image inspect 100229ba687e

docker image remove mysql

docker image remove in28min/hello-world-java:0.0.1.RELEASE

docker container rm 3e657ae9bd16

docker container ls -a

docker container pause 832

docker container unpause 832

docker container stop 832

docker container inspect ff521fa58db3

docker container prune

docker system

docker system df

docker system info

docker system prune -a

docker top 9009722eac4d

docker stats 9009722eac4d

docker container run -p 5000:5000 -d -m 512m in28min/hello-world-java:0.0.1.RELEASE

docker container run -p 5000:5000 -d -m 512m --cpu-quota=50000 in28min/hello-world-java:0.0.1.RELEASE

docker system events



docker container stats 4faca1ea914e3e4587d1d790948ec6cb8fa34f26e900c12632fd64d4722fd59a

docker stats 42f170966ce613d2a16d7404495af7b3295e01aeb9142e1fa1762bbdc581f502



cd /in28Minutes/git/devops-master-class/projects/hello-world/hello-world-python

docker build -t in28min/hello-world-python:0.0.2.RELEASE .

docker run -p 5000:5000 -d in28min/hello-world-python:0.0.2.RELEASE

docker history e66dc383f7a0

docker push in28min/hello-world-python:0.0.2.RELEASE



cd ../hello-world-nodejs/

docker build -t in28min/hello-world-nodejs:0.0.2.RELEASE .

docker container run -d -p 5000:5000 in28min/hello-world-nodejs:0.0.2.RELEASE

docker push in28min/hello-world-nodejs:0.0.2.RELEASE



cd ../hello-world-java/

docker build -t in28min/hello-world-java:0.0.2.RELEASE .

docker run -d -p 5000:5000 in28min/hello-world-java:0.0.2.RELEASE

docker push in28min/hello-world-java:0.0.2.RELEASE



docker run -d -p 5001:5000 in28min/hello-world-nodejs:0.0.3.RELEASE ping google.com



docker run -d -p 8000:8000 --name=currency-exchange in28min/currency-exchange:0.0.1-RELEASE

docker run -d -p 8100:8100 --name=currency-conversion in28min/currency-conversion:0.0.1-RELEASE

docker network ls

docker network inspect bridge



docker run -d -p 8100:8100 --env CURRENCY\_EXCHANGE\_SERVICE\_HOST=http://currency-exchange --name=currency-conversion --link currency-exchange in28min/currency-conversion:0.0.1-RELEASE



docker network create currency-network

docker container stop currency-exchange

docker container stop currency-conversion

docker run -d -p 8000:8000 --name=currency-exchange --network=currency-network in28min/currency-exchange:0.0.1-RELEASE

docker run -d -p 8100:8100 --env CURRENCY\_EXCHANGE\_SERVICE\_HOST=http://currency-exchange --name=currency-conversion --network=currency-network in28min/currency-conversion:0.0.1-RELEASE



docker-compose --version

cd ../../microservices/

docker-compose up

docker-compose up -d

docker container ls

docker network ls

docker network inspect microservices\_currency-compose-network

docker-compose down

docker container ls -a

docker system prune -a

docker-compose config

docker-compose images

docker-compose ps

docker-compose top



docker build -t in28min/hello-world-java:0.0.1.RELEASE .

docker push in28min/hello-world-java:0.0.1.RELEASE

docker build -t in28min/hello-world-python:0.0.1.RELEASE .

docker push in28min/hello-world-python:0.0.1.RELEASE

docker build -t in28min/hello-world-nodejs:0.0.1.RELEASE .

docker push in28min/hello-world-nodejs:0.0.1.RELEASE