



Marcos López

Melisa Ait

ASIXB

# Pràctica 10

## Bases de dades en Docker

### Lliuraments

Els resultats d'aquesta part de la pràctica s'hauran d'entregar en **format PDF** i l'entrega pot ser a través de GIT\* o el Moodle.

\* S'ha d'entregar l'enllaç del GIT al Moodle.

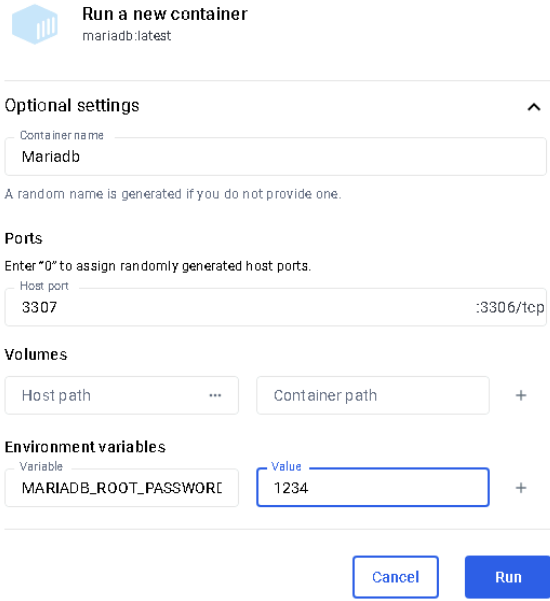
### MariaDB en un Docker

1. Crea un contenidor Docker de **MariaDB** i executa'l al **port 3307**. Pots descarregar una imatge de MariaDB del DockerHub.

Pots consultar la documentació de la imatge de docker de MariaDB:

[https://hub.docker.com/\\_/mariadb](https://hub.docker.com/_/mariadb)

A més, recorda que has d'establir la variable d'entorn **MARIADB\_ROOT\_PASSWORD** i escollir una contrasenya que després faràs servir per connectar-te a la base de dades.



**Run a new container**  
mariadb:latest

**Optional settings** ^

Container name  
Mariadb  
A random name is generated if you do not provide one.

**Ports**  
Enter "0" to assign randomly generated host ports.  
Host port  
3307 :3306/tcp

**Volumes**  
Host path ... Container path +

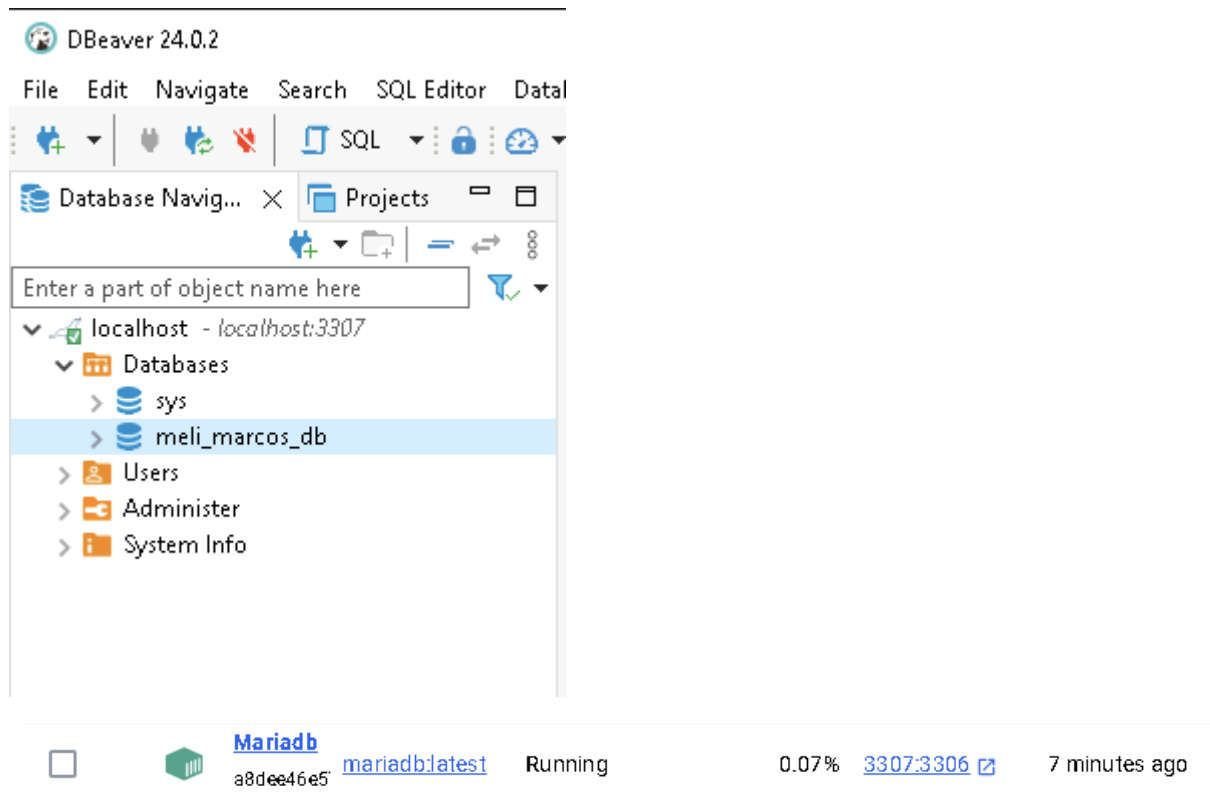
**Environment variables**  
Variable Value  
MARIADB\_ROOT\_PASSWORD 1234 +

