

# UNIDAD TRABAJO 5:

## VIRTUALIZACIÓN.

## CONTENEDORES. DOCKER.

### CONTENEDORES – EJERCICIOS DOCKER

Esta unidad didáctica se ha desarrollado para el Curso de Especialización de Ciberseguridad en entornos de las Tecnologías de la Información. En concreto para el módulo de Puesta en Producción Segura, cuyos aspectos básicos del currículo vienen recogidos en el Real Decreto 479/2020.

**Licencia:** *Creative Commons Atribución - No Comercial -Compartir Igual (CC BY-NC-SA 4.0)*

**José Gaspar Sánchez García.**

**Pedro Antonio Santiago Santiago.**

## Normas de entrega:

### Corrección:

-  Por cada día de retraso sin justificación en la entrega, se restará 1 punto a la nota.

### Forma de Entrega:

- A través de la plataforma Moodle GVA AULAS: <https://aules.edu.gva.es/> **Documentos a entregar:**
  - Este mismo fichero con las preguntas rellenas (usar fuente Verdana de tamaño 12 color azul en cursiva – **Estilo Respuesta**): *[Respuesta]*.
  - Añade la dirección URL del repositorio GitHub creado al documento que contiene las respuestas.
  - Documento con la respuesta a las cuestiones y explicación al desarrollo de la práctica, en formato del documento **PDF**.
  - Cambia el nombre al documento, y añade tu PRIMER APELLIDO y tu NOMBRE.

## Índice:

---

1. Ejercicios.....	4
1.1. Crear imágenes a partir de contenedores.....	4
2. Enlaces	
vídeos.....	5

# 1. Ejercicios

1. Explorar la web de [dockerhub.com](https://hub.docker.com) y crear **5 contenedores** en segundo plano con imagenes diferentes según tus intereses.

*[Instalación desde la página Docker Hub creación de 5 contenedores]*

***docker run NombreAplicacion :***

*Los contenedores seleccionados son nginx,alpine,Python,homeassistant y node.*

*]*

2. **Listar** los contenedores en ejecución.

*[Listamos contenedores en ejecución con **docker ps** y todos los contenedores con **docker ps -a** :*

```
sh-3.2# docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
1087630a36de	nginx	"/docker-entrypoint..."	2 days ago	Up 2 days	80/tcp	cool_aryabhata

```
josefranciscmurciafuentes@MacBook-Pro-de-Jose ~ % docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
3a7c7097292f	homeassistant/amd64-addon-mosquitto	"/init"	4 minutes ago	Exited (1) 4 minutes ago		friendly_hoover
b713aeed476d	node	"docker-entrypoint.s..."	11 hours ago	Exited (0) 11 hours ago		sleepy_austin
4d00d042c8f	alpine:latest	"/bin/sh"	11 hours ago	Exited (0) 11 hours ago		gracious_kare
85633085ced2	my-alpine-python	"pytest"	11 hours ago	Created		quirky_dhawan
254b4565f694	alpine	"/bin/sh"	11 hours ago	Exited (0) 11 hours ago		angry_vaughan
88d01c921078	ubuntu	"/bin/bash"	11 hours ago	Exited (0) 11 hours ago		beautiful_taussig
1087630a36de	nginx	"/docker-entrypoint..."	2 days ago	Exited (0) 19 minutes ago		cool_aryabhata
cf258c3641ca	python	"python3"	2 days ago	Exited (0) 2 days ago		competent_jang
bfe6ed25715c	alpine	"/bin/sh"	2 days ago	Exited (0) 2 days ago		dreamy_lalande
cdc683854b62	my-alpine-python	"pytest"	8 days ago	Created		goofy_shirley
ec2169352329	alpine-python:v1.0	"pytest"	8 days ago	Exited (5) 8 days ago		dazzling_haibt
a636aba0c9dc	alpine	"/bin/sh"	8 days ago	Exited (130) 8 days ago		jfmf-alpine-python

*]*

3. **Borrar** todos los contenedores (ejecutándose y terminados).

