

# Actividad Evaluable 3 - Docker - EJERCICIO 1 - Contenedores en red y Docker Desktop

Tarea creada por Aida Montes Cabello para el módulo Despliegue de Aplicaciones Web.

## Actividad Evaluable 3 - Docker - EJERCICIO 1 - Contenedores en red y Docker Desktop

### Introducción

#### Instalación extensión PortNavigator

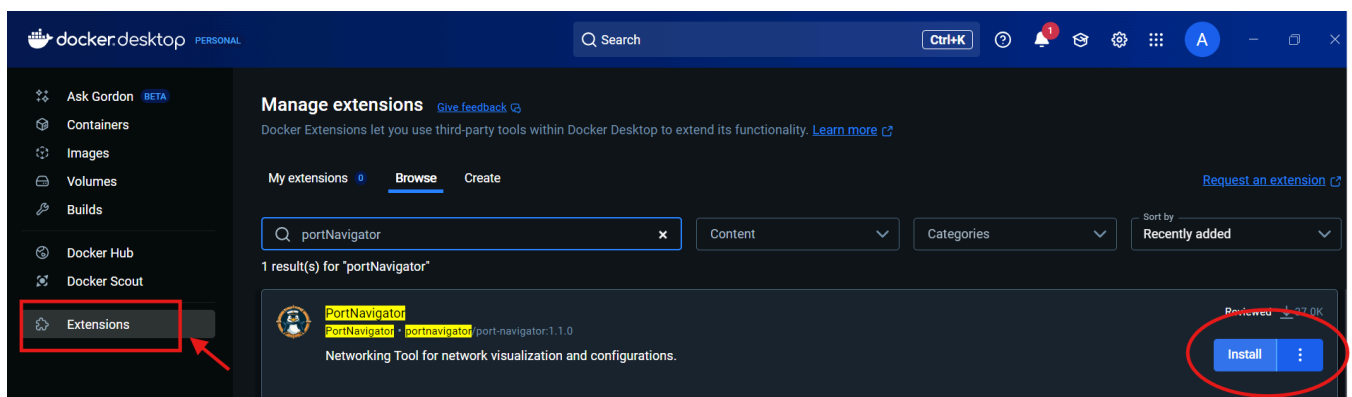
1. Creación red bridge
2. Creación contenedor MariaDB  
Configuración contenedor  
Script SQL
3. Creación contenedor con Adminer
4. Conexión a la BD.
5. Instalación Disk Usage  
Borrado de elementos. Comprobación espacio.

## Introducción

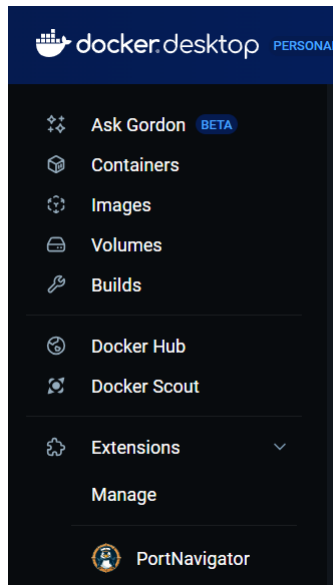
En este documento se detallarán los pasos a seguir para la resolución del Ejercicio 1 de la Actividad Evaluable 3 de Despliegue de Aplicaciones Web sobre Docker. Para la realización de dicho ejercicio, utilizaremos la extensión PortNavigator para poder gestionar las redes desde Docker Desktop.

## Instalación extensión PortNavigator

Para el desarrollo de este ejercicio, es necesario instalar la extensión PortNavigator en nuestro Docker Desktop, para ello abrimos la aplicación y vamos a la pestaña **Extension** y buscamos dicha extensión y la instalamos.

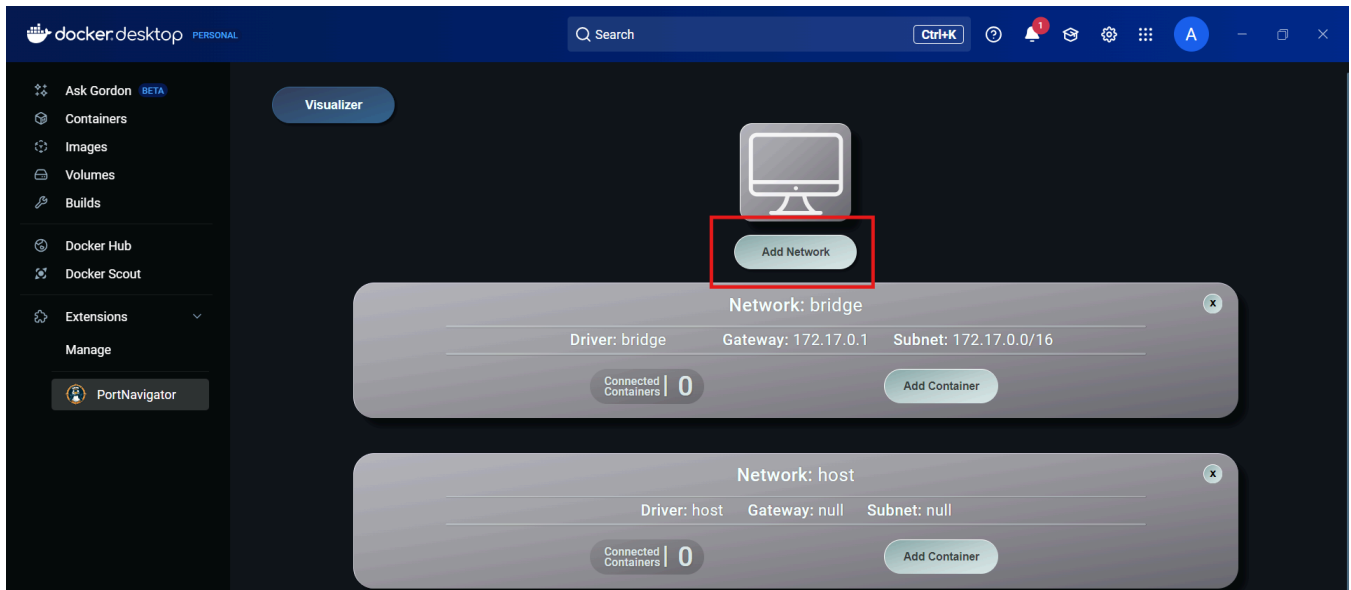


Acto seguido nos aparecerá en el panel izquierdo. Con esto ya podemos crear nuestra red a través de esta extensión.



## 1. Creación red bridge

Para la creación de nuestra red bridge `redej1`, abrimos PortNavigator y pulsamos en el botón `Add Network`



Como podemos ver a continuación, nos saldrá una pestaña donde añadiremos el nombre de nuestra red y pulsamos `Create Network`.

