

Ejercicio 1 - Contenedores en red y Docker Desktop

Adrián García de la Cera

Ejercicio 1 - Contenedores en red y Docker Desktop

Enunciado

1. Creación de red bridge redej1
2. Crea un contenedor con una imagen de `mariaDB`
Definir credenciales
3. Crear un contenedor con `Adminer`
4. Conectar contenedores a redej1
Configurar redes
Conexión a interfaz grafica
Ejecución del script SQL
5. Instalación y comprobaciones con `Disk Usage`
Borrado de volúmenes, contenedores y la red

Enunciado

Este ejercicio se resolverá con Docker Desktop, si se soluciona con comandos la nota será 0

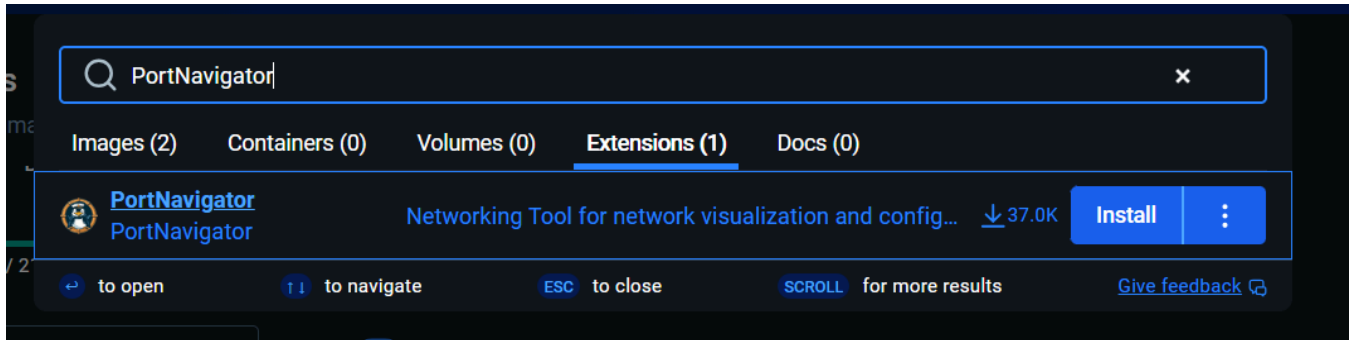
Si necesitas usar comandos, usa la Terminal integrada en Desktop

SUGERENCIA: Utiliza la extensión PortNavigator para gestionar las redes en Docker Desktop

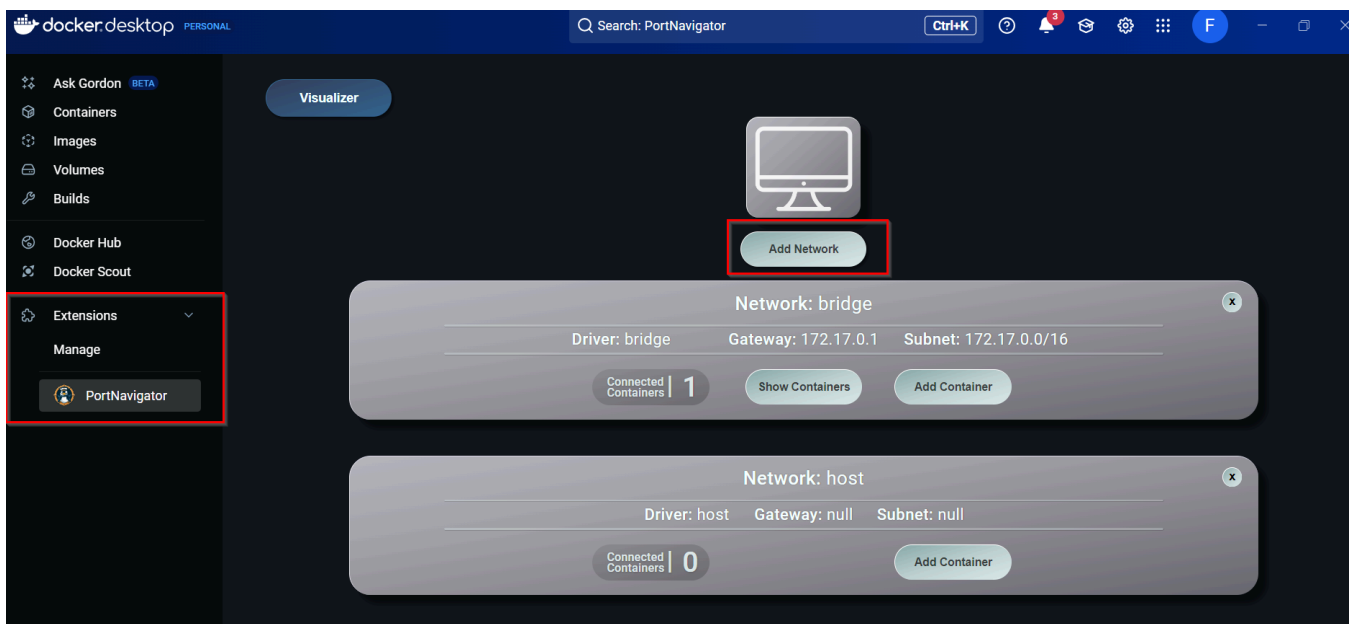
1. Crea una red bridge `redej1`
2. Crea un contenedor con una imagen de `mariaDB` que estará en la red `redej1`. Este contenedor se ejecutará en segundo plano, y será accesible a través del puerto 3306.
 - Definir una contraseña para el usuario `root`, y un usuario con tu nombre de pila y con contraseña. La BD por defecto será `DAW`.
 - Genera un script SQL que cree una tabla `módulos` con algunos registros con los nombres de los módulos que estás estudiando.
3. Crear un contenedor con `Adminer` o con `phpMyAdmin` que se pueda conectar al contenedor de la BD.
4. Desde la interfaz gráfica elegida, conéctate a la BD con tu usuario personal, ejecuta el script con los datos de los módulos y muestra la BD y la tabla creados.
5. Elige entre estas dos opciones:
 - Instala la extensión `Disk Usage`, muestra el espacio ocupado, borra algo...
 - Instala la extensión `Resource usage` y muestra su salida cuando estén activos los contenedores.

1.Creación de red bridge redej1

Para crear la red usaremos como aconseja el título la extensión **PortNavigator** , para ello lo primero que debemos realizar es instalarla, en la parte central superior buscamos la extensión y pulsamos sobre el botón **Install**.



Una vez instalada, nos aparecerá en la parte izquierda, en la sección de extensiones, pulsamos sobre ella y en el botón central "Add Network"



La creamos con nombre **redej1** como solicita el enunciado:

