

Copia BBDD Docker a local y restauración a nuevo contenedor

Nuestra base de datos está **ligada al contenedor** de Docker por lo que si eliminamos el contenedor, perderemos la base de datos entera. La idea es **externalizar la base de datos**(en este caso el host) para poder importarla en cualquier contenedor:

Método 1

1. crearemos con un comando un script que almacene esas bases de datos desde nuestro contenedor en el bash:
`mysqldump -u root -p root --all-databases > /root/backup.sql`
2. exportaremos del contenedor al host el script.(desde la consola del host para guardar el archivo más fácilmente poniendo el . al final):
`docker cp f_mysql:/root/backup.sql .`
3. crearemos una imagen de nuestro contenedor y la activaremos:
`docker commit f_mysql mi_mysql`
`docker run --name ejercicio -e MYSQL_ROOT_PASSWORD=root -p 3366:3306 -dt mi_mysql`
4. copiaremos nuestro script al contenedor nuevo y lo ejecutaremos para la reconstrucción de las bases de datos:
`docker exec -i ejercicio mysql -uroot -proot < backup.sql`

Método 2

1. crearemos con un comando un script que almacene esas bases de datos desde nuestro contenedor en el bash:
`mysqldump -u root -p root --all-databases > /root/backup.sql`
2. exportaremos del contenedor al host el script.(desde la consola del host para guardar el archivo más fácilmente poniendo el . al final):
`docker cp f_mysql:/root/backup.sql .`
3. crearemos una imagen de nuestro contenedor y la activaremos:
`docker commit f_mysql mi_mysql`
4. docker build es a partir de un archivo que contiene los comandos para construir una imagen.Crearemos un archivo con editor de texto(asegurándonos de que el texto no tiene formato) que contenga:
`FROM mi_mysql`
`COPY backup.sql /docker-entrypoint-initdb.d/`
5. salvamos el archivo sin extensión y ejecutaremos desde consola del host estando en la ruta donde contenemos el archivo de texto:
`docker build -t f_mysql_datos .`
6. levantaremos un contenedor a partir de esa nueva imagen:
`docker run --name ejercicios -p3366:3306 -d nombreimagen`