volumes

intro

- persistencia : si lanzamos un contenedor indicando un volumen sin dar el path del equipo local, montará una estructura en **/var/lib/docker/volumes/<volumen>/_data**
- BBDD: `./init.sql:/docker-entrypoint-initdb.d/init.sql` : si carecemos de un volumen que de persistencia, con esto montaríamos el sistema para que al levantar el contenedor se estableciese los valores **mínimos**
- no deja de ser un alias entre el directorio creado

comandos

- `docker volume ls`
- `docker volume inspect <volume>` :información JSON del volumen
- `docker run -v <volumen>:/data …` : vincula el volumen al contenedor
- `docker volume rm <volumen>` : elimina el volumen
  - si está en el path por defecto **/var/lib/docker/volumes…** lo elimina todo
  - si es un volumen creado en otra ruta (comando de abajo) no eliminará nada (aunque el volumen ya no estará disponible desde **docker volume**
- `docker volume create –driver local –opt type=none –opt o=bind –opt device=/home/vagrant/vol <volumen>`
- `docker volume create <volumen>`
- `docker volume create –driver local –opt type=nfs –opt o=addr=10.0.0.2,rw –opt device=:/path nfs <volumen>`
  - no recomendado, como mucho desarrollo
- `docker run -v nfs:/app`
  - `docker run -v nfs:/app:ro`
  - `docker run -v nfs:/app:rw`
- `docker volume prune` : borra volúmenes que no tienen contenedor asociado
  - si estan fuera del default_path, no lo borra (tampoco los de NFS)

From:
https://matebcn.eu/ - **miguel angel torres egea**

Permanent link:
https://matebcn.eu/info:cursos:altran:docker:volumes2

Last update: **17/07/2018 18:08**