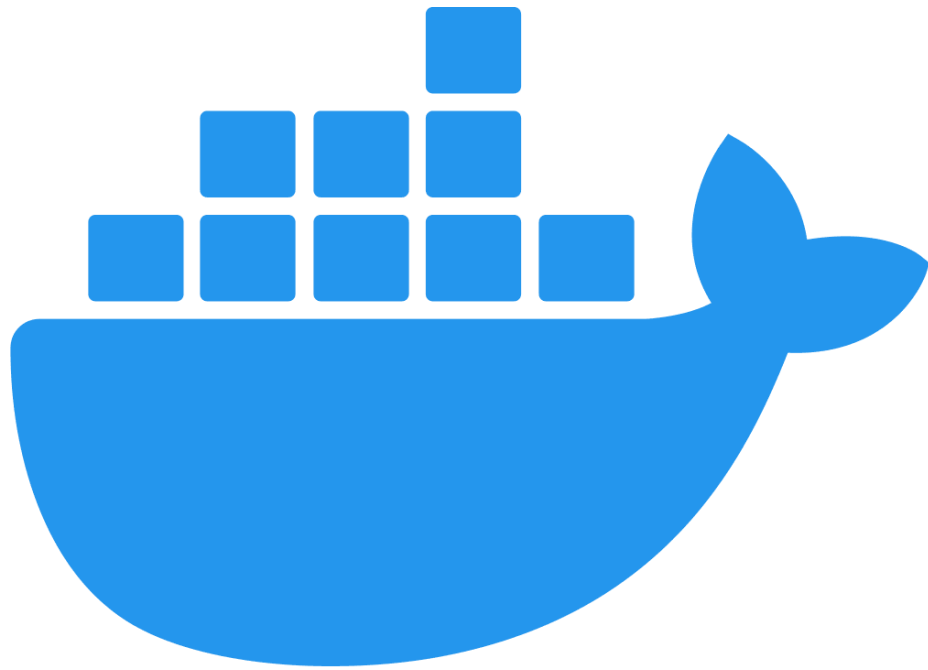


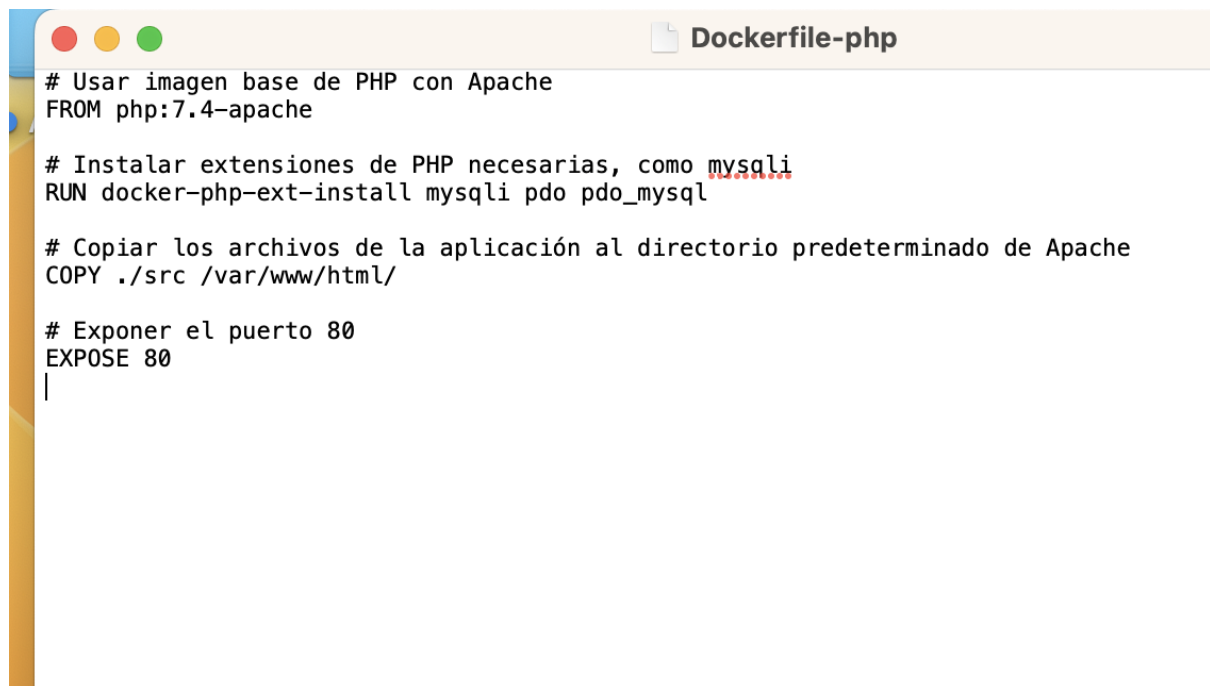
Tutorial - Actividad

Docker Compose



docker®

Primero, necesito un Dockerfile para construir una imagen de PHP.



```
Dockerfile-php

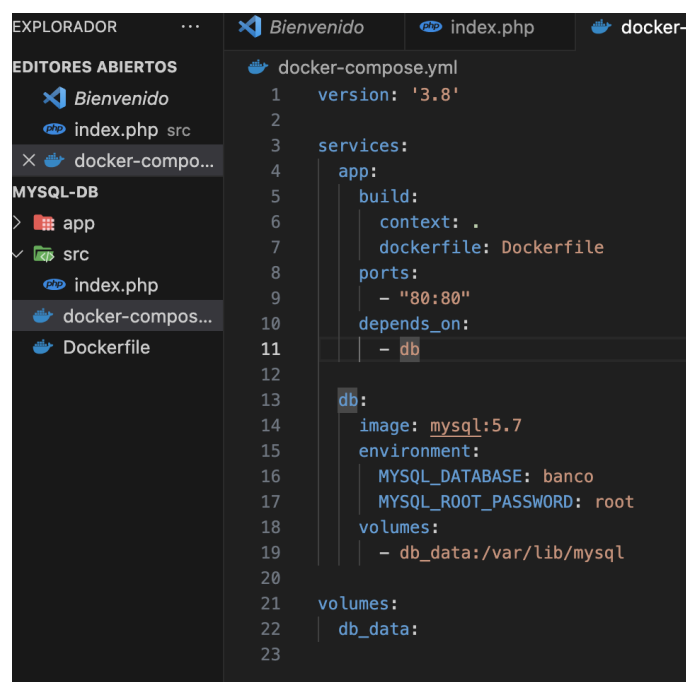
# Usar imagen base de PHP con Apache
FROM php:7.4-apache

# Instalar extensiones de PHP necesarias, como mysqli
RUN docker-php-ext-install mysqli pdo pdo_mysql

# Copiar los archivos de la aplicación al directorio predeterminado de Apache
COPY ./src /var/www/html/

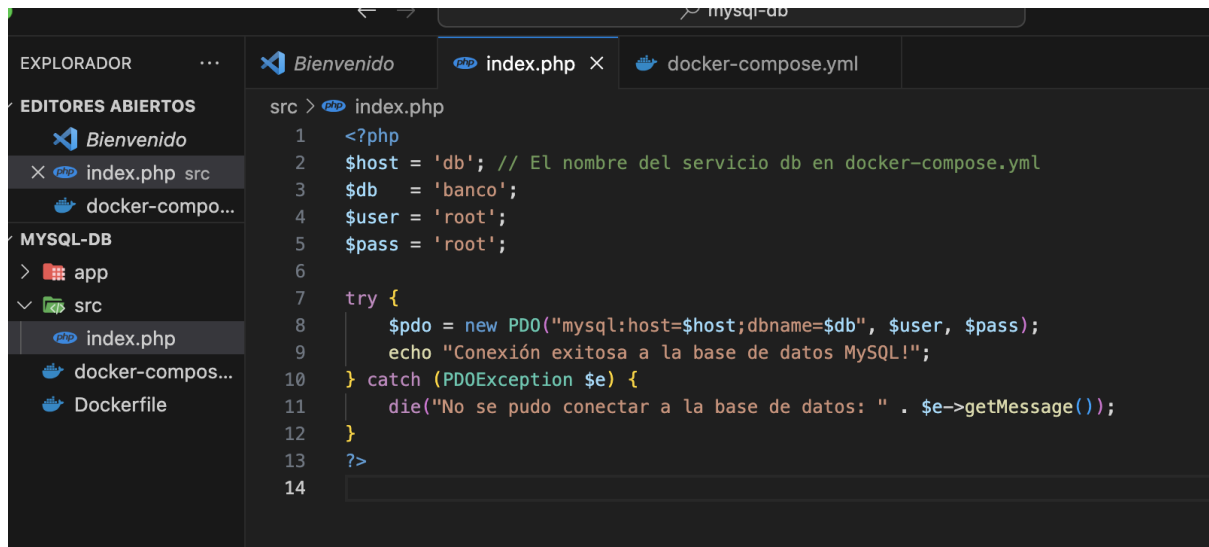
# Exponer el puerto 80
EXPOSE 80
|
```

Ahora creo docker-compose.yml para incluir el servicio PHP y MySQL.



```
EXPLORADOR  ...  Bienvenido  index.php  docker-
EDITