IMÁGENES

**Docker pull**

Descargar una imagen

**docker run -it -d --name mysql8 -p 3306:3306 mysql:8.0.22**

Supondremos que es la PRIMERA VEZ que vamos a usar esa imagen y no la hemos descargado

**docker images**

lista de imágenes descargadas

**INFORMACIÓN DEL DOCKER IMAGES**

**REPOSITORY:** Nombre de la imagen en el repositorio.

**TAG**

Versión de la imagen que hemos descargado

**IMAGE ID**

Un identificador que es único para cada imagen

**CREATED**

Hace cuánto se creó la imagen.

**SIZE**

Tamaño de la imagen.

**docker pull mysql:5.7**

Actualiza la versión mysql:5.7

**docker pull -a php o docker pull --all-tags php**

Descargamos todas las versiones

**docker pull -q httpd**

**docker pull --quiet**

No muestro la información de las capas al descargarse

**docker rmi**

Borrar un contenedor de docker

Para eliminar mas de una imagen se ponen los nombres de las imágenes una detrás de otra separada por un espacio

**docker rmi -f**

borra una imagen aunque estuviera usándose el contenedor

**-a o --all**

para borrar todas las imágenes que no están siendo usadas por contenedores

**-f o --force**

para que no nos solicite confirmación

**--filter**

para especificar ciertos filtros a las imágenes.

**docker image prune -a**

 Borrar todas las imágenes sin usar

**docker image inspect**

Obtener información de una imagen

**docker inspect --format '{{.Architecture}} es la arquitectura y el SO es {{.Os}}' mysql:8.0.22**

  Mostrar la arquitectura y el sistema

**docker inspect --format '{{.Config.ExposedPorts}}' mysql:8.0.22**

Mostrar lista de puertos expuestos

**docker image build**

para construir una imagen desde un fichero Dockerfile

**docker image history**

para que se nos muestre por pantalla la evolución de esa imagen.

**docker image save**

para guardar imágenes en fichero y cargarlas desde fichero

**docker image tag ( docker tag)**

para añadir TAGs (versiones) a las distintas imágenes.