



## Ejercicio 3 - Contenedores en Red Adminer y MariaBD

1. Crea una red bridge redbd

```
docker network create --driver bridge redbd
```

```
PS C:\Users\alumno> docker network create --driver bridge redbd
90a6f041c41849b160726198c11d7670ef7da87245f9a798a230ce8ee4f4521f
PS C:\Users\alumno>
```

2. Crea un contenedor con una imagen de mariaDB que estará en la red redbd. Este contenedor se ejecutará en segundo plano, y será accesible a través del puerto 3306. (Es necesario definir la contraseña del usuario root y un volumen de datos persistente)

```
docker run -d --name mariadb-container --network redbd -e MYSQL_ROOT_PASSWORD=123 -v maria
```

```
PS C:\Users\alumno> docker run -d --name mariadb-container --network redbd -e MYSQL_ROOT_PASSWORD=123 -v maria
bd_data:/var/lib/mysql -p 3306:3306 mariadb:latest
c37ee6b7d65bf0f56366d5f128383f7d0d65400deff46e848f91c6e73dca4b11
```

Comprobamos que el contenedor se creo correctamente haciendo un  
docker ps

```
PS C:\Users\alumno> docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
c37ee6b7d65b	mariadb:latest	"docker-entrypoint.s..."	7 seconds ago	Up 7 seconds	0.0.0.0:3306->3306/tcp
mariadb-container					

3. Crear un contenedor con Adminer o con phpMyAdmin que se pueda  
conectar al contenedor de la BD

```
docker run -d --name adminer-container --network redbd -p 8080:8080 adminer:latest
```

```
PS C:\Users\alumno> docker run -d --name adminer-container --network redbd -p 8080:8080 adminer:latest
9f4fa646270e0cb85aee04a205161dbba14c4e41a7804ddaaa6c5d7a7863bf9b
```

Comprobamos que el contenedor se creo correctamente haciendo un  
docker ps

```
PS C:\Users\alumno> docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
9f4fa646270e	adminer:latest	"entrypoint.sh php -..."	About a minute ago	Up About a minute	0.0.0.0:8080->8080/tcp
adminer-container					

4. Desde la interfaz gráfica, crear una base de datos y una tabla en el servidor  
de base de  
datos

Idioma: Español

Adminer 4.8.1

Login

<b>Motor de base de datos</b>	MySQL
<b>Servidor</b>	db
<b>Usuario</b>	root
<b>Contraseña</b>	...
<b>Base de datos</b>	Base de Datos

☒ Guardar contraseña

Adminer 4.8.1

Base de datos: BaseDocker

DB: BaseDocker

[Comando SQL](#)
[Importar](#)
[Exportar](#)
[Crear tabla](#)

No existen tablas.

[Base de datos creada. 11:56:55](#)
[Comando SQL](#)

[Modificar Base de datos](#)
[Esquema de base de datos](#)
[Privilegios](#)

### Tablas y vistas

No existen tablas.

[Crear tabla](#)
[Crear vista](#)

### Procedimientos

[Crear procedimiento](#)
[Crear función](#)

### Eventos

[Crear Evento](#)

Adminer 4.8.1

Crear tabla

DB: BaseDocker

[Comando SQL](#)
[Importar](#)
[Exportar](#)
[Crear tabla](#)

No existen tablas.

Nombre de la tabla: Imagen (motor) (colación)

Nombre de columna	Tipo	Longitud	Opciones	NULL	AI?	+
Tamaño	int	10		<input type="checkbox"/>	<input type="radio"/>	+ ↑ ↓ ×
	int			<input type="checkbox"/>	<input type="radio"/>	+ ↑ ↓ ×

Incremento automático:  ☐ Valores predeterminados ☐ Comentario

[Particionar por](#)

Eliminamos los contenedores utilizados usando `docker rm` y el nombre del contendor si el contenedor que queremos borrar esta en ejecucion deberemos

hacer un docker stop y el nombre del contendor

```
docker stop adminer-container
```

```
PS C:\Users\alumno> docker stop adminer-container  
adminer-container
```

```
docker rm adminer-container
```

```
PS C:\Users\alumno> docker rm adminer-container  
adminer-container _
```

```
docker stop mariadb-container
```

```
PS C:\Users\alumno> docker stop mariadb-container  
mariadb-container _
```

```
docker rm mariadb-container
```

```
PS C:\Users\alumno> docker rm mariadb-containe  
mariadb-container
```