



# docker

## COMANDOS BÁSICOS DOCKER

Javier Martínez Gallego

## Contenido

# Mostrar la versión de Docker:.....	2
# Patrón general de Docker.....	2
# Listado de imágenes.....	2
# Descargar una imagen.....	2
# Crear un contenedor .....	2
# Ver qué contenedores tenemos.....	3
#Arrancar y parar contenedor.....	3
#Ejecutar contenedor.....	3
#Borrar contenedor.....	3
# Borrar imágenes .....	3
# Ver info de un contenedor .....	3
#Ver logs de un contenedor .....	4
# Actualizar contenedor .....	4
#Crear volumen.....	4
#Listar y ver volúmenes.....	4
#Levantar con docker compose .....	4

### # Mostrar la versión de Docker:

docker --versión :Muestra la versión instalada de Docker

```
PS C:\WINDOWS\system32> docker --version
Docker version 20.10.17, build 100c701
```

### # Patrón general de Docker

docker <command> [--options] <object>

### # Listado de imágenes

docker images: Muestra las imágenes descargadas en el equipo

```
PS C:\WINDOWS\system32> docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
busybox              latest             2bd29714875d       2 weeks ago        1.24MB
postgres            latest             75993dd36176       2 weeks ago        376MB
ubuntu              latest             2dc39ba059dc       4 weeks ago        77.8MB
alpine              latest             9c6f07244728       7 weeks ago        5.54MB
bitnami/wildfly      latest             a5628b1c73ae       4 months ago       691MB
bitnami/mariadb      10.6               f2e0e6710630       4 months ago       335MB
```

### # Descargar una imagen

docker pull "imagen": descargamos una imagen que podemos usar para un contenedor

```
PS C:\WINDOWS\system32> docker pull httpd
Using default tag: latest
latest: Pulling from library/httpd
31b3f1ad4ce1: Already exists
f29089ecfcbf: Pull complete
a9fcd580ef1c: Pull complete
a19138bf3164: Pull complete
5bfb2ce98078: Pull complete
Digest: sha256:71e882df50adc606c57e46e5deb3c93328
Status: Downloaded newer image for httpd:latest
docker.io/library/httpd:latest
PS C:\WINDOWS\system32>
```

### # Crear un contenedor

\$ docker run [OPTIONS] IMAGE[:TAG|@DIGEST] [COMMAND] [ARG...]

Esta es la sintáxis general para crear un contenedor (docker run "micontenedor") , si es la primera vez que lo ejecutamos podemos añadirle además parámetros de configuración de almacenamiento, red, puertos,etc.

- **-d**: se utiliza por el comando Docker Run para **arrancar en segundo plano**, también conocido como *background*.
- **-rm**: para destruir el **docker** al terminar su ejecución.
- **-p**: cumple la función de mapear puertos del *host* hacia el contenedor.
- **-it**: interactivo (*stdin*) + *tty* (pseudo terminal).
- **-v**: hace referencia a los **volúmenes del sistema**.
- **-c**: esta opción del comando Docker Run está relacionado con la **cuota de CPU**.
- **-m**: opción que se refiere al **límite de memoria**.
- **-name**: indica el **nombre del Docker** en el sistema.
- **-e**: hace referencia a las variables de entorno dentro del comando Docker Run.
- **--network**: se encarga de la función de **asociar el contenedor a la red** que quiera el usuario.
- **--entrypoint**: se utiliza con el objetivo de **cambiar el punto de entrada** del **docker**.
- **--restart**: se refiere a la **política de reinicio en caso de error** o [healthcheck](#) fallido. Esta opción tiene funciones predeterminadas como no

## # Ver qué contenedores tenemos

docker ps: Si hacemos docker ps vemos los contenedores que tenemos arrancados, si hacemos docker ps -a vemos todos los contenedores que tenemos.

```
PS C:\WINDOWS\system32> docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
610a33bde0b	ubuntu	"bash"	2 days ago	Exited (0) 2 days ago	
f2306f5d245a	nostalgic_jackson	"docker-entrypoint.s..."	2 days ago	Exited (1) 2 days ago	
839add3eed5	postgres	"sh"	2 days ago	Exited (0) 2 days ago	
c7d455cdaa9f	vigilant_shannon	"docker-entrypoint.s..."	3 days ago	Exited (0) 19 hours ago	
8ad150757a50	adoring_neumann	"catalina.sh run"	3 months ago	Exited (255) 2 days ago	0.0.0.0:8080->8080/tcp

## #Arrancar y parar contenedor

Con docker start "container" arrancamos y con docker stop "container" lo paramos.