x

Add Network

Name: redej1

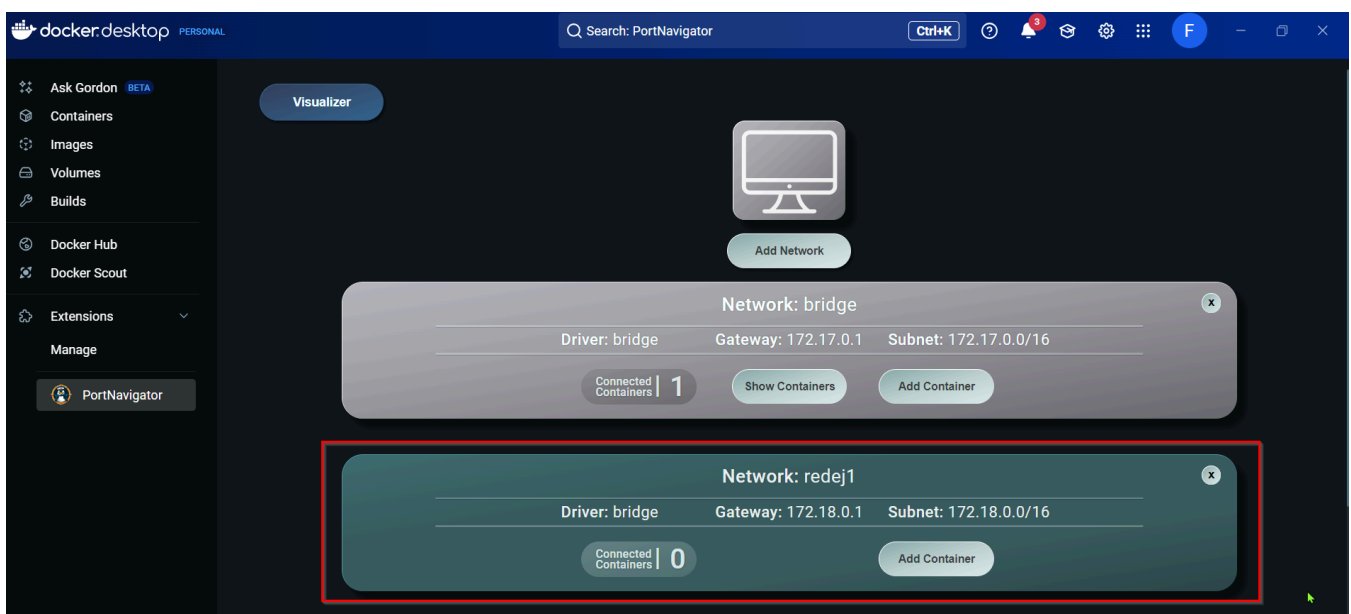
☐ Advanced network settings

Subnetwork: Add a subnetwork (optional)

Gateway: Add a gateway (optional)

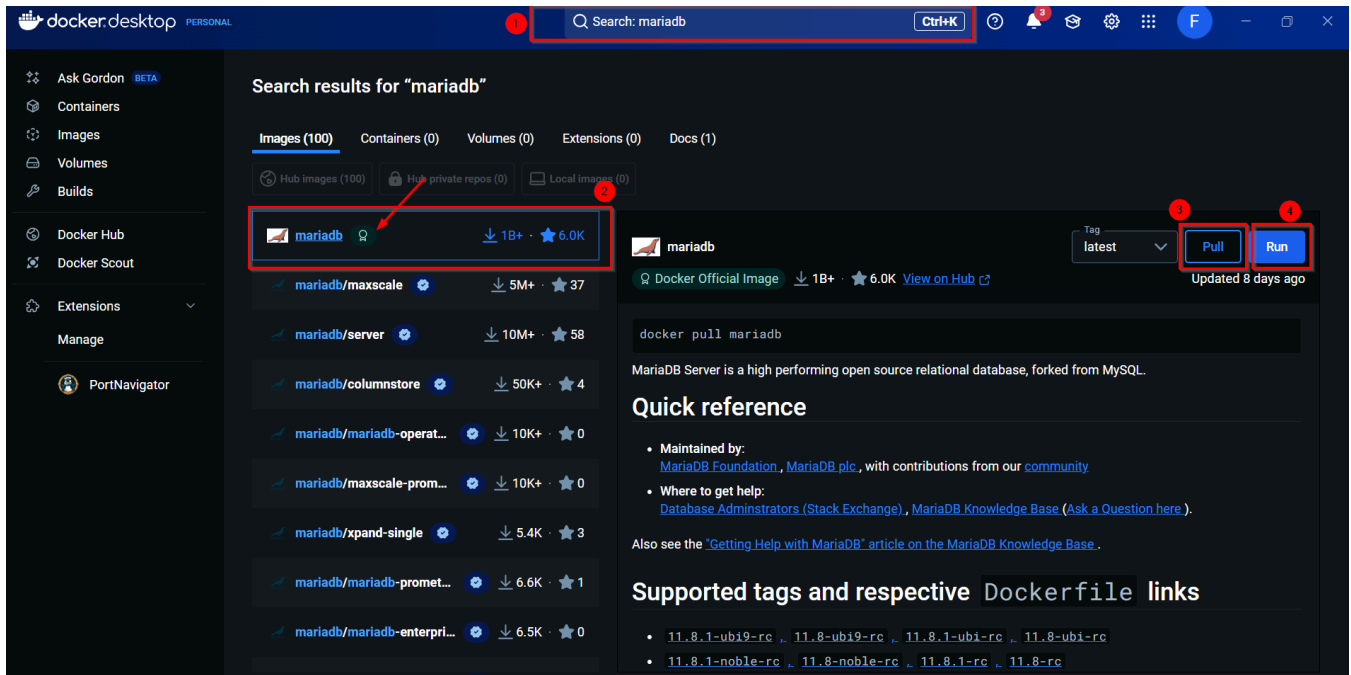
IP-Range: Add an IP-Range (optional)

Create Network

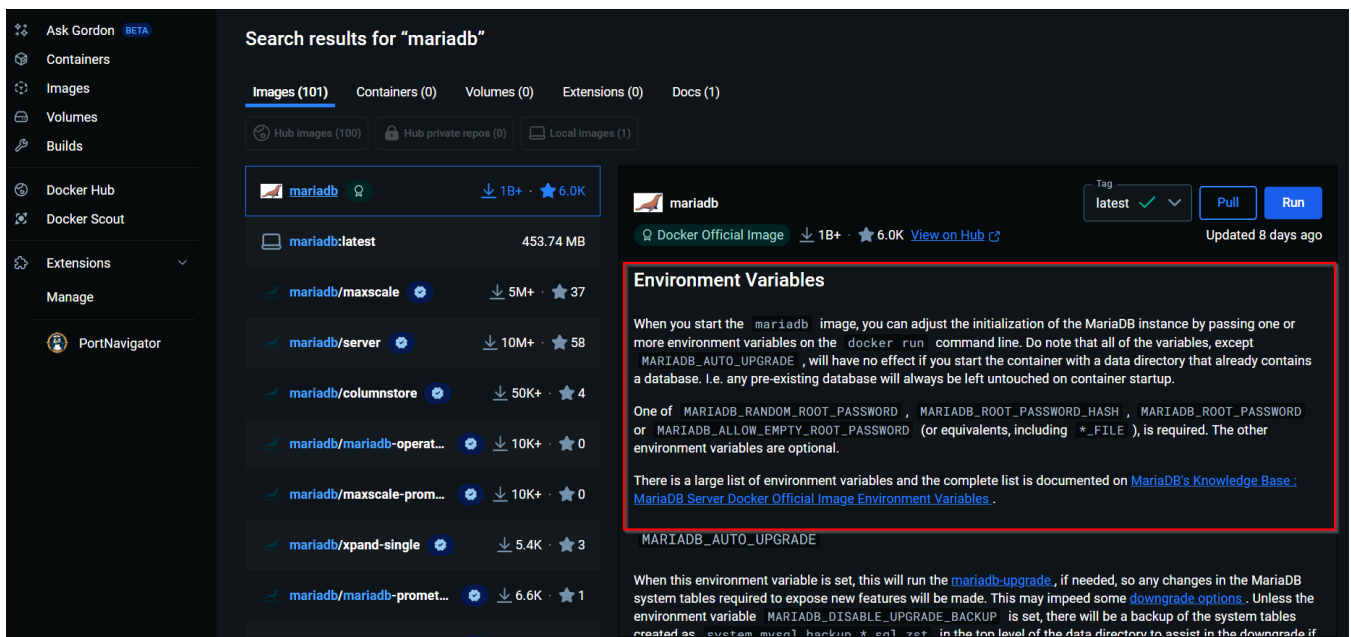


2. Crea un contenedor con una imagen de mariadb

Para crear el contenedor, el primer paso es usar el buscador de Docker Desktop para buscar la imagen oficial de `mariadb` (Descargaremos la imagen oficial, que se indica con un símbolo de una medalla junto al nombre), la descargamos con el botón `Pull` y creamos el contenedor con el botón `Run`.



