

EJERCICIO 1 - Trabajo con imágenes

1.1 Servidor web

En primer lugar voy a arrancar un contenedor que va a ejecutar una instancia de la imagen `php:7.4-apache`, se va a llamar `servidor` y va a ser accesible desde un navegador en el puerto 1234.

```
begogutierrez@oClienteLinux:~$ docker run -d --name servidor -p 1234:80 php:7.4-apache
```

```
begogutierrez@oClienteLinux:~$ docker run -d --name servidor -p 1234:80 php:7.4-apache
Unable to find image 'php:7.4-apache' locally
7.4-apache: Pulling from library/php
a603fa5e3b41: Pull complete
c428f1a49423: Pull complete
156740b07ef8: Pull complete
fb5a4c8af82f: Pull complete
25f85b498fd5: Pull complete
9b233e420ac7: Pull complete
fe42347c4ecf: Pull complete
d14eb2ed1e17: Pull complete
66d98f73acb6: Pull complete
d2c43c5efbc8: Pull complete
ab590b48ea47: Pull complete
80692ae2d067: Pull complete
05e465aaa99a: Pull complete
Digest: sha256:c9d7e608f73832673479770d66aacc8100011ec751d1905ff63fae3fe2e0ca6d
Status: Downloaded newer image for php:7.4-apache
f59c05d45a22efda88fce2a70a62dc0dde948b27ece652e514b6af1f343f0b8d
begogutierrez@oClienteLinux:~$
```

En la segunda parte de este ejercicio me pide que coloque en el directorio raíz del contenedor creado, un sitio web. Lo primero que necesito para copiar los archivos de la plantilla que he creado, es saber la id de mi contenedor. Para ello debo escribir los siguientes comandos:

```
begogutierrez@oClienteLinux:~$ docker start servidor
```

Para iniciar el contenedor y,

```
begogutierrez@oClienteLinux:~$ docker container ls
```

para que muestre de los datos de los contenedores, en este caso el que está remarcado corresponde a mi contenedor `servidor`.

```
begogutierrez@oClienteLinux:~$ docker start servidor
servidor
begogutierrez@oClienteLinux:~$ docker container ls
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
f59c05d45a22	php:7.4-apache	"docker-php-entrypoi..."	18 minutes ago	Up 18 minutes	0.0.0.0:1234->80/tcp, :::1234->80/tcp
ab586365994c	mongo:3.0.1	"/entrypoint.sh mong..."	13 days ago	Up 22 minutes	27017/tcp

```
begogutierrez@oClienteLinux:~$
```

A continuación copio el directorio con mi web al contenedor `servidor`

1-1_ServidorWeb

```
begogutierrez@oClienteLinux:~$ docker cp web servidor:/var/www/html
```

```
begogutierrez@oClienteLinux:~$ docker cp web servidor:/var/www/html
```

Y compruebo que dentro del contenedor en la ruta `/var/www/html` efectivamente se encuentra el directorio de la web personalizada.

```
begogutierrez@oClienteLinux:~$ docker exec -it servidor bash
root@f59c05d45a22:/var/www/html# ls
web
root@f59c05d45a22:/var/www/html# cd web
root@f59c05d45a22:/var/www/html/web# ls
'ABOUT THIS TEMPLATE.txt'  css  font-awesome-4.5.0  img  index.html  js  slick
root@f59c05d45a22:/var/www/html/web# ls -la
total 44
drwxrwxr-x 7      1000      1000  4096 Jun  6  2019  .
drwxrwxrwx 1 www-data www-data  4096 Mar 28 19:05  ..
-rw-rw-r-- 1      1000      1000   453 Oct 19  2018 'ABOUT THIS TEMPLATE.txt'
drwxrwxr-x 2      1000      1000  4096 Jun  6  2019  css
drwxrwxr-x 4      1000      1000  4096 Jun  6  2019 font-awesome-4.5.0
drwxrwxr-x 2      1000      1000  4096 Jun  6  2019  img
-rw-rw-r-- 1      1000      1000 11230 Mar 28 18:54 index.html
drwxrwxr-x 2      1000      1000  4096 Jun  6  2019  js
drwxrwxr-x 3      1000      1000  4096 Jun  6  2019  slick
root@f59c05d45a22:/var/www/html/web#
```

Como habia copiado la carpeta completa llamada web, ahora tengo que mover todos los ficheros de dentro de esa carpeta a la raíz del servidor `/var/www/html` porque de la manera actual me da un error porque no lo encuentra al no estar en el raíz.