Cancel Run

2. Instal·la't **DBeaver Community**<sup>1</sup> i connecta't al port **3307** del localhost, on s'executa el servei de la base de dades. **Contraseña: la misma que pusimos en Dockers**
3. Crea una base de dades amb el **nom dels integrants del grup**.
4. **Entrega una screenshot** on es vegi la base de dades i el contenedor docker.

---

<sup>1</sup>DBeaver Community: <https://dbeaver.io/>



## MariaDB en un Docker mitjançant un fitxer YML

5. Crea un contenidor Docker de **MariaDB** fent servir un **fitxer de configuració YML**.
  - a. Crea una [carpeta buida al teu ordinador](#).
  - b. [Afegeix-hi un fitxer amb el nom \*\*docker-compose.yml\*\* en \*\*Virtual Studio Code\*\*](#)
  - c. **Modifica** el següent contingut perquè tingui la configuració per crear un contenidor a partir de la imatge de MariaDB i s'executi al **port 3308** i el **nom** del contenidor sigui ***mariadb-sense-volum***.

```
version: "3.9"
services:
  nom_servei:
    image: url_imatge_dockerhub
    container_name: nom_del_contenidor
    environment:
```

```
#variables d'entorn si calen
ports:
  - port_teva_maquina:port_servei_docker
```

<input type="checkbox"/>		<a href="#">mariadb-sense-volum</a>	Running (1/1)	0.02%	19 seconds ago				
<input type="checkbox"/>		<a href="#">nmariadb-sense-volum</a> 7bbbb0618070	<a href="#">mariadb:latest</a> Running	0.02%	<a href="#">3309:3306</a>	19 seconds ago			

- d. Executa la següent instrucció i comprova, amb DBeaver, que et pots connectar a la base de dades del 3308.

```
docker-compose up -d
```

```
C:\Users\argel\OneDrive\Documents\Asix\lenguaje de marcas\DOCKER\mariadb-sense-volum>docker-compose up -d
[+] Running 1/1
✓ Container nmariadb-sense-volum Started 0.3s
C:\Users\argel\OneDrive\Documents\Asix\lenguaje de marcas\DOCKER\mariadb-sense-volum>
```

- e. **Entrega una captura** del fitxer **docker-compose.yml**. Títol de la captura: **compose\_mariadb**

```
version: "3.9"
services:
  mariadb_test:
    image: mariadb:latest
    container_name: nmariadb-sense-volum
    environment:
      MARIADB_ROOT_PASSWORD: 1234
    ports:
      - 3309:3306
```

>> 3309:3306

En DBeaver:

(Disconnect)

New Database Connection

Port: 3306 y Password

## MariaDB en un Docker mitjançant YAML amb un volume

DOCKER COMPOSE = Ficher/Guión en virtual studio code o similar repositorio para no tener que configurar cada container manualmente. Importante las tabulaciones.

Siempre se llamará **docker-compose.yml**.

Acceder: En la carpeta del ordenador con el archivo de repositorio que hemos creado, escribiendo cmd en la barra de ruta + en terminal: docker-compose up -d

En DBeaver: conecta > mariadb > localhost > port 3308 > password 1234 > Finish

Volumenes:

Crea una bbdd en dbeaver, para el docker y en dbeaver desconect y conect = error porque docker apagado. Cuando reactivas docker, la bbdd vuelve a aparecer.

Si elimina el docker y vuelve a crearlo con el compose, la bbdd ya no estará. Elimina docker = eliminar todo.

