



Pràctica 10

Bases de dades en Docker

Grup 1r DAM B: Lishi JL, Andreu Merino, Carolina Melguizo, Jordi Tijeras

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.

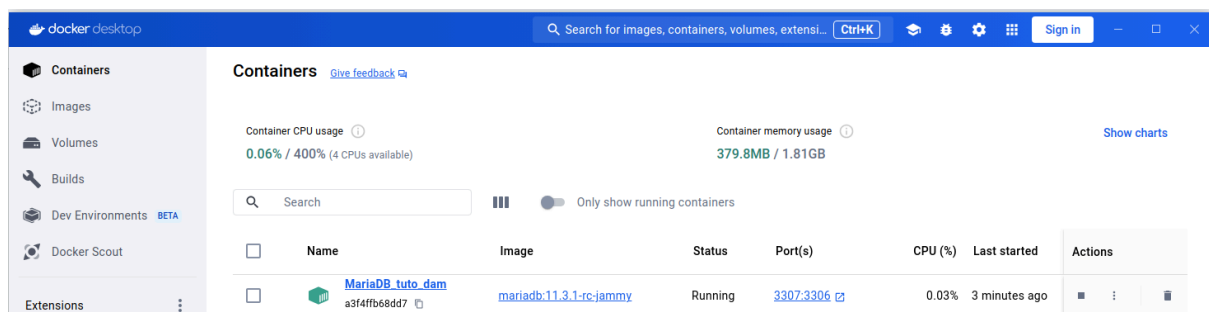
MariaDB en un Docker

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

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

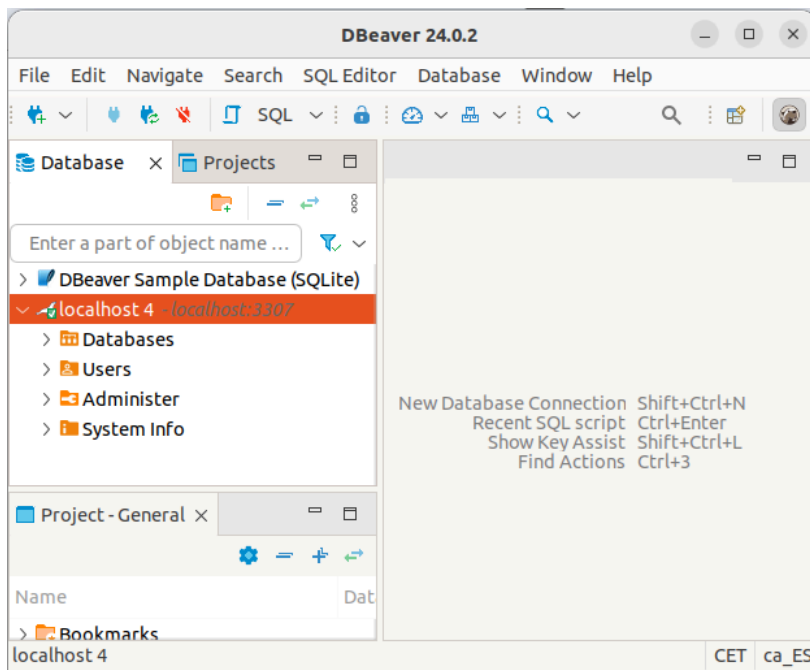
https://hub.docker.com/_/mariadb

A més, recorda que has d'establir la variable d'entorn **MARIADB_ROOT_PASSWORD** i escollir una contrasenya(1234) que després faràs servir per connectar-te a la base de dades.

