

Creación y manejo de contenedores  
utilizando Docker

# Examen Primer Parcial

Base de datos

RAFAEL ALEJANDRO RAYGOSA MENA

---



## Contenido

Objetivo.....	3
Alcance .....	3
Proceso.....	4
Descripción del código en Docker-compose: .....	4
Pasos .....	5
Diferencias entre procesos de instalación .....	6
Contenedor .....	6
Host .....	6
Nube.....	6
Conclusiones .....	7

## **Objetivo**

El objetivo del documento es explicar detalladamente como es el proceso para montar un contenedor utilizando las herramientas de Docker, el cual contendrá un gestor de base de datos. Para eso se utilizará un archivo llamado Docker-Compose.yml que contiene todas las características de los servicios del contenedor, así como especificaciones del mismo

## **Alcance**

Primeramente, es necesario tener instalado el servicio de Docker en tu dispositivo, con el cual se montará el contenedor en el archivo yml. Este archivo puede ser modificado utilizando un editor de texto o código. Para montarlo, es necesario seguir una serie de comandos en CMD mientras se ejecuta Docker.

## Proceso

### Descripción del código en Docker-compose:

version: "3.3" //version del archivo Docker

services: // lista de servicios que va a contenedor

basedatos:

image: mariadb //imagen que se monta en el contenedor

ports:

- "3806:3306" //Puerto que se utilizará para conectarse a la bd

environment: //es El entorno donde se va a trabajar

- MYSQL\_ROOT\_PASSWORD=123456 //contraseña del administrador

- MYSQL\_DATABASE=myfirstdb //Nombre que tendrá la base de datos

- MYSQL\_USER=myfirstuser //Nombre de usuario que pueda acceder a la bd

- MYSQL\_PASSWORD=myfirstpassword //contraseña para acceder a la bd

volumes: // direcciones donde se pondrán y extraerán datos

- "./config:/etc/mysql/conf.d" // configuración del contenedor

- "./files:/var/lib/mysql" //dirección donde guardará las bds

- "./log:/var/log/mysql" // registros de auditoria

## Pasos

Primeramente, en ventana de comandos se accede a la carpeta donde estará el archivo y se ejecutará el comando:

```
C:\clase>docker-compose up -d
```

Con esto el contenedor ya estará creado, siendo posible correrlo desde ventana de comandos o directamente en la app de Docker. Después con el siguiente comando se deberá acceder a los archivos del contenedor:

```
C:\clase>docker exec -i -t clase_basedatos_1 /bin/bash
```

Desde aquí nos dirigiremos a la carpeta etc/mysql para extraer el archivo my.cnf que será guardado dentro de la carpeta Config en la misma dirección donde esté el archivo yml.

Después se sigue con los pasos para la configuración de los registros. Para esto se escribió la línea de código que dice “. /log:/var/log/mysql” en el archivo yml. Después en el archivo my.cnf se debe descomentar las siguientes líneas:

```
53 general_log_file = /var/log/mysql/mysql.log.
```

```
60 log_error = /var/log/mysql/error.log)
```

```
62 slow_query_log_file = /var/log/mysql/mariadb-slow.log
```

```
72-log_bin = /var/log/mysql/mysql-bin.log
```

Con todo esto listo, es posible crear respaldos de los datos de la base de datos en el contenedor con el siguiente comando:

```
C:\clase>docker exec -i clase_basedatos_1 mysqldump -u root -p123456 myfirstdb > respaldo1.sql
```

## **Diferencias entre procesos de instalación**

### **Contenedor**

En lugar de virtualizar la pila de hardware como en el enfoque de las máquinas virtuales, los contenedores realizan la virtualización en el nivel del sistema operativo, con varios contenedores que se ejecutan directamente en el kernel del SO. Esto significa que los contenedores son mucho más livianos: comparten el kernel del SO, se inician con bastante más rapidez y usan una fracción de la memoria en comparación con el arranque de todo el SO.

### **Host**

A diferencia de otros tipos de base de datos en este caso todo está concentrado en un mismo equipo, lo que le confiere un gran sentido de la unidad. Se utiliza una CPU individual, una máquina y un entorno de trabajo que no depende del uso de otros ordenadores. Así mismo, no requiere el uso de sistemas interconectados entre varios dispositivos, dado que la unidad central lo unifica todo.

### **Nube**

Una base de datos en la nube es una colección de contenido, estructurado o no estructurado, que reside en una plataforma de infraestructura de computación en la nube privada, pública o híbrida. Las ventajas que ofrece son la eliminación de una infraestructura física y el ahorro de costos.

### **Conclusiones**

Actualmente tener una base de datos directamente en el host podría ser menos eficaz ya que utilizando los contenedores es más fácil utilizar varios servicios para su administración sin tener la necesidad de instalar muchas aplicaciones, además que el uso de contenedores consume menos recursos ya que este lo distribuye entre cada uno