Para crear el contenedor con los datos requeridos en el enunciado, accedemos a la documentación de la imagen la cual podemos revisar o desde la web de Docker Hub o debajo de los propios botones de `Pull` y `Run`

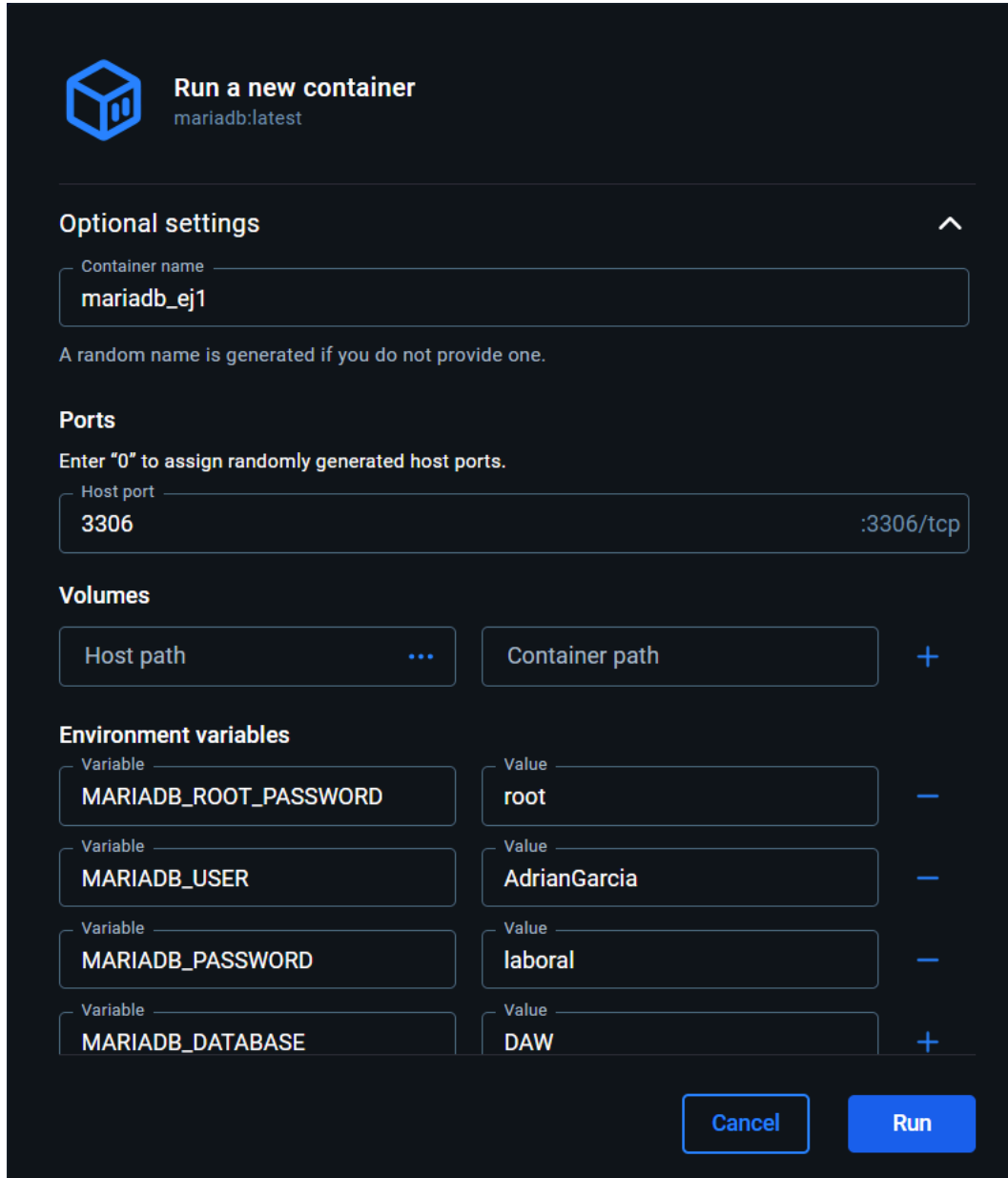


Definir credenciales

Al pulsar el botón `Run` desplegamos `Optional settings` y procedemos a definir las opciones:

- Definir la contraseña para el usuario root: Revisando la documentación, observamos que la variable se llama `MARIADB_ROOT_PASSWORD`, le estableceremos de contraseña `root`.
- Definir usuario con mi nombre y contraseña: En la documentación encontramos un enlace ([MariaDB's Knowledge Base : MariaDB Server Docker Official Image Environment Variables](#)) a la web oficial de `mariadb` en esta observamos el nombre de las variables para asignar valores que deseamos.
 - Nombre de usuario: La variable de entorno según documentación es `MARIADB_USER`, a la cual le daremos como valor mi nombre y apellido, `AdrianGarcia`.

- Contraseña: La variable de entorno se llama `MARIADB_PASSWORD`, a la que le daremos como valor `laboral`.
- Nombre de la base de datos `DAW`, para ello le asignaremos a la variable de entorno `MARIADB_DATABASE` el valor `DAW`.



Run a new container
mariadb:latest

Optional settings ^

Container name
`mariadb_ej1`

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

Ports
Enter "0" to assign randomly generated host ports.

Host port
`3306` :3306/tcp

Volumes

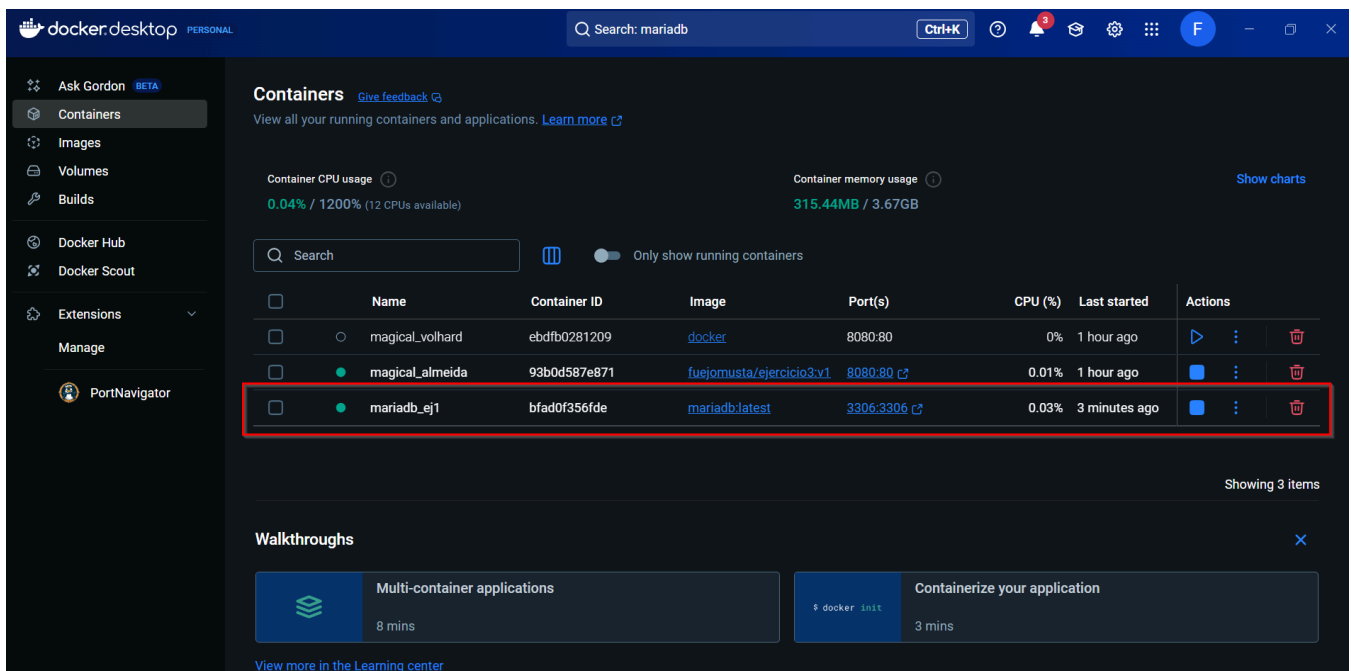
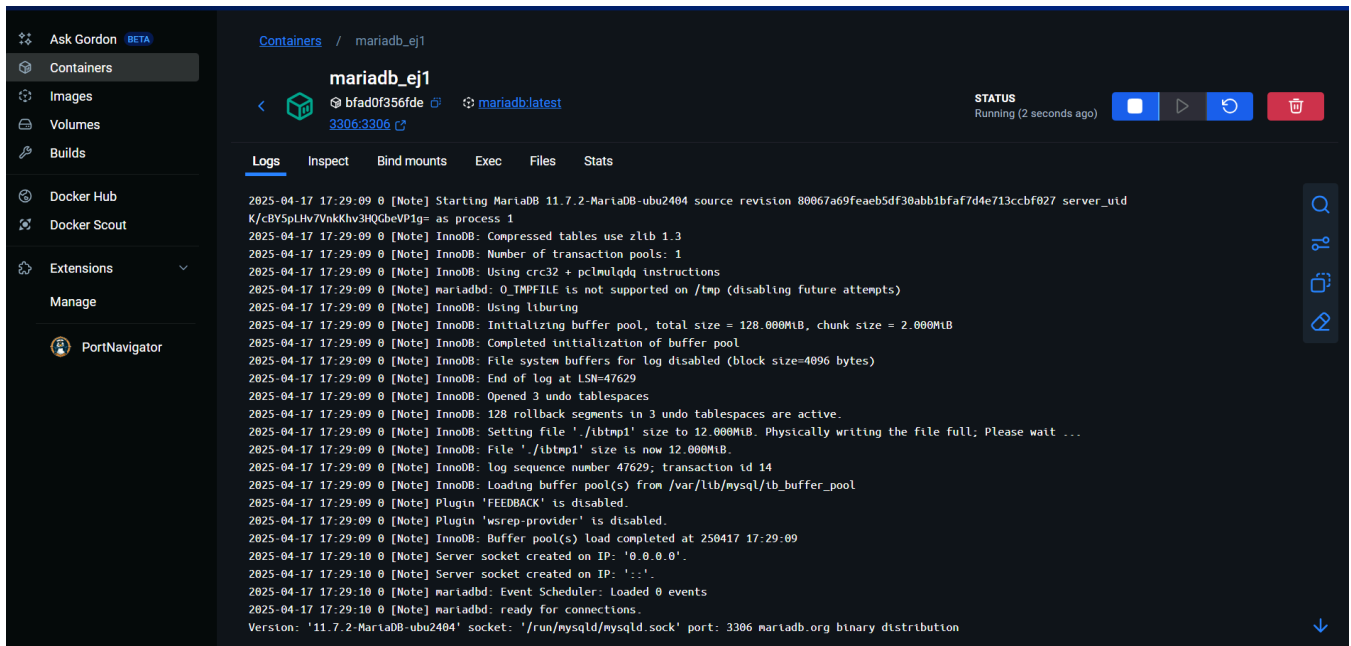
Host path	...	Container path	+