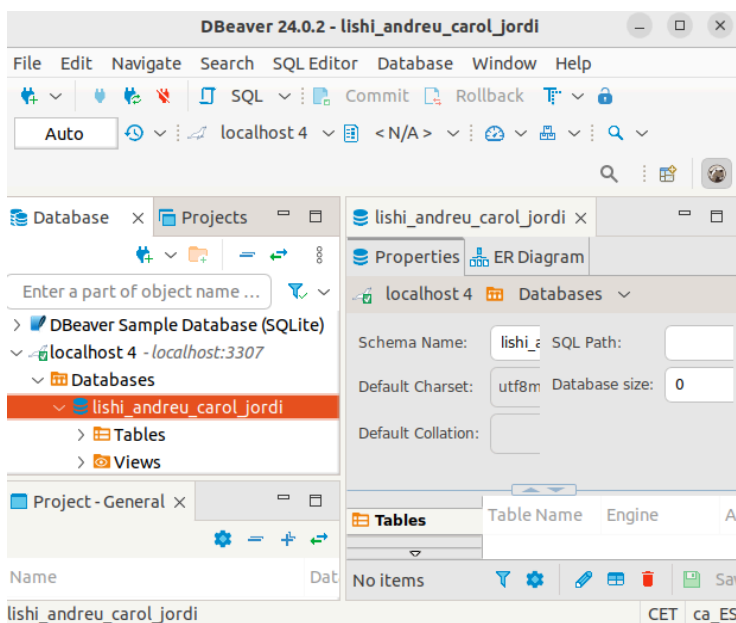


2. Instal·la't **DBeaver Community**¹ i connecta't al port 3307 del localhost, on s'executa el servei de la base de dades.

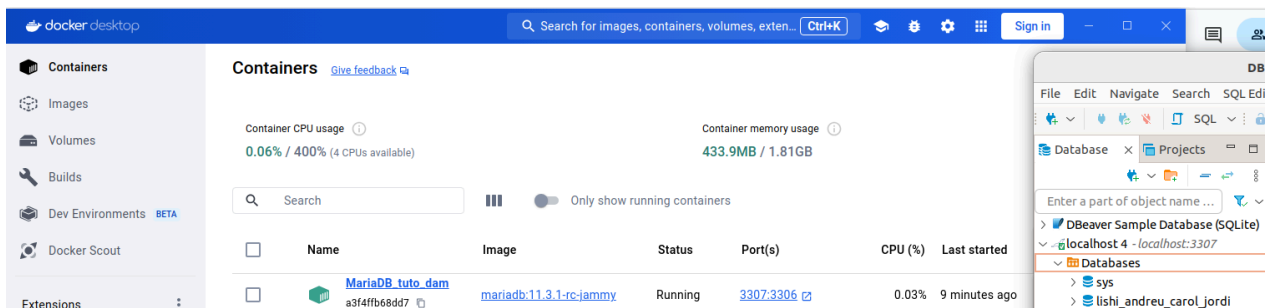
¹DBeaver Community: <https://dbeaver.io/>



3. Crea una base de dades amb el nom dels integrants del grup.



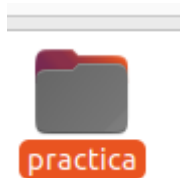
4. Entrega una screenshot on es vegi la base de dades i el contenedor docker.



MariaDB en un Docker mitjançant un fitxer YML

5. Crea un contenidor Docker de **MariaDB** fent servir un **fitxer de configuració YML**.

a. Crea una carpeta buida al teu ordinador.



b. Afegeix-hi un fitxer amb el nom **docker-compose.yml**

```
isard@ubuntu:~/Baixades/practica$ ls
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
```

```
isard@ubuntu:~/Baixades/practica$ cat docker-compose.yml
version: "3.9"
services:
  tuto_dam_yaml:
    image: mariadb:latest
    container_name: mariadb-sense-volum
    environment:
      #variables d'entorn si calen
      MARIADB_ROOT_PASSWORD: 1234
    ports:
      - 3308:3306
```

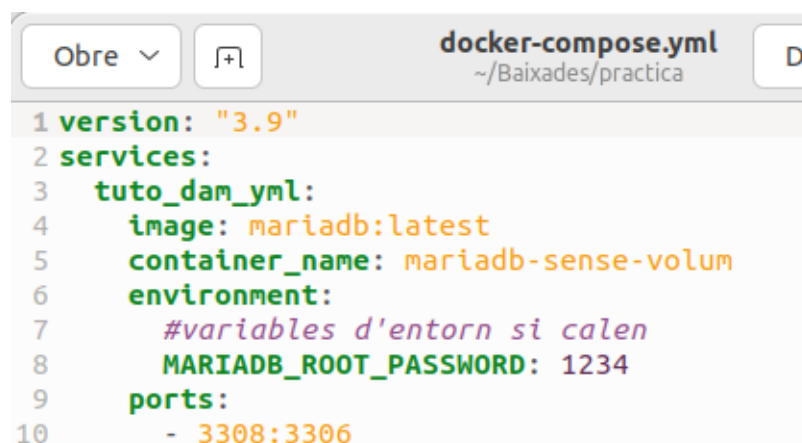
d. Executa la següent instrucció i comprova, amb DBeaver, que et pots connectar a la base de dades del port 3308.

