

Práctica 10. Docker

Lee detenidamente cada uno de los puntos antes de realizar las tareas solicitadas. Revisa los recursos incluidos.

Al adjuntar ficheros, introdúcelos en carpetas con el nombre de la agrupación a la que corresponde. Revisa el apartado “A entregar” al final del doc.

Parte 1: Docker (Dockerfile)

Actividades 1-5 se realizan en conjunto: Crear imagen docker para web y arrancar.

1. Crea un directorio **src/** en tu equipo y dentro coloca un fichero **index.php** de prueba con el siguiente código (y muestra capturas):
<?php phpinfo(); ?>

```
ajbeltran94@BloodWorm in repo: School/DespliegueWEB/Unit5/01Actividad/Actividad 10 on P main [x?] as  took 0s
λ sudo mkdir src
PS C:\Users\USUARIO> src>

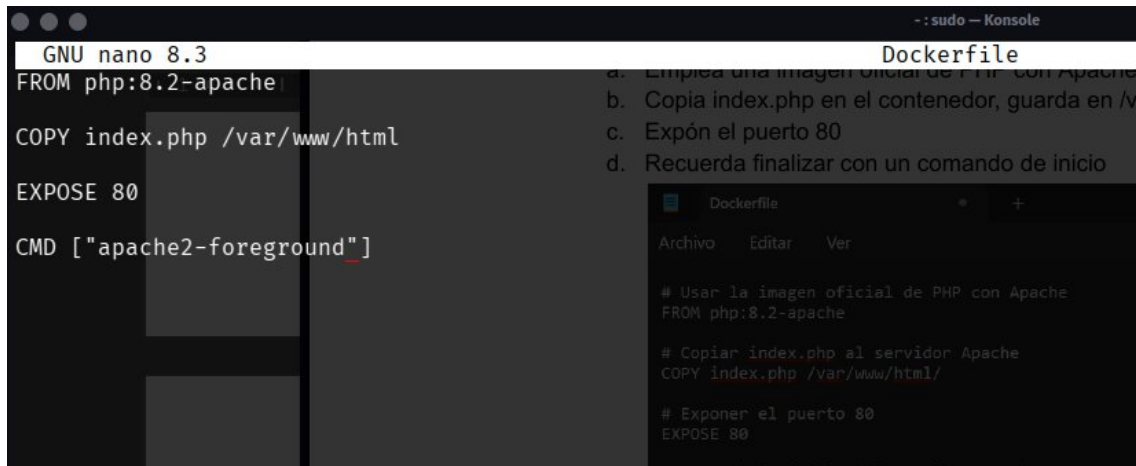
ajbeltran94@BloodWorm in repo: School/DespliegueWEB/Unit5/01Actividad/Actividad 10 on P main [x?] as  took 0s
λ tree
.
├── Práctica10_Docker_24_25.docx
└── src
    2 directories, 1 file
PS C:\Users\USUARIO> echo "<?php phpinfo();"
?>" > src/index.php
PS C:\Users\USUARIO> Get-Content index.php

GNU nano 8.3 index.php
<?php phpinfo(); ?>

ajbeltran94@BloodWorm in repo: School/DespliegueWEB/Unit5/01Actividad/Actividad 10/src on P main [x?] as  took 0s
λ sudo nano index.php
PS C:\Users\USUARIO> cd src
PS G:\Users\USUARIO> cd src
PS C:\Users\USUARIO> echo "<?php phpinfo();"
?>" > src/index.php
PS C:\Users\USUARIO> Get-Content index.php
<?php phpinfo(); ?>

ajbeltran94@BloodWorm in repo: School/DespliegueWEB/Unit5/01Actividad/Actividad 10 on P main [x?] as  took 0s
λ tree
.
├── Práctica10_Docker_24_25.docx
└── src
    ├── index.php
    2 directories, 2 files
```

2. Prepara un contenedor con **Apache y PHP** y el programa .php anterior usando **Dockerfile**. Crea el dockerfile en /src (adjunta o copia el contenido de este fichero).
 - a. Emplea una imagen oficial de PHP con Apache (ej. php:8.2-apache)
 - b. Copia index.php en el contenedor, guarda en /var/www/html
 - c. Expón el puerto 80
 - d. Recuerda finalizar con un comando de inicio



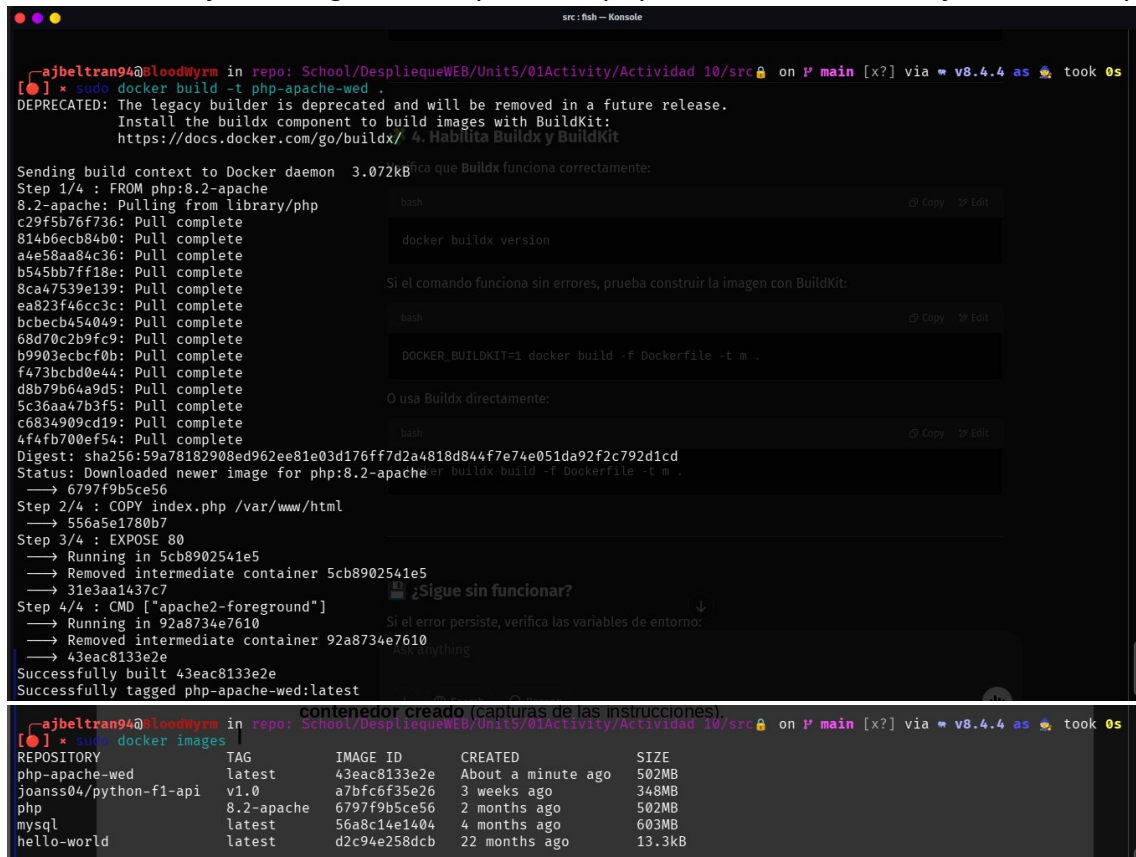
```
GNU nano 8.3 Dockerfile
FROM php:8.2-apache

COPY index.php /var/www/html

EXPOSE 80

CMD ["apache2-foreground"]
```

3. Construye la **imagen** de la aplicación (capturas del docker build y su resultado)

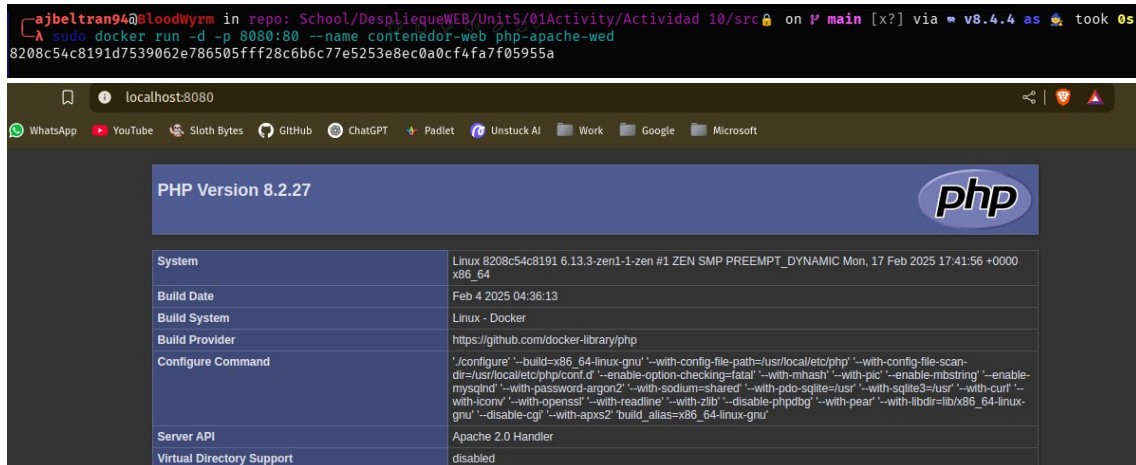


```
ajbeltran94@bloodwyrn in repo: School/DespliegueWEB/Unit5/01Activity/Actividad 10/src on P main [x?] via v8.4.4 as took 0s
[●] * sudo docker build -t php-apache-wed .
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
Install the buildx component to build images with BuildKit:
https://docs.docker.com/go/buildx/ 4. Habilita Buildx y BuildKit

Sending build context to Docker daemon 3.072kB
Step 1/4 : FROM php:8.2-apache
8.2-apache: Pulling from library/php
c29f5b76f736: Pull complete
814b6ecb84b0: Pull complete
a4e58aa84c36: Pull complete
b545bb7ff18e: Pull complete
8ca47539e139: Pull complete
ea823f46cc3c: Pull complete
bcbecb454049: Pull complete
68d70c2b9fc9: Pull complete
b9903ecbcf0b: Pull complete
f473bcbd0e44: Pull complete
d8b79b64a9d5: Pull complete
5c36aa47b3f5: Pull complete
c6834909cd19: Pull complete
4f4fb700ef54: Pull complete
Digest: sha256:59a78182908ed962ee81e03d176ff7d2a4818d844f7e74e051da92f2c792d1cd
Status: Downloaded newer image for php:8.2-apache
Step 2/4 : COPY index.php /var/www/html
Step 3/4 : EXPOSE 80
Step 4/4 : CMD ["apache2-foreground"]
Successfully built 43eac8133e2e
Successfully tagged php-apache-wed:latest

ajbeltran94@bloodwyrn in repo: School/DespliegueWEB/Unit5/01Activity/Actividad 10/src on P main [x?] via v8.4.4 as took 0s
[●] * sudo docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
php-apache-wed      latest             43eac8133e2e       About a minute ago 502MB
joanss04/python-fl-api v1.0              a7bfc6f35e26       3 weeks ago        348MB
php                 8.2-apache        6797f9b55e56       2 months ago       502MB
mysql               latest             56a8c14e1404       4 months ago       603MB
hello-world         latest             d2c94e258dcb       22 months ago      13.3kB
```

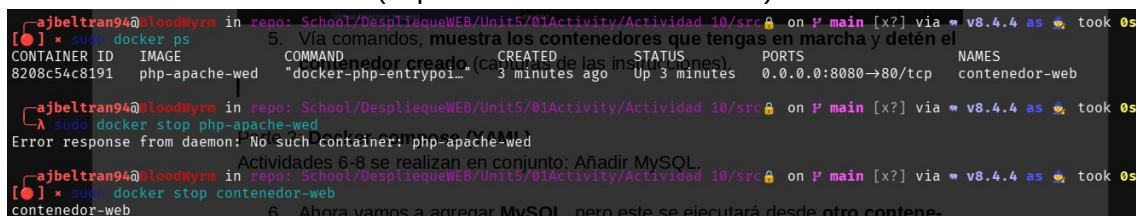
- Ejecuta el **contenedor** (captura del run). Establece vínculos de puertos y un nombre específico. Abre `http://localhost:8080` y verifica que se muestra la información de PHP (captura).



The screenshot shows a terminal window with the command `sudo docker run -d -p 8080:80 --name contenedor-web php-apache-wed` being executed. Below the terminal, a web browser displays the PHP version 8.2.27 page. The page includes a table with system information:

System	Linux 8208c54c8191 6.13.3-zen1-1-zen #1 ZEN SMP PREEMPT_DYNAMIC Mon, 17 Feb 2025 17:41:56 +0000 x86_64
Build Date	Feb 4 2025 04:36:13
Build System	Linux - Docker
Build Provider	https://github.com/docker-library/php
Configure Command	./configure '--build=x86_64-linux-gnu' '--with-config-file-path=/usr/local/etc/php' '--with-config-file-scan-dir=/usr/local/etc/php/conf.d' '--enable-option-checking=fatal' '--with-mhash' '--with-pic' '--enable-mbstring' '--enable-mysqld' '--with-password-argon2' '--with-sodium=shared' '--with-pdo-sqlite=/usr' '--with-sqlite3=/usr' '--with-curl' '--with-iconv' '--with-openssl' '--with-readline' '--with-zlib' '--disable-phpdbg' '--with-pear' '--with-libdir=lib/x86_64-linux-gnu' '--disable-cgi' '--with-apxs2' 'build_alias=x86_64-linux-gnu'
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled

- Vía comandos, muestra los contenedores que tengas en marcha y detén el contenedor creado (capturas de las instrucciones).



The screenshot shows terminal commands and their output. The first command is `sudo docker ps`, which lists the running container 'contenedor-web'. The second command is `sudo docker stop php-apache-wed`, which results in an error: 'No such container: php-apache-wed'. The third command is `sudo docker stop contenedor-web`, which successfully stops the container.

Parte 2: Docker-compose (YAML)

Actividades 6-8 se realizan en conjunto: Añadir MySQL.

- Ahora vamos a agregar **MySQL**, pero este se ejecutará desde **otro contenedor** diferente al creado previamente. Para que puedan trabajar en conjunto, crea un archivo **docker-compose.yml**. Establece las *variables de entorno* de MySQL para definir una contraseña al usuario root (a tu elección, pero toma nota), crear una base de datos con tu nombre y `_db` al final (formato [nombre_db]) y un usuario con tu nombre (contraseña también a tu elección).
- Continuando con el punto anterior, asegúrate de definir un **volumen** para poder tener persistencia a nivel local con el contenedor de MySQL. Para esto, dentro de `src/`, en local, crea una **subcarpeta llamada "datadir"**, a vincular con la carpeta correspondiente del contenedor (ver documentación – en recomendaciones). Explica cómo has introducido esta configuración.
- Crea un nuevo fichero de PHP llamado **"db_test.php"** con el código:

```
<?php
$conn = new mysqli("db", "user", "password", "[nombre_db]");
if ($conn->connect_error) {
    die("Conexión fallida: " . $conn->connect_error);
}
echo "Conexión establecida con MySQL!";
?>
```

Cambia la información de "db" (nombre del servicio), "user", "password", "[nom-

bre_db]" por los datos definidos en los puntos anteriores para establecer la conexión con MySQL.

Edita **dockerfile** y genera nuevamente la **imagen** de tu aplicación, para incluir db_test.php.

Arranca los contenedores y verifica que http://localhost:8080/db_test.php funciona (captura del comando de *docker-compose up*, captura de navegador, y captura de *logs del contenedor de MySQL* donde se vea la conexión realizada desde el navegador).

```
...ajbeltran94@bloodwyrn in repo: School/DespliegueWEB/Units/04Activitiy/Actividad 10/Actividades_6-8/src on P' main [x?] as [?] took 32s  
[●] * sudo docker-compose up -d --build Professional cdfpfrancesdeborja@gmail.com educaibv.es  
[+] Building 10.4s (8/8) FINISHED  
=> [web internal] load build definition from Dockerfile 0.0s  
=> transferring Dockerfile: 19kB 0.0s  
=> [web internal] load metadata for mcr.microsoft.com/windows/servercore/lts2022 0.0s  
=> [web internal] load Dockerfile from Dockerfile 0.0s  
=> transferring context: 2B 0.0s  
=> CACHED [web 1/3] FROM node:16-alpine 0.0s  
=> [web internal] load build context 0.0s  
=> transferring context: 25kB 0.0s  
=> [web 2/3] RUN docker-php-ext-install pdo_mysql 0.0s  
=> [web 3/3] COPY db_test.php /usr/local/bin/db_test.php 0.1s  
=> exporting layers 0.7s  
=> writing image sha256:ccf0e7ba3c0ebaf1706b7bcb131519eb 0.0s  
=> naming to docker.io/library/src-web 0.0s  
[+] Running 3/3  
✔ Network src_default Created 0.1s  
✔ Container mysql_db Started 0.4s  
✔ Container apache_php Started 0.5s
```

Conexión establecida con MySQL!

```

... bloodwyrn in repo: School/Desktop/Workbooks/MySQL/mysqlmain to /usr/share/nginx/html/images de la aplicacion P main [x] Devia v8.4.4 as took 0s
└─ sudo docker logs mysql_db db_test.php.
2025-02-24 18:00:11+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 8.0.41-1.el9 started. php fun-
2025-02-24 18:00:11+00:00 [Note] [Entrypoint]: Switching to dedicated user 'mysql' captura de navegador y
2025-02-24 18:00:11+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 8.0.41-1.el9 started.
2025-02-24 18:00:12+00:00 [Note] [Entrypoint]: Initializing database files donde se vea la conexión realizada
2025-02-24 18:00:12.054509Z [Warning] [MY-011068] [Server] The syntax '--skip-host-cache' is deprecated and will be removed in a f
uture release. Please use SET GLOBAL host_cache_size=0 instead.
2025-02-24 18:00:12.054597Z [System] [MY-013169] [Server] /usr/sbin/mysqld (mysqld 8.0.41) initializing of server in progress as p
rocess 81
2025-02-24 18:00:12.062037Z 1 [System] [MY-013576] [InnoDB] InnoDB initialization has started.
2025-02-24 18:00:12.999021Z 1 [System] [MY-013577] [InnoDB] InnoDB initialization has ended.
2025-02-24 18:00:14.936694Z 6 [Warning] [MY-010453] [Server] root@localhost is created with an empty password ! Please consider swit
ching off the --initialize-insecure option.
2025-02-24 18:00:20+00:00 [Note] [Entrypoint]: Database files initialized
2025-02-24 18:00:20+00:00 [Note] [Entrypoint]: Starting temporary server
2025-02-24 18:00:20.969768Z 0 [Warning] [MY-011068] [Server] The syntax '--skip-host-cache' is deprecated and will be removed in a f
uture release. Please use SET GLOBAL host_cache_size=0 instead.
2025-02-24 18:00:20.970673Z 0 [System] [MY-010116] [Server] /usr/sbin/mysqld (mysqld 8.0.41) starting as process 125
2025-02-24 18:00:20.986031Z 1 [System] [MY-013576] [InnoDB] InnoDB initialization has started.
2025-02-24 18:00:21.263997Z 1 [System] [MY-013577] [InnoDB] InnoDB initialization has ended.
2025-02-24 18:00:21.594089Z 0 [Warning] [MY-010068] [Server] CA certificate ca.pem is self signed.
2025-02-24 18:00:21.594119Z 0 [System] [MY-013602] [Server] Channel mysql main configured to support TLS. Encrypted connections are

```


Parte 3: Datos persistentes y despliegue de App

Actividades 9-10 se realizan en conjunto: Añadir PHPMyAdmin y desplegar app web.

9. **Detén** los contenedores con *docker-compose down* (muestra captura). Edita *docker-compose.yml* para incluir **phpMyAdmin**. Configura las variables de entorno con la contraseña de root que declaraste anteriormente e indicando el nombre del servicio/contenedor de MySQL (ver documentación). Vincula los puertos 8081:80.

Una vez configurado, haz *docker-compose up* y accede a PHPMyAdmin en <http://localhost:8081> con usuario root y verifica que aparece la base de datos con tu nombre (captura de PHPMyAdmin).

```
...BloodWym in repo: School/Despliegue de Docker/01-Adicionando phpMyAdmin a la configuraci... [x?] via v8.4.4 as took 0s
[+] Running 3/3
  Container apache_php Removed
  Container mysql_db Removed
  Network src_default Removed

Edita el inicio del código PHP y agrega los datos que has definido previamente, concretamente:
// Configuración de la conexión a la base de datos
$host = "localhost:8080"; // A DEFINIR POR EL DESARROLLADOR
$user = "nombre"; // A DEFINIR POR EL DESARROLLADOR
```

Browser tabs: dex.php - Google Drive | Práctica10_Docker_24_25.docx - Google Drive | phpMyAdmin

Browser bookmarks: GitHub | ChatGPT | Padlet | Unstuck AI | Work | Google | Microsoft



Language

English ▼

Log in

Username:

Password:

Log in

10. Detén los contenedores nuevamente. **Descarga el fichero index.php adjunto** al enunciado. Se debe desplegar con docker como una aplicación web:

- Edita el inicio del código PHP según los datos que has definido previamente, concretamente:

// Configuración de la conexión a la base de datos

\$host = "localhost:8080"; // A DEFINIR POR EL DESARROLLADOR

\$user = "nombre"; // A DEFINIR POR EL DESARROLLADOR

\$pass = "password"; // A DEFINIR POR EL DESARROLLADOR

\$db = "nombre_db"; // A DEFINIR POR EL DESARROLLADOR

(Adjunta e fichero index.php modificado)

- Vuelve a editar el dockerfile para incluir este fichero index.php en la copia (aíñadelo a la carpeta src) y genera nuevamente la imagen (muestra los cambios y la creaci3n de la imagen).

```
GNU nano 8.3 dockerfile
FROM php:7.4-apache

RUN docker-php-ext-install mysqli

COPY index.php /var/www/html/index.php$db = "nombre_db"; // A DEFINIR POR
(Adjunta e fichero index.php modificado)

...odWyrn in repo: School/DespliegueWEB/Unit5/01Activity/Actividad 10/Actividades_9-10/src on P main [x?] via v8.4.4 as took 0s
[!] * sudo docker build -t webapp_image .
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
Install the buildx component to build images with BuildKit:
https://docs.docker.com/go/buildx/
genera nuevamente la imagen: Navega al directorio donde est1 el 'Dockerfile' y ejecuta:

Sending build context to Docker daemon 197MB
Step 1/3 : FROM php:7.4-apache
7.4-apache: Pulling from library/php
a603fa5e3b41: Pull complete
c428f1a49423: Pull complete
156740b07ef8: Extracting [
fb5a4c8af82f: Download complete
25f85b498fd5: Download complete
9b233e420ac7: Download complete
fe42347c4ecf: Download complete
d14eb2ed1e17: Download complete
66d98f73acb6: Download complete
d2c43c5efbc8: Download complete
ab590b48ea47: Download complete
80692ae2d067: Waiting
05e465aaa99a: Waiting

Si todo est1 correctamente configurado, no necesitar1s hacer m1s cambios en 'docker-
compose.yml'.

6. Levanta los contenedores nuevamente:

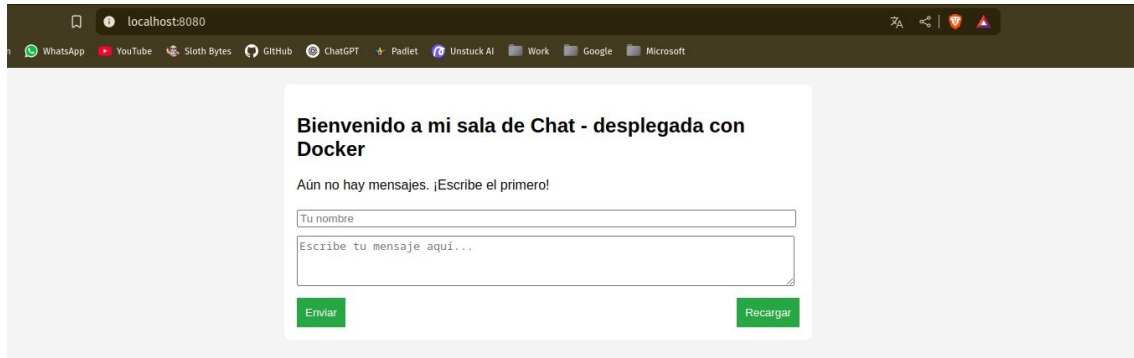
docker-compose up
```

- Edita docker-compose.yml si es necesario (indica si has tenido que cambiar algo y muestra los cambios).

- Arranca los contenedores empleando el fichero YAML (captura de la instrucci3n up y de los contenedores en funcionamiento).

```
...odWyrn in repo: School/DespliegueWEB/Unit5/01Activity/Actividad 10/Actividades_9-10/src on P main [x?] via v8.4.4 as took 0s
[+] Running 3/0
  Container mysql_db Created 0.0s
  Container phpmyadmin Created 0.0s
  Container apache_php Created 0.0s
Attaching to apache_php, mysql_db, phpmyadmin
mysql_db | 2025-02-24 19:04:03:00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 8.0.41-1.el9 started.
mysql_db | 2025-02-24 19:04:03:00:00 [Note] [Entrypoint]: Switching to dedicated user 'mysql'
mysql_db | 2025-02-24 19:04:03:00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 8.0.41-1.el9 started.
apache_php | AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 172.18.0.4. Set the 'Se
rverName' directive globally to suppress this message
mysql_db | '/var/lib/mysql/mysql.sock' -> '/var/run/mysqld/mysqld.sock'
phpmyadmin | AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 172.18.0.3. Set the 'Se
rverName' directive globally to suppress this message
apache_php | AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 172.18.0.4. Set the 'Se
rverName' directive globally to suppress this message
apache_php | [Mon Feb 24 19:04:03.464168 2025] [mpm_prefork:notice] [pid 1:tid 1] AH00163: Apache/2.4.62 (Debian) PHP/8.2.27 config
ured -- resuming normal operations
```

- e. Comprueba si puedes acceder a index.php vía navegador y trabajar con la aplicación (captura del navegador). Introduce mensajes nuevos.



- f. Haz docker-compose down y luego otro up. Comprueba si los mensajes creados previamente siguen existiendo en la aplicación desplegada.



El resultado de esta práctica es una pila LAMP montada con Docker.

A entregar:

- Carpeta Actividad 1-5:
 - o src/index.php (solo con phpinfo())
 - o src/dockerfile
 - o imagen creada (busca en /var/lib/docker)
- Carpeta Actividad 6-8:
 - o src/dockerfile
 - o src/db_test.php
 - o src/docker-compose.yml (con apache, php, mysql y la app web)
 - o imagen creada (busca en /var/lib/docker)
 - o Contenido de src/datadir
- Carpeta Actividad 9-10:
 - o src/dockerfile
 - o src/index.php (adjunto modificado)
 - o src/docker-compose.yml (con apache, php, mysql, PHPMyAdmin y la app web)
 - o imagen creada (busca en /var/lib/docker)
 - o Contenido de src/datadir
- Documento con las capturas y las explicaciones solicitadas, en orden y claramente enumeradas.

Recursos:

- <https://docs.docker.com/>
- https://hub.docker.com/_/mysql
- https://hub.docker.com/_/phpmyadmin

Recordatorio:

- Docker image = aplicación y sus recurso preparados para desplegar
- Docker container = imagen desplegada en entorno virtualizado
- Dockerfile = Para poder construir una imagen
- Docker-compose.yaml = para poder arrancar multiples contenedores en conjunto (en una misma red)

Condiciones de entrega:

- La práctica se **debe** entregar de forma **individual**, cada uno debe presentar sus propias respuestas. Sin embargo, se puede trabajar en equipo.
- Se debe entregar un documento de texto (.pdf, .docx, .odt, etc.) con los ejercicios correctamente ordenados, identificados y **numerados**.
- En cada página del documento debe aparecer el nombre completo del alumno.
- La nota comprenderá un valor numérico entre 0 y 10.
- **La fecha límite de entrega es la indicada en Google Classroom.**
- **Se podrá entregar hasta 72 horas más tarde de la fecha límite pero con una penalización sobre su puntuación (no será posible aspirar al 10).**