Environment variables

Variable	Value	
<code>MARIADB_ROOT_PASSWORD</code>	<code>root</code>	—
<code>MARIADB_USER</code>	<code>AdrianGarcia</code>	—
<code>MARIADB_PASSWORD</code>	<code>laboral</code>	—
<code>MARIADB_DATABASE</code>	<code>DAW</code>	+

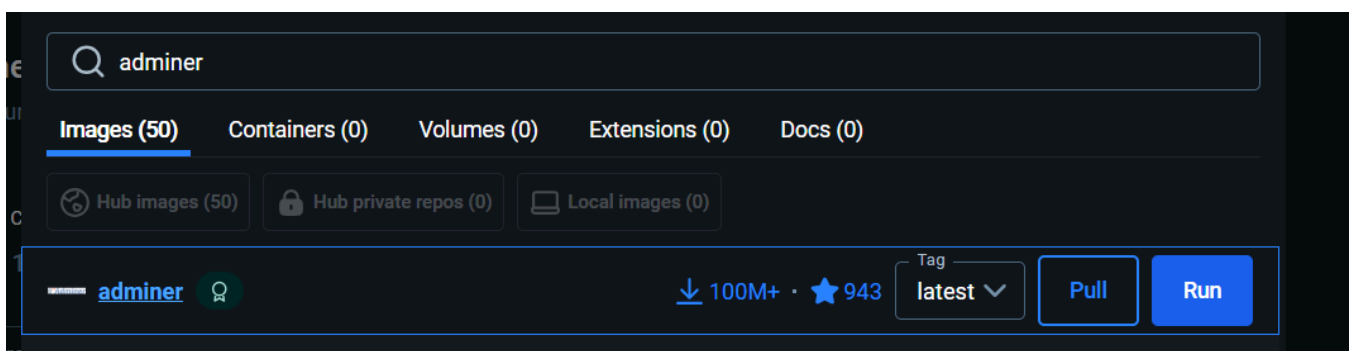
Cancel **Run**

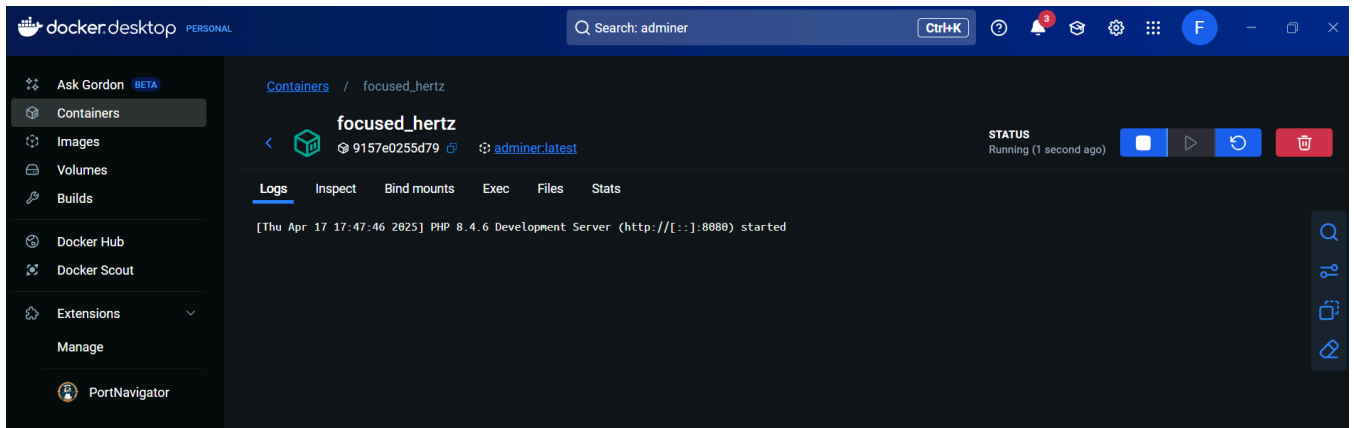
Una vez finalizada la configuración, pulsamos sobre **Run** y comienza a descargarse y configurarse.



3. Crear un contenedor con Adminer

Al igual que se hizo anteriormente con mariadb en el buscador buscamos Adminer, lo descargamos con el botón Pull y creamos el contenedor con Run.