Con volúmenes consigues PERSISTENCIA. La bbdd se guarda aunque se elimine el docker. En Docker Hub web, imagen mariadb oficial image = path a incluir

version: '3.9'

services:

mariadb-volum:

image: mariadb:latest

container\_name: mariadb-volum

environment:

MYSQL\_ROOT\_PASSWORD: 1234

ports:

- 3308:3306

volumes:

- data\_melimarcos:/var/lib/mysql

volumes:

data\_melimarcos:

6. Crea un nou fitxer de configuració **docker-compose** que creï un contenidor de MariaDB amb un **volum**. Tingues en compte que la informació de MariaDB s'emmagatzema a la ruta **/var/lib/mysql** → **dónde almacena mariadb por defecto y permite la persistencia: la información se mantiene ahí aunque la perdamos localmente**

- a. El **nom del container** ha de ser **mariadb-volum**
- b. El **nom del volum** ha de ser **data\_nom\_del\_grup**
- c. **Entrega una captura** del fitxer **docker-compose.yml**. Títol de la captura: **compose\_mariadb\_volume**

```










version: "3.9"
services:
  mariadb-volum:
    image: mariadb:latest
    container_name: mariadb-volum
    environment:
      MYSQL_ROOT_PASSWORD: 1234
    volumes:
      - data_melimarcos:/var/lib/mysql
    ports:
      - 3308:3306
volumes:
  data_melimarcos:

```

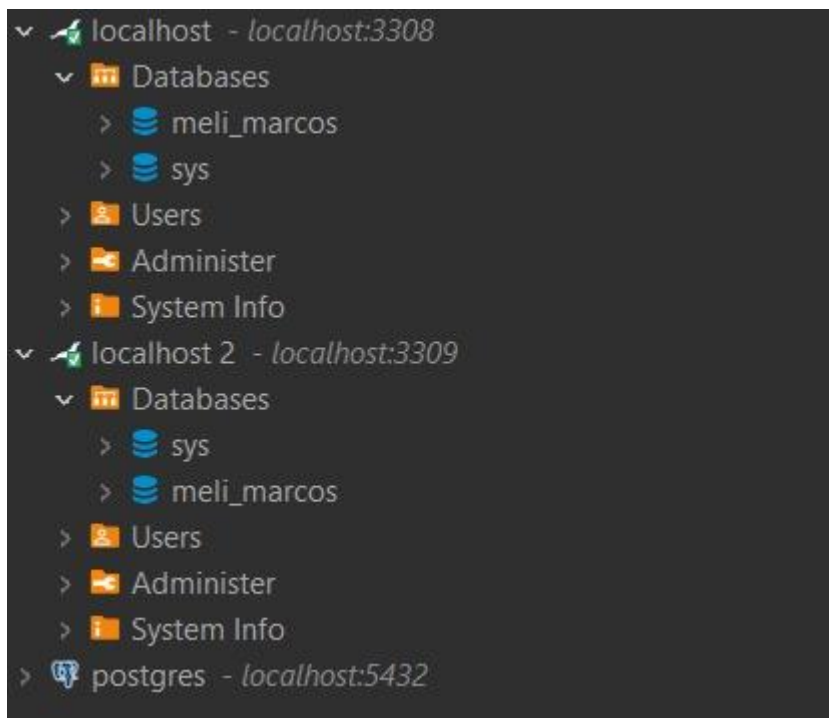
```

C:\Users\argel\OneDrive\Documentos\Asix\Lenguaje de marcas\DOCKER\mariadb_volum>docker-compose up -d
[+] Running 1/3
- Network mariadb_volum_default      Created                                0.6s
- Volume "mariadb_volum_data_melimarcos" Created                            0.5s
✓ Container mariadb-volum            Started                               0.5s
C:\Users\argel\OneDrive\Documentos\Asix\Lenguaje de marcas\DOCKER\mariadb_volum>

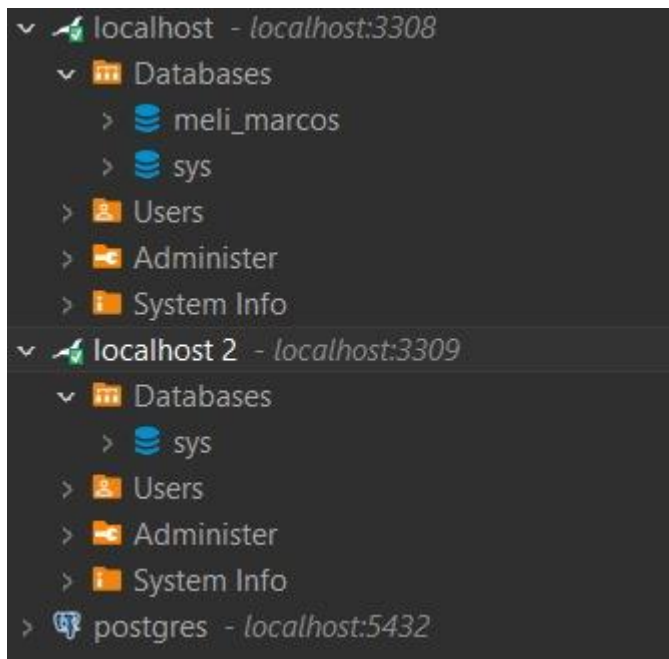
```