X

## Add Network

Name:

---

☐ Advanced network settings

---

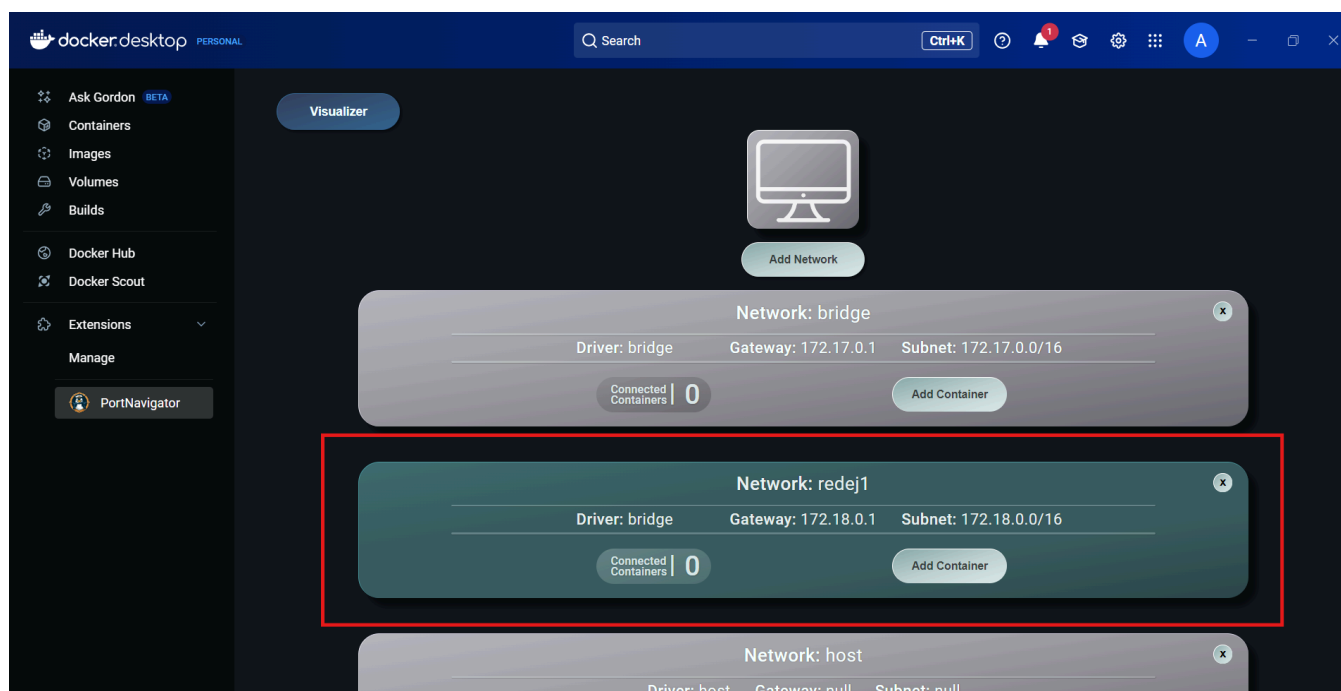
Subnetwork:

Gateway:

IP-Range:

**Create Network**

Después de crearla, confirmamos que se haya generado correctamente y comprobamos que es una red bridge.

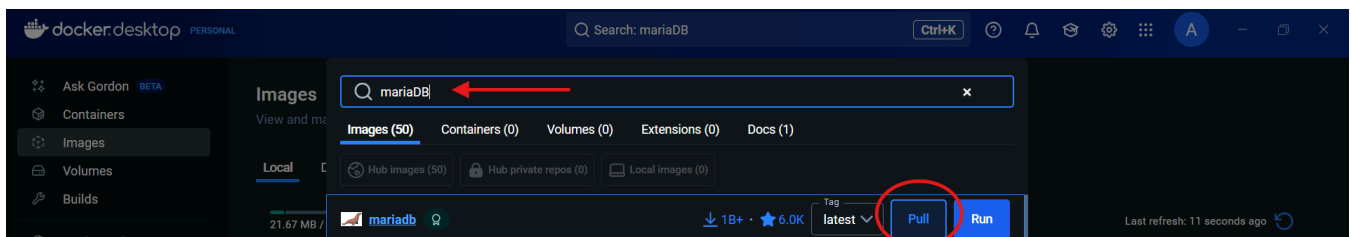


## 2. Creación contenedor MariaDB

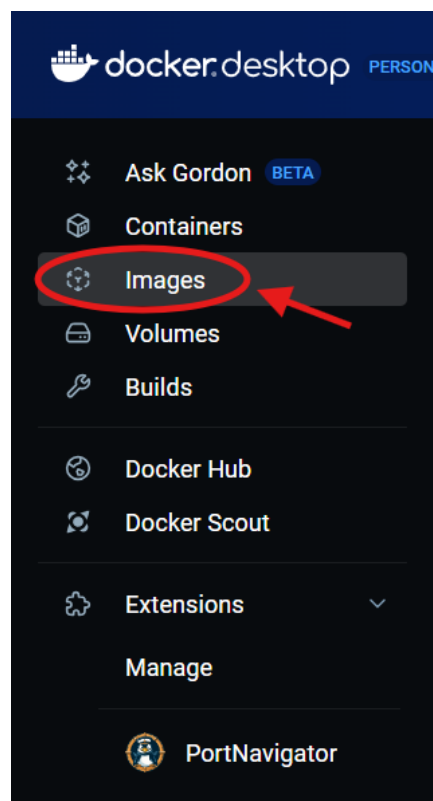
En este apartado vamos a crear un contenedor que se ejecutará en segundo plano utilizando una imagen de `mariadb` y que estará conectado a la red configurada en el punto anterior. Este contenedor también tendrá que ser accesible a través del puerto 3306. En este punto también realizaremos las siguientes tareas:

- Definición de una contraseña para el usuario `root`, y un usuario que será el nombre de pila y una contraseña para éste (BD: DAW).
- Creación de un script SQL que defina una tabla llamada `módulos` e inserción de algunos registros con nombres de módulos que se esté cursando.

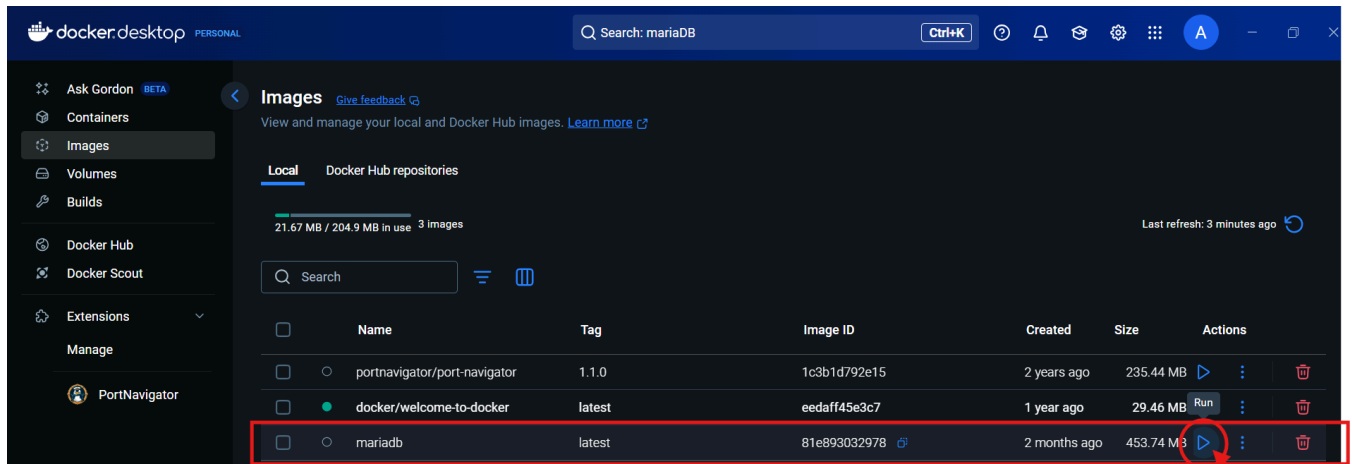
Lo primero que debemos hacer antes de crear el contenedor, es descargar la imagen de `mariadb`, para ello utilizaremos el buscador para buscarla. Una vez encontrada, pulsaremos el botón `Pull`.



