



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

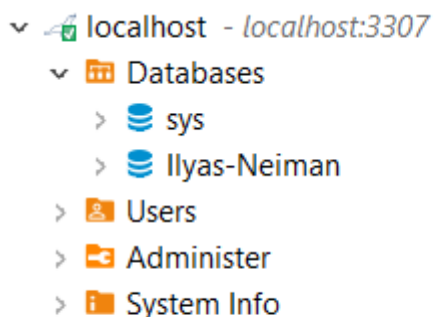
1. Crea un contenidor Docker de **MariaDB** i executa'l al **port 3307**. Pots descarregar una imatge de MariaDB del DockerHub.

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

https://hub.docker.com/_/mariadb

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**¹ i connecta't al port 3307 del localhost, on s'executa el servei de la base de dades.
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 contenidor docker.



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

MariaDB en un Docker mitjançant un fitxer YML

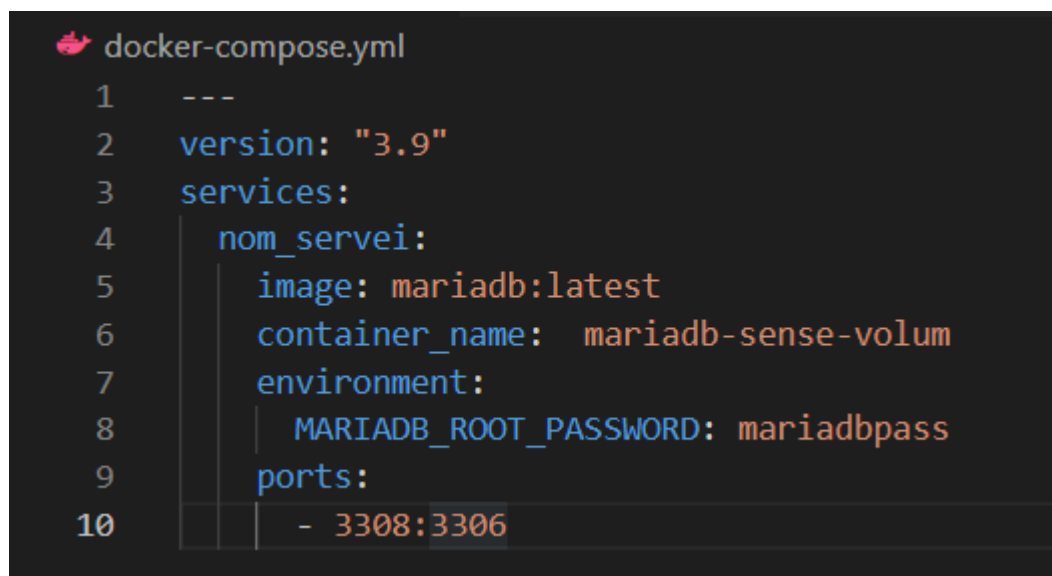
5. Crea un contenidor Docker de **MariaDB** fent servir un **fitxer de configuració YML**.
 - a. Crea una carpeta buida al teu ordinador.
 - b. Afegeix-hi un fitxer amb el nom **docker-compose.yml**
 - c. **Modifica** el següent contingut perquè tingui la configuració per crear un contenidor a partir 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
```

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

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

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



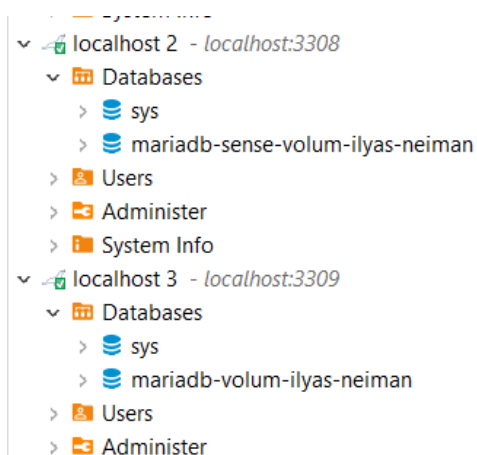
```
🐳 docker-compose.yml
1  ---
2  version: "3.9"
3  services:
4    nom_servei:
5      image: mariadb:latest
6      container_name: mariadb-sense-volum
7      environment:
8        MARIADB_ROOT_PASSWORD: mariadbpass
9      ports:
10     - 3308:3306
```

MariaDB en un Docker mitjançant YML amb un volume

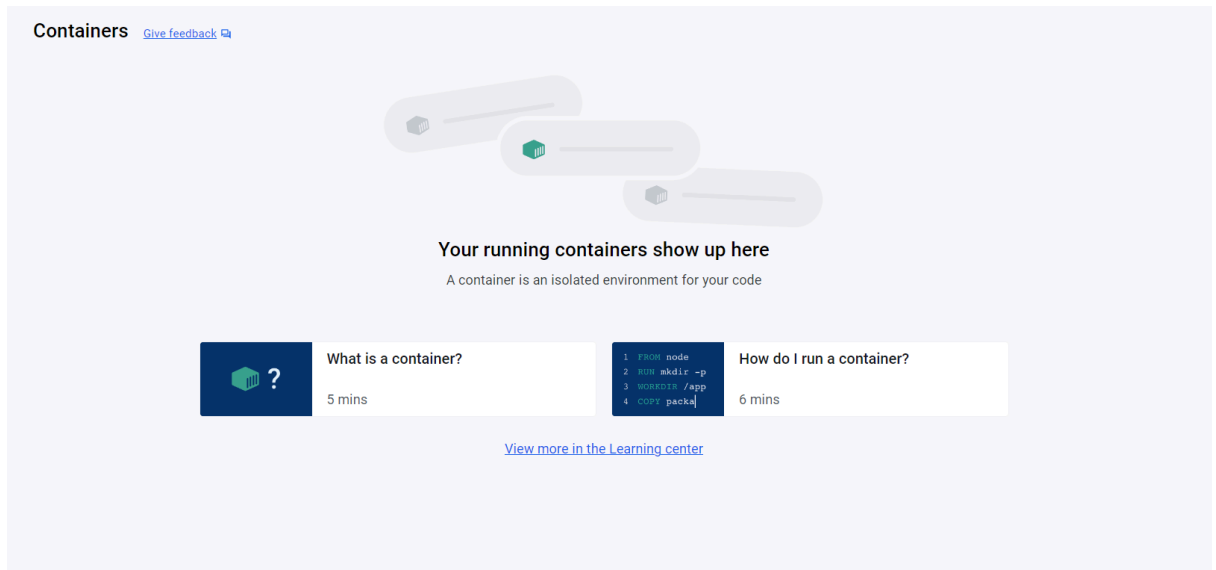
6. Crea un nou fitxer de configuració **docker-compose** que creï un contenidor de MariaDB amb un **volume**. Tingues en compte que la informació de MariaDB s'emmagatzema a la ruta **/var/lib/mysql**
 - a. El **nom del container** ha de ser ***mariadb-volum***
 - b. El **nom del volum** ha de ser ***data_nom_del_grup***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