<input type="checkbox"/>		<a href="#">mariadb_volum</a>	Running (1/1)	0%	48 seconds ago				
<input type="checkbox"/>		<a href="#">mariadb-volum</a> 5b3d74598107 	<a href="#">mariadb:latest</a>	Running	0% <a href="#">3308:3306</a> 	48 seconds ago			

7. Crea una base de dades al contenidor *mariadb-sense-volum* i una altra a *mariadb-volum*.



8. Suprimeix els dos contenidors dockers i torneu-los a crear.



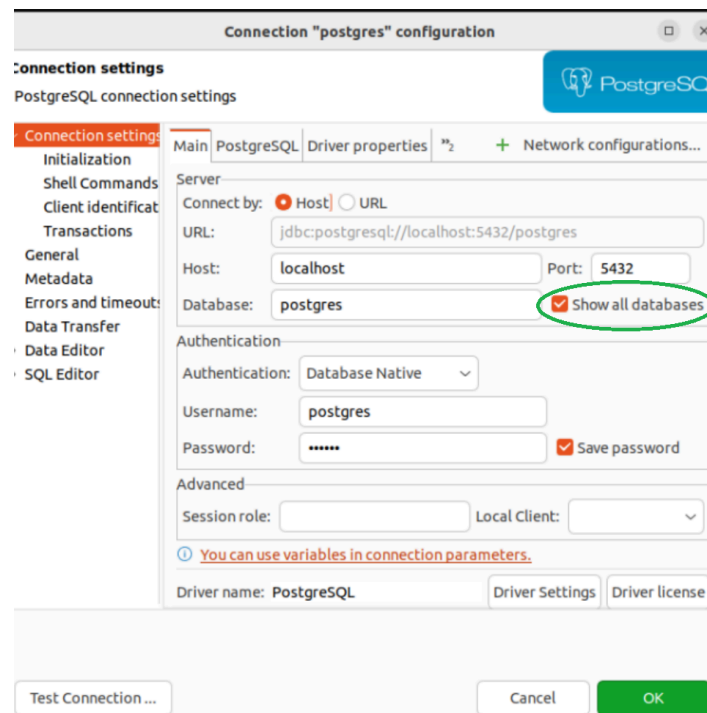
9. **Explica quina diferència** veus una vegada estableixes la connexió als dos dockers.

La base de datos creada con el contenedor mariadb-volum se mantiene a pesar de eliminar el docker, esto es lo que se conoce como **persistencia**. La base de datos en el contenedor sense-volum se elimina al eliminar el docker.

# PostgreSQL en un Docker mitjançant YML amb un volume











10. Crea un nou fitxer de configuració **docker-compose** que creï un contenidor de PostgreSQL amb un **volume**. Busca tota la informació en la documentació del dockerhub i fes servir un port estàndard.

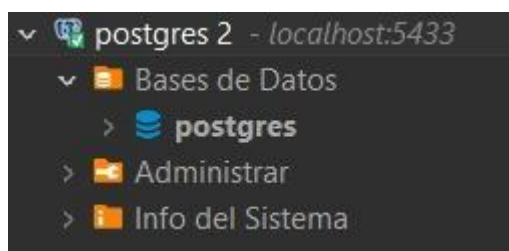
[https://hub.docker.com/\\_/postgres](https://hub.docker.com/_/postgres)



*Ves amb compte tenir marcada l'opció “**show all databases**” quan configuris el postgres. Sense aquesta opció, el DBeaver no mostra totes les bases de dades i sembla que s'esborrin quan es reinicia el docker.*

```
C:\Users\argel\OneDrive\Documentos\Asix\lenguaje de marcas\DOCKER\posgresSQL>docker-compose up -d
[+] Running 1/2
  - Network posgressql_default Created                                0.4s
  ✓ Container posgressql-db-1 Started                                0.3s
C:\Users\argel\OneDrive\Documentos\Asix\lenguaje de marcas\DOCKER\posgresSQL>
```

