

Proyecto 4 - Enlazar phpMyAdmin con un contenedor Docker con MySQL

Objetivo

Creación de una base de datos, usuario y password del usuario así como el root

MYSQL_DATABASE=bd_biblioteca

MYSQL_USER=user_biblioteca

MYSQL_PASSWORD=pass_biblioteca

MYSQL_ROOT_PASSWORD=root_pass

- creación de la estructura y su carga inicial en una imagen mysql
- Creamos un contenedor MySQL con persistencia de datos
- Creación de un contenedor con phpMyAdmin este enlazado al anterior
- Comprobar el funcionamiento de phpMyAdmin

Crea la siguiente estructura

```
10xFCProyecto4/  
  
├─ Dockerfile  
├─ my.cnf  
└─ init/  
  
    └─ 1-schema.sql  
  
        └─ 2-datos.sql
```

Clonar repositorio

```
git clone url
```

Cambia al directorio donde se ha clonado.

```
cd nombreCarpeta
```

Definir una network bridge

```
docker network create mi_red
```

Desde la raíz del proyecto, construye la imagen

```
docker build -t mysql_biblioteca_personal .
```

Ejecutar el contenedor con MySQL indicamos que está en la red --network mi-red

```
docker run -d --rm --name mysql_biblioc --network mi_red -p 3306:3306 -v mysql_data_biblio:/var/lib/mysql mysql_biblioteca_personal
```

Comprobación del contenedor en ejecución (running o up)

```
D:\AMOR\docker\10xFCProyecto4>docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
62b9c4a82b6e	mysql_biblioteca_personal	"docker-entrypoint.s..."	14 seconds ago	Up 7 seconds	0.0.0.0:3306->3306/tcp, 33060/tcp

Crear un contenedor phpMyAdmin indicando que este --network mi-red

phpMyAdmin es una herramienta web escrita en PHP que permite **administrar bases de datos MySQL o MariaDB** a través de un interfaz gráfico fácil de usar, en lugar de hacerlo desde línea de comandos.

Variable de entorno **PMA_HOST=mysql_biblioc** es un argumento que se utiliza en la ejecución de contenedores Docker, especialmente cuando se usa una imagen phpMyadmin.

PMA_HOST: se especifica el host del servidor MySQL o MariaDB al que phpMyAdmin debe conectarse.

Con el siguiente comando creamos el conteneodr phpadmin_biblioc:

```
docker run -d --rm --name phpadmin_biblioc --network mi_red -e PMA_HOST=mysql_biblioc -p 8080:80 phpmyadmin
```

Comprobar que los dos contenedores en ejecución

```
D:\AMOR\docker\10xFCProyecto4>docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
ac3e94054dd7	phpmyadmin	"/docker-entrypoint..."	25 seconds ago	Up 21 seconds	0.0.0.0:8080->80/tcp	phpadmin_biblioc
62b9c4a82b6e	mysql_biblioteca_personal	"docker-entrypoint.s..."	4 minutes ago	Up 4 minutes	0.0.0.0:3306->3306/tcp, 33060/tcp	mysql_biblioc

