

Trabajo con Imágenes - Docker

Trabajo con Imágenes - Docker

SERVIDOR WEB

SERVIDOR DE BASE DE DATOS

1. Arrancar un contenedor que se llame bbdd y que ejecute una instancia de la imagen mariadb para que sea accesible desde el puerto 3336.

Antes de arrancarlo estable las variables de entorno necesarias para que:

Pantallazo que desde el navegador muestre el fichero `index.html`

Pantallazo que desde un navegador muestre la salida del script `mes.php`

Pantallazo donde se vea el tamaño del contenedor `web` después de crear los dos ficheros.

Pantallazo donde desde un cliente de base de datos (instalado en tu ordenador) se pueda observar que hemos podido conectarnos al servidor de base de datos con el usuario creado y que se ha creado la base de datos prueba (`show databases`). El acceso se debe realizar desde el ordenador que tenéis instalado docker, no hay que acceder desde dentro del contenedor, es decir, no usar `docker exec` .

Pantallazo donde se comprueba que no se puede borrar la imagen `mariadb` mientras el contenedor `bbdd` está creado.

SERVIDOR WEB

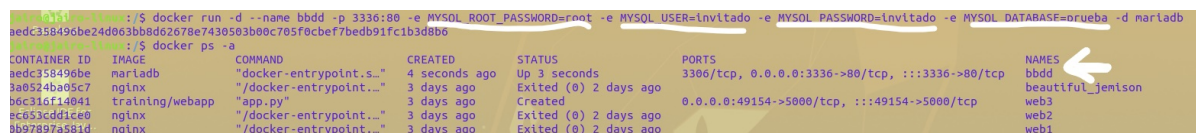
SERVIDOR DE BASE DE DATOS

1. Arrancar un contenedor que se llame bbdd y que ejecute una instancia de la imagen mariadb para que sea accesible desde el puerto 3336.

Antes de arrancarlo estable las variables de entorno necesarias para que:

- La contraseña de root sea `root`
- Crear una base de datos automáticamente al arrancar que se llame `prueba`
- Crear el usuario `invitado` con la contraseña `invitado`

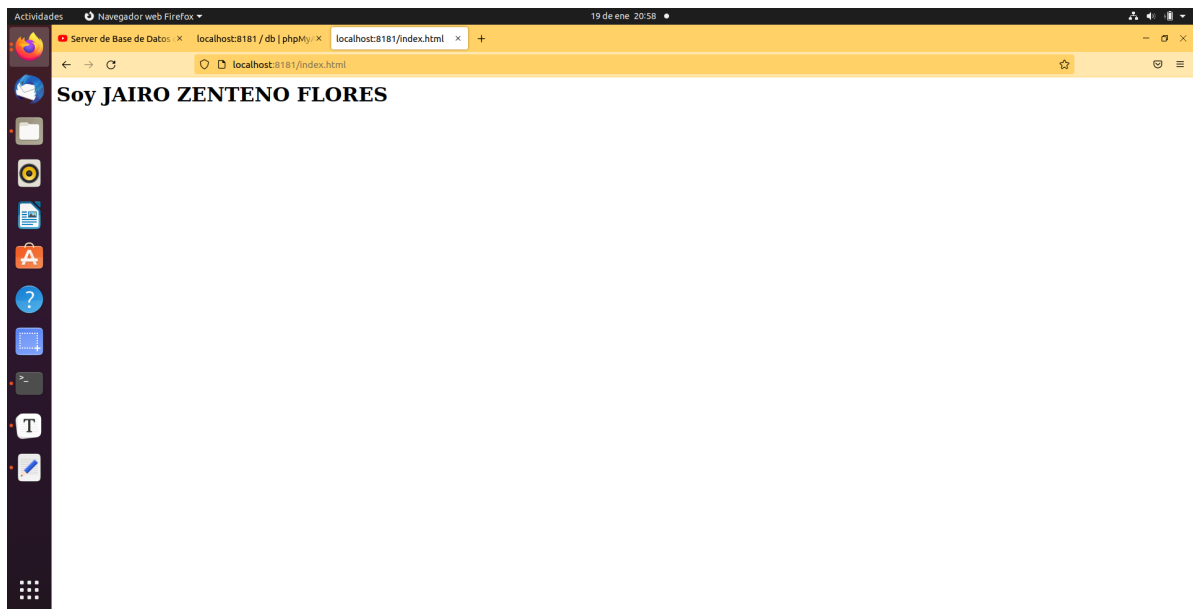
```
docker run -d --name bbdd -p 3336:80 -e MYSQL_ROOT_PASSWORD=root -e MYSQL_USER=invitado -e MYSQL_PASSWORD=invitado -e MYSQL_DATABASE=prueba -d mariadb
```