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

```
lsard@ubuntu:~/practica$ docker-compose up -d
Creating network "practica_default" with the default driver
Pulling tuto_dam_yml (mariadb:latest)...
latest: Pulling from library/mariadb
3c645031de29: Pull complete
b9a09dbb5569: Pull complete
3785a15fbe9c: Pull complete
e06cee351ba0: Pull complete
e3993f93abd6: Pull complete
67b5fb2a7a58: Pull complete
76b744a46697: Pull complete
9f2ab02ffe77: Pull complete
Digest: sha256:78d0c3b8c39b47e91cc32df64ef1e26f797b54b1dd762c850e7350095ed4715f
Status: Downloaded newer image for mariadb:latest
Creating mariadb-sense-volum ... done
```

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

`compose_mariadb`



```
1 version: "3.9"
2 services:
3   tuto_dam_yml:
4     image: mariadb:latest
5     container_name: mariadb-sense-volum
6     environment:
7       #variables d'entorn si calen
8       MARIADB_ROOT_PASSWORD: 1234
9     ports:
10      - 3308:3306
```

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`

```

Obre  ▾  docker-compose.yml
~/Baixades/mariadb-volum

1 version: "3.9"
2 services:
3   tuto_dam_ym1:
4     image: mariadb:latest
5     container_name: mariadb-volum
6     environment:
7       #variables d'entorn si calen
8       MARIADB_ROOT_PASSWORD: 1234
9     ports:
10      - 3309:3306
11
12     volumes:
13      - 16CALJ:/var/lib/mysql
14 volumes:
15   16CALJ:

```

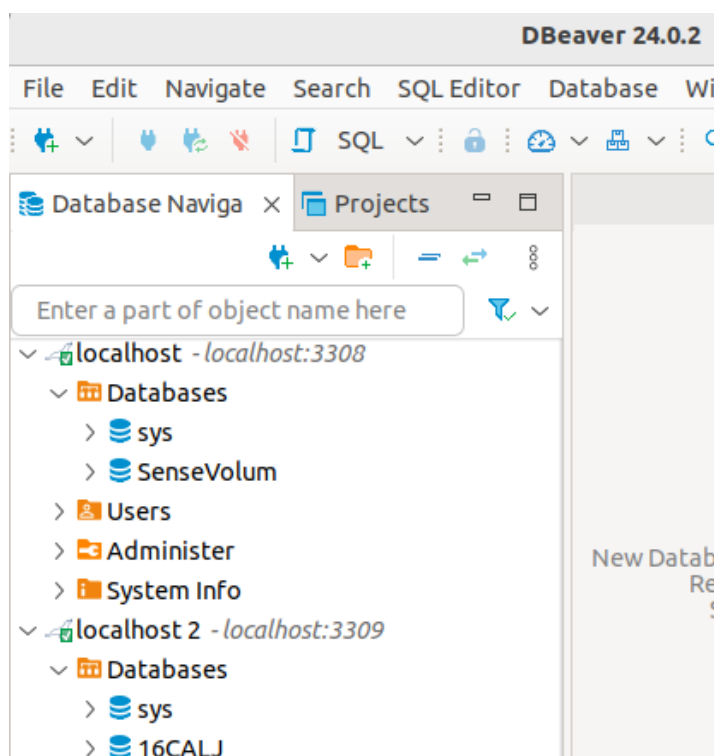
```

isard@ubuntu:~/Baixades/mariadb-volum$ docker-compose up -d
Creating network "mariadb-volum_default" with the default driver
Creating volume "mariadb-volum_CALJ" with default driver
Creating mariadb-volum ... done

```

<input type="checkbox"/>	practica	Running (1/1)	
<input type="checkbox"/>	mariadb-sense-volum 596c736a6fbc	mariadb:lat Running 3308:3306	
<input type="checkbox"/>	mariadb-volum	Running (1/1)	
<input type="checkbox"/>	mariadb-volum e1162ade6dac	mariadb:lat Running 3309:3306	

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.

The screenshot shows the Docker Desktop interface. In the first part, a modal dialog titled "Delete compose project?" is displayed over a list of projects. The dialog states: "The 'practica' compose project is selected for deletion." Below the text are two buttons: "Cancel" and "Delete forever". The background shows a list of Docker projects including "MariaDB_tuto_dam", "practica", "mariadb-sense-volum", "mariadb-volum", and "mariadb-volum".

