



# 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.

```
this_andres@this:~$ sudo docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS                               NAMES
356b04322552   mariadb   "docker-entrypoint.s..." 13 minutes ago Up 40 seconds 3306/tcp, 0.0.0.0:3307->3307/tcp, :::3307->3307/tcp   mi_mariadb
this_andres@this:~$
```

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

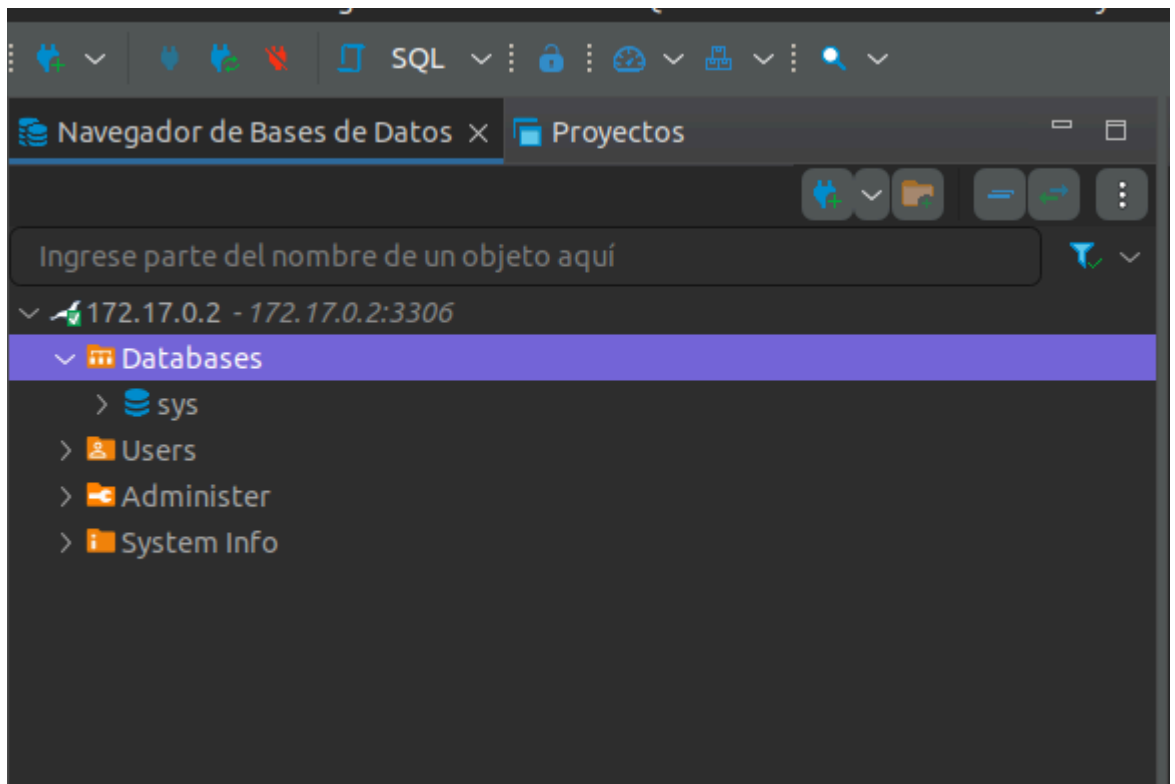
[https://hub.docker.com/\\_/mariadb](https://hub.docker.com/_/mariadb)

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.