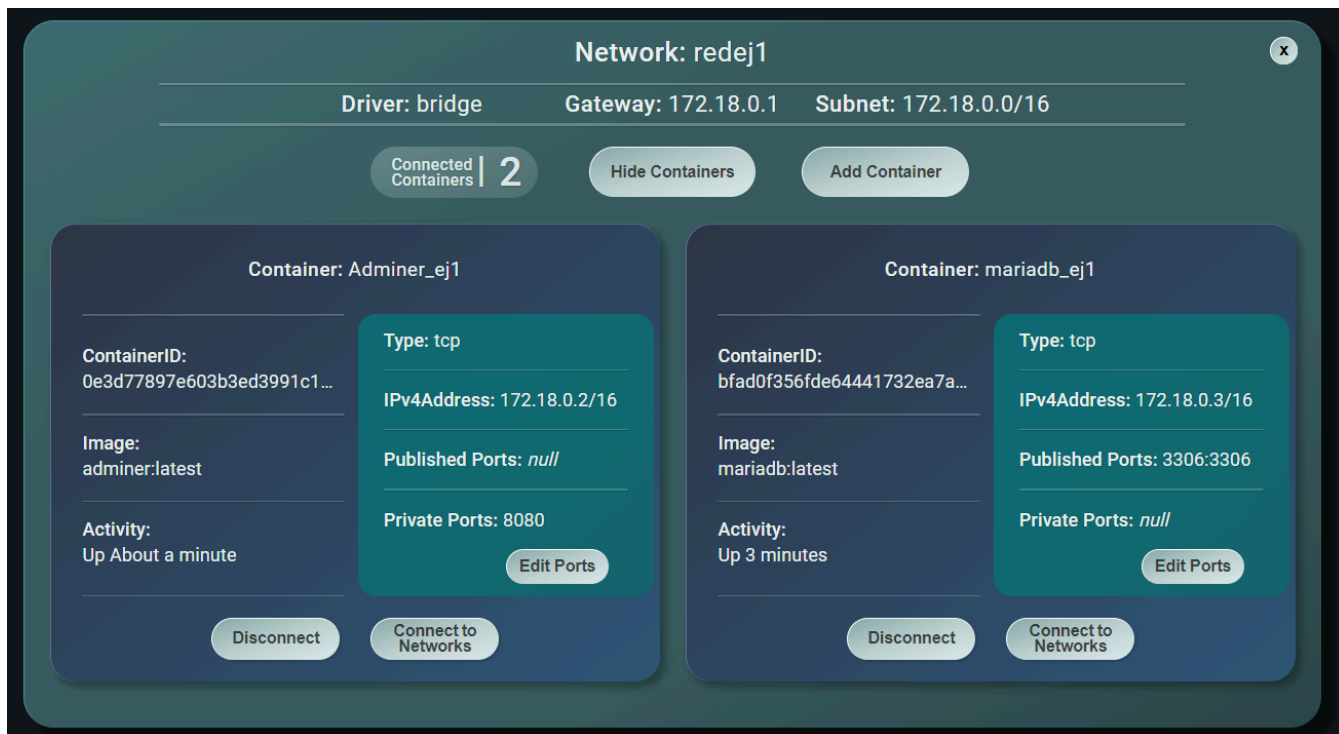


4. Conectar contenedores a redej1

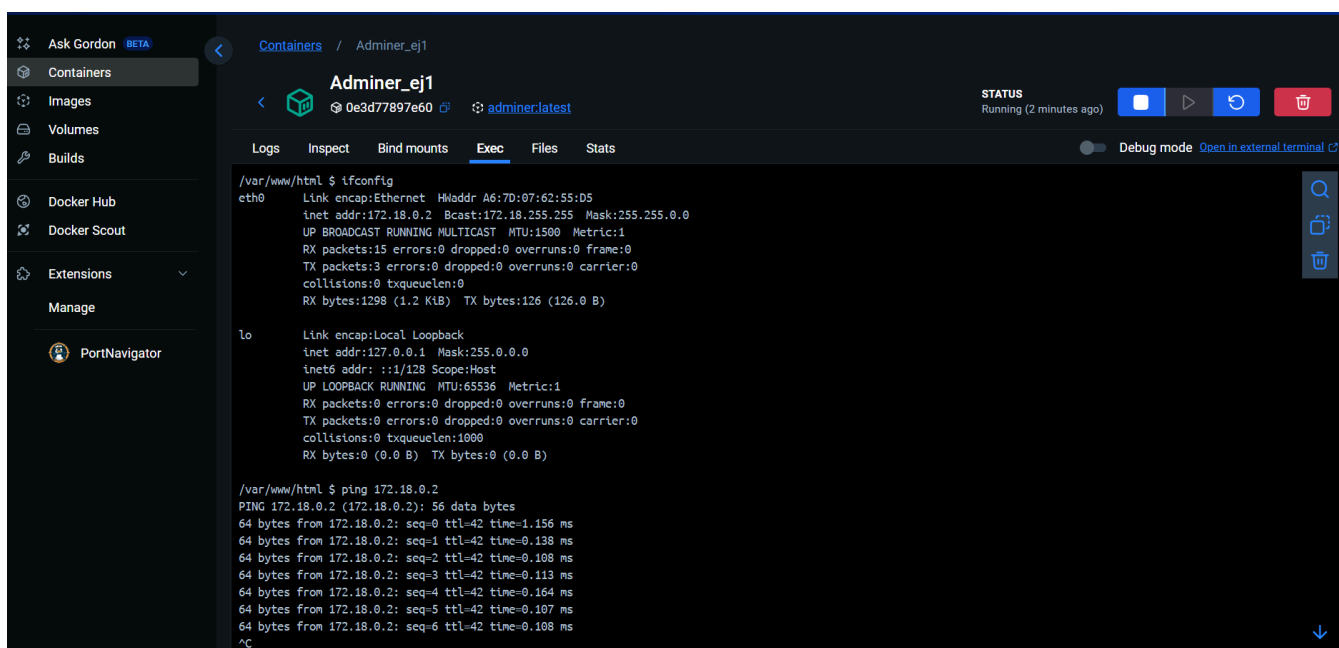
Configurar redes

Como podemos ver en la siguiente imagen, actualmente tenemos ambos contenedores conectados a la red `bridge` por defecto, debemos desconectarlos y conectarlos a la red creada `redej1`