In the second part, another modal dialog is shown: "Delete compose project?" with the text: "The 'mariadb-volum' compose project is selected for deletion." It also has "Cancel" and "Delete forever" buttons. The background shows the same list of projects, but "practica" is no longer visible.

The third part shows the Docker Desktop interface after the deletions. The "practica" and "mariadb-volum" projects are no longer listed. The remaining projects are "MariaDB_tuto_dam" and "mariadb-volum".

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

The top part of the screenshot shows terminal commands for connecting to Docker containers:

```
> localhost -localhost:3308
> localhost 2 -localhost:3309
> localhost 4 -localhost:3307
```

The bottom part shows a terminal window with the following commands and output:

```
isard@ubuntu:~/Baixades/mariadb-volum$ docker-compose up -d
Creating network "mariadb-volum_default" with the default driver
Creating mariadb-volum ... done
isard@ubuntu:~/Baixades/mariadb-volum$ cd ../practica/
isard@ubuntu:~/Baixades/practica$ docker-compose up -d
Creating network "practica_default" with the default driver
Creating mariadb-sense-volum ... done
```

The bottom part shows the Docker Desktop interface with three hosts connected:

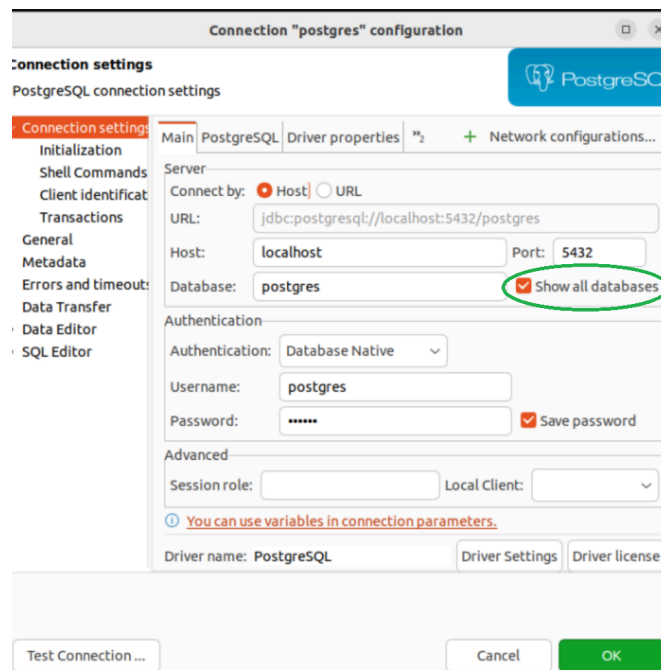
- localhost -localhost:3308
 - Databases
 - sys
- localhost 2 -localhost:3309
 - Databases
 - CALJ
 - sys
 - Users
 - Administer
 - System Info
- localhost 4 -localhost:3307
 - Databases
 - lishi_andreu_carol_jordi

Doncs al esborrar, i tornar a fer docker-compose up -d pels dos dockers i establint connexió, comprovem que al reconectar, una de les bases de dades es conserva que es la de **amb volum** mentre que l'altre s'ha eliminat, pel fet de no tenir **volum** en el fitxer a configurar docker-compose.yml del [localhost:3308](#).