*[Ejecutamos comando **docker stop**, parar contenedores en ejecución borrar todos los contenedores captura:*

```
80ac89d18005 hello-world 7hello
josefranciscmurciafuentes@MacBook-Pro-de-Jose ~ % docker rm 3a7c7097292f
3a7c7097292f
josefranciscmurciafuentes@MacBook-Pro-de-Jose ~ % doker rm b713aeed476d
zsh: command not found: doker
josefranciscmurciafuentes@MacBook-Pro-de-Jose ~ % docker rm b713aeed476d
b713aeed476d
josefranciscmurciafuentes@MacBook-Pro-de-Jose ~ % docker rm 254b4565f694
254b4565f694
josefranciscmurciafuentes@MacBook-Pro-de-Jose ~ % docker rm 1087630a36de
1087630a36de
josefranciscmurciafuentes@MacBook-Pro-de-Jose ~ % docker rm 85633085ced2
85633085ced2
```

*]*

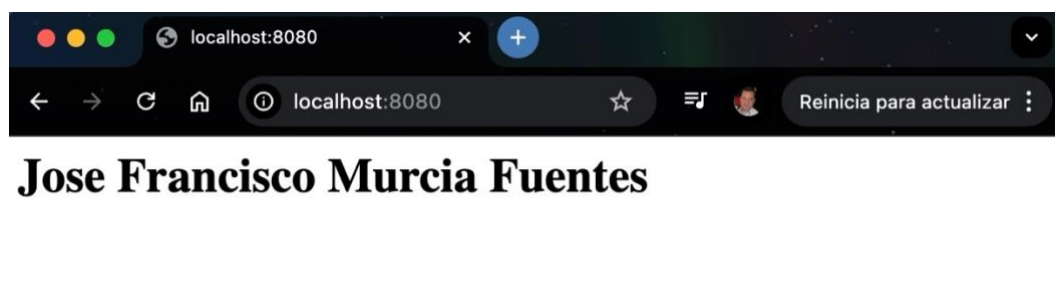
4. Crear un contenedor con la imagen **nginx** que esté ofreciendo una página por defecto personalizada con tu nombre y apellido

*[Creamos el contenedor y copiamos el archivo en el directorio del contenedor para servir la página en los puertos 80:8080 con nuestro nombre llamaremos al contenedor miweb:]*

***docker run -d -p 8080:80 --name miweb nginx***

```
josefranciscomurciafuentes@MacBook-Pro-de-Jose ~ % echo "<h1>Jose Francisco Murcia Fuentes</h1>" > index.html
josefranciscomurciafuentes@MacBook-Pro-de-Jose ~ % docker run -d -p 8080:80 --name miweb nginx
784c167668dcf84bc7167cc2e960215a080e2dd46eb1f32b278f27e493a0e9e9
josefranciscomurciafuentes@MacBook-Pro-de-Jose ~ % docker cp index.html 784c167668dcf84bc7167cc2e960215a080e2dd46eb1f32b278f27e493a0e9e9:/usr/share/nginx/html
Successfully copied 2.05kB to 784c167668dcf84bc7167cc2e960215a080e2dd46eb1f32b278f27e493a0e9e9:/usr/share/nginx/html
```