Una vez descargada, accederemos a la pestaña `Images`, que se encuentra en el panel izquierdo de la aplicación.




A continuación podemos observar que se ha descargado correctamente, por lo que pulsamos sobre `Run` para la creación del contenedor.



## Configuración contenedor

Ahora comenzaremos con la configuración del contenedor cuyo nombre va a ser `db-DAW` y la accesibilidad del puerto 3306. También agregaremos las variables de entorno: la contraseña para el usuario `root` (`MYSQL_ROOT_PASSWORD`), la base de datos DAW (`MYSQL_DATABASE`), un usuario (`MYSQL_USER`) y su correspondiente contraseña (`MYSQL_PASSWORD`). Una vez que se hayan añadido, presionaremos el botón `Run`.

 **Run a new container**  
mariadb:latest

**Optional settings** ^

Container name  
**db-DAW** ←

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

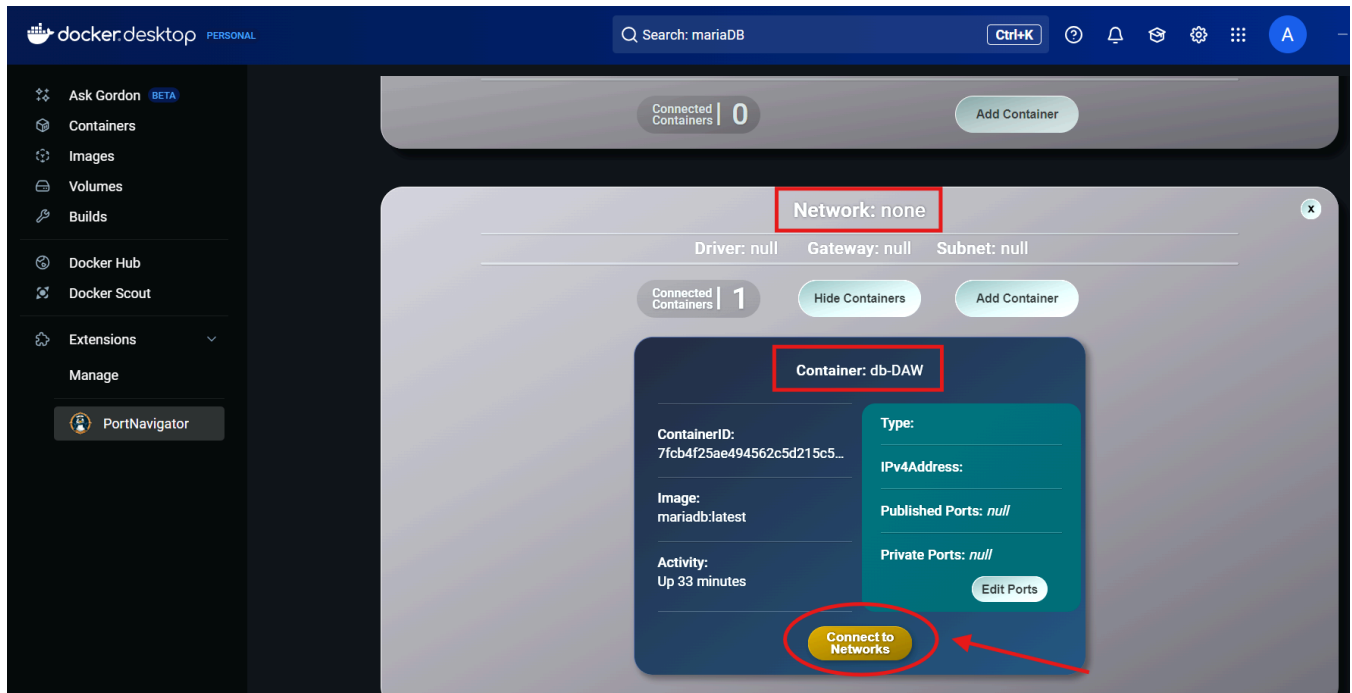
**Volumes**  
Host path ... Container path +

**Environment variables**

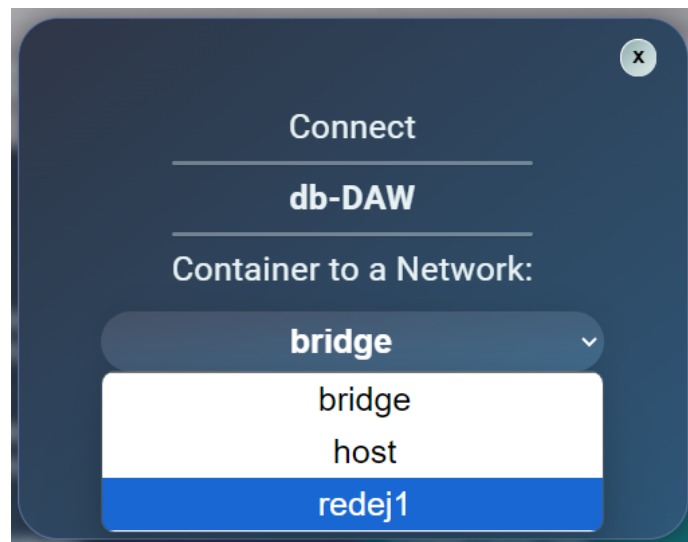
Variable <b>MYSQL_ROOT_PASSWORD</b>	Value <b>contraseñaroot</b> —
Variable <b>MYSQL_DATABASE</b>	Value <b>DAW</b> —
Variable <b>MYSQL_USER</b>	Value <b>Aida</b> —
Variable <b>MYSQL_PASSWORD</b>	Value <b>contraseñaaida</b> +