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



Per a la variable d'entorn de la contrasenya del administrador hem utilitzat l'etiqueta POSTGRES_PASSWORD, sense el ROOT com a MariaDB.

```
Obre  docker-compose.yml
~/Baixades/PostgreSQL

1 version: "3.9"
2 services:
3   tuto_dam_yml:
4     image: postgres:latest
5     container_name: postgres-volum
6     environment:
7       #variables d'entorn si calen
8       POSTGRES_PASSWORD: 1234
9     ports:
10      # per mantenir el port estandard 5432 de
11      # Postgres, ho indicarem a la dreta i
12      # un host personalitzat a l'esquerra,
13      # normalment l'estandard es posaria a
14      # l'esquerra si el fem coincidir amb el
15      # port del contenidor, que seria 5432:5432
16      # en aquest cas, hem volgut que el local
17      # fos 3310 per conservar l'ordre de creació
18      # dels dockers.(Canviem el host a 5432)
19      - 5432:5432
20
21   volumes:
22     - PSQL:/var/lib/postgresql/data
23 volumes:
24   PSQL:
```

```
lsard@ubuntu:~/Baixades/PostgreSQL$ docker-compose up -d
Pulling tuto_dam_yml (postgres:latest)...
latest: Pulling from library/postgres
13808c22b207: Pull complete
e59e71301578: Pull complete
30258475ebe6: Pull complete
dc0c37c3ab3c: Pull complete
80df7455ae43: Pull complete
c200c670beef: Pull complete
d051a616a810: Pull complete
87c27dce0c4d: Pull complete
9dae96d37274: Pull complete
9b9e5e72a98e: Pull complete
73301da916ba: Pull complete
673a1691030a: Pull complete
16d03b1cb8d8: Pull complete
9d680d6449b5: Pull complete
Digest: sha256:f4b0987cb4ba8bcc2b90aa33ad8b5786669bec4dc633fc93d1418275e3627b34
Status: Downloaded newer image for postgres:latest
Creating postgres-volum ... done
```

Cal tenir en compte tenir marcada l'opció “show all databases” quan es configura la connexió amb el docker de Postgres. Sense aquesta opció, el DBeaver no mostra totes les bases de dades i sembla que s'esborrin quan es reinicia el docker.

Connect to a database

Connection Settings

PostgreSQL connection settings

PostgreSQL

Main

PostgreSQL

Driver properties

SSH

SSL

+ Network configurations...

Server

Connect by: ☒ Host ☐ URL

URL: jdbc:postgresql://localhost:5432/postgres

Host: localhost Port: 5432

Database: postgres ☒ Show all databases

Authentication

Authentication: Database Native

Username: postgres

Password:

....

☒ Save password

Advanced

Session role:

Local Client: <not present>

You can use variables in connection parameters.

Connection details (name, type, ...)

Driver name: PostgreSQL

Driver Settings

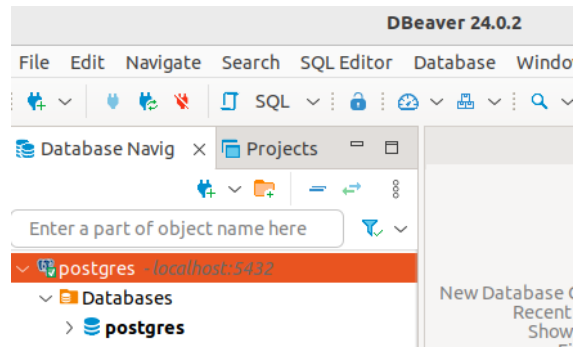
Test Connection ...

< Back

Next >

Cancel

Finish



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

compose_postgres

```

1 version: "3.9"
2 services:
3   tuto_dam_yml:
4     image: postgres:latest
5     container_name: postgres-volum
6     environment:
7       #variables d'entorn si calen
8       POSTGRES_PASSWORD: 1234
9     ports:
10      # per mantenir el port estandard 5432 de
11      # Postgres, ho indicarem a la dreta i
12      # un host personalitzat a l'esquerra,
13      # normalment l'estandard es posaria a
14      # l'esquerra si el fem coincidir amb el
15      # port del contenidor, que seria 5432:5432
16      # en aquest cas, hem volgut que el local
17      # fos 3310 per conservar l'ordre de creació
18      # dels dockers.(Canviem el host a 5432)
19      - 5432:5432
20
21   volumes:
22     - PSQL:/var/lib/postgresql/data
23 volumes:
24   PSQL:
  
```

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.

```

1 version: "3.9"
2
3 services:
4   tuto_dam_yml:
5     image: mariadb:latest
6     container_name: mariadb-volum2
7     environment:
8       #variables d'entorn si calen
9       MARIADB_ROOT_PASSWORD: 1234
10    ports:
11      - 3310:3306
12    volumes:
13      - 16CALJ:/var/lib/mysql
14
15   tuto_dam_yml_pstgs:
16     image: postgres:latest
17     container_name: postgres-volum2
18     environment:
19       POSTGRES_PASSWORD: 1234
20     ports:
21       - 5433:5432
22     volumes:
23       - PSQL:/var/lib/postgresql/data
24
25 volumes:
26   16CALJ:
27   PSQL:
  
