

# Despliegue de Aplicaciones Web - Práctica entregable 3 - Ejercicio 1. Contenedores en red y Docker Desktop (resolución con Docker Desktop) - Roberto Rodríguez González

---

## Tabla de contenidos

---

1. [Previa](#)
2. [Crea una red bridge `redej1`](#)
3. [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](#)
4. [Crea un contenedor con Adminer o con phpMyAdmin que se pueda conectar al contenedor de la BD](#)
5. [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](#)
6. [Instala la extensión Disk Usage y muestra el espacio ocupado, borra algo](#)

## 1. Previa

---

Lo primero es crear el repositorio en GitHub:

### Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere?  
[Import a repository.](#)

Required fields are marked with an asterisk (\*).

Owner *	Repository name *
 roberrod0 ▾	/ tarea3Docker
 tarea3Docker is available.	

Luego, crear el repositorio en local, crear carpetas y añadirlas:

```

MINGW64:/c:/iCloudDrive/FP_DESARROLLO_WEB/24-25/DESPLIEGU
Roberto@DESKTOP-G34A0MP MINGW64 /c:/iCloudDrive/FP_DESARROLLO_WEB/24-25/DESPLIEGU
E/WORKSPACE.DESPLIEGUE/Tarea 3 - Docker/repositorioTarea3Docker
$ git init
Initialized empty Git repository in C:/iCloudDrive/FP_DESARROLLO_WEB/24-25/DESPLIEGU
E/WORKSPACE.DESPLIEGUE/Tarea 3 - Docker/repositorioTarea3Docker/.git/

Roberto@DESKTOP-G34A0MP MINGW64 /c:/iCloudDrive/FP_DESARROLLO_WEB/24-25/DESPLIEGU
E/WORKSPACE.DESPLIEGUE/Tarea 3 - Docker/repositorioTarea3Docker (main)
$ git status
On branch main

No commits yet

nothing to commit (create/copy files and use "git add" to track)

Roberto@DESKTOP-G34A0MP MINGW64 /c:/iCloudDrive/FP_DESARROLLO_WEB/24-25/DESPLIEGU
E/WORKSPACE.DESPLIEGUE/Tarea 3 - Docker/repositorioTarea3Docker (main)
$ git add .

Roberto@DESKTOP-G34A0MP MINGW64 /c:/iCloudDrive/FP_DESARROLLO_WEB/24-25/DESPLIEGU
E/WORKSPACE.DESPLIEGUE/Tarea 3 - Docker/repositorioTarea3Docker (main)
$ git commit -m "Creadas carpetas y .md"
[main (root-commit) 06392a2] Creadas carpetas y .md
3 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 ejercicio1/ejercicio1.md
create mode 100644 ejercicio2/Nuevo Documento de texto.md
create mode 100644 ejercicio3/Nuevo Documento de texto.md

Roberto@DESKTOP-G34A0MP MINGW64 /c:/iCloudDrive/FP_DESARROLLO_WEB/24-25/DESPLIEGU
E/WORKSPACE.DESPLIEGUE/Tarea 3 - Docker/repositorioTarea3Docker (main)
$ |

```

Vinculamos repositorios:


```

Roberto@DESKTOP-G34A0MP MINGW64 /c:/iCloudDrive/FP_DESARROLLO_WEB/24-25/DESPLIEGU
E/WORKSPACE.DESPLIEGUE/Tarea 3 - Docker/repositorioTarea3Docker (main)
$ git remote add origin https://github.com/roberrod0/tarea3Docker.git




Roberto@DESKTOP-G34A0MP MINGW64 /c:/iCloudDrive/FP_DESARROLLO_WEB/24-25/DESPLIEGU
E/WORKSPACE.DESPLIEGUE/Tarea 3 - Docker/repositorioTarea3Docker (main)
$ git remote -v
origin https://github.com/roberrod0/tarea3Docker.git (fetch)
origin https://github.com/roberrod0/tarea3Docker.git (push)

Roberto@DESKTOP-G34A0MP MINGW64 /c:/iCloudDrive/FP_DESARROLLO_WEB/24-25/DESPLIEGU
E/WORKSPACE.DESPLIEGUE/Tarea 3 - Docker/repositorioTarea3Docker (main)
$ git push -u origin main
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (5/5), 366 bytes | 183.00 KiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/roberrod0/tarea3Docker.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.