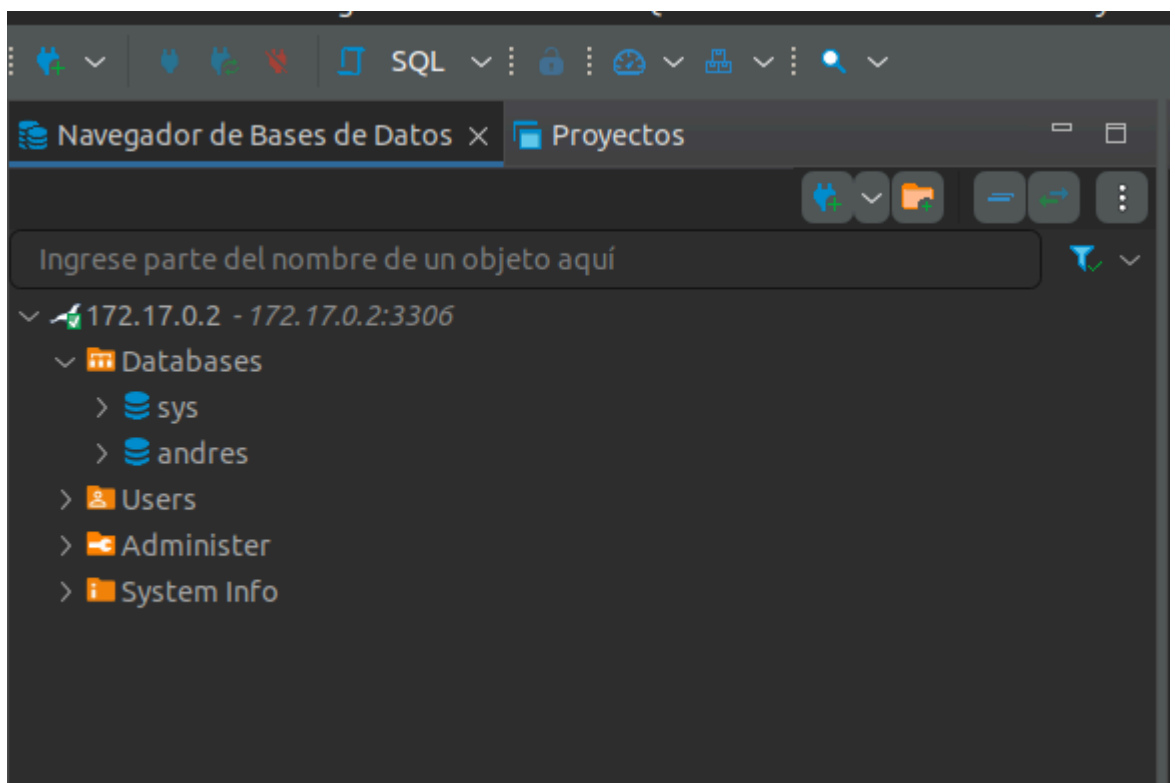
2. Instal·la't **DBeaver Community**<sup>1</sup> i connecta't al port 3307 del localhost, on s'executa el servei de la base de dades.

---

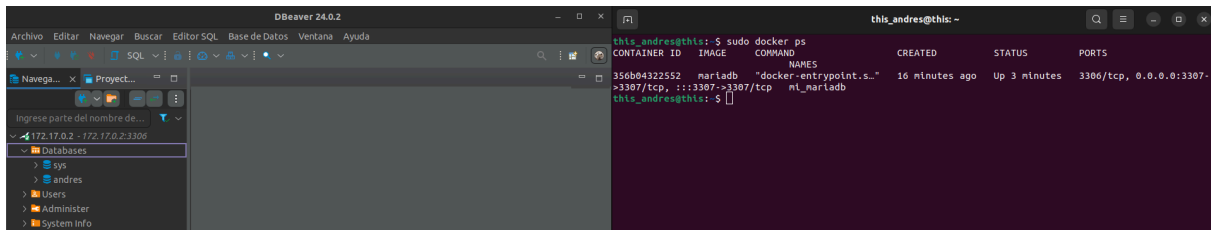
<sup>1</sup>DBeaver Community: <https://dbeaver.io/>



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.



## MariaDB en un Docker mitjançant un fitxer YML

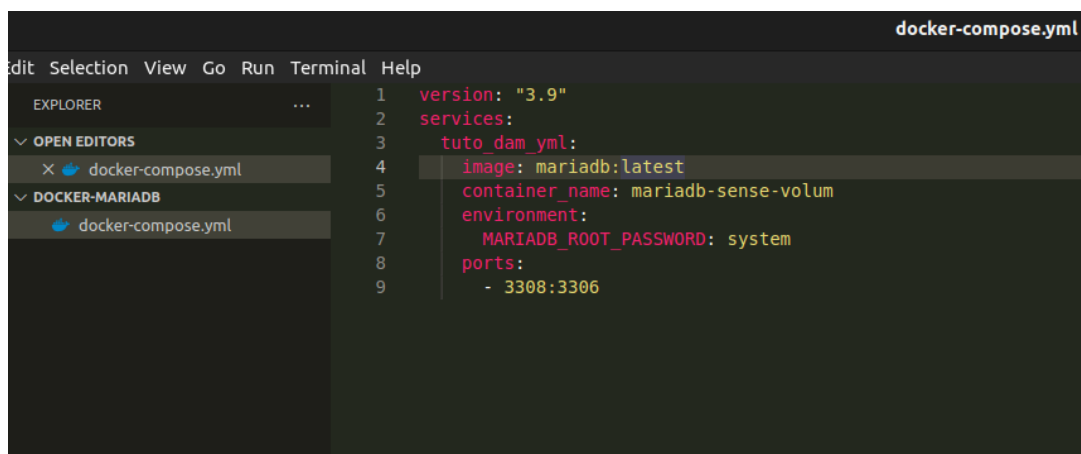
5. Crea un contenidor Docker de **MariaDB** fent servir un **fitxer de configuració YML**.
- Crea una carpeta buida al teu ordinador.
  - Afegeix-hi un fitxer amb el nom **docker-compose.yml**
  - Modifica** el següent contingut perquè tingui la configuració per crear un contenidor a partir de la imatge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

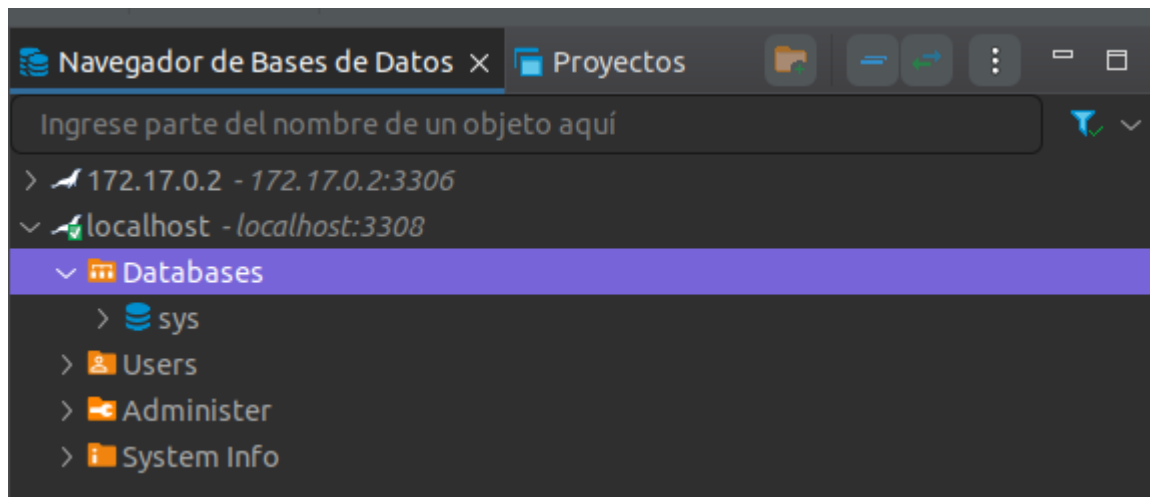
```
version: "3.9"
services:
  nom_servei:
    image: url_imatge_dockerhub
    container_name: nom_del_contenidor
    environment:
      #variables d'entorn si calen
    ports:
      - port_teva_maquina:port_servei_docker
```

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

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

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





## MariaDB en un Docker mitjançant YAML amb un volume

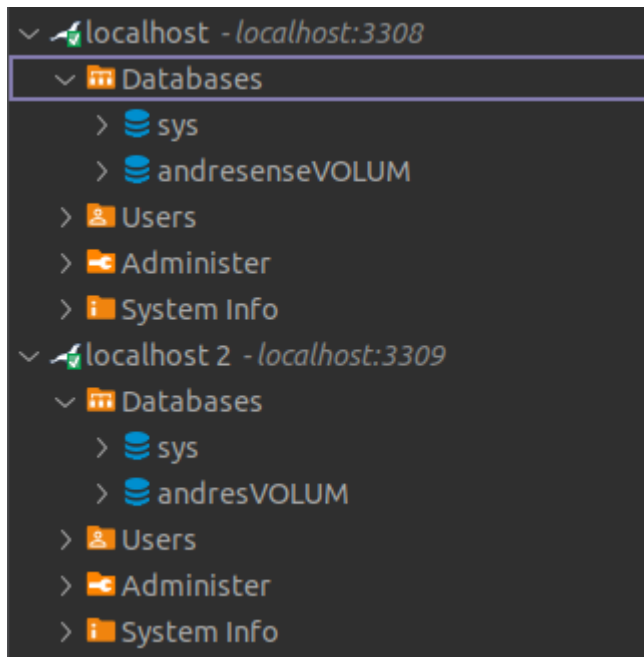
6. Crea un nou fitxer de configuració **docker-compose** que creï un contenidor de MariaDB amb un **volum**. Tingues en compte que la informació de MariaDB s'emmagatzema a la ruta **/var/lib/mysql**
  - a. El nom del container ha de ser ***mariadb-volum***
  - b. El nom del volum ha de ser ***data\_nom\_del\_grup***
  - c. **Entrega una captura** del fitxer **docker-compose.yml**. Títol de la captura: **compose\_mariadb\_volume**