```

```
isard@ubuntu:~/Baixades/MariaDB+PostgreSQL$ docker-compose up -d
Creating network "mariadbpostgresql_default" with the default driver
Creating mariadb-volum2 ... done
Creating postgres-volum2 ... done
```

<input type="checkbox"/>		mariadbpostgresql		Running (2/2)	  
<input type="checkbox"/>		postgres-volum2	postgres:la	Running	5433:5432   
<input type="checkbox"/>		mariadb-volum2	mariadb:lat	Running	3310:3306   

Connect to a database

Connection Settings

MariaDB connection settings

Main

Driver properties

SSH

SSL

+ Network configurations

Server

Connect by: ☒ Host ☐ URL

URL:

Server Host: Port:

Database:

Authentication (Database Native)

Username:

Password: ☒ Save password

Advanced

Server Time Zone:

Local Client:

[You can use variables in connection parameters.](#)

Connection details (name, type, ...)

Driver name: MariaDB

Driver Settings

Driver license

Test Connection ...

< Back

Next >

Cancel

Finish

Connect to a database

Connection Settings

PostgreSQL connection settings

Main

PostgreSQL

Driver properties

SSH

SSL

+ Network configurations

Server

Connect by: ☒ Host ☐ URL

URL:

Host: Port:

Database: ☒ Show all databases

Authentication

Authentication:

Username:

Password: ☒ Save password

Advanced

Session role:

Local Client:

[You can use variables in connection parameters.](#)

Connection details (name, type, ...)

Driver name: PostgreSQL

Driver Settings

Driver license

Test Connection ...