<input type="checkbox"/>		posgressql	Running (1/1)	0.01%	38 minutes ago				
<input type="checkbox"/>		db-1 e52475361a68 	Running	0.01%	5433:5432 	38 minutes ago			





11. Entrega una captura del fitxer **docker-compose.yml**. Títol de la captura: **compose\_postgres**

Para conseguir el nombre de la imagen ir a buscar en docker hub

version: '3.9'

services:

db:

image: postgres

environment:

POSTGRES\_PASSWORD: 1234

volumes:

- **data\_melimarcos**:/var/lib/mysql

ports:

- 5433:5432

volumes:

**data\_melimarcos**:

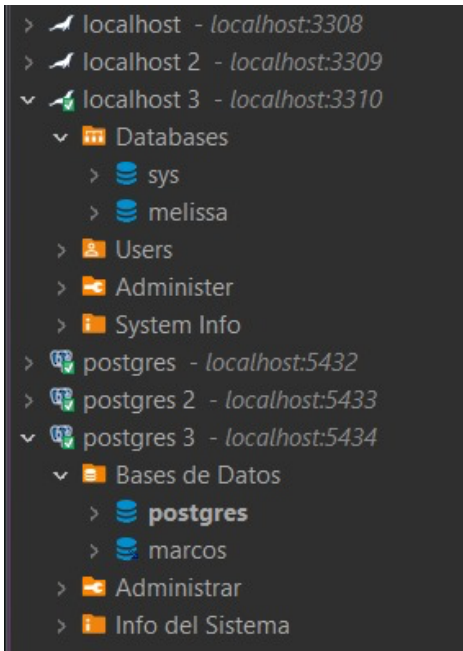
```
version: '3.9'
services:
  db:
    image: postgres
    environment:
      POSTGRES_PASSWORD: 1234
    volumes:
      - data_melimarcos:/var/lib/postgresql/data
    ports:
      - 5433:5432
volumes:
  data_melimarcos:
```

## Docker-compose MariaDB + PostgreSQL

12. Crea un nou fitxer de configuració **docker-compose** que creï un contenidor de **PostgreSQL** i un contenidor **MariaDB** amb un **volum** per cada un.
13. Crea una **base de dades** a cada sistema gestor.
14. Suprimeix els contenidors dockers i torneu-los a crear i comprova que les bases de dades continuen persistint.
15. Entrega una captura del fitxer **docker-compose.yml**. Títol de la captura: **compose\_maria\_postgres**

```
C:\Users\argel\OneDrive\Documentos\Asix\lenguaje de marcas\DOCKER\maria_posgres>docker-compose up -d
[+] Running 2/3
- Network maria_posgres_default Created 0.5s
✓ Container maria_posgres-db-1 Started 0.5s
✓ Container mariadb-posgres Started 0.3s
C:\Users\argel\OneDrive\Documentos\Asix\lenguaje de marcas\DOCKER\maria_posgres>
```

		maria_posgres		Running (2/2)	0.03%	43 seconds ago				
		mariadb-posgres	13a681c97433	<a href="#">mariadb:latest</a>	Running	0.01% <a href="#">3310:3306</a>	43 seconds ago			
		db-1	fbb831e4d979	<a href="#">postgres</a>	Running	0.02% <a href="#">5434:5432</a>	43 seconds ago			



version: '3.9'

services:

db:

image: postgres

environment:

POSTGRES\_PASSWORD: 1234

volumes:

- data\_melimarcos:/var/lib/postgresql/data

ports:

- 5434:5432

mariadb-volume:

image: mariadb:latest

container\_name: mariadb-postgres

environment:

MYSQL\_ROOT\_PASSWORD: 1234

ports:

- 33010:3306

volumes:

- data\_melimarcos2:/var/lib/mysql

volumes:

data\_melimarcos:

data\_melimarcos2:

```
version: '3.9'
services:
  db:
    image: postgres
    environment:
      POSTGRES_PASSWORD: 1234
    volumes:
      - data_melimarcos:/var/lib/postgresql/data
    ports:
      - 5434:5432
  mariadb-volum:
    image: mariadb:latest
    container_name: mariadb-posgres
    environment:
      MYSQL_ROOT_PASSWORD: 1234
    ports:
      - 3310:3306
    volumes:
      - data_melimarcos2:/var/lib/mysql
volumes:
  data_melimarcos:
  data_melimarcos2:
```