*Mostramos el servicio web:*



5. Muestra mensajes de **log** del contenedor de **nginx**.

*[ docker logs miweb*

```
josefranciscomurciafuentes@MacBook-Pro-de-Jose ~ % docker logs miweb
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
/docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
/docker-entrypoint.sh: Configuration complete; ready for start up
2024/12/20 20:00:54 [notice] 1#1: using the "epoll" event method
2024/12/20 20:00:54 [notice] 1#1: nginx/1.27.3
2024/12/20 20:00:54 [notice] 1#1: built by gcc 12.2.0 (Debian 12.2.0-14)
2024/12/20 20:00:54 [notice] 1#1: OS: Linux 6.10.14-linuxkit
2024/12/20 20:00:54 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2024/12/20 20:00:54 [notice] 1#1: start worker processes
2024/12/20 20:00:54 [notice] 1#1: start worker process 28
2024/12/20 20:00:54 [notice] 1#1: start worker process 29
2024/12/20 20:00:54 [notice] 1#1: start worker process 30
2024/12/20 20:00:54 [notice] 1#1: start worker process 31
2024/12/20 20:00:54 [notice] 1#1: start worker process 32
2024/12/20 20:00:54 [notice] 1#1: start worker process 33
2024/12/20 20:00:54 [notice] 1#1: start worker process 34
2024/12/20 20:00:54 [notice] 1#1: start worker process 35
172.17.0.1 - - [20/Dec/2024:20:10:35 +0000] "GET / HTTP/1.1" 200 39 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/131.0.0.0 Safari/537.36" "-"
2024/12/20 20:10:35 [error] 32#32: *3 open() "/usr/share/nginx/html/favicon.ico" failed (2: No such file or directory), client: 172.17.0.1, server: localhost, request: "GET /favicon.ico HTTP/1.1", host: "localhost:8080", referrer: "http://localhost:8080/"
172.17.0.1 - - [20/Dec/2024:20:10:35 +0000] "GET /favicon.ico HTTP/1.1" 404 555 "http://localhost:8080/" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/131.0.0.0 Safari/537.36" "-"
172.17.0.1 - - [20/Dec/2024:20:13:02 +0000] "GET / HTTP/1.1" 200 40 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/131.0.0.0 Safari/537.36" "-"
```

***docker logs -n1 miweb***

```
josefranciscomurciafuentes@MacBook-Pro-de-Jose ~ % docker logs -n1 miweb
172.17.0.1 - - [20/Dec/2024:20:13:02 +0000] "GET / HTTP/1.1" 200 40 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/131.0.0.0 Safari/537.36" "-"
```

*]*



6. Crea un contenedor con imagen de Ubuntu. Instala tus paquetes preferidos y crea imagen llamada **MiUbuntu** basada en dicho contenedor.

### *[-Creación contenedor con terminal interactivo instalamos actualizaciones*

```

root@fbf12e8f7c2f:/# apt install net-tools
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  net-tools
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 207 kB of archives.
After this operation, 1229 kB of additional disk space will be used.
Get:1 http://ports.ubuntu.com/ubuntu-ports noble/main arm64 net-tools arm64 2.10-0.1ubuntu4
Fetched 207 kB in 0s (622 kB/s)
debconf: delaying package configuration, since apt-utils is not installed
Selecting previously unselected package net-tools.
(Reading database ... 4374 files and directories currently installed.)
Preparing to unpack .../net-tools_2.10-0.1ubuntu4_arm64.deb ...
Unpacking net-tools (2.10-0.1ubuntu4) ...
Setting up net-tools (2.10-0.1ubuntu4) ...
root@fbf12e8f7c2f:/# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 65535
    inet 172.17.0.3 netmask 255.255.0.0 broadcast 172.17.255.255
    ether 02:42:ac:11:00:03 txqueuelen 0 (Ethernet)
    RX packets 1462 bytes 27563387 (27.5 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 818 bytes 59713 (59.7 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