```





**area3Docker** Public
Pin Unwatch

---

 main
  1 Branch
  0 Tags
 
Add file
Code

---

 **roberrod0** Creadas carpetas y .md 06392a2 · 6 minutes ago 1 Commit

 ejercicio1	Creadas carpetas y .md	6 minutes ago
 ejercicio2	Creadas carpetas y .md	6 minutes ago
 ejercicio3	Creadas carpetas y .md	6 minutes ago

Creamos la rama ejercicio1:

```
Roberto@DESKTOP-G34A0MP MINGW64 /c/iCloudDrive/FP_DESARROLLO_WEB/24-25/DESPLIEGU
E/WORKSPACE.DESPLIEGU/Tarea 3 - Docker/repositorioTarea3Docker (main)
$ git checkout -b ejercicio1
Switched to a new branch 'ejercicio1'
```

## 2. Crea una red brige redej1

Instalada la extensión PortNavigator, creamos la red bridge como se nos pide:

X

### Add Network

Name:

---

☐ Advanced network settings

---

Subnetwork:

Gateway:

IP-Range:

Create Network

Network: redej1

X

---

Driver: bridge    Gateway: 172.18.0.1    Subnet: 172.18.0.0/16

---

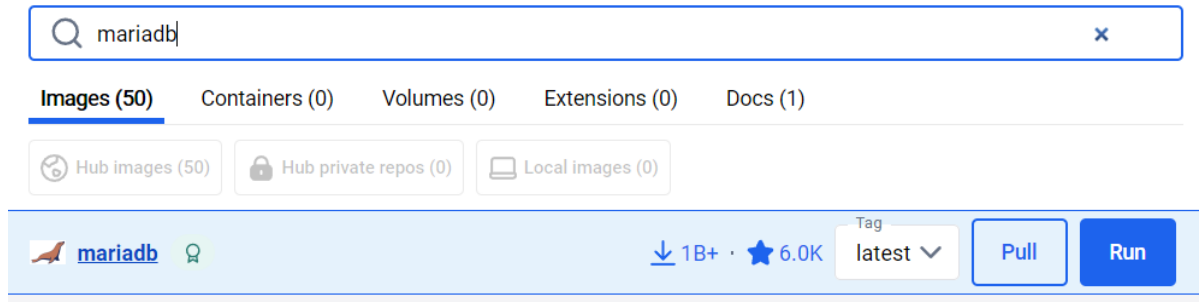
Connected Containers

0

Add Container

### 3. 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

Buscamos la imagen de mariaDB, la descargamos (Pull) y ejecutamos (Run):



La configuramos tal y como se nos pide. Los nombres de las variables los encontramos en la web <https://mariadb.com/kb/en/mariadb-server-docker-official-image-environment-variables/>:



## 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

MARIADB\_ROOT\_PASSWORD

Value

1234



Variable

MARIADB\_USER

Value

ROBERTO



Variable

MARIADB\_PASSWORD

Value

1234



Variable

MARIADB\_DATABASE

Value

DAW



Cancel

Run

El contenedor corriendo (creado en el puerto 3307 porque en el 3306 tenía mysql):

### Containers [Give feedback](#)

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

Container CPU usage ⓘ

0.03% / 800% (8 CPUs available)

Container memory usage ⓘ

180.3MB / 6.56GB

[Show charts](#)

Q Search



Only show running containers

<input type="checkbox"/>	Name	Container ID	Image	Port(s)	CPU (%)	Last started	Actions
<input type="checkbox"/>	<span style="color: green;">●</span> mariadb-ej1	6fa1fab353f8	<a href="#">mariadb:late</a>	3307:3306 <a href="#">↗</a>	0.03%	41 seconds ago	<div> </div>

Generamos un script SQL:

```
CREATE TABLE modulos (
  id INT AUTO_INCREMENT PRIMARY KEY,
  nombre VARCHAR(100) NOT NULL
);
```

```
INSERT INTO modulos (nombre) VALUES
('DESPLIEGE'),
('ENTORNO CLIENTE'),
('ENTORNO SERVIDOR'),
('PROGRAMACION');
```