Mostrar volumen

```
D:\AMOR\docker\10xFCProyecto4>docker volume ls
```

DRIVER	VOLUME NAME
local	mysql_data_biblio

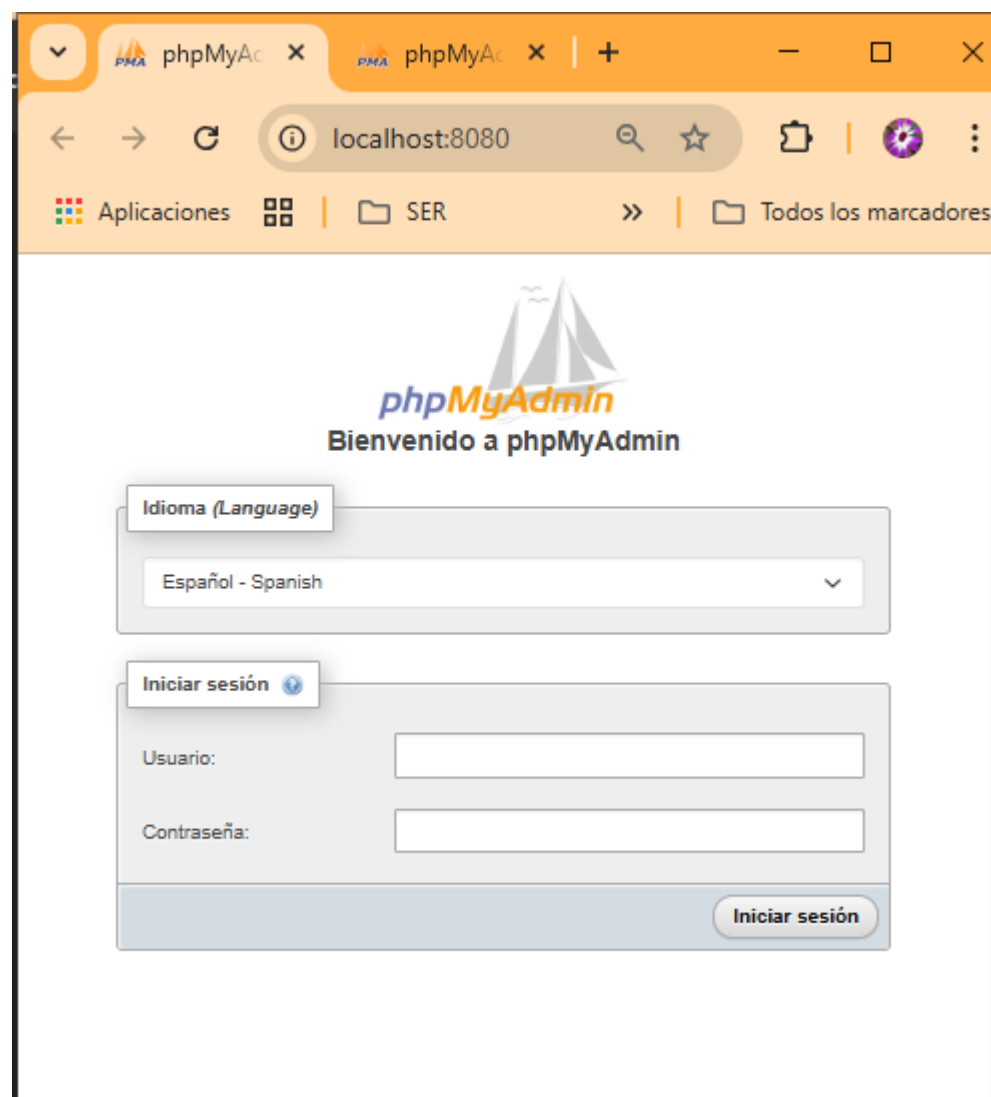
Mostrar network

```
D:\AMOR\docker\10xFCProyecto4>docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
1fdf3b0b5c4b        bridge              bridge              local
e37b47e392a5        host                host                local
f929d5c0eb7e        mi_red              bridge              local
88486f5b3523        my_network          bridge              local
9aea824e132b        none                null                local
D:\AMOR\docker\10xFCProyecto4\
```

Usar base de datos, que tablas y su contenido

```
D:\AMOR\docker\10xFCProyecto4>docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
1fdf3b0b5c4b        bridge              bridge              local
e37b47e392a5        host                host                local
f929d5c0eb7e        mi_red              bridge              local
88486f5b3523        my_network          bridge              local
9aea824e132b        none                null                local
D:\AMOR\docker\10xFCProyecto4\
```

Cliente MySQL: WorkBench, Visual Studio Code y con <http://localhost:8080>



Mostrar la base de datos bd_biblioteca