```

### *-Instalación del paquete net-tools y creación de la imagen miubuntu:*

```

~ — root@fbf12e8f7c2f: / — docker run -it --name ubuntu ubuntu
josefranciscoturciafuentes@MacBook-Pro-de-Jose ~ % docker run -it --name ubuntu ubuntu
root@fbf12e8f7c2f:/# apt update
Get:1 http://ports.ubuntu.com/ubuntu-ports noble InRelease [256 kB]
Get:2 http://ports.ubuntu.com/ubuntu-ports noble-updates InRelease [126 kB]
Get:3 http://ports.ubuntu.com/ubuntu-ports noble-backports InRelease [126 kB]
Get:4 http://ports.ubuntu.com/ubuntu-ports noble-security InRelease [126 kB]
Get:5 http://ports.ubuntu.com/ubuntu-ports noble/multiverse arm64 Packages [274 kB]
Get:6 http://ports.ubuntu.com/ubuntu-ports noble/restricted arm64 Packages [113 kB]
Get:7 http://ports.ubuntu.com/ubuntu-ports noble/universe arm64 Packages [19.0 MB]
Get:8 http://ports.ubuntu.com/ubuntu-ports noble/main arm64 Packages [1776 kB]
Get:9 http://ports.ubuntu.com/ubuntu-ports noble-updates/main arm64 Packages [948 kB]
Get:10 http://ports.ubuntu.com/ubuntu-ports noble-updates/multiverse arm64 Packages [15.1 kB]
Get:11 http://ports.ubuntu.com/ubuntu-ports noble-updates/universe arm64 Packages [1222 kB]
Get:12 http://ports.ubuntu.com/ubuntu-ports noble-updates/restricted arm64 Packages [771 kB]
Get:13 http://ports.ubuntu.com/ubuntu-ports noble-backports/universe arm64 Packages [11.9 kB]
Get:14 http://ports.ubuntu.com/ubuntu-ports noble-security/main arm64 Packages [703 kB]
Get:15 http://ports.ubuntu.com/ubuntu-ports noble-security/multiverse arm64 Packages [13.3 kB]
Get:16 http://ports.ubuntu.com/ubuntu-ports noble-security/restricted arm64 Packages [750 kB]
Get:17 http://ports.ubuntu.com/ubuntu-ports noble-security/universe arm64 Packages [997 kB]
Fetched 27.3 MB in 2s (13.4 MB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
All packages are up to date.

```

```

josefranciscoturciafuentes@MacBook-Pro-de-Jose ~ % docker commit ubuntu miubuntu:latest
sha256:b4a46d8167d6c157ed385e9069da6fc4f47d86e1ae02c628aad5bba24d92808c
josefranciscoturciafuentes@MacBook-Pro-de-Jose ~ % docker run -it miubuntu
root@29d192b21ca6:/# docker images
bash: docker: command not found
root@29d192b21ca6:/# exit
exit
josefranciscoturciafuentes@MacBook-Pro-de-Jose ~ % docker images

```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
miubuntu	latest	b4a46d8167d6	About a minute ago	216MB

7. Crea un **Dockerfile** para personalizar una imagen Alpine. Consulta las capas de la nueva imagen.

### [Creación de Dockerfile

```
josefranciscumurciafuentes@MacBook-Pro-de-Jose Ejer2 % CAT Dockerfile
FROM alpine
ARG WEB
ENV WEBC=$WEB
CMD ping -c1 $WEBC
josefranciscumurciafuentes@MacBook-Pro-de-Jose Ejer2 %
```

### y construcción de la imagen:

```
josefranciscumurciafuentes@MacBook-Pro-de-Jose ~ % cd Docker
josefranciscumurciafuentes@MacBook-Pro-de-Jose Docker % mkdir Ejer2
josefranciscumurciafuentes@MacBook-Pro-de-Jose Docker % cd Ejer2
josefranciscumurciafuentes@MacBook-Pro-de-Jose Ejer2 % nano Dockerfile
josefranciscumurciafuentes@MacBook-Pro-de-Jose Ejer2 % ls
Dockerfile      Texto de Terminal.txt
josefranciscumurciafuentes@MacBook-Pro-de-Jose Ejer2 % docker build -t mialpine:latest .
[+] Building 8.8s (7/7) FINISHED                                docker:desktop-linux
=> [internal] load build definition from Dockerfile              0.0s
=> => transferring dockerfile: 72B                               0.0s
=> [internal] load metadata for docker.io/library/alpine:latest  0.0s
=> [internal] load .dockerignore                                0.0s
=> => transferring context: 2B                                     0.0s
=> CACHED [1/2] FROM docker.io/library/alpine:latest@sha256:21dc6063fd678b478f57c0e13f47560d0ea4eeba26dfc947b2a4 8.4s
=> => resolve docker.io/library/alpine:latest@sha256:21dc6063fd678b478f57c0e13f47560d0ea4eeba26dfc947b2a4f81f686 8.4s
=> [auth] library/alpine:pull token for registry-1.docker.io    0.0s
=> [2/2] RUN echo "Hola Mundo"                                   0.2s
=> exporting to image                                           0.1s
=> => exporting layers                                           0.1s
=> => exporting manifest sha256:5e1fe95442e051a4b7f8e454e103ff4efac06630923115080ca1db1e7ed1799d 0.0s
=> => exporting config sha256:e2264eddee818f75eae20aa8272ec751786780a4b44772da516599e5c22df67 0.0s
=> => exporting attestation manifest sha256:4a6fc438eb8e3fa47e89bf5a49ba14d44eec453de720468a7f4bef0dead8cb5e 0.0s
=> => exporting manifest list sha256:d261f41acd191260e82483175241ff32f0d4feeb0d84a2bf35c9f291db214644 0.0s
=> => naming to docker.io/library/mialpine:latest               0.0s
=> => unpacking to docker.io/library/mialpine:latest             0.0s

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/iw3w3trvucwx51crdbaoerv63
```