**Cancel** **Run** ←

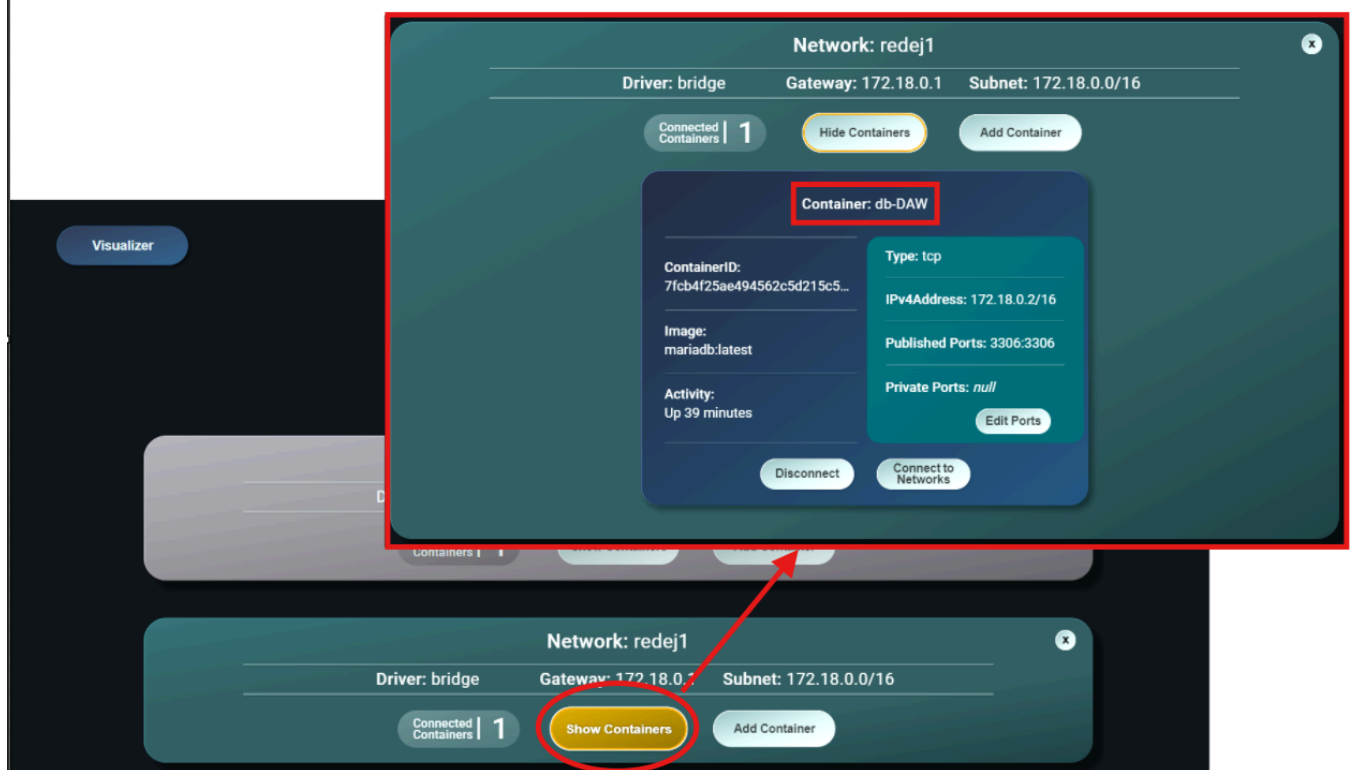
Con esto, el contenedor se iniciará, pero todavía no estaría en la red `redej1`, para ello accederemos a la extensión `PortNavigator` y buscamos el contenedor creado, el cual vemos que no tiene asignada ninguna red, por lo tanto hacemos clic en `Connect to Networks`.



Aquí seleccionamos la red redej1 y conectamos.



Ahora vamos a comprobar que el contenedor está conectado a nuestra red. En PortNavigator buscamos la red y observamos que hay un contenedor conectado, por lo que pulsamos Show Containers para verificarlo. En efecto, el contenedor db-DAW está conectado a redej1 usando PortNavigator.



## Script SQL

Por último vamos a crear un script SQL que defina una tabla cuyo nombre será `módulos` e insertaremos algunos registros con los nombres de los módulos que se esté cursando.

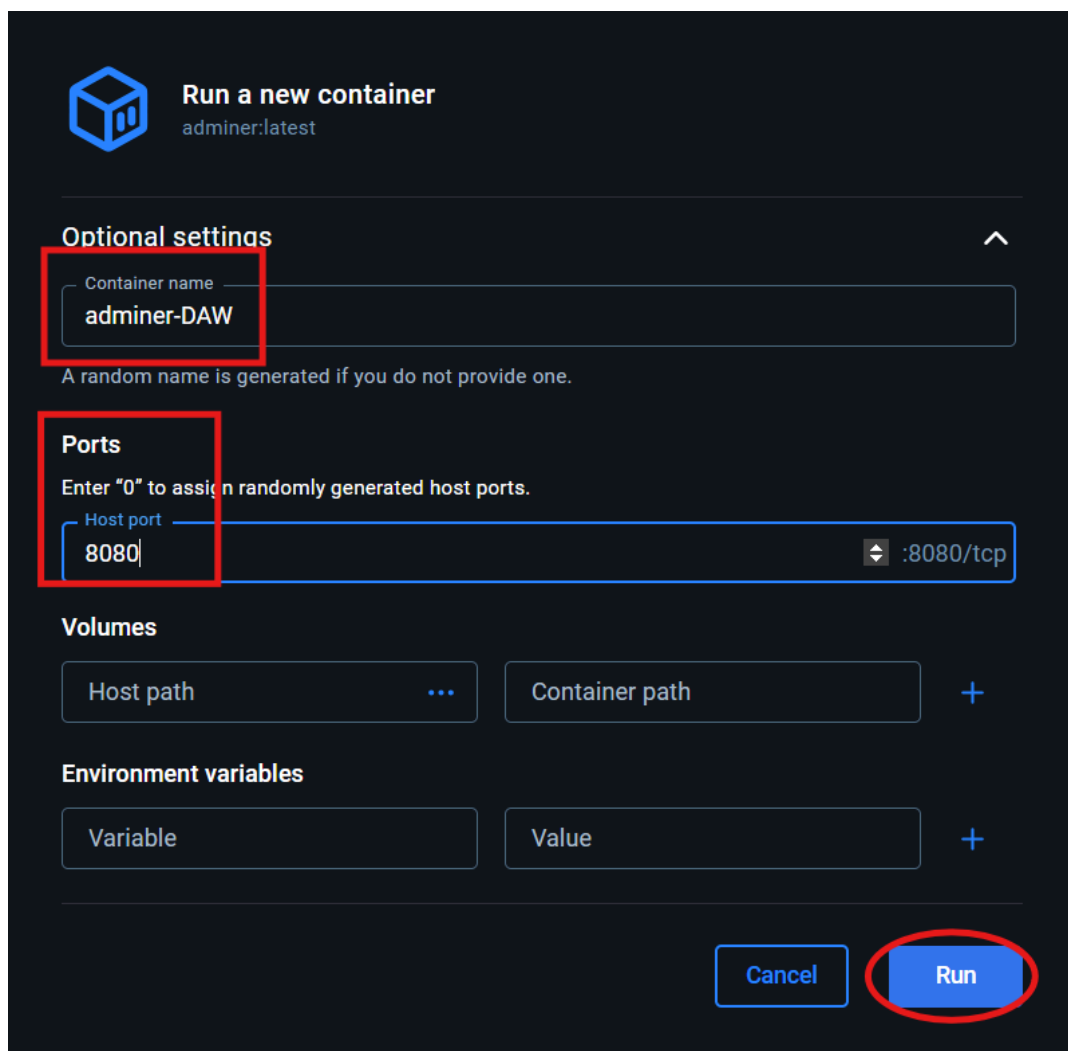
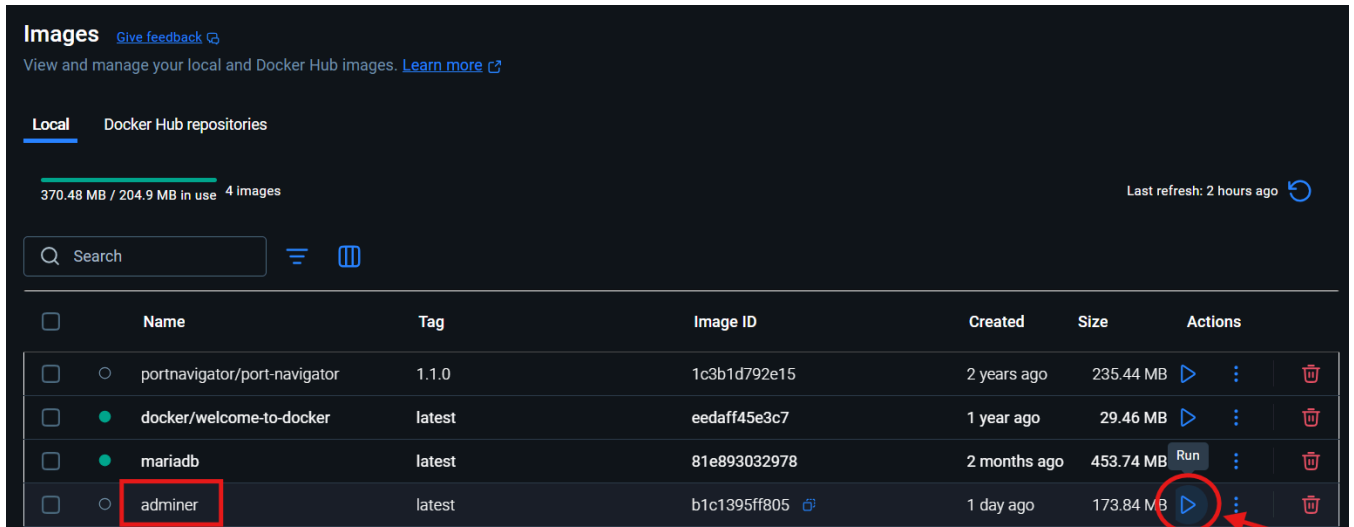
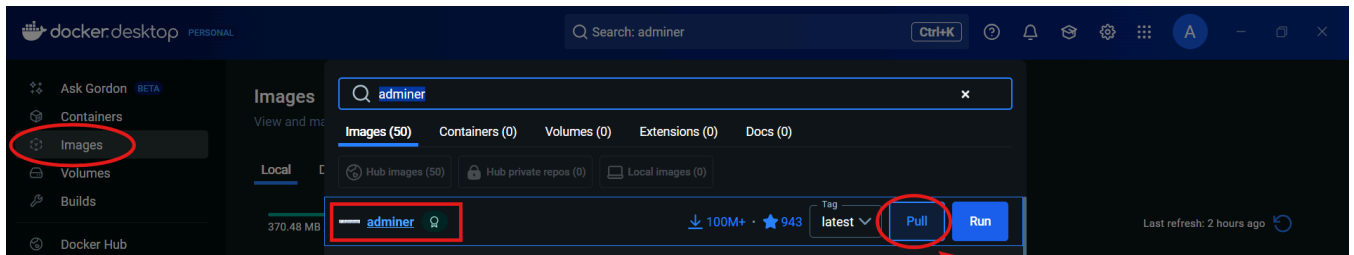
```
CREATE TABLE módulos (  
  id INT AUTO_INCREMENT PRIMARY KEY,  
  nombre VARCHAR(50) NOT NULL  
);  
  
INSERT INTO módulos (nombre) VALUES  
(  
'Despliegue de Aplicaciones web',  
'Desarrollo web en Entorno Cliente',  
'Desarrollo web en Entorno Servidor',  
'Diseño de Interfaces web',  
'Empresa e Iniciativa Emprendedora');
```

