

Docker - Comandos básicos

Para crear imagen a partir del Dockerfile del directorio actual

```
docker build -t nombre_imagen .
```

Este comando creará un contenedor a partir de la imagen pasada por parámetro

-p para especificar por cual puerto será accesible desde localhost, -p [puerto_localhost]:[puerto_contenedor]

```
docker run -p 9080:9080 --name nombre_contenedor -d nombre_imagen
```

Lista los contenedores activos

```
docker ps
```

Lista los contenedores activos e inactivos

```
docker ps -a
```

Iniciar contenedor

```
docker start [nombre_contenedor | id_contenedor]
```

Reiniciar contenedor

```
docker restart [nombre_contenedor | id_contenedor]
```

Detener contenedor

```
docker stop [nombre_contenedor | id_contenedor]
```

Remover contenedor

```
docker rm [nombre_contenedor | id_contenedor]
```

Exportar imagen docker a un .tar

```
docker save imagen > imagen.tar
```

Para iniciar todos los contenedores y configuraciones especificadas en el docker-compose.yml

```
docker-compose up -d
```

Entrar a la linea de comando de un contenedor especifico

```
docker exec -ti container_id /bin/bash
```

```
docker run -i -t --rm image bash
```

```
docker inspect -f '{{range .NetworkSettings.Networks}}{{.IPAddress}}{{end}}' mariadbtest
```

```
docker commit upbeat_panini testredis:example
```

```
#reiniciar todos los contenedores
```

```
docker restart $(docker ps -a -q)
```

```
#--- Docker compose
```

```
docker-compose down --rmi all
```

```
# --no-deps - Don't start linked services. --build - Build images before starting containers.
```

```
docker-compose up -d --no-deps --build <service_name>
```

```
# to remove all the images which are not used by existing containers
```

```
docker system prune -a
```

Revision #11

Created 12 July 2019 19:46:59 by Juan Navas

Updated 18 January 2023 20:43:20 by Juan Navas