### Historial de imágenes:

```
josefranciscumurciafuentes@MacBook-Pro-de-Jose Ejer2 % docker history mialpine:latest
IMAGE          CREATED        CREATED BY          SIZE      COMMENT
d261f41acd19   2 minutes ago RUN /bin/sh -c echo "Hola Mundo" # buildkit 4.1kB     buildkit.dockerfile.v0
<missing>      2 weeks ago   CMD ["/bin/sh"]     0B        buildkit.dockerfile.v0
<missing>      2 weeks ago   ADD alpine-minirootfs-3.21.0-aarch64.tar.gz ... 8.84MB    buildkit.dockerfile.v0
josefranciscumurciafuentes@MacBook-Pro-de-Jose Ejer2 % docker history alpine
IMAGE          CREATED        CREATED BY          SIZE      COMMENT
21dc6063fd67   2 weeks ago   CMD ["/bin/sh"]     0B        buildkit.dockerfile.v0
<missing>      2 weeks ago   ADD alpine-minirootfs-3.21.0-aarch64.tar.gz ... 8.84MB    buildkit.dockerfile.v0
```

]

8. Utiliza **Dockerfile** y las variables necesarias para que al crear una imagen, los contenedores basados en esa imagen, ejecuten directamente un sólo **ping** (*ping -c 1*) a la web que le introduzca por parámetro al crear la imagen. Ejemplo: *docker run ping:google* se ejecute un sólo ping a google.com.



## [ Captura del resultado ejecutando contenedor:

```
josefranciscomurciafuentes@MacBook-Pro-de-Jose Ejer2 % docker run ping:google
PING www.google.com (216.58.209.68): 56 data bytes
64 bytes from 216.58.209.68: seq=0 ttl=63 time=15.941 ms

--- www.google.com ping statistics ---
1 packets transmitted, 1 packets received, 0% packet loss
round-trip min/avg/max = 15.941/15.941/15.941 ms
josefranciscomurciafuentes@MacBook-Pro-de-Jose Ejer2 %
```

]

- Posteriormente, cree otra imagen ejecutando **docker run ping:youtube** se ejecute un *ping* a *youtube.com* (el nombre de la etiqueta no tiene porqué corresponder con el dominio al que se hace *ping*, deste dominio se pasará por una variable cuando se crea la imagen).

## [Captura de creación de la imagen y ejecución del contenedor:

```
josefranciscomurciafuentes@MacBook-Pro-de-Jose Ejer2 % docker run ping:youtube
PING www.youtube.com (172.217.168.174): 56 data bytes
64 bytes from 172.217.168.174: seq=0 ttl=63 time=38.177 ms

--- www.youtube.com ping statistics ---
1 packets transmitted, 1 packets received, 0% packet loss
round-trip min/avg/max = 38.177/38.177/38.177 ms
josefranciscomurciafuentes@MacBook-Pro-de-Jose Ejer2 %
```

```
1 packets transmitted, 1 packets received, 0% packet loss
round-trip min/avg/max = 15.941/15.941/15.941 ms
josefranciscomurciafuentes@MacBook-Pro-de-Jose Ejer2 % docker build -t ping:youtube --build-arg WEB=www.youtube.com .
[+] Building 1.3s (6/6) FINISHED
=> [internal] load build definition from Dockerfile                                0.0s
=> => transferring dockerfile: 90B                                                0.0s
=> WARN: JSONArgsRecommended: JSON arguments recommended for CMD to prevent unintended behavior related to OS si 0.0s
=> [internal] load metadata for docker.io/library/alpine:latest                  0.0s
=> [internal] load .dockerignore                                                  0.0s
=> => transferring context: 2B                                                    0.0s
=> CACHED [1/1] FROM docker.io/library/alpine:latest@sha256:21dc6063fd678b478f57c0e13f47560d0ea4eeba26dfc947b2a4 1.1s
=> => resolve docker.io/library/alpine:latest@sha256:21dc6063fd678b478f57c0e13f47560d0ea4eeba26dfc947b2a4f81f686 1.1s
=> [auth] library/alpine:pull token for registry-1.docker.io                    0.0s
=> exporting to image                                                            0.1s
=> => exporting layers                                                            0.0s
=> => exporting manifest sha256:38b46f24676dfdbe5113aafba6bb30370f7850bd8a7874f1b213f51a9e68cc68          0.0s
=> => exporting config sha256:1f9658129363e2e6ff71c0b88130b2fad0420814f53e81870b6beb02b6c60441            0.0s
=> => exporting attestation manifest sha256:36ceff5540aa8506160185b629829f10118fb63ffddb2c83f469c2c5e665bd7e 0.0s
=> => exporting manifest list sha256:a254755dd847060f183f4976c754f5e6827f21279b08c6d12548aec80a49b5a2        0.0s
=> => naming to docker.io/library/ping:youtube                                  0.0s
=> => unpacking to docker.io/library/ping:youtube                                0.0s

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/3e5u46gdbntb1az2juqp3e7eb

1 warning found (use docker --debug to expand):
- JSONArgsRecommended: JSON arguments recommended for CMD to prevent unintended behavior related to OS signals (line 4)
```

]

## 1.1. Crear imágenes a partir de contenedores

10. Ejecuta un contenedor de nombre **midebian-<iniciales>** con la imagen de **debian:latest**.

[

```
josefranciscomurciafuentes@MacBook-Pro-de-Jose Ejer2 % docker run --name midebian-jfmf -it debian
Unable to find image 'debian:latest' locally
latest: Pulling from library/debian
Digest: sha256:17122fe3d66916e55c0cbd5bbf54bb3f87b3582f4d86a755a0fd3498d360f91b
Status: Downloaded newer image for debian:latest
```

].

11. Instala **apache2**.

[*instalación de apt install apache2:*

```
root@3a4d729c59bd:/# apt install apache2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils ca-certificates krb5-locales libapr1 libaprutil1 libaprutil1-dbd-sqlite3
  libaprutil1-ldap libbrotli1 libcurl4 libexpat1 libgdbm-compat4 libgdbm6 libgpm2 libgssapi-krb5-2 libicu72
  libjansson4 libk5crypto3 libkeyutils1 libkrb5-3 libkrb5support0 libldap-2.5-0 libldap-common liblua5.3-0
  libncursesw6 libnghttp2-14 libperl5.36 libproc2-0 libpsl5 librtmp1 libsasl2-2 libsasl2-modules libsasl2-modules-db
  libsqlite3-0 libssh2-1 libssl3 libxml2 media-types netbase openssl perl perl-modules-5.36 procps psmisc publicsuffix
  ssl-cert
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom www-browser gdbm-l10n gpm krb5-doc krb5-user
  sensible-utils libsasl2-modules-gssapi-mit | libsasl2-modules-gssapi-heimdal libsasl2-modules-ldap
  libsasl2-modules-otp libsasl2-modules-sql perl-doc libterm-readline-gnu-perl | libterm-readline-perl-perl make
  libtap-harness-archive-perl
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils ca-certificates krb5-locales libapr1 libaprutil1
  libaprutil1-dbd-sqlite3 libaprutil1-ldap libbrotli1 libcurl4 libexpat1 libgdbm-compat4 libgdbm6 libgpm2
  libgssapi-krb5-2 libicu72 libjansson4 libk5crypto3 libkeyutils1 libkrb5-3 libkrb5support0 libldap-2.5-0
  libldap-common liblua5.3-0 libncursesw6 libnghttp2-14 libperl5.36 libproc2-0 libpsl5 librtmp1 libsasl2-2
  libsasl2-modules libsasl2-modules-db libsqlite3-0 libssh2-1 libssl3 libxml2 media-types netbase openssl perl
  perl-modules-5.36 procps psmisc publicsuffix ssl-cert
0 upgraded, 48 newly installed, 0 to remove and 0 not upgraded.
Need to get 26.8 MB of archives.
After this operation, 127 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://deb.debian.org/debian bookworm/main arm64 perl-modules-5.36 all 5.36.0-7+deb12u1 [2815 kB]
Get:2 http://deb.debian.org/debian bookworm/main arm64 libgdbm6 arm64 1.23-3 [70.9 kB]
```

].

11. Inicialo con el comando **/etc/init.d/apache2 start**.

[*Iniciamos el servicio :*

```
Running hooks in /etc/ca-certificates/update.d...
done.
root@3a4d729c59bd:/# /etc/init.d/apache2 start
Starting Apache httpd web server: apache2AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 172.17.0.3. Set the 'ServerName' directive globally to suppress this message
.
root@3a4d729c59bd:/#
```

].



13. Copia una web estática que muestre tu nombre y apellidos como página por defecto para apache.

*[Copiamos el archivo en el directorio de apache mostrar web]*

```
[root@3a4d729c59bd:/var/www/html# hostname -I
172.17.0.3
[root@3a4d729c59bd:/var/www/html# chmod -R 755 /var/www/index.html
chmod: cannot access '/var/www/index.html': No such file or directory
[root@3a4d729c59bd:/var/www/html# chmod -R 755 index.html
[root@3a4d729c59bd:/var/www/html# ls -l
total 4
-rwxr-xr-x 1 501 dialout 40 Dec 20 20:12 index.html
```

].

14. Comprueba que el servidor web funciona desde el anfitrión accediendo a la IP del contenedor (recuerda que no funcionará con HTTPS).
15. Realiza todos estos pasos de manera que se pueda dejar el terminal libre para seguir ejecutando comandos desde el anfitrión.

*[Para que funcione desde equipo anfitrión habilitamos redirección del puerto 80 al puerto 8080 cuando creamos el contenedor accedemos a localhost:8080 la ip **Hostname -I** es 172.17.0.3 de nuestro contenedor, la administración del contenedor se ejecuta desde el terminal del equipo anfitrión :*

```
josefranciscumurciafuentes@MacBook-Pro-de-Jose ~ % docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                    NAMES
91ac7dae2482   ubuntu:latest  "/bin/bash"             2 hours ago   Up 2 hours   0.0.0.0:8080->80/tcp     debianmiweb-jfmf
josefranciscumurciafuentes@MacBook-Pro-de-Jose ~ % docker cp index.html debianmiweb-jfmf:/var/www/html
Successfully copied 2.05kB to debianmiweb-jfmf:/var/www/html
josefranciscumurciafuentes@MacBook-Pro-de-Jose ~ %

josefranciscumurciafuentes@MacBook-Pro-de-Jose ~ % docker run --name debianmiweb-jfmf -it -p 8080:80 ubuntu:latest
root@91ac7dae2482:/# apt update
```



Vista de la web `index.html`:



].

16. Crea una imagen llamada **debianmiweb-<iniciales>** a partir del contenedor **midebian<iniciales>**.

*[Creamos una imagen a partir del contenedor creado lo he llamado debian-apachemiweb-jfmf]*

```
josefranciscumurciafuentes@MacBook-Pro-de-Jose ~ % docker commit debianmiweb-jfmf debian-apachemiweb-jfmf:latest
sha256:b8aa9d7c1af21607f276c949f13f5387471239c23092aa360128f557b910829c
```

].

17. Elimina el contenedor **midebian-<iniciales>**.

*[Paramos y borramos contenedor:]*

```
josefranciscumurciafuentes@MacBook-Pro-de-Jose ~ % docker commit debianmiweb-jfmf debian-apachemiweb-jfmf:latest
sha256:b8aa9d7c1af21607f276c949f13f5387471239c23092aa360128f557b910829c
josefranciscumurciafuentes@MacBook-Pro-de-Jose ~ % docker stop debianmiweb-jfmf
debianmiweb-jfmf
josefranciscumurciafuentes@MacBook-Pro-de-Jose ~ % docker rm debianmiweb-jfmf
debianmiweb-jfmf
```

].

18. Lanza otro contenedor para que su web siga estando disponible.

*[Lanzamos otro contenedor a partir de la imagen eliminada]*

```
josefranciscumurciafuentes@MacBook-Pro-de-Jose ~ % docker run -it --name debianmiweb-jfmf -p 80:8080 debian-apachemiweb-jfmf
root@fda3ad71b3a0:/# hostname -I
172.17.0.2
```

].

19. ¿Funciona el servidor web en el nuevo contenedor? *[No]*. ¿Por qué? *[Se deden ejecutar arrancar los servicios de ejecución en memoria que no se guardan cuando instalamos el contenedor solo se guarda la configuración]*.

20. Realiza lo necesario para que su web siga disponible.

*[Iniciamos el servidor apache : **/etc/init.d/apache2 start***

```
[root@fda3ad71b3a0:/# nmap 172.17.0.2
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-12-29 11:54 UTC
Nmap scan report for fda3ad71b3a0 (172.17.0.2)
Host is up (0.0000020s latency).
Not shown: 998 closed tcp ports (reset)
PORT      STATE SERVICE
80/tcp    open  http
8080/tcp  open  http-proxy
```

Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds

```
[root@fda3ad71b3a0:/# apache2 -v
Server version: Apache/2.4.58 (Ubuntu)
Server built:   2024-10-02T12:40:51
[root@fda3ad71b3a0:/# /etc/init.d/apache2 start
* Starting Apache httpd web server apache2
*
```

*].*

## 2. Enlaces vídeos

- [📺 Ejercicios Docker: Ejecutar instancias, revisarlas, pararlas | 🧠 Prácticas guiadas y fáciles.](#)
- [📺 Ejercicios Docker: Crear imágenes basadas en un contenedor y uso de Dockerfile .](#)
- [Ejercicios Docker: Crear imágenes a partir de contenedores | 🧠 Entiende este concepto importante](#)



## Resultados de Aprendizaje y Criterios de Evaluación

Contenidos:	
<b>5. Implantación de sistemas seguros de despliegado de software:</b> <ul style="list-style-type: none"> <li>• Puesta segura en producción.</li> <li>• Prácticas unificadas para el desarrollo y operación del software (DevOps).</li> <li>• Sistemas de control de versiones.</li> <li>• Sistemas de automatización de construcción (build).</li> <li>• Escalado de servidores. Virtualización. Contenedores.</li> <li>• Gestión automatizada de configuración de sistemas.</li> <li>• Orquestación de contenedores.</li> </ul>	
Resultados de Aprendizaje	Criterios de Evaluación
<b>RA5. Implanta sistemas seguros de despliegado de software, utilizando herramientas para la automatización de la construcción de sus elementos.</b>	<p>a) Se han identificado las características, principios y objetivos de la integración del desarrollo y operación del software.</p> <p>b) Se han implantado sistemas de control de versiones, administrando los roles y permisos solicitados.</p> <p>c) Se han instalado, configurado y verificado sistemas de integración continua, conectándolos con sistemas de control de versiones.</p> <p>d) Se han planificado, implementado y automatizado planes de despliegado de software.</p> <p>e) Se ha evaluado la capacidad del sistema desplegado parareaccionar de forma automática a fallos.</p> <p>f) Se han documentado las tareas realizadas y los procedimientos a seguir para la recuperación ante desastres.</p> <p>g) Se han creado bucles de retroalimentación ágiles entre los miembros del equipo.</p>