## 3. Creación contenedor con Adminer

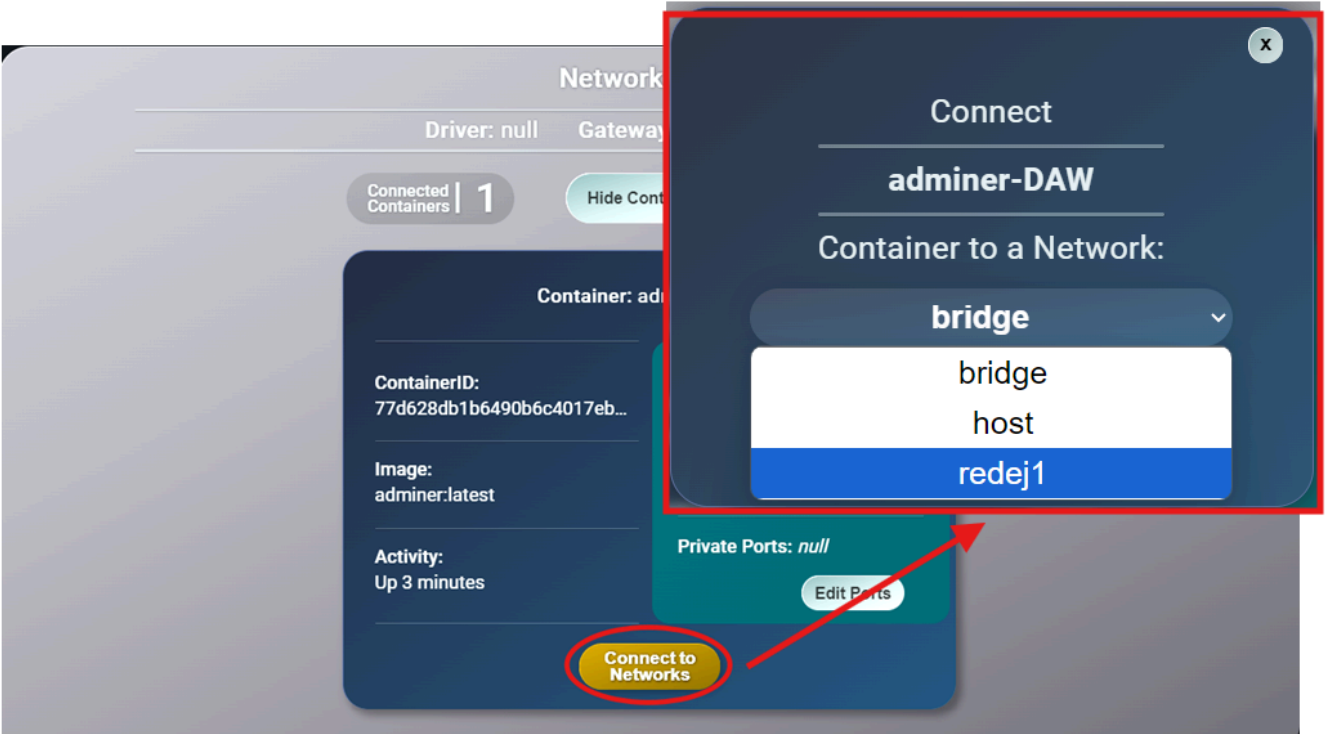
El punto 3 de este ejercicio, nos pide crear un contenedor con `Adminer` o con `phpMyAdmin` que podamos conectar al contenedor de la BD, pues bien, vamos a escoger crear el contenedor `Adminer`, que es una herramienta que gestiona bases de datos desde el navegador y que es similar a `phpMyAdmin`.

Para ello, vamos a `Images` y en el buscador escribimos `adminer`, seguiremos los mismos pasos realizados con la imagen de `mariadb` en el punto número 2.