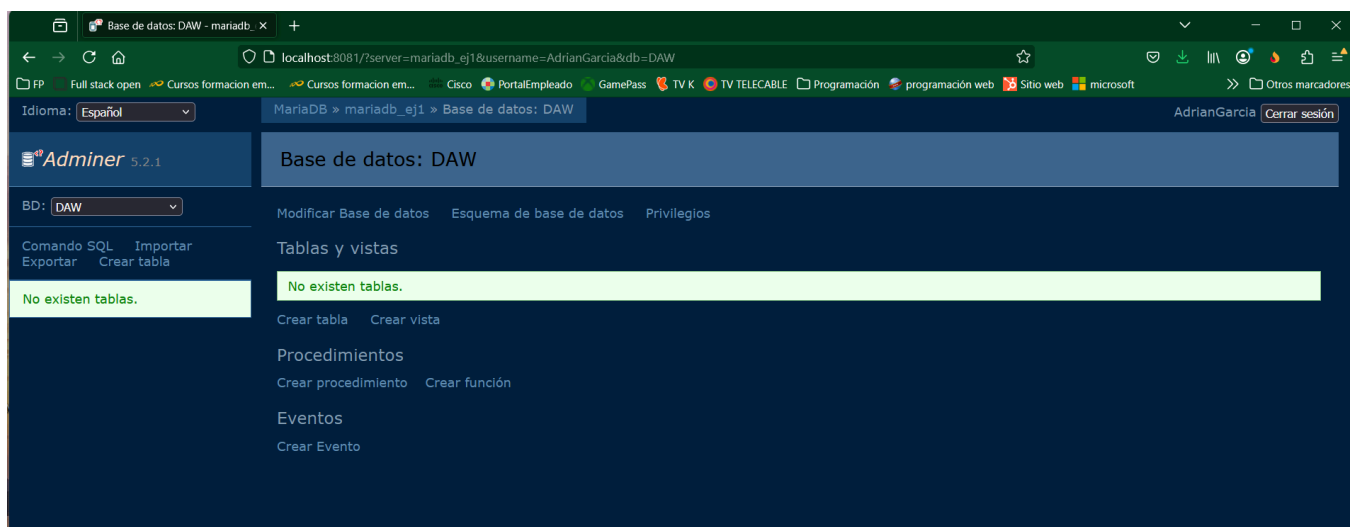
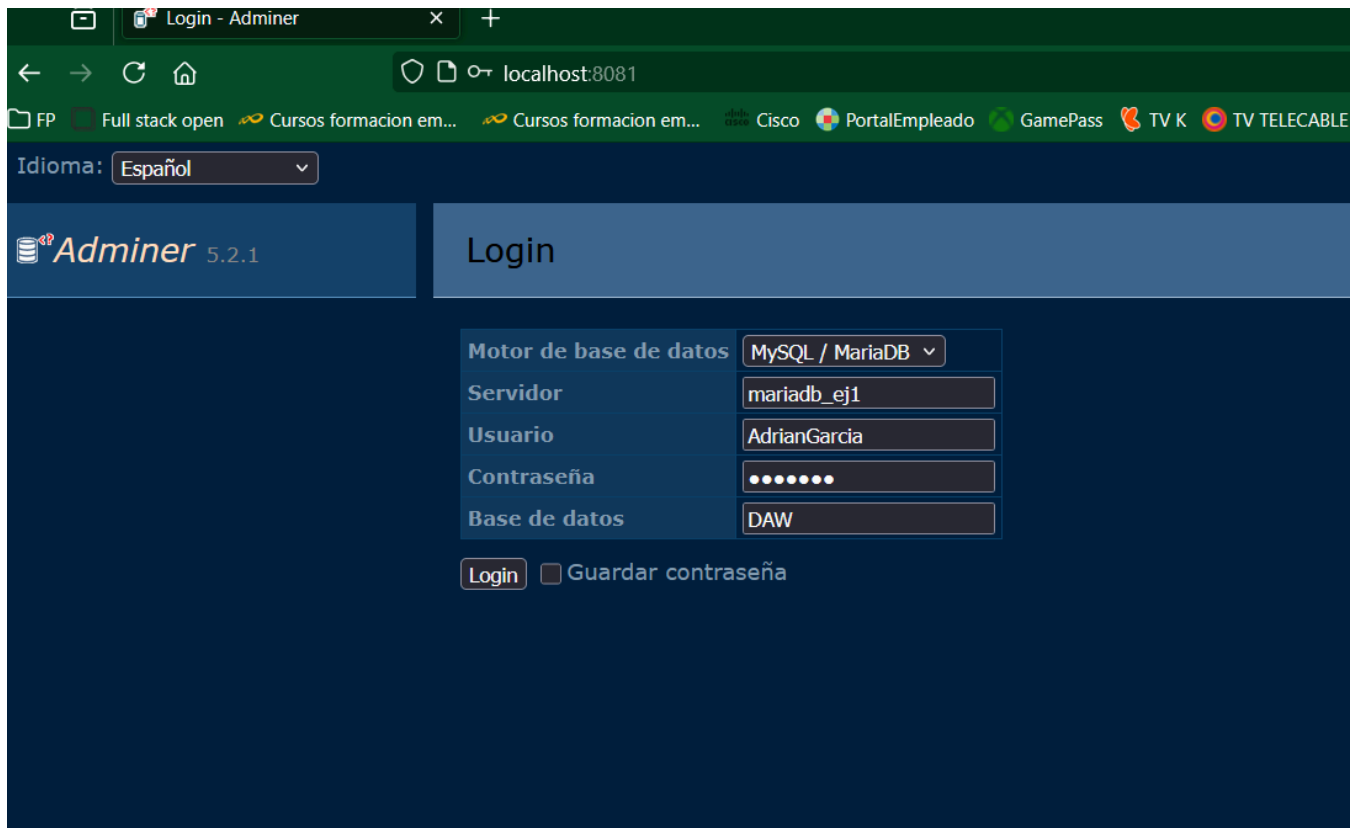
Una vez conectados comprobamos su ip con `ifconfig` y realizamos un ping para comprobar la conexión:



Conexión a interfaz grafica

Para conectarnos a través del navegador indicamos:

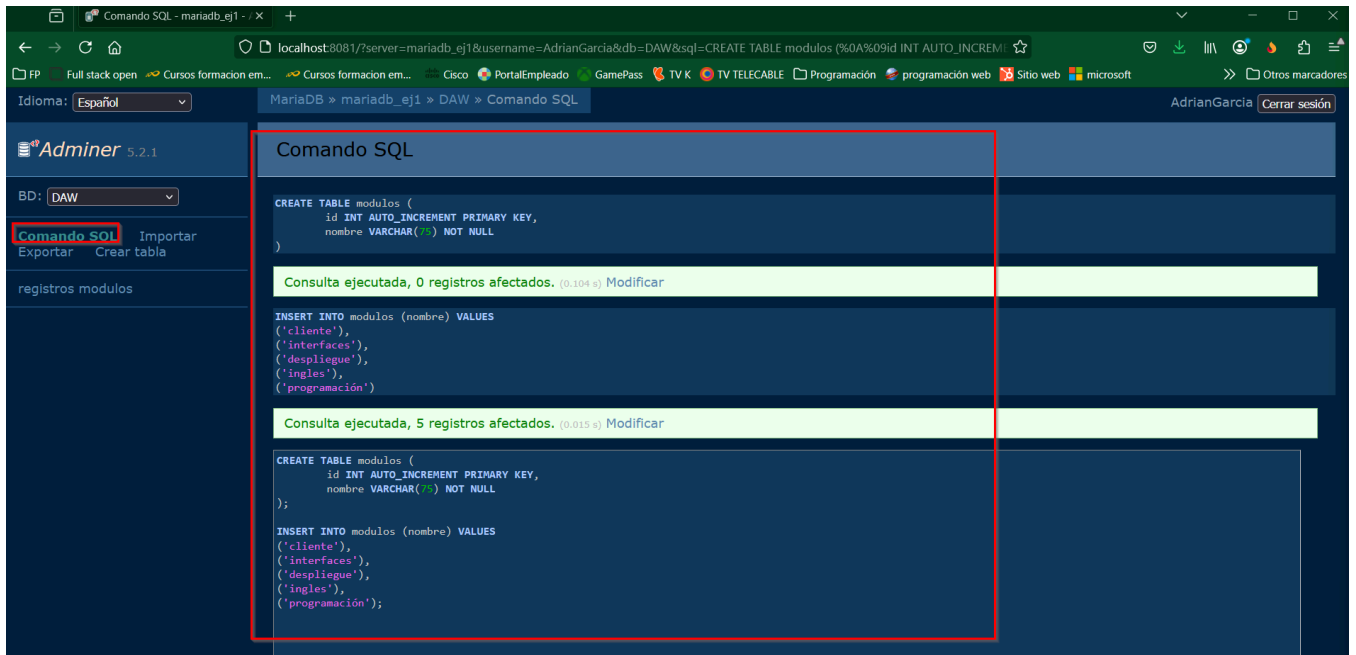
```
http://localhost:8081/
```

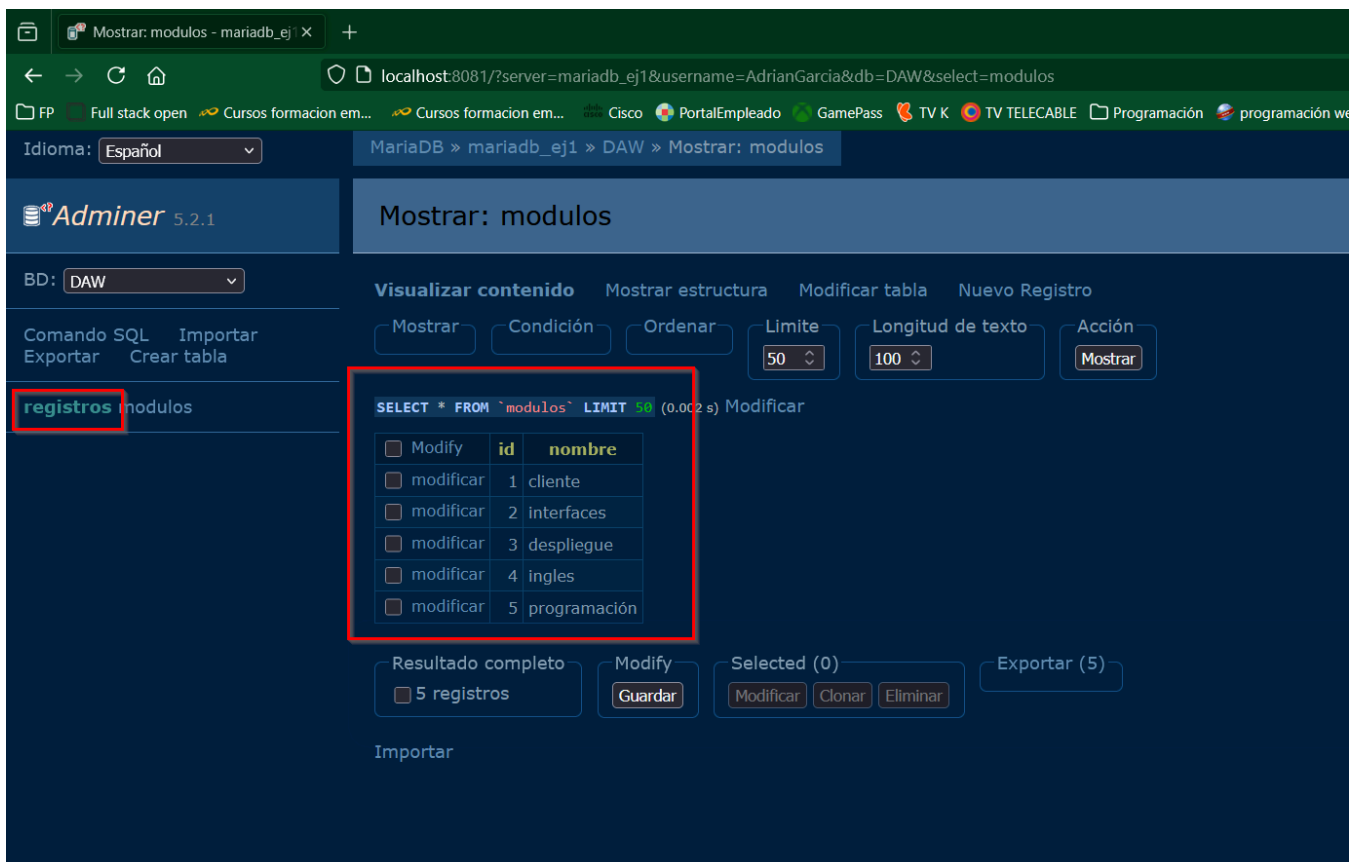
Ejecución del script SQL

Una vez dentro seleccionaremos Comando SQL y ejecutaremos el script:

```
CREATE TABLE modulos (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    nombre VARCHAR(75) NOT NULL  
);  
  
INSERT INTO modulos (nombre) VALUES  
(  
    'cliente',  
    'interfaces',  
    'despliegue',  
    'ingles',  
    'programación');
```

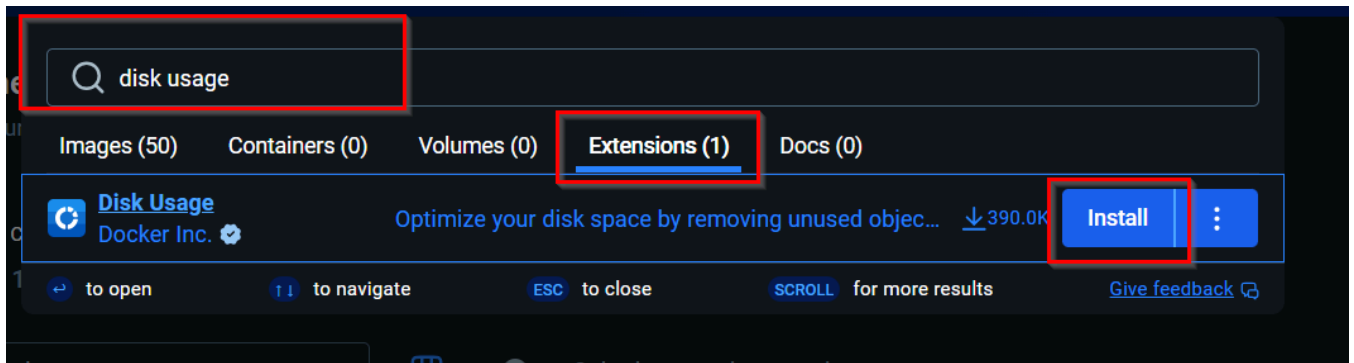


Con el script ya ejecutado, podemos comprobar en registros que realmente se han añadido:

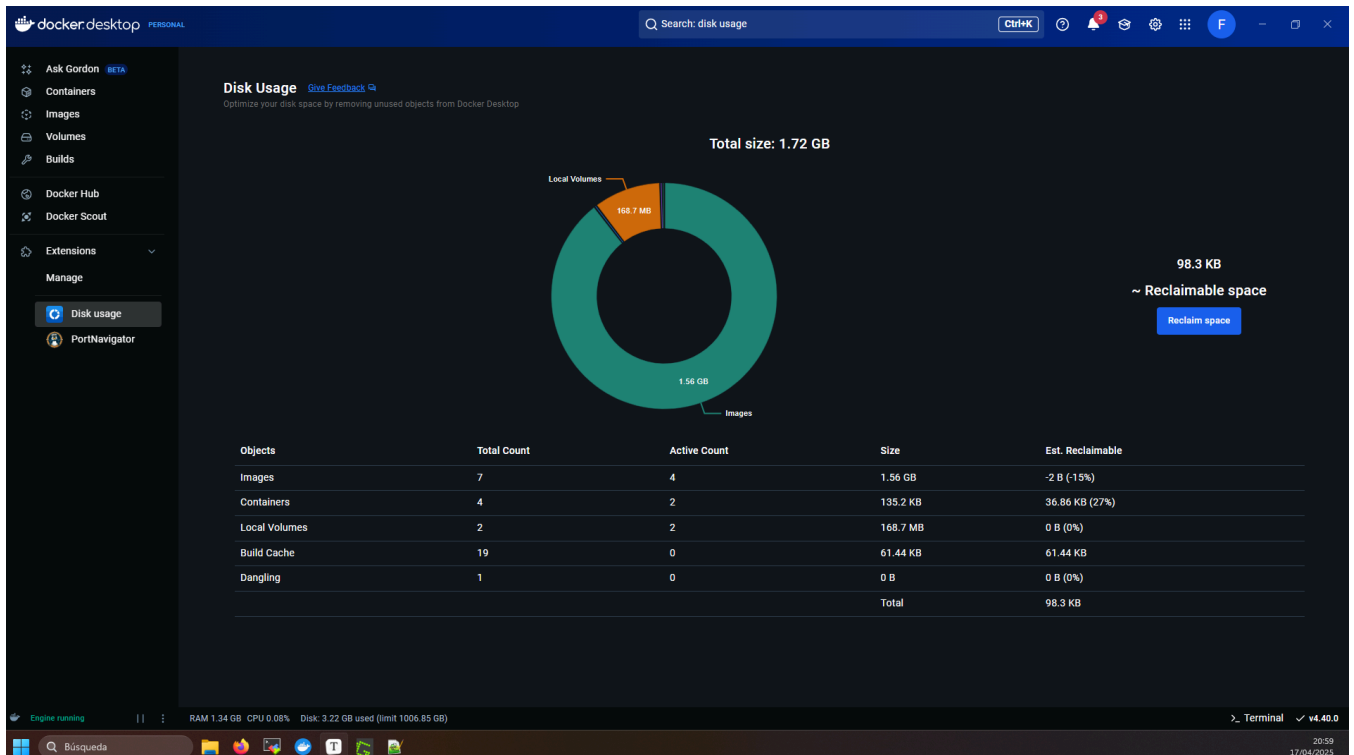


5.Instalación y comprobaciones con Disk Usage

Lo primero que debemos hacer al igual que con mariadb y adminer es buscarlo en el buscador, nos vamos a la pestaña **Extensions** y pulsamos **Install**