```
root@cd659c9d2d25:/var/www/html/web# mv * ..
```

De esta manera accede a la pagina perfectamente.



Despliegue de Aplicaciones Web

Tarea de Begoña Gutierrez Ceñal

Menu



Contacta

Begoña Gutierrez Ceñal

CIFP Sectores Industrial y de Servicios

A continuación me pide que en el directorio raíz, cree un archivo llamado `cabeceras.php` que muestre las cabeceras de petición HTTP y que muestre la salida del script en el navegador. En primer lugar tuve que actualizar los repositorios con `apt update` para luego instalar `nano` (podría haberlo hecho sin descargar nada, pero esta opción es menos compleja y mas rápida).

```
root@cd659c9d2d25:/var/www/html# apt update root@cd659c9d2d25:/var/www/html# apt install nano
```

Creo el archivo `cabeceras.php` con `nano`.

```
root@cd659c9d2d25:/var/www/html# nano cabeceras.php
```

Dentro del archivo, escribo el código para sacar las cabeceras de la petición web como pide el ejercicio.

```

GNU nano 3.2 cabeceras.php

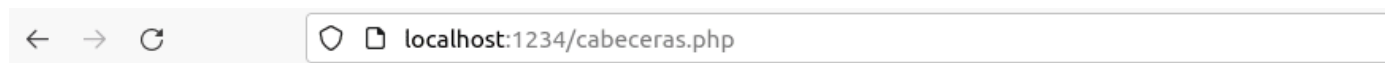
<?php
    print("<h1>Script Cabeceras - tarea de Bego</h1>");

    foreach (getallheaders() as $name => $value)
        echo "$name: $value<br>";
?>

[ Read 7 lines ]
^G Get Help      ^O Write Out     ^W Where Is      ^K Cut Text      ^J Justify       ^C Cur Pos
^X Exit          ^R Read File     ^\ Replace       ^U Uncut Text    ^T To Spell     ^_ Go To Line

```

En el navegador, vemos la salida de esta manera:



Script Cabeceras - tarea de Bego

```

Host: localhost:1234
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/111.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: es-ES,es;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: none
Sec-Fetch-User: ?1

```

El tamaño del contenedor se muestra con

```
begogutierrez@oClienteLinux:~$ docker container ls -s
```

1-1_ServidorWeb

```
begogutierrez@oClienteLinux:~$ docker container ls -s
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORT
S		NAMES	SIZE		
793b4d80e289	php:7.4-apache	"docker-php-entrypoi..."	4 minutes ago	Up 4 minutes	0.0.
0.0:1234->80/tcp, :::1234->80/tcp		servidor	2.55MB (virtual 455MB)		
ab586365994c	mongo:3.0.1	"/entrypoint.sh mong..."	3 weeks ago	Up About an hour	2701
7/tcp		mongo	0B (virtual 256MB)		

```
begogutierrez@oClienteLinux:~$
```

Por último me pide el ejercicio que borre el contenedor. Primero lo paro (compruebo que no se está ejecutando ya) y luego lo borro.

```
begogutierrez@oClienteLinux:~$ docker stop servidor
begogutierrez@oClienteLinux:~$ docker ps
begogutierrez@oClienteLinux:~$ docker rm servidor
```

```
begogutierrez@oClienteLinux:~$ docker stop servidor
servidor
begogutierrez@oClienteLinux:~$ docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
NAMES					
ab586365994c	mongo:3.0.1	"/entrypoint.sh mong..."	3 weeks ago	Up 43 minutes	27017/t
cp	mongo				

```
begogutierrez@oClienteLinux:~$ docker rm servidor
servidor
```

Compruebo efectivamente que ha sido borrado con:

```
begogutierrez@oClienteLinux:~$ docker ps -all
```

```
begogutierrez@oClienteLinux:~$ docker ps -all
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
cbd02ea6995a	sdelements/lets-chat	"npm start"	3 weeks ago	Created		letsc
hat						

```
begogutierrez@oClienteLinux:~$
```