

Sistemas Informáticos

Tarea Docker 3: Trabajo con volúmenes

En este ejercicio trabajaremos con volúmenes en Docker:

1. Crearemos un volumen llamado *miweb*.
2. Crearemos un contenedor con la imagen *php:7.4-apache* que descargamos en el ejercicio anterior, en cuyo directorio */var/www/html* montaremos el volumen creado.
3. Crearemos y copiaremos un fichero en el directorio */var/www/html* como en el ejercicio anterior. Después accederemos al contenedor desde el navegador.
4. Borraremos el contenedor, crearemos uno nuevo y montaremos el mismo volumen.
5. Al acceder nuevamente al fichero creado desde el navegador comprobaremos si sigue existiendo.

Creación del volumen y el fichero

Primero crearemos el volumen llamado “*miweb*”, lo mostraremos en la lista y después crearemos y copiaremos el fichero.

```
osboxes@osboxes:~$ docker volume create miweb 1
miweb
osboxes@osboxes:~$ docker volume ls 2
DRIVER    VOLUME NAME
local     miweb
```

1) Utilizamos ***docker volume create miweb*** para crear el volumen *miweb*.

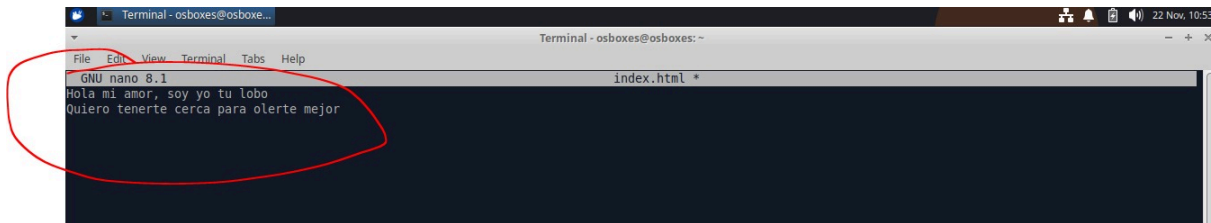
2) Utilizamos ***docker volume ls*** para mostrar el volumen en la lista.

```
osboxes@osboxes:~$ docker run -d --name servidor_tipo2 -p 8282:80 -v miweb:/var/www/html php:7.4-apache 3
e1c9dadf272b3910f6a51db5645e81734a36c1bdb09a7d3996aa6b7648b180f3
osboxes@osboxes:~$ docker ps 4
CONTAINER ID   IMAGE     COMMAND                  CREATED    STATUS    PORTS                               NAMES
e1c9dadf272b   php:7.4-apache  "docker-php-entrypoi..."  7 seconds ago    Up 7 seconds    0.0.0.0:8282->80/tcp, :::8282->80/tcp    servidor_tipo2
osboxes@osboxes:~$ nano index.html 5
```

3) Usamos ***docker -d --name servidor_tipo2 -p 8282:80 -v miweb:/var/www/html php:7.4-apache*** creamos el contenedor (al que llamaremos *servidor_tipo2*) en la imagen de *php:7.4-apache*, al que asignamos el puerto **8282** y montamos el volumen *miweb* en su directorio *var/www/html*.

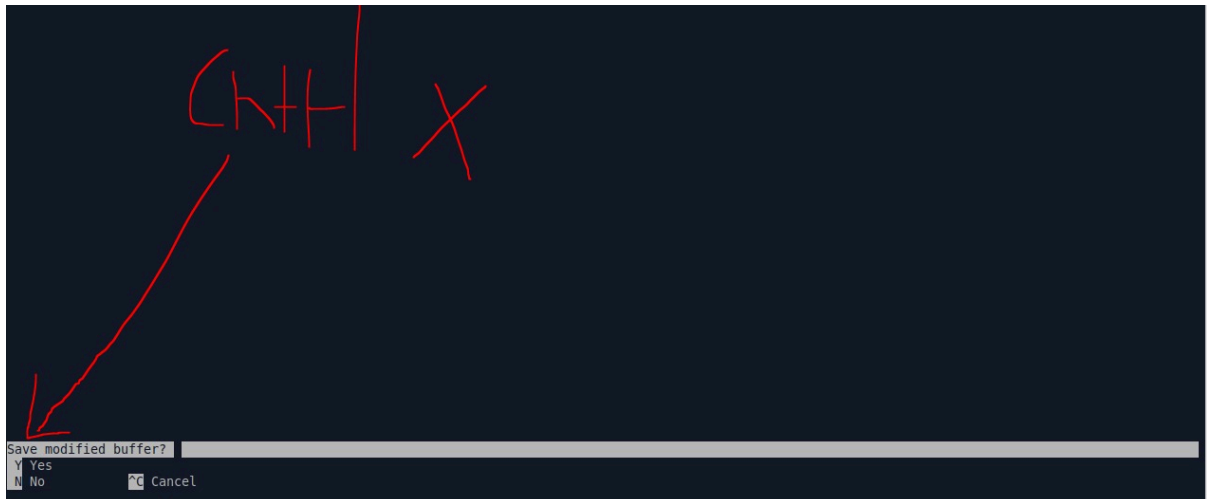
4) Utilizamos ***docker ps*** para mostrar el contenedor.

