**EXPONIENDO CONTENEDORES**

Comandos:

$ docker run -d --name proxy nginx (corro un nginx)

$ docker stop proxy (apaga el contenedor)

$ docker rm proxy (borro el contenedor)

$ docker rm -f <contenedor> (lo para y lo borra)

$ docker run -d --name proxy -p 8080:80 nginx (corro un nginx y expongo el puerto 80 del contenedor en el puerto 8080 de mi máquina)

localhost:8080 (desde mi navegador compruebo que funcione)

$ docker logs proxy (veo los logs)

$ docker logs -f proxy (hago un follow del log)

$ docker logs --tail 10 -f proxy (veo y sigo solo las 10 últimas entradas del log)

**BIND MOUNTS**

Comandos:

$ mkdir dockerdata (creo un directorio en mi máquina)

$ docker run -d --name db mongo

$ docker ps (veo los contenedores activos)

$ docker exec -it db bash (entro al bash del contenedor)

$ mongo (me conecto a la BBDD)

shows dbs (listo las BBDD)

use platzi ( creo la BBDD platzi)

db.users.insert({“nombre”:“guido”}) (inserto un nuevo dato)

db.users.find() (veo el dato que cargué)

$ docker run -d --name db -v <path de mi maquina>:<path dentro del contenedor(/data/db mongo)> (corro un contenedor de mongo y creo un bind mount

docker run -d --name thiago -p 27017:27017 -v "C:/Users/Eder Mena/Desktop/Docker/data:/data/db" -e MONGO\_INITDB\_ROOT\_USERNAME=<usuario> -e MONGO\_INITDB\_ROOT\_PASSWORD=<contraseña> mongo

**VOLUMES**

Comandos:

$ docker volume ls (listo los volumes)

$ docker volume create dbdata (creo un volume)

$ docker run -d --name db --mount src=dbdata,dst=/data/db mongo (corro la BBDD y monto el volume)

$ docker inspect db (veo la información detallada del contenedor)

$ mongo (me conecto a la BBDD)

shows dbs (listo las BBDD)

use platzi ( creo la BBDD platzi)

db.users.insert({“nombre”:“guido”}) (inserto un nuevo dato)

db.users.find() (veo el dato que cargué)

**Comandos:**

$ touch prueba.txt (creo un archivo en mi máquina)

$ docker run -d --name copytest ubuntu tail -f /dev/null (corron un ubuntu y le agrego el tail para que quede activo)

$ docker exec -it copytest bash (entro al contenedor)

$ mkdir testing (creo un directorio en el contenedor)

$ docker cp prueba.txt copytest:/testing/test.txt (copio el archivo dentro del contenedor)

$ docker cp copytest:/testing localtesting (copio el directorio de un contenedor a mi máquina)

con “docker cp” no hace falta que el contenedor esté corriendo