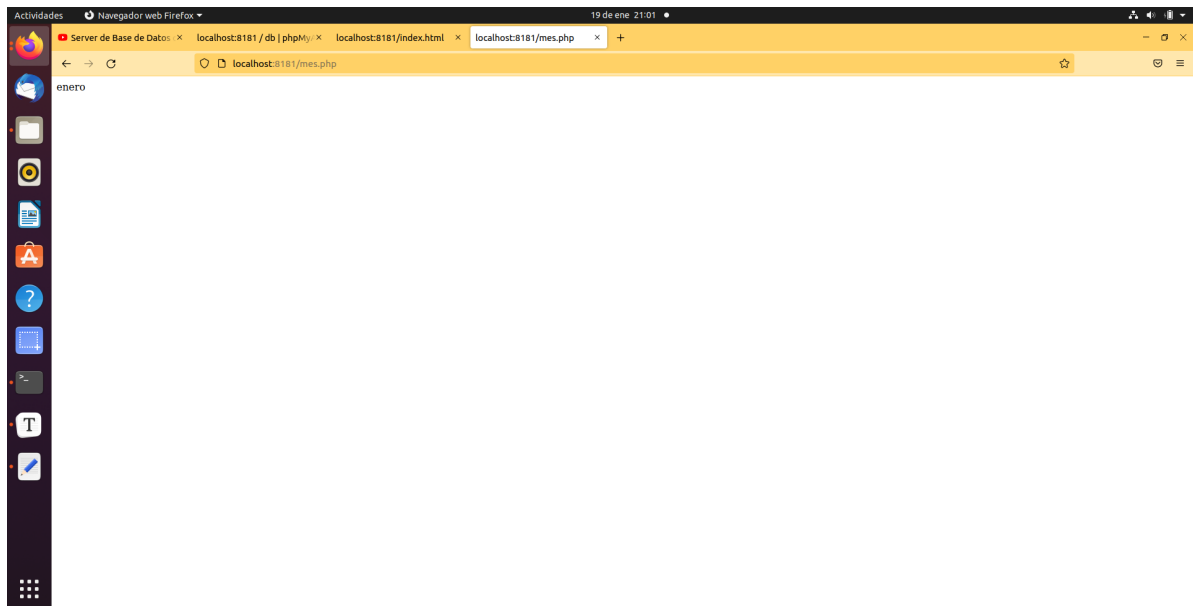
The screenshot shows a terminal window with the command `docker run -d --name bbdd -p 3336:80 -e MYSQL_ROOT_PASSWORD=root -e MYSQL_USER=invitado -e MYSQL_PASSWORD=invitado -e MYSQL_DATABASE=prueba -d mariadb` being executed. Below the command, the output of `docker ps -a` is shown, listing several containers. The container named 'bbdd' is highlighted with a white arrow pointing to its 'NAME' column.