```
docker-compose.yml
1  ---
2  version: "3.9"
3
4  services:
5
6
7    db:
8      container_name: mariadb-sense-volum2
9      image: mariadb:latest
10     volumes:
11       - Ilyas_Neiman2:/var/lib/mysql
12     environment:
13       MARIADB_ROOT_PASSWORD: mariadbpass
14     ports:
15       - 3309:3306
16   volumes:
17     Ilyas_Neiman2:
```

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.



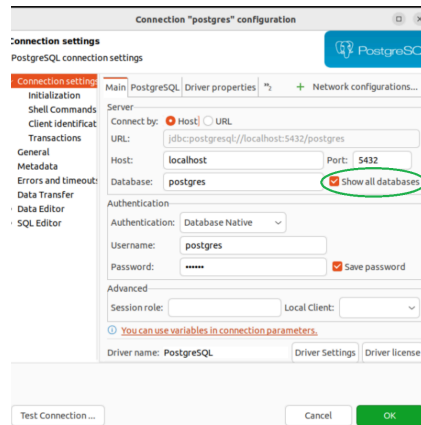
9. **Explica quina diferència veus una vegada estableixes la connexió als dos dockers.**

El contenidor amb volum conserva la base de dades en canvi el contenidor sense volum no ho fa.

PostgreSQL en un Docker mitjançant YML amb un volum

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



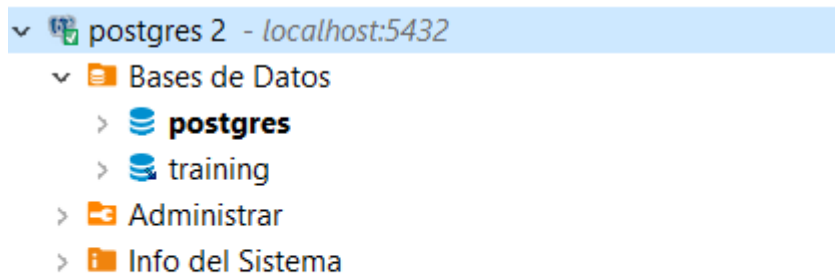
Ves amb compte tenir marcada l'opció **“show all databases”** quan configuris el postgres. Sense aquesta opció, el DBeaver no mostra totes les bases de dades i sembla que s'esborrin quan es reinicia el docker.

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

```

1
2 version: '3.9'
3
4 services:
5
6   db:
7     container_name: my_postgres
8     image: postgres:latest
9     environment:
10      POSTGRES_PASSWORD: postgrespass
11   adminer:
12     image: adminer
13     restart: always
14     ports:
15      - 8080:8080
16     volumes:
17      - Ilyas_Neiman2:/var/lib/postgresql/data
18 volumes:
19   Ilyas_Neiman2:
20
21


```




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`


```
docker-compose.yml X
docker-compose.yml
1  version: '3.9'
2
3  services:
4    postgres:
5      container_name: my_postgres_Ilyas_Neiman
6      image: postgres:latest
7      environment:
8        POSTGRES_PASSWORD: pirineus
9      ports:
10       - 5432:5432
11      volumes:
12       - Ilyas_Neiman2:/var/lib/postgresql/data
13
14    mariadb:
15      container_name: mariadbvolum_Ilyas_Neiman
16      image: mariadb:latest
17      environment:
18        MARIADB_ROOT_PASSWORD: mariadbpass
19      ports:
20       - 3308:3306
21      volumes:
22       - Ilyas_Neiman2_MARIADB:/var/lib/postgresql/data
23
24  volumes:
25    Ilyas_Neiman2:
26    Ilyas_Neiman2_MARIADB:
```


▼  localhost - localhost:3308


▼  Databases


>  sys

>  Users


>  Administer


>  System Info


▼  postgres - localhost:5432

▼  Bases de Datos

>  **postgres**

>  training

>  Administrar

>  Info del Sistema