

# Solución Ejercicio 1

---

## Solución Ejercicio 1

### Servidor web

- Descargar imagen
- Crear contenedor
- Insertar index.html
- Insertar mes.php
- Ver tamaño del contenedor
- Borrar contenedor

### Servidor de base de datos

- Iniciar contenedor
- Conexión al servidor
- Intentando borrar imagen "mariadb" mientras el contenedor "mariadb" está creado

### Otros

- Listar imágenes
- Borrar contenedores

## Servidor web

---

### Descargar imagen

Lo primero que debo hacer es descargar la imagen de Docker que me servirá para generar el contenedor. En este caso, descargo la versión indicada en el enunciado de la tarea: `php:7.4-apache`.

Para ello ejecuto el siguiente comando:

```
sudo docker pull php:7.4-apache
```

```
samuel@soto: ~
Archivo Editar Ver Buscar Terminal Ayuda
samuel@soto:~$ sudo docker pull php:7.4-apache
7.4-apache: Pulling from library/php
c229119241af: Pull complete
47e86af584f1: Pull complete
e1bd55b3ae5f: Pull complete
1f3a70af964a: Pull complete
0f5086159710: Pull complete
7d9c764dc190: Pull complete
ec2bb7a6eead: Pull complete
9d9132470f34: Pull complete
fb23ab197126: Pull complete
cbdd566be443: Pull complete
be224cc1ae0f: Pull complete
629912c3cae4: Pull complete
f1bae9b2bf5b: Pull complete
Digest: sha256:ef46a8784e3f71cd19fa3833850a91e80aa592e7d8339194defd8adb17ed3fa1
Status: Downloaded newer image for php:7.4-apache
docker.io/library/php:7.4-apache
samuel@soto:~$
```

## Crear contenedor

A continuación debo crear el contenedor. En el enunciado de la tarea se especifica que debo darle el nombre "web" y habilitar el puerto "8000" de mi host para que pueda comunicarse a través del puerto "80" del contenedor. Además, debo sincronizar un directorio de mi host con el directorio "/www/html" del contenedor. Para hacerlo todo en uno, ejecuto el siguiente comando:

```
sudo docker run -d -p 8000:80 -it --name web -v "$(pwd)"/src:/var/www/html php:7.4-apache
```

```
samuel@soto: ~/Escritorio/docker/ejercicio1/servidor-web
Archivo Editar Ver Buscar Terminal Ayuda
samuel@soto:~/Escritorio/docker/ejercicio1/servidor-web$ sudo docker run -d -p 8000:80 -it --name web -v "$(pwd)"/src:/var/www/html php:7.4-apache
7e36bdc73678bde13c7ddf58e3c167ae745ae0c090f1ae2bf2574b9eb77a2825
samuel@soto:~/Escritorio/docker/ejercicio1/servidor-web$
```

## Insertar index.html

Actualmente, al acceder a la URL "<http://localhost:8000>" me da un "Forbidden" ya que, al no existir archivo alguno en el directorio raíz del servidor, no puedo acceder al mismo. Por ello creo un archivo "index.html" en mi directorio local "/src". De esta forma, se sincronizará con el directorio raíz del servidor Apache del contenedor y ya me permitirá ver la página.

El código que inserto en ese archivo es el siguiente:

```
<h1>HOLA, SOY SAMUEL SOTO LOPEZ</h1>
```

Si ahora acceso al navegador a través de "<http://localhost:8000>" ya puedo ver mi index.html funcionando sin problema:

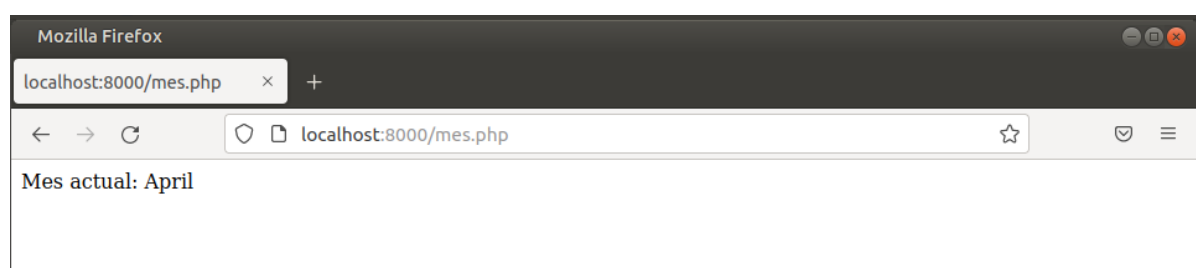


## Insertar mes.php

A continuación procedo a insertar un archivo "mes.php" que me muestre el mes de la fecha actual. Para ello utilizo el siguiente script PHP:

```
<?php echo "Mes actual: " . date("F"); ?>
```