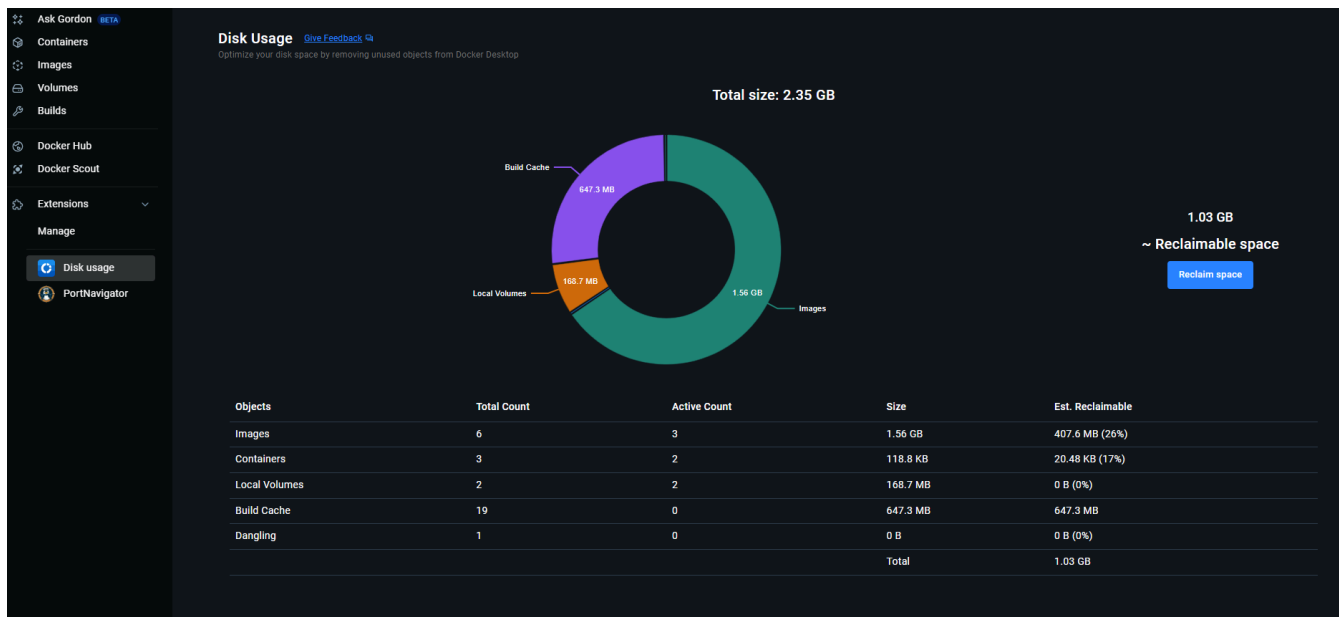


Una vez instalado, en el menú de la izquierda nos vamos a **Extensions** y seleccionamos **Disk usage**, nos mostrará un grafico con el espacio utilizado por las imágenes y los volúmenes y un listado de los mismos:

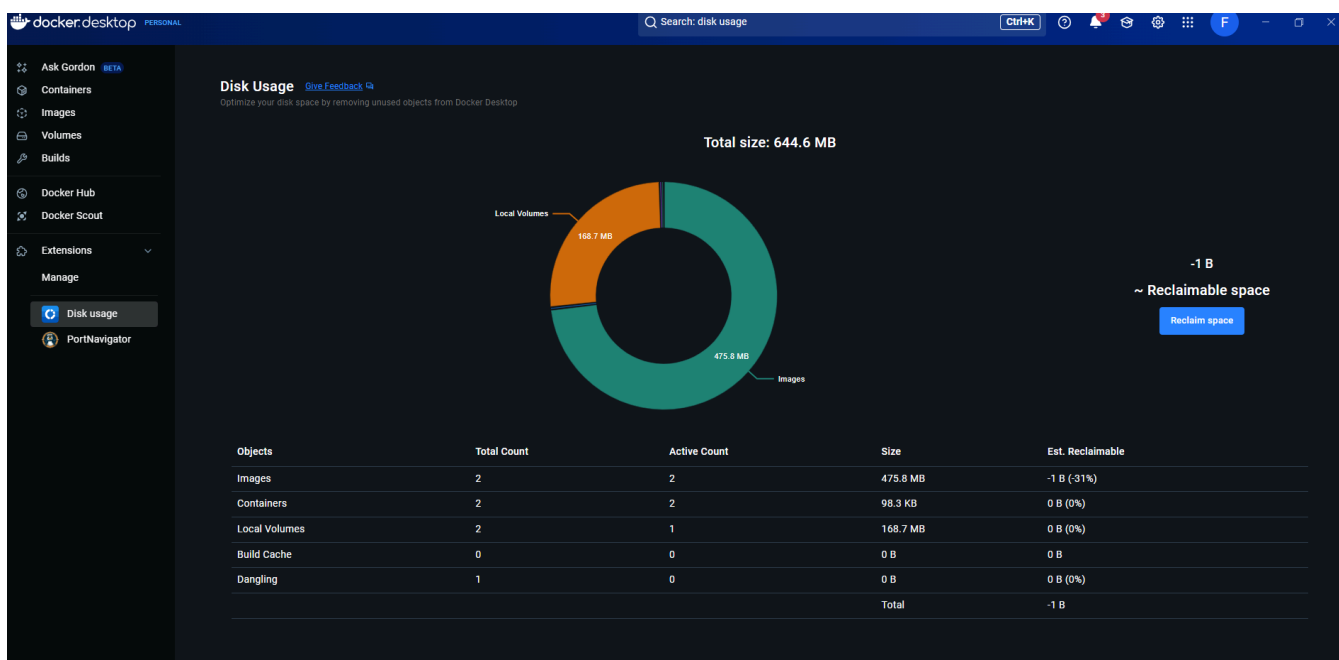


Borramos algún contenedor y alguna imagen

Name	Container ID	Image	Port(s)	CPU (%)	Last started	Actions
magical_volhard	ebdfb0281209	docker	8080:80	0%	3 hours ago	Delete Play Stop
magical_almeida	93b0d587e871	fuejomusta/ejercicio3.v1	8080:80	0%	3 hours ago	Delete Play Stop
mariaadb_ej1	bfad0f356fde	mariaadb:latest	3306:3306	0.02%	33 minutes ago	Delete Play Stop
Adminer_ej1	ee941df42a9d	adminer:latest	8081:8080	0%	21 minutes ago	Delete Play Stop



Vemos que tenemos mucho almacenamiento usado en cache, pero usando la herramienta **Reclaim space** (botón azul a la derecha de la imagen) la cual nos indica que tenemos 1.03GB para liberar, lo liberamos:



Borrado de volúmenes, contenedores y la red

Para eliminar los contenedores, en el menú contenedores usamos el botón **delete** con forma de cubo de basura:

Containers [Give feedback](#)

View all your running containers and applications. [Learn more](#)

Container CPU usage

0.03% / 1200% (12 CPUs available)

Container memory usage

294.85MB / 3.67GB

Show charts

Only show running containers

<input type="checkbox"/>	Name	Container ID	Image	Port(s)	CPU (%)	Last started	Actions
<input type="checkbox"/>	 mariadb_ej1	bfad0f356fde	mariadb:latest	3306:3306	0.02%	41 minutes ago	<div></div> <div></div> <div></div>
<input type="checkbox"/>	 Adminer_ej1	ee941df42a9d	adminer:latest	8081:8080	0.01%	29 minutes ago	<div></div> <div></div> <div></div>

docker:desktop PERSONAL

Search: disk usage Ctrl+K

Ask Gordon BETA

Containers

Images

Volumes

Builds

Docker Hub

Docker Scout

Extensions

Manage

Disk usage

PortNavigator

Containers [Give feedback](#)

View all your running containers and applications. [Learn more](#)

Your running containers show up here

A container is an isolated environment for your code

What is a container?

5 mins

How do I run a container?

6 mins

[View more in the Learning center](#)

Para borrar la red, nos dirigimos a **Extensions**, y en **PortNavigator** nos mostrará todas las redes, y en la red **redej1** pulsamos en la **X** de arriba a la derecha para borrarla.

Network: redej1

Driver: bridge

Gateway: 172.18.0.1

Subnet: 172.18.0.0/16

Connected Containers

0

Add Container