5) Creamos el archivo ***index.html*** con el comando ***nano index.html***.



```
Terminal - osboxes@osboxe...
GNU nano 8.1
index.html *
Hola mi amor, soy yo tu lobo
Quiero tenerte cerca para olerte mejor
```

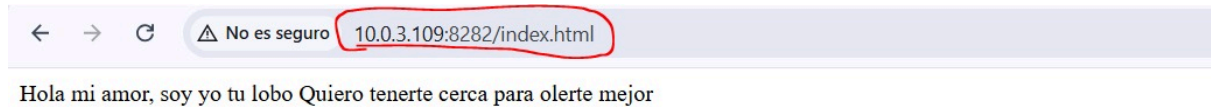
6) Añadimos texto al archivo. En este caso hemos elegido el que hemos elegido.



7) ***Control + x*** para guardar el contenido. Pulsamos ***Y*** para Sí cuando aparezca la pregunta ***“save modifier buffer?”***

Copia del archivo y muestra en el navegador

Lo último que queda por hacer es acceder al archivo desde el navegador web. Después copiaremos el archivo creado al directorio que se nos ha pedido en el ejercicio. Finalmente borraremos el contenedor pero crearemos uno nuevo y comprobaremos que el archivo sigue ahí.



Como en los ejercicios anteriores, introducimos nuestra dirección IP en el navegador web seguido de /y el nombre del archivo. En este caso, **10.0.3.109:8282/index.html**. Al hacerlo, vemos que se muestra el contenido del archivo.

```
elc9dadf272b php:7.4-apache "docker-php-entrypoi..." 7 seconds ago Up 7 seconds 0.0.0.0:8282->80/tcp, :::8282->80/tcp servidor_tipo2
osboxes@osboxes:~$ nano index.html 5
osboxes@osboxes:~$ docker cp /home/osboxes/index.html servidor_tipo2:/var/www/html/index.html
Successfully copied 2.05kB to servidor_tipo2:/var/www/html/index.html
```

- 8) Con el comando **docker cp /home/osboxes/index.html servidor_tipo2:/var/www/html** copiamos el archivo creado al directorio requerido en el contenedor que hemos creado.

```
osboxes@osboxes:~$ docker stop servidor_tipo2
servidor_tipo2
osboxes@osboxes:~$ docker rm servidor_tipo2
servidor_tipo2
```

- 9) Con el comando **docker stop servidor_tipo2** detendremos el contenedor. Esto es necesario si queremos borrarlo.

- 10) Con el comando **docker rm servidor_tipo2** eliminaremos el contenedor.

Si escribimos de nuevo el comando **docker ps** veremos que la lista está vacía, consideréis como un paso **11)** opcional.

```
osboxes@osboxes:~$ docker run -d --name another_server -p 8282:80 -v miweb:/var/www/html php:7.4-apache
8adcc64bd86abe0008a81525d9f9b4a9005d5d285d4cdfadb70bf09e41360e8
osboxes@osboxes:~$ docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS                               NAMES
8adcc64bd86   php:7.4-apache   "docker-php-entrypoi..."   5 seconds ago   Up 4 seconds   0.0.0.0:8282->80/tcp, :::8282->80/tcp   another_server
osboxes@osboxes:~$ docker exec -it servidor bash
Error response from daemon: No such container: servidor
osboxes@osboxes:~$ docker exec -it another_server bash
root@8adcc64bd86:/var/www/html# ls
index.html
root@8adcc64bd86:/var/www/html#
```

12) Con el comando **`docker -d --name another_sercontenedor`** en la imagen **`php:7.4-apache`** exactamente igual que en el paso 3, esta vez con el nombre **`another_server`**, y de nuevo lo asignamos al puerto **`8282`** y montamos el volumen **`miweb`** en su directorio **`var/www/html`**.

13) Usamos **`docker ps`** para mostrar el nuevo contenedor.

14) ***Accedemos al contenedor utilizando `docker exec -it another_server bash`***. Una vez dentro, **`ls`** mostrará si el archivo **`index.html`** se encuentra allí.

15) En efecto, comprobamos que sigue ahí.



Hola mi amor, soy yo tu lobo Quiero tenerte cerca para olerte mejor

Al volver a escribir **`10.0.3.109:8282/index.html`** comprobamos que el contenido del archivo sigue mostrándose. Con esto, el ejercicio ha sido completado.