Se muestra así:



## Ver tamaño del contenedor

Una vez he agregado estos dos archivos, visualizo el tamaño actual de mi contenedor. Para ello utilizo el siguiente comando:

```
sudo docker ps -s
```

```

samuel@soto:~$ sudo docker ps -s
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                               NAMES    SIZE
ba92caef4c48   php:7.4-apache  "docker-php-entrypoi..."  About a minute ago  Up About a minute  0.0.0.0:8000->80/tcp, :::8000->80/tcp  web      2B (virtual 452MB)

```

## Borrar contenedor

Antes de borrar el contenedor debo detenerlo, ya que Docker no permite eliminar contenedores que se encuentran en ejecución. Para detener el contenedor "web" utilizo el siguiente comando:

```
sudo docker stop web
```

```

samuel@soto: ~/Escritorio/docker/ejercicio1/servidor-web
Archivo Editar Ver Buscar Terminal Ayuda
samuel@soto:~/Escritorio/docker/ejercicio1/servidor-web$ sudo docker stop web
web
samuel@soto:~/Escritorio/docker/ejercicio1/servidor-web$

```

Finalmente, utilizo este otro comando para eliminar el contenedor:

```
sudo docker rm web
```

```

samuel@soto: ~/Escritorio/docker/ejercicio1/servidor-web
Archivo Editar Ver Buscar Terminal Ayuda
samuel@soto:~/Escritorio/docker/ejercicio1/servidor-web$ sudo docker rm web
web
samuel@soto:~/Escritorio/docker/ejercicio1/servidor-web$

```

Para confirmar que he borrado el contenedor correctamente, ejecuto el siguiente comando, que me lista todos los contenedores que tengo en el sistema:

```
sudo docker ps
```

```

samuel@soto: ~/Escritorio/docker/ejercicio1/servidor-web
Archivo Editar Ver Buscar Terminal Ayuda
samuel@soto:~/Escritorio/docker/ejercicio1/servidor-web$ sudo docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                               NAMES
samuel@soto:~/Escritorio/docker/ejercicio1/servidor-web$

```

Como se puede apreciar, ya no existe dicho contenedor.

# Servidor de base de datos

## Iniciar contenedor

Para iniciar un contenedor desde la imagen "mariadb", con nombre "bbdd", contraseña de root "root", con nombre de base de datos por defecto "prueba" y con creación de un nuevo usuario "invitado" que utilice la contraseña "invitado" ejecuto el siguiente comando:

```
$ sudo docker run --detach --name mariadb --env MARIADB_DATABASE=prueba --env MARIADB_USER=invitado --env MARIADB_PASSWORD=invitado --env MARIADB_ROOT_PASSWORD=root mariadb:latest
```

```
samuel@soto: ~
Archivo Editar Ver Buscar Terminal Ayuda
samuel@soto:~$ sudo docker run --detach --name mariadb --env MARIADB_DATABASE=prueba --env MARIADB_USER=invitado --env MARIADB_PASSWORD=invitado --env MARIADB_ROOT_PASSWORD=root mariadb:latest
[sudo] contraseña para samuel:
Unable to find image 'mariadb:latest' locally
latest: Pulling from library/mariadb
4d32b49e2995: Pull complete
83231033a6d8: Pull complete
bbf8bcd9a929: Pull complete
6c260950e90f: Pull complete
35baff4c5363: Pull complete
16cdd7e19197: Pull complete
813bf88c4cde: Pull complete
e5fe98b3cc0e: Pull complete
98f9c19436dd: Pull complete
9ee83168eeb9: Pull complete
b79ef8792161: Pull complete
Digest: sha256:89f52efa423413027694120540ef03da857f1105e393d728c198db9160db49d1
Status: Downloaded newer image for mariadb:latest
5390c82569fc510ac7885276312a0cc80aa59f65f5e9c0587cc22605ae8f78a
samuel@soto:~$
```

El comando anterior contiene las distintas variables de entorno para establecer la configuración que se requiere en esta tarea:

- `MARIADB_DATABASE`: nombre de la base de datos por defecto.
- `MARIADB_USER`: nombre del usuario por defecto para la base de datos.
- `MARIADB_PASSWORD`: contraseña para el usuario por defecto.
- `MARIADB_ROOT_PASSWORD`: contraseña para el usuario "root".