< Back

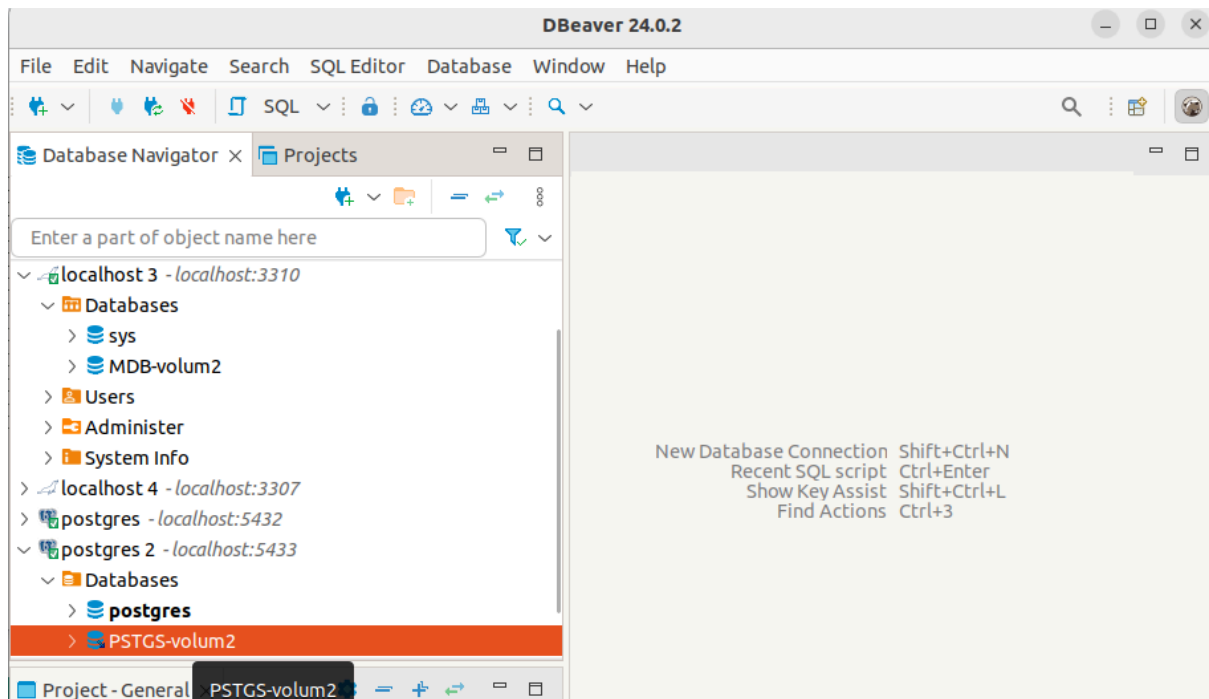
Next >

Cancel

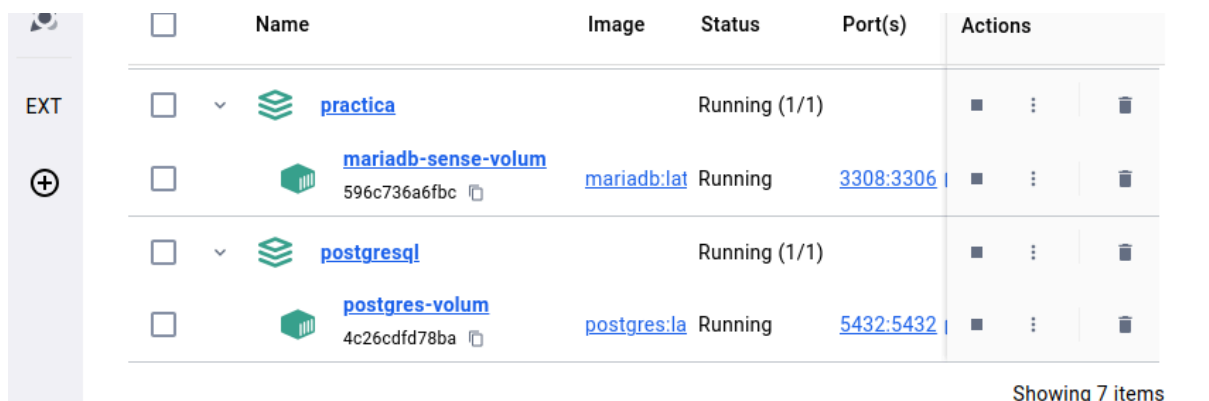
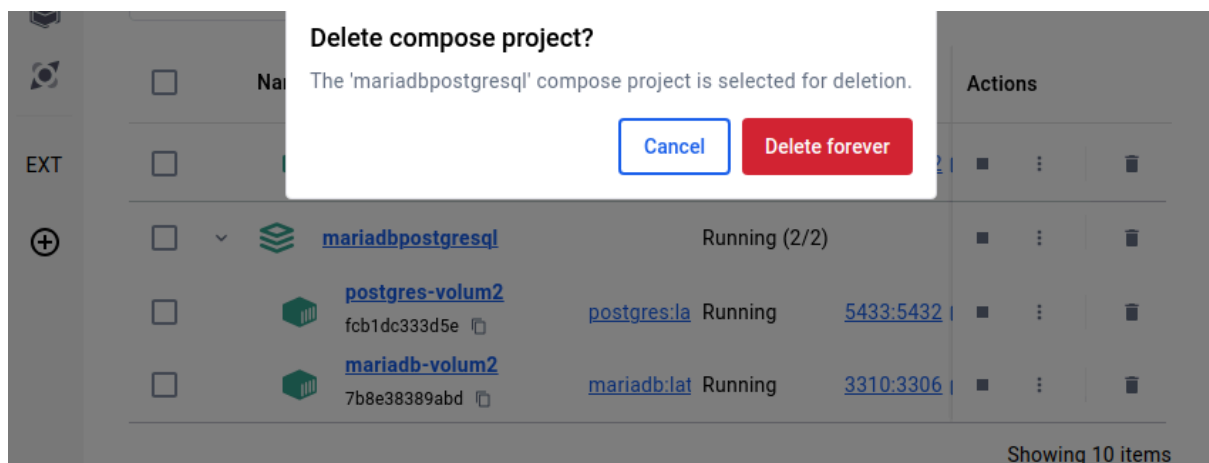
Finish

Show all databases in database navigator.
If not set then only one database will be visible.
This option works in manual connection configuration type only.
Options to show templates or non-available databases will not be disabled.