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
aedc358496be	mariadb	"docker-entrypoint.s..."	4 seconds ago	Up 3 seconds	3306/tcp, 0.0.0.0:3336->80/tcp, :::3336->80/tcp	bbdd
3a0524ba05c7	nginx	"/docker-entrypoint..."	3 days ago	Exited (0) 2 days ago		beautiful_jemison
b6c316f14041	training/webapp	"app.py"	3 days ago	Created	0.0.0.0:49154->5000/tcp, :::49154->5000/tcp	web3
ec653cddice0	nginx	"/docker-entrypoint..."	3 days ago	Exited (0) 2 days ago		web2
bb97897a581d	nginx	"/docker-entrypoint..."	3 days ago	Exited (0) 2 days ago		web1

Pantallazo que desde el navegador muestre el fichero `index.html`

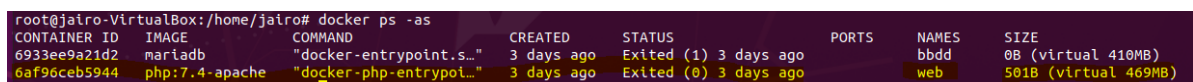


Pantallazo que desde un navegador muestre la salida del script `mes.php`

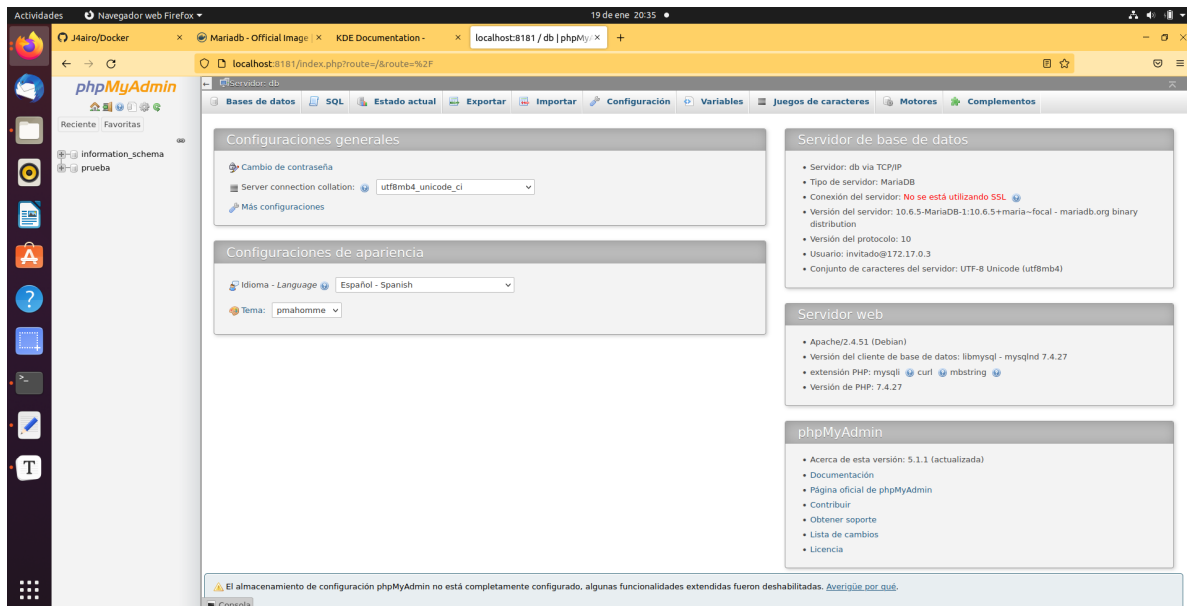


Pantallazo donde se vea el tamaño del contenedor `web` después de crear los dos ficheros.

```
docker ps -as
```



Pantallazo donde desde un cliente de base de datos (instalado en tu ordenador) se pueda observar que hemos podido conectarnos al servidor de base de datos con el usuario creado y que se ha creado la base de datos prueba (show databases). El acceso se debe realizar desde el ordenador que tenéis instalado docker, no hay que acceder desde dentro del contenedor, es decir, no usar docker exec .



Pantallazo donde se comprueba que no se puede borrar la imagen mariadb mientras el contenedor bbdd está creado.

```
docker rmi mariadb
```