**compose\_mariadb\_volume:**

```
EXPLORER    ...
OPEN EDITORS
  X docker-compose.yml
DOCKER-MARIADB2
  docker-compose.yml

1  version: "3.9"
2  services:
3    tuto_dam_yaml:
4      image: mariadb:latest
5      container_name: mariadb-volum
6      volumes:
7        - data_nom_del_grup:/var/lib/mysql
8      environment:
9        MARIADB_ROOT_PASSWORD: system
10     ports:
11       - 3309:3306
12
13  volumes:
14    data_nom_del_grup:
15
```

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.

```
this_andres@this:~/docker-mariadb2$ sudo docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS
813a42443887   mariadb:latest "docker-entrypoint.s..." 2 minutes ago  Up 2 minutes  0.0.0.0:
3309->3306/tcp, :::3309->3306/tcp
66b0dc181b46   mariadb:latest "docker-entrypoint.s..." 27 minutes ago  Up 27 minutes  0.0.0.0:
3308->3306/tcp, :::3308->3306/tcp
356b04322552   mariadb       "docker-entrypoint.s..." 50 minutes ago  Up 38 minutes  3306/tcp
, 0.0.0.0:3307->3307/tcp, :::3307->3307/tcp
7617772ce47c   hello-world    "/hello"                 About an hour ago  Exited (0) About an hour ago
vibrant_meninsky

this_andres@this:~/docker-mariadb2$ sudo docker rm mariadb-sense-volum
Error response from daemon: You cannot remove a running container 66b0dc181b46c0bf7bf3bc4d47a7bdc26da2b6dd1f3c62715b
82eab3cc2c71c8. Stop the container before attempting removal or force remove
this_andres@this:~/docker-mariadb2$ sudo docker rm -f mariadb-sense-volum
mariadb-sense-volum
this_andres@this:~/docker-mariadb2$ sudo docker rm -f mariadb-volum
mariadb-volum
this_andres@this:~/docker-mariadb2$ sudo docker-compose up -d
[+] Running 1/1
 ✓ Container mariadb-volum Started
0.0s

this_andres@this:~/docker-mariadb2$ cd ../docker-mariadb/
this_andres@this:~/docker-mariadb$ sudo docker-compose up -d
[+] Running 1/1
 ✓ Container mariadb-sense-volum Started
0.0s

this_andres@this:~/docker-mariadb$
```

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

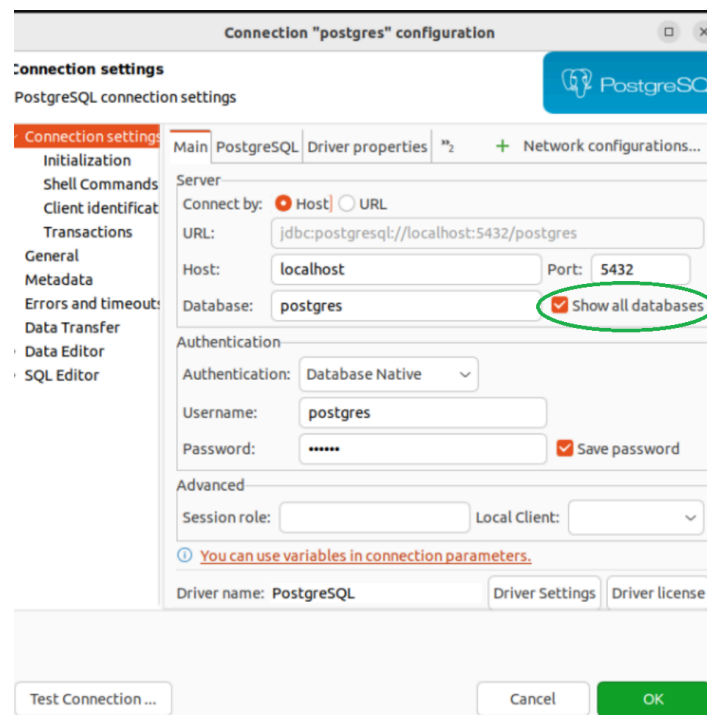
Quan tens un contenidor amb volum, tot allò que creem en aquest contenidor, com tenim un volum, tot es guardarà inclusivament si esborrem el contenidor.

En canvi, en el contenidor sense volum, les bases de dades que creem no es guardaran en cap lloc i quan esborrem el contenidor, s'esborrarà tot.

# 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 **volum**. Busca tota la informació en la documentació del dockerhub i fes servir un port estàndard.