13. Crea una base de dades a cada sistema gestor. **MDB-volum2** i **PSTGS-volum2**



14. Suprimeix els contenidors dockers i torneu-los a crear i comprova que les bases de dades continuen persistint.



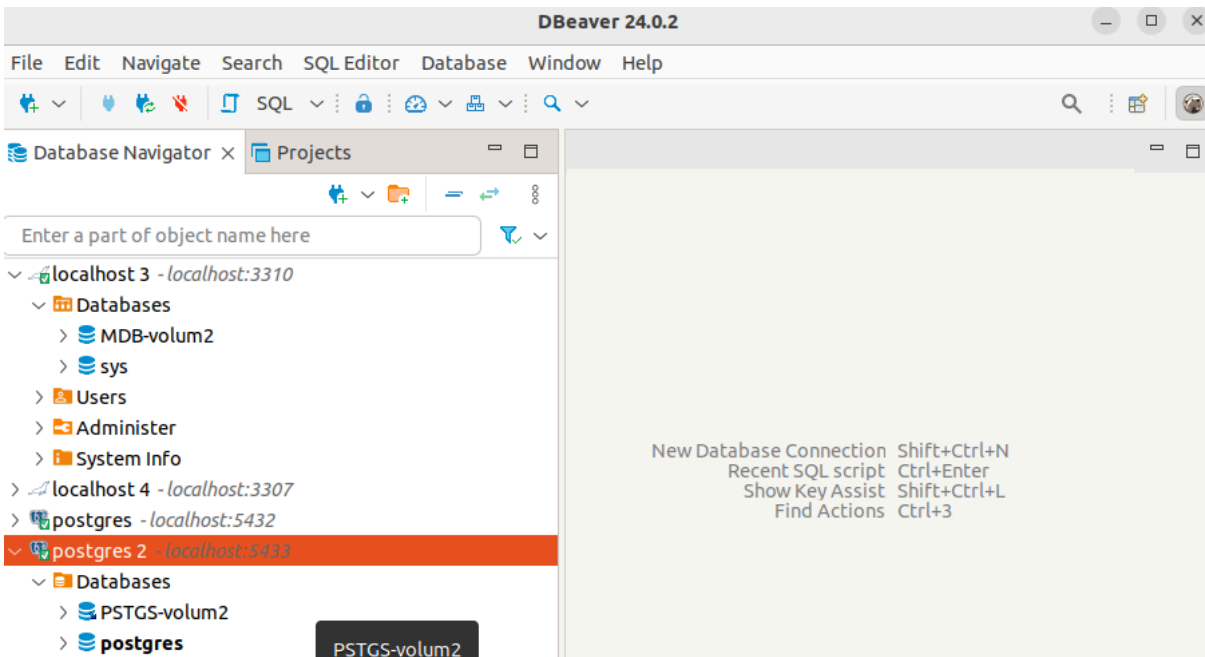
```
> localhost 3 - localhost:3310
> localhost 4 - localhost:3307
> postgres - localhost:5432
> postgres 2 - localhost:5433
```

```
isard@ubuntu:~/Baixades/MariaDB+PostgreSQL$ docker-compose up -d
Creating network "mariadbpostgresql_default" with the default driver
Creating mariadb-volum2 ... done
Creating postgres-volum2 ... done
```

<input type="checkbox"/>		mariadbpostgresql	Running (2/2)			
<input type="checkbox"/>		mariadb-volum2 bb6ac3f765fd	mariadb:lat Running 3310:3306			
<input type="checkbox"/>		postgres-volum2 60b8125f42fe	postgres:la Running 5433:5432			


Showing 10 items

Durant la creació de la connexió amb postgres com que hem marcat mostrar totes les BBDD el reconectar després d'eliminar-se, es conserva la visibilitat de PSTGS-volum2, en cas de que no s'hagués marcat, només seria visible “**postgres**” que apareix en negreta, tot i que la BBDD de PSTGS-volum2 seguiria estant, però que no seria visible.



15. Entrega una captura del fitxer `docker-compose.yml`. Títol de la captura: `compose_maria_postgres`

compose_maria_postgres

```
Obre ▾  docker-compose.yml  
~/Baixades/MariaDB+PostgreSQL
```

```
1 version: "3.9"  
2  
3 services:  
4   tuto_dam_yml:  
5     image: mariadb:latest  
6     container_name: mariadb-volum2  
7     environment:  
8       #variables d'entorn si calen  
9     MARIADB_ROOT_PASSWORD: 1234  
10    ports:  
11      - 3310:3306  
12    volumes:  
13      - 16CALJ:/var/lib/mysql  
14  
15    tuto_dam_yml_pstgs:  
16      image: postgres:latest  
17      container_name: postgres-volum  
18      environment:  
19        POSTGRES_PASSWORD: 1234  
20      ports:  
21        - 5433:5432  
22      volumes:  
23        - PSQL:/var/lib/postgresql/data  
24  
25 volumes:  
26   16CALJ:  
27   PSQL:
```