## Conexión al servidor

Para conectarme al servidor de bases de datos desde el exterior del contenedor docker, necesito obtener primero la IP de dicho contenedor. Lo hago a través del siguiente comando:

```
sudo docker inspect -f '{{range .NetworkSettings.Networks}}{{.IPAddress}}{{end}}' mariadb
```

```
samuel@soto: ~
Archivo Editar Ver Buscar Terminal Ayuda
samuel@soto:~$ sudo docker inspect -f '{{range .NetworkSettings.Networks}}{{.IPAddress}}{{end}}' mariadb
172.17.0.2
samuel@soto:~$
```

Tras averiguar la IP utilizo el siguiente comando para conectarme al servidor de bases de datos:

```
mariadb -h 172.17.0.2 -u invitado -p
```

```

samuel@soto: ~
Archivo Editar Ver Buscar Terminal Ayuda
samuel@soto:~$ mariadb -h 172.17.0.2 -u invitado -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 4
Server version: 10.7.3-MariaDB-1:10.7.3+maria~focal mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>

```

Para poder realizar la conexión, previamente he instalado el cliente de MariaDB en mi host Linux. El nombre del paquete es `mariadb-client-core-10.3`.

A continuación muestro conexión completa al servidor de bases de datos, utilizando la sentencia `SHOW DATABASES;` en MariaDB.

```

samuel@soto: ~
Archivo Editar Ver Buscar Terminal Ayuda
samuel@soto:~$ mariadb -h 172.17.0.2 -u invitado -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 6
Server version: 10.7.3-MariaDB-1:10.7.3+maria~focal mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> SHOW DATABASES;
+-----+
| Database |
+-----+
| information_schema |
| prueba |
+-----+
2 rows in set (0.000 sec)

MariaDB [(none)]>

```

## Intentando borrar imagen "mariadb" mientras el contenedor "mariadb" está creado

Para intentar eliminar la imagen "mariadb" (con una instancia de dicha imagen levantada) utilizo el siguiente comando:

```
sudo docker rm "mariadb"
```

```

samuel@soto: ~
Archivo Editar Ver Buscar Terminal Ayuda
Error response from daemon: You cannot remove a running container 5390c82569fc510ac7885276312a0cc86aa59f65fd5e9c0587cc22605ae8f78a. Stop the container before attempting removal or force remove
samuel@soto:~$

```

Docker no me permite borrar esta imagen ya que existe un contenedor en ejecución que la utiliza. Me pide detener el contenedor antes de intentar eliminarla de nuevo.

## Otros

## Listar imágenes

Para listar las imágenes que tengo en mi registro local, ejecuto el siguiente comando:

```
sudo docker image ls
```

```
samuel@soto: ~
Archivo Editar Ver Buscar Terminal Ayuda
samuel@soto:~$ sudo docker image ls
REPOSITORY    TAG                IMAGE ID           CREATED          SIZE
mariadb        latest             665218ee5fdf      4 days ago      414MB
php            7.4-apache         eca4459c9842      7 days ago      452MB
samuel@soto:~$
```

## Borrar contenedores

Si bien el contenedor del servidor web ya ha sido eliminado (a requerimiento del enunciado de la tarea) incluyo este apartado genérico que también se solicita. En este caso, borro únicamente el contenedor "mariadb", que es el único que queda (el proceso de borrado del contenedor "web" ya se ha detallado previamente en su apartado correspondiente).

Antes de borrar el contenedor "mariadb", primero debo detenerlo:

```
sudo docker stop mariadb
```

```
samuel@soto: ~
Archivo Editar Ver Buscar Terminal Ayuda
samuel@soto:~$ sudo docker stop mariadb
mariadb
samuel@soto:~$
```

Posteriormente, ejecuto el siguiente comando para eliminarlo:

```
sudo docker rm mariadb
```

```
samuel@soto: ~
Archivo Editar Ver Buscar Terminal Ayuda
samuel@soto:~$ sudo docker rm mariadb
mariadb
samuel@soto:~$
```

Contenedor borrado. Ejecuto el siguiente comando, que lista los contenedores existentes en mi sistema, para asegurarme:

```
sudo docker ps
```

```

samuel@soto: ~
Archivo Editar Ver Buscar Terminal Ayuda
samuel@soto:~$ sudo docker ps
CONTAINER ID    IMAGE    COMMAND    CREATED    STATUS    PORTS    NAMES
samuel@soto:~$
```

Como puede apreciarse, no existen contenedores.