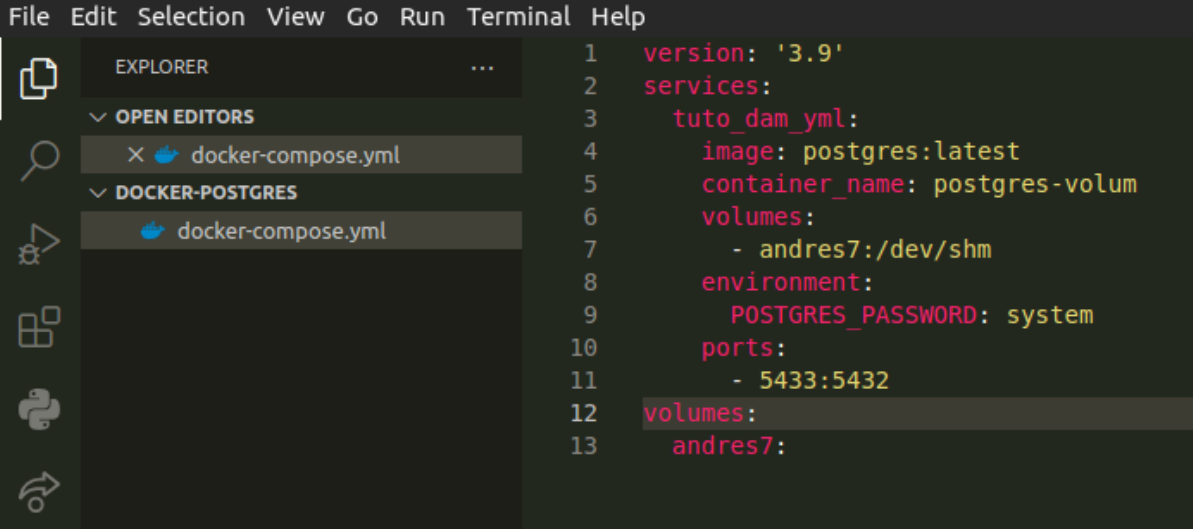
[https://hub.docker.com/\\_/postgres](https://hub.docker.com/_/postgres)



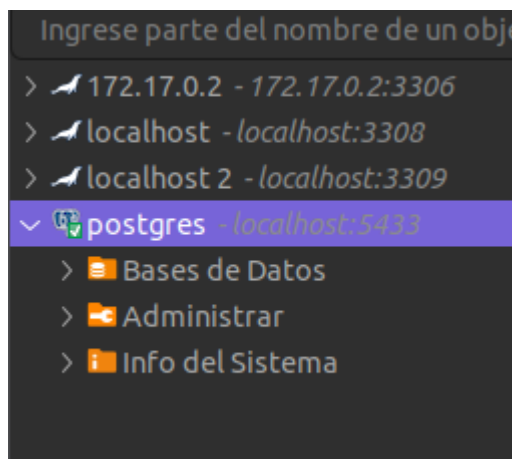
**Ves amb compte** tenir marcada l'opció **“show all databases”** quan configuris el postgres. Sense aquesta opció, el DBeaver no mostra totes les bases de dades i sembla que s'esborrin quan es reinicia el docker.

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

**compose\_postgres:**



```
File Edit Selection View Go Run Terminal Help
EXPLORER
OPEN EDITORS
  X docker-compose.yml
DOCKER-POSTGRES
  docker-compose.yml
1 version: '3.9'
2 services:
3   tuto_dam_yml:
4     image: postgres:latest
5     container_name: postgres-volum
6     volumes:
7       - andres7:/dev/shm
8     environment:
9       POSTGRES_PASSWORD: system
10    ports:
11      - 5433:5432
12 volumes:
13   andres7:
```



## 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**

### compose\_maria\_postgres:

```
EXPLORER ...
└─ OPEN EDITORS
  └─ docker-compose.yml
  └─ DOCKER-MARIPOST
    └─ docker-compose.yml

1 version: "3.9"
2 services:
3   postgres:
4     image: postgres:latest
5     container_name: postgres-volum2
6     environment:
7       POSTGRES_PASSWORD: system
8     volumes:
9       - 16_andres_postgres:/dev/shm
10    ports:
11      - 5435:5432
12
13   mariadb:
14     image: mariadb:latest
15     container_name: mariadb-volum2
16     environment:
17       MARIADB_ROOT_PASSWORD: system
18     volumes:
19       - 16_andres:/var/lib/mysql
20    ports:
21      - 3312:3306
22
23 volumes:
24   16_andres_postgres:
25   16_andres:
26
```