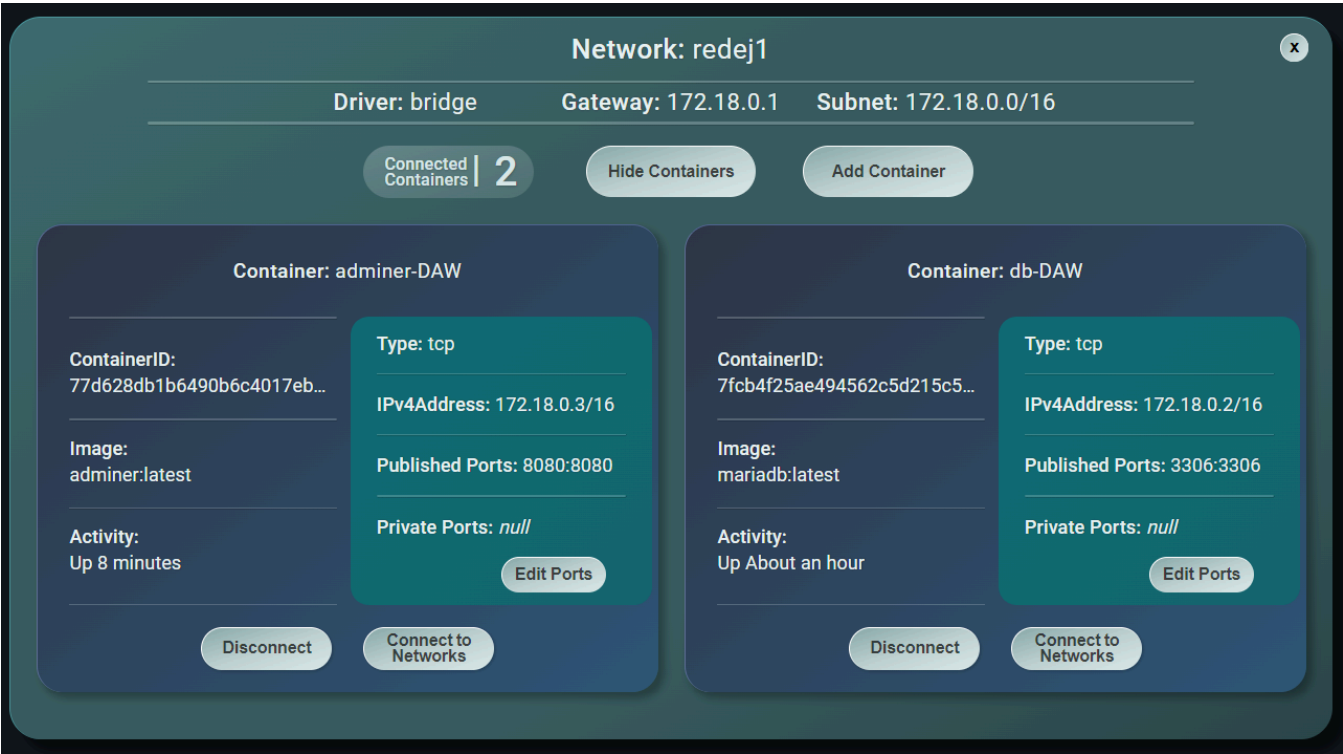




Como nos pasó con anterioridad con mariaDB, el contenedor `adminer-DAW` se iniciará, pero aún no estará conectado a `redej1`. Por lo tanto seguiremos también los pasos seguidos con db-DAW. Buscaremos el contenedor y pulsaremos `Connect to Networks` donde seleccionaremos `redej1`.



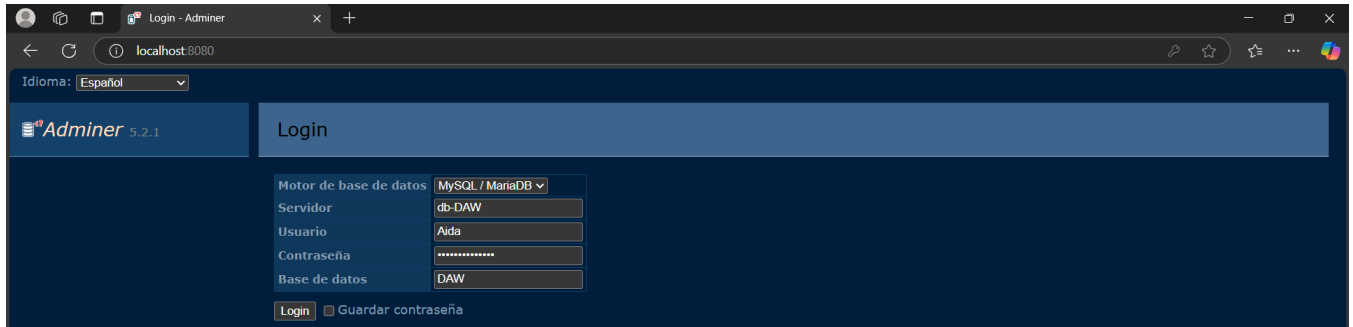
Ahora, tanto Maria DB ( `db-DAW` ) como Adminer ( `adminer-DAW` ) están conectadas en la misma red `bridge`, lo que les permite comunicarse entre sí usando sus nombres como si fueran direcciones IP.



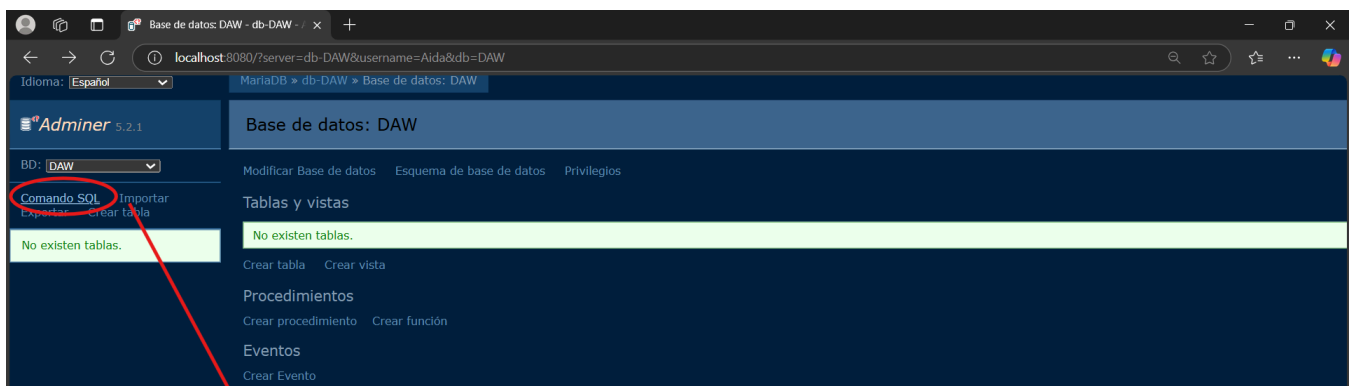
## 4. Conexión a la BD.

En este punto, desde Microsoft Edge nos conectaremos a la BD con el usuario creado en el punto 2, ejecutaremos el script con los datos de los módulos mostrando la BD y la tabla creados.

Para conectarnos a la BD escribiremos en nuestra interfaz <http://localhost:8080> y nos aparecerá lo siguiente, que rellenaremos con nuestros datos.



Una vez dentro, hacemos clic en Comando SQL y pegamos el script creado en el apartado 2 y lo ejecutamos.



Al ejecutar el script nos sale lo siguiente:



Esto quiere decir que se han ejecutado las consultas correctamente, y que en el caso de la inserción de los módulos, se han hecho 5 registros.