## 4. Crea un contenedor con Adminer que se pueda conectar al contenedor de la BD

Buscamos la imagen de adminer y creamos un contenedor, de la misma forma que hicimos con la imagen de mariadb previamente:

### Containers [Give feedback](#)

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

Container CPU usage ⓘ

0.03% / 800% (8 CPUs available)

Container memory usage ⓘ

188.02MB / 6.56GB

Q Search



Only show running containers

<input type="checkbox"/>	Name	Container ID	Image	Port(s)
<input type="checkbox"/>	● mariadb-ej1	6fa1fab353f8	<a href="#">mariadb:latest</a>	<a href="#">3307:3306</a> ↗
<input type="checkbox"/>	● adminer-ej1	effb53546cf1	<a href="#">adminer:latest</a>	

Vamos a desconectar los contenedores de la red bridge que nos sale por defecto y conectarla a la que hemos creado ( `redej1` ):

Network: bridge

Driver: bridgeGateway: 172.17.0.1Subnet: 172.17.0.0/16

Connected Containers | 2

Hide Containers

Add Container

Container: adminer-ej1

ContainerID: effb53546cf1c93263b0583b...

Image: adminer:latest

Activity: Up 2 minutes

Type: tcp

IPv4Address: 172.17.0.3/16

Published Ports: null

Private Ports: 8080

Edit Ports

Disconnect

Connect to Networks

Container: mariadb-ej1

ContainerID: 6fa1fab353f82363eed56bb3...

Image: mariadb:latest

Activity: Up 10 minutes

Type: tcp

IPv4Address: 172.17.0.2/16

Published Ports: 3307:3306

Private Ports: null

Edit Ports

Disconnect

Connect to Networks

Connect

mariadb-ej1

Container to a Network:

redej1

Connect

Network: bridge

Driver: bridgeGateway: 172.17.0.1Subnet: 172.17.0.0/16

Connected Containers | 0

Add Container

Network: redej1

Driver: bridgeGateway: 172.18.0.1Subnet: 172.18.0.0/16

Connected Containers | 2

Show Containers

Add Container

## 5. 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

Conectamos a la BD:

Antes de llevar a cabo nada:

Creamos la tabla en Comando SQL:



Comando SQL - mariadb-ej1 - / x +

localhost:8081/?server=mariadb-ej1&username=ROBERTO&db=DAW&sql=CREATE%20TABLE%20modulos%20(%0A%2...

Idioma: Español MariaDB » mariadb-ej1 » DAW » Comando SQL ROBERTO

Adminer 5.2.1

BD: DAW

Comando SQL Importar Exportar Crear tabla

registros modules

### Comando SQL

```
CREATE TABLE `modulos` (
  id INT AUTO_INCREMENT PRIMARY KEY,
  nombre VARCHAR(100) NOT NULL
)
```

Consulta ejecutada, 0 registros afectados. (0.065 s) [Modificar](#)

```
INSERT INTO `modulos` (nombre) VALUES
('DESPLIEGE'),
('ENTORNO CLIENTE'),
('ENTORNO SERVIDOR'),
('PROGRAMACION')
```

Consulta ejecutada, 4 registros afectados. (0.010 s) [Modificar](#)

```
CREATE TABLE `modulos` (
  id INT AUTO_INCREMENT PRIMARY KEY,
  nombre VARCHAR(100) NOT NULL
);

INSERT INTO `modulos` (nombre) VALUES
('DESPLIEGE'),
('ENTORNO CLIENTE'),
('ENTORNO SERVIDOR'),
('PROGRAMACION');
```

Ejecutar Limit rows:  ☐ Parar en caso de error ☐ Mostrar solamente errores

Estos son los datos que muestra la tabla:

Mostrar: modules - mariadb-ej1 x +

localhost:8081/?server=mariadb-ej1&username=ROBERTO&db=DAW&select=modulos

Idioma: Español MariaDB » mariadb-ej1 » DAW » Mostrar: modules ROBERTO

Adminer 5.2.1

BD: DAW

Comando SQL Importar Exportar Crear tabla

registros modules

### Mostrar: modules

Visualizar contenido Mostrar estructura Modificar tabla Nuevo Registro

Mostrar Condición Ordenar Limite Longitud de texto Acción

50 100 Mostrar

SELECT \* FROM `modulos` LIMIT 50 (0.002 s) [Modificar](#)

