

# Isma's Blog

Blog sobre tecnologías de desarrollo

## Usando MySQL con Docker

### Introducción

En muchas ocasiones, cuando estamos desarrollando una aplicación, necesitamos persistir información en una base de datos, generalmente.

Y para hacer nuestras pruebas manuales o automáticas necesitamos una base de datos más completa que una simple base de datos en memoria como H2 o HSQL, y recurrimos a instalar en nuestra máquina local bases de datos como MySQLServer, Postgres, etc... Esto, entre otras cosas, tiene dos desventajas principales:

- nos llena nuestra máquina de software que usaremos puntualmente
- nos hace perder tiempo en descargar e instalar el producto, etc...

Con Docker, tenemos una solución más limpia a estas desventajas. Existen imágenes Docker oficiales de muchos servidores de base de datos, basta con tener instalado Docker en nuestro equipo y bajarnos la imagen adecuada del servidor de base de datos que deseemos. En este post mostraré cómo tener una imagen Docker con MySQLServer.

### Instalando la imagen Docker

En este punto, supongo que ya tenemos instalado Docker en nuestro equipo.

A continuación descargamos la imagen Docker de MySQL con el siguiente comando

```
$ docker pull mysql:5.7.17
```

Este proceso puede tardar algunos minutos, y una vez descargada la imagen podemos comprobar que la tenemos instalada con el comando

```
$ docker images
```

y veremos las imágenes que tenemos en nuestra máquina Docker

```
$ docker images
```

| REPOSITORY | TAG    | IMAGE ID     | CREATED    | SIZE     |
|------------|--------|--------------|------------|----------|
| postgres   | 9.6.1  | 0e24dd8079dc | 6 days ago | 264.9 MB |
| mysql      | 5.7.17 | 594dc21de8de | 7 days ago | 400.2 MB |

- El siguiente paso es ejecutar la imagen **mysql** y para ello ejecutamos el comando

```
$ docker run --name container-name -e MYSQL_ROOT_PASSWORD=secret -p 3306:3306 -d mysql:5.7.17
```

Con este comando estamos ejecutando la imagen **mysql** con el tag **5.7.1** y se crea un contenedor con el nombre **container-name** y expone el puerto 3306 para que podamos conectarnos con un cliente a la base de datos. Además, cuando se crea la instancia de MySQL, ésta se crea por defecto con el usuario **root** y la password la que hayamos establecido en **MYSQL\_ROOT\_PASSWORD**.

## Conectar a la base de datos

En cualquier momento nos puede interesar conectarnos con la base de datos levantada en el contenedor. Tenemos dos opciones para realizar esto:

- por línea de comandos
- con un cliente

### Por línea de comandos

Con el comando

```
$ docker exec -it my-container-name mysql -uroot -p
```

donde **my-container-name** es el nombre del contenedor de la base de datos MySQL, y después de introducir la password del usuario **root** deberíamos acceder a la consola de mysql para manipular la base de datos.

```
$ winpty docker exec -it mysql mysql -uroot -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 169
Server version: 5.7.17 MySQL Community Server (GPL)

Copyright (c) 2000, 2016, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

### Con un cliente MySQL

Para conectar con un cliente MySQL de nuestra elección debemos conocer el host o IP del contenedor donde está la base de datos. Este host o IP es la de la máquina Docker donde se está ejecutando el contenedor, y ésto lo podemos averiguar utilizando el comando

```
$ docker-machine ip machine-name
```

donde **machine-name** es el nombre de la máquina Docker.

Con esta información ya podemos configurar nuestro cliente para conectar con la base de datos.

## Ver los logs del servidor de base de datos

Para ver los logs del servidor de base de datos debemos ejecutar el siguiente comando para acceder al contenedor

```
$ docker exec -it my-container-name bash
```

done **my-container-name** es el nombre del contenedor. De esta forma accederemos a la consola del contenedor, y en **/var/log/mysqld.log**

**Nota:** En entorno Windows probablemente necesites iniciar el comando con **winpty**.

Anuncios

Esta entrada fue publicada en Uncategorized el diciembre 21, 2016

[<https://pajarokillo.wordpress.com/2016/12/21/usando-mysql-con-docker/>] .