Finalmente, vamos a comprobar que la base de datos y la tabla se han creado, mostrando el contenido. Para ello, accederemos a la tabla `módulos` que aparecerá en la parte izquierda del menú. Inicialmente se mostrará su estructura, pero lo que queremos es ver son los datos de la tabla, por lo que pulsamos en `visualizar contenido`. De este modo, podemos observar los registros insertados en la tabla.

The image shows a screenshot of the Adminer 5.2.1 web interface. On the left sidebar, the 'registros' menu item is circled in red and labeled with a red '1'. The main panel shows the 'Tabla: módulos' view, which is also labeled with a red '2'. The 'Visualizar contenido' tab is selected and circled in red. Below this, the table structure is shown with columns 'id' (int(11) Incremento automático) and 'nombre' (varchar(50)). The 'Índices' section shows a PRIMARY index on 'id'. The 'Claves' section is empty. The 'Mostrar: módulos' view is shown below, with the 'Visualizar contenido' tab selected. It displays a table with 5 records, each with a 'Modificar' button. The table has columns 'id' and 'nombre'. The records are:

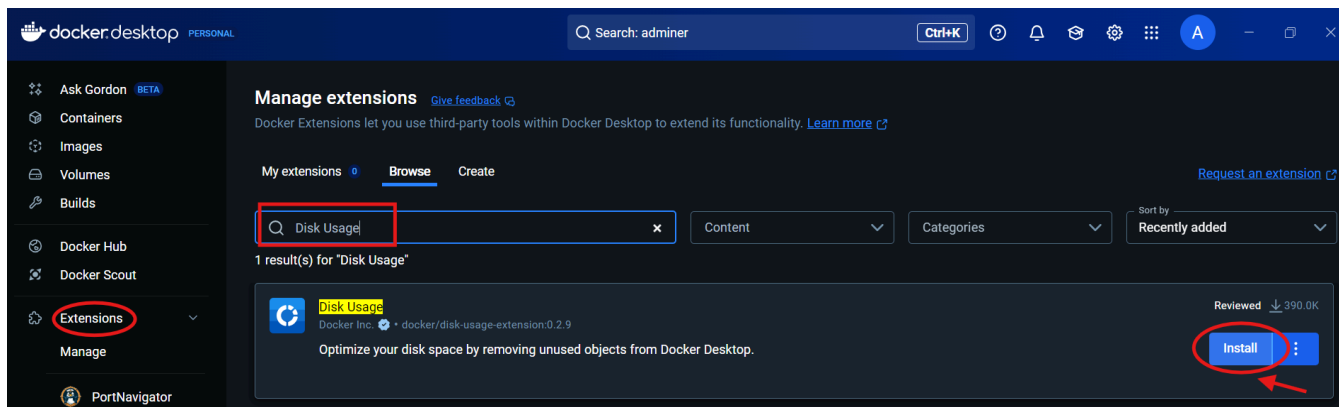
	id	nombre
Modificar	1	Despliegue de Aplicaciones Web
Modificar	2	Desarrollo Web en Entorno Cliente
Modificar	3	Desarrollo Web en Entorno Servidor
Modificar	4	Diseño de Interfaces Web
Modificar	5	Empresa e Iniciativa Emprendedora

At the bottom of the 'Mostrar: módulos' view, there are buttons for 'Resultado completo' (5 registros), 'Modificar' (Guardar), 'Selected (0)' (Modificar, Clonar, Eliminar), and 'Exportar (5)'.

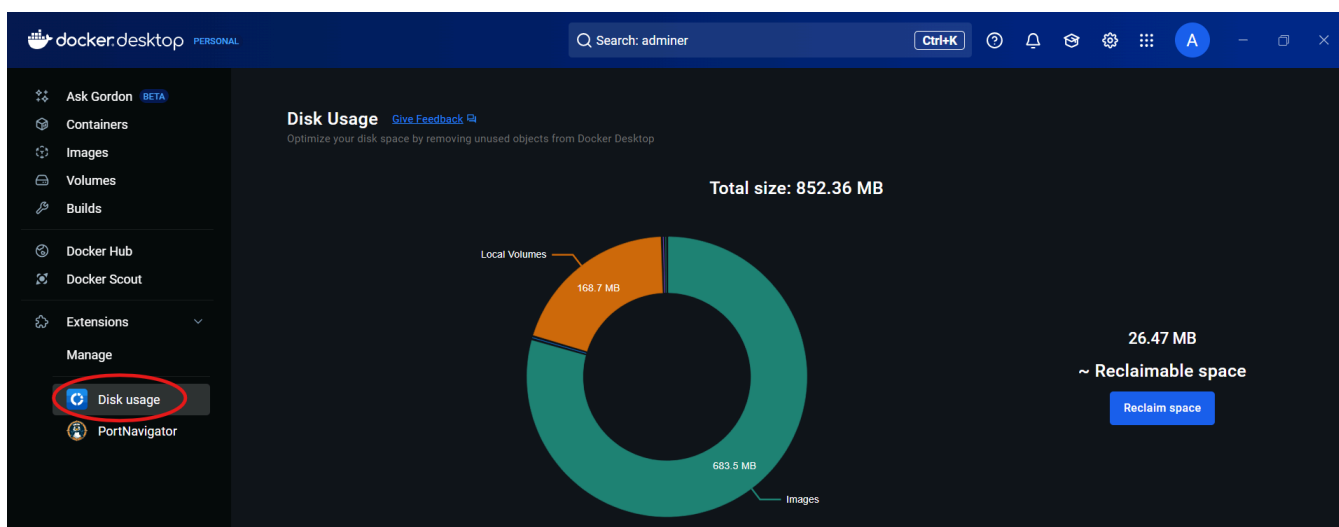
## 5. Instalación Disk Usage

En este punto instalaremos la extensión `Disk Usage` para comprobar el espacio ocupado y posteriormente eliminar los contenedores, las redes y los volúmenes que no se estén usando.

Lo primero que haremos será ir a `Extensions`, buscar `Disk Usage` y pulsamos `Install`.



Una vez instalada, nos aparecerá en el lateral izquierdo, junto a PortNavigator, pulsamos sobre ella y podremos ver el espacio ocupado por imágenes, contenedores y volúmenes en forma de gráfico y tabla.

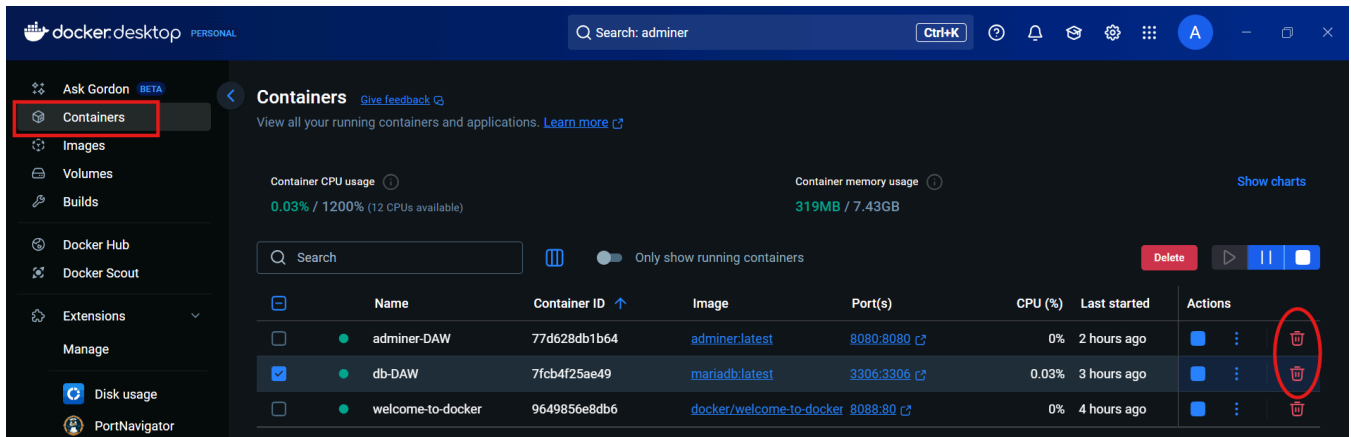


Objects	Total Count	Active Count	Size	Est. Reclaimable
Images	5	3	683.5 MB	26.47 MB (3%)
Containers	3	3	163.8 KB	0 B (0%)
Local Volumes	2	1	168.7 MB	0 B (0%)
Build Cache	0	0	0 B	0 B
Dangling	1	0	0 B	0 B (0%)
<b>Total</b>				<b>26.47 MB</b>

## Borrado de elementos. Comprobación espacio.

Ahora vamos a eliminar los contenedores ( `db-DAW` y `adminer-DAW` ), la red ( `redelj1` ) y los volúmenes.

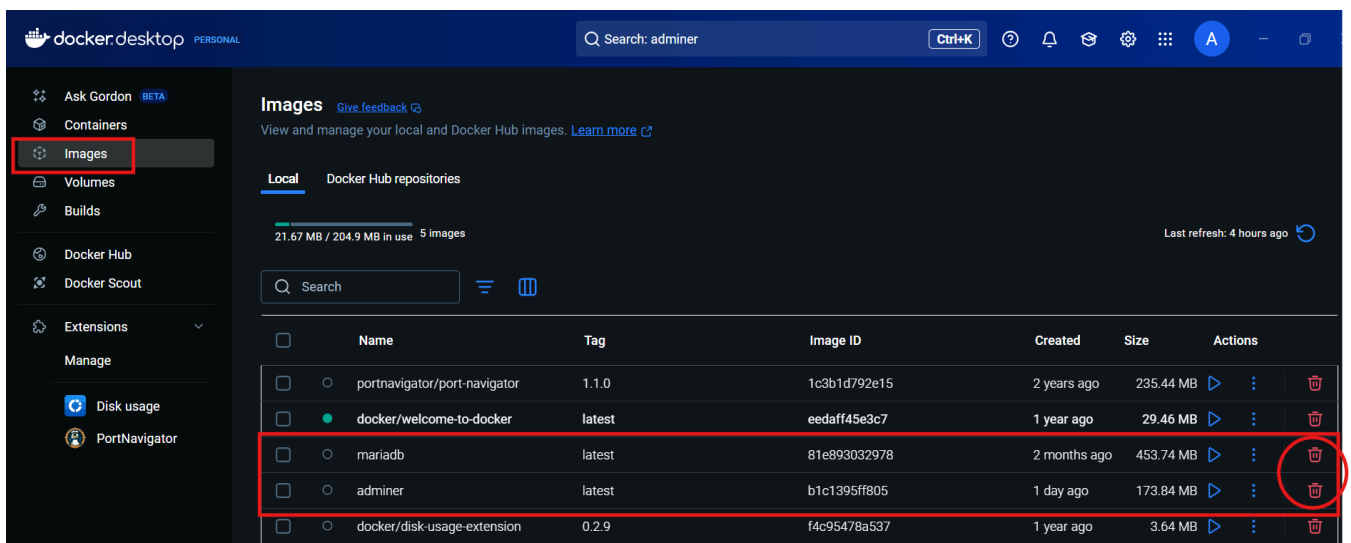
Primero vamos a acceder a `Containers` y eliminar los contenedores creados con anterioridad pulsando sobre la papelera que aparece a sus derechas.



Tanto en `db-DAW` como en `adminer-DAW` nos saldrá la siguiente advertencia en la cual pondremos `Delete forever`.

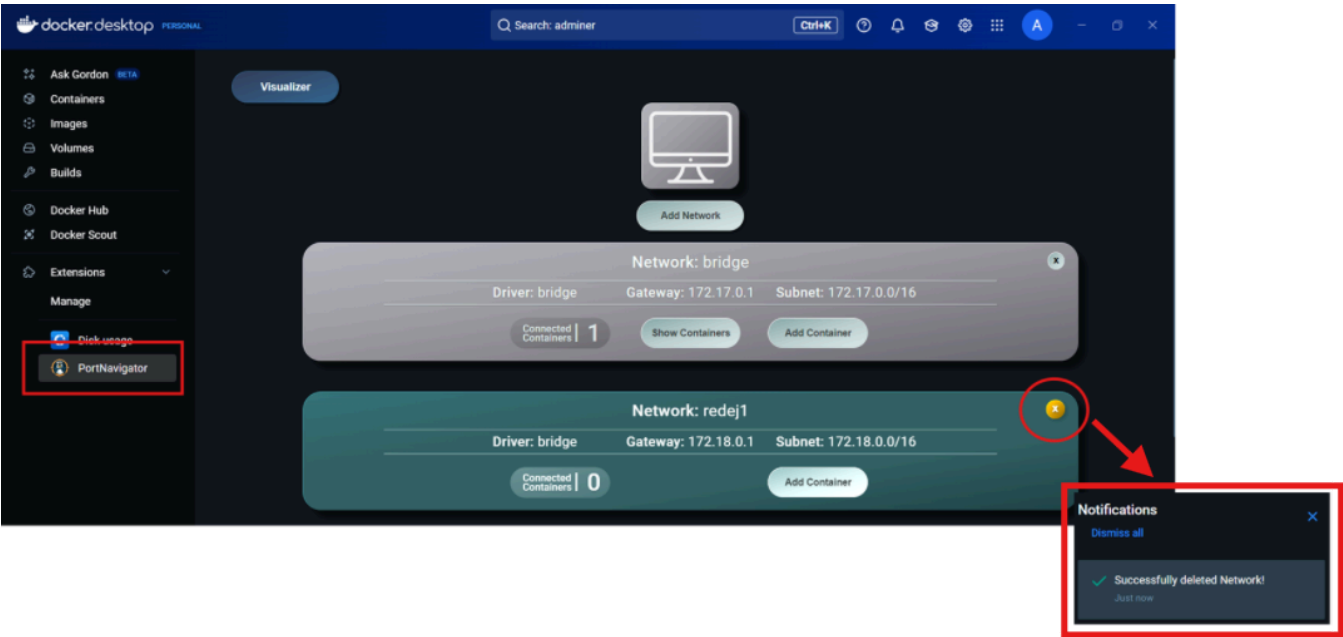


Lo mismo haremos con las imágenes tanto de `mariadb` como `adminer`.



En volúmenes ya no saldría nada ya que no hay nada que lo vincule.

Para eliminar la red, iremos a PortNavigator y buscaremos la red redej1 y pulsaremos sobre la cruz.



Por último, vamos a comprobar en Disk usage que el espacio ha disminuido con el borrado de estos elementos.