```
PS C:\WINDOWS\system32> docker start tomcat
tomcat
PS C:\WINDOWS\system32> docker stop tomcat
tomcat
PS C:\WINDOWS\system32>
```

## #Ejecutar contenedor

Con docker exec "container" ejecutamos un contenedor, aunque la opción más interesante aquí es docker exec -it "container" /bin/bash con esto el contenedor lanzará una consola para poder interactuar con él.

## #Borrar contenedor

Con docker rm "container" podemos borrar un contenedor

```
PS C:\WINDOWS\system32> docker rm wildfly
wildfly
```

## # Borrar imágenes

Con docker rmi borramos imágenes que tengamos descargadas

```
PS C:\WINDOWS\system32> docker rmi 4681988c73f3
Untagged: quay.io/wildfly/wildfly:latest
Untagged: quay.io/wildfly/wildfly@sha256:2e2bcb224012e8c6e07479cf5dde44128acb1089067215f18635ca202e8e1675
Deleted: sha256:4681988c73f3a51c7763d7f0ebb41a385048287d823d69cdb25ea183eb490cc9
Deleted: sha256:e525a8b89c5c9ae334ea061e8c076384e567928cb5a042a2f8e5b2c13fbc4b5e
```

## # Ver info de un contenedor

docker inspect lista la información de un contenedor

```
PS C:\WINDOWS\system32> docker inspect mysql
[
  {
    "Id": "c7d455cdaa9fbed47864cdf16b558b74d58a02d6171b9367df72126a8352762",
    "Created": "2022-09-26T16:52:32.333421455Z",
    "Path": "docker-entrypoint.sh",
    "Args": [
      "mysqld"
    ],
    "State": {
      "Status": "exited",
      "Running": false,
      "Paused": false,
      "Restarting": false,
      "OOMKilled": false,
      "Dead": false,
      "Pid": 0,
      "ExitCode": 0,
      "Error": "",
      "StartedAt": "2022-09-29T06:09:08.534007994Z",
      "FinishedAt": "2022-09-29T10:57:57.035854478Z"
    },
    "Image": "sha256:6bb891430fb6e2d3b4db41fd1f7ece08c5fc769d8f4823ec33c7c7ba99679213",
    "Labels": {}
  }
]
```

## #Ver logs de un contenedor

Con docker logs “container” vemos los logs que se han ido registrando en ese contenedor

```
PS C:\WINDOWS\system32> docker logs mysql
Initializing database
2022-09-26T16:52:44.833487Z 0 [Warning] TIMESTAMP with implicit DEFAULT value is deprecated. Please use --explicit_defaults_for_timestamp server option (see documentation for more details).
2022-09-26T16:52:45.182722Z 0 [Warning] InnoDB: New log files created, LSN=4
2022-09-26T16:52:45.294509Z 0 [Warning] InnoDB: Creating foreign key constraint and dropping existing constraint.
2022-09-26T16:52:45.329054Z 0 [Warning] No existing UUID has been found, so this server has been started. Generating a new UUID: a4e961ae-3dbb-11ed-bc26-000000000000.
2022-09-26T16:52:45.334231Z 0 [Warning] Gtid table is not ready to be used. Table was not created at server startup.
2022-09-26T16:52:45.335765Z 1 [Warning] root@localhost is created with an empty password. To avoid password leaks, it is recommended to use the --initialize-insecure option.
2022-09-26T16:52:46.928041Z 1 [Warning] 'user' entry 'root@localhost' ignored.
2022-09-26T16:52:46.928174Z 1 [Warning] 'user' entry 'mysql.session@localhost' ignored.
2022-09-26T16:52:46.928189Z 1 [Warning] 'user' entry 'mysql.sys@localhost' ignored.
2022-09-26T16:52:46.928222Z 1 [Warning] 'db' entry 'performance_schema mysql.session' ignored.
2022-09-26T16:52:46.928229Z 1 [Warning] 'db' entry 'sys mysql.sys@localhost' ignored.
2022-09-26T16:52:46.928250Z 1 [Warning] 'proxies_priv' entry '@ root@localhost' ignored.
2022-09-26T16:52:46.928322Z 1 [Warning] 'tables_priv' entry 'user mysql.session' ignored.
2022-09-26T16:52:46.928336Z 1 [Warning] 'tables_priv' entry 'sys_config mysql.session' ignored.
Database initialized
Initializing certificates
Generating a 2048 bit RSA private key
+++
```

## # Actualizar contenedor

Con docker update “container” podemos actualizar un contenedor y añadirle los parámetros que nos sean necesarios.

## #Crear volumen

Con docker volumen create creamos un nuevo volumen de almacenamiento

## #Listar y ver volúmenes

Con docker ls volume podemos ver los que tenemos y con inspect podemos tener información acerca de uno en concreto

```
PS C:\WINDOWS\system32> docker volume ls
DRIVER      VOLUME NAME
local       766c1bcecd05be291f0e9b68f6814f45c948e5402c72beaaf114673cb28a599a
local       escritorio_mariadb_data
local       escritorio_prestashop_data
local       f6f5b721c6aa59658779792af3356eef3b5f0318bcdce574897457804332b0cf

PS C:\WINDOWS\system32> docker inspect 766c1bcecd05be291f0e9b68f6814f45c948e5402c72beaaf114673cb28a599a
[
  {
    "CreatedAt": "2022-09-29T10:57:56Z",
    "Driver": "local",
    "Labels": null,
    "Mountpoint": "/var/lib/docker/volumes/766c1bcecd05be291f0e9b68f6814f45c948e5402c72beaaf114673cb28a599a/_data",
    "Name": "766c1bcecd05be291f0e9b68f6814f45c948e5402c72beaaf114673cb28a599a",
    "Options": null,
    "Scope": "local"
  }
]
```

## #Levantar con docker compose

Con docker compose up -d levantamos un contenedor a partir de un fichero docker-compose.yml y el -d lo deja arrancado.