<input type="checkbox"/> Modify	id	nombre
<input type="checkbox"/> modificar	1	DESPLIEGE
<input type="checkbox"/> modificar	2	ENTORNO CLIENTE
<input type="checkbox"/> modificar	3	ENTORNO SERVIDOR
<input type="checkbox"/> modificar	4	PROGRAMACION

Resultado completo ☐ 4 registros  Selected (0)    [Exportar \(4\)](#)

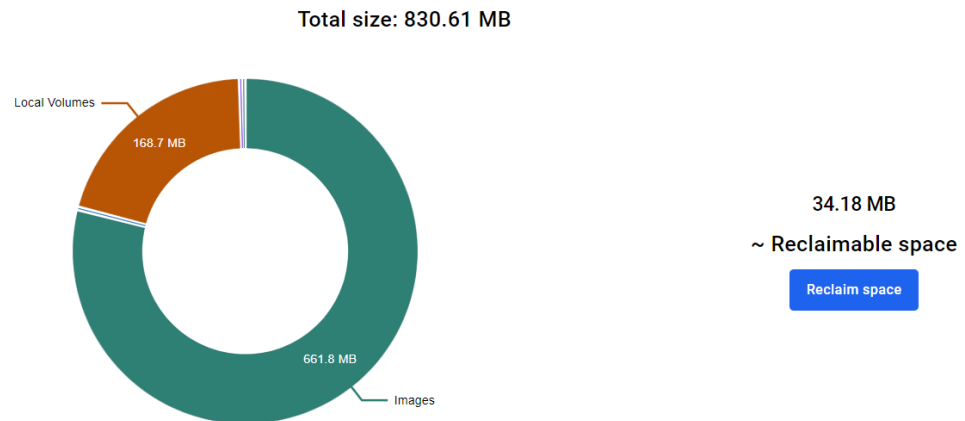
[Importar](#)

## 6. Instala la extensión Disk Usage y muestra el espacio ocupado, borra algo

Nos decidimos por la extensión Disk Usage básicamente por ser la primera opción. La instalamos y mostramos el espacio ocupado:

### Disk Usage [Give Feedback](#)

Optimize your disk space by removing unused objects from Docker Desktop



Objects	Total Count	Active Count	Size	Est. Reclaimable
Images	4	2	661.8 MB	34.18 MB (5%)
Containers	2	2	110.6 KB	0 B (0%)
Local Volumes	1	1	168.7 MB	0 B (0%)
Build Cache	0	0	0 B	0 B
Dangling	1	0	0 B	0 B (0%)
			Total	34.18 MB

Borramos un contenedor y mostramos Disk Usage después:

### Delete container?

The 'mariadb-ej1' container is selected for deletion. Any anonymous volumes associated with this container are also deleted.

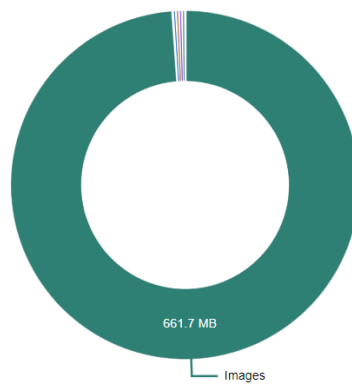
Cancel

Delete forever

**Disk Usage** [Give Feedback](#)

Optimize your disk space by removing unused objects from Docker Desktop

Total size: 661.78 MB



487.9 MB

~ Reclaimable space

[Reclaim space](#)

Objects	Total Count	Active Count	Size	Est. Reclaimable
Images	4	1	661.7 MB	487.9 MB (73%)
Containers	1	1	86.02 KB	0 B (0%)
Local Volumes	0	0	0 B	0 B
Build Cache	0	0	0 B	0 B
Dangling	1	0	0 B	0 B (0%)
			Total	487.9 MB

Ahora, tal y como se nos pide, procedemos a borrar todo:

**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



Add Network

Network: bridge



Driver: bridge

Gateway: 172.17.0.1

Subnet: 172.17.0.0/16

Connected  
Containers | 0

Add Container

Network: host



Driver: host

Gateway: null

Subnet: null

Connected  
Containers | 0

Add Container

Network: none



Driver: null

Gateway: null

Subnet: null

Connected  
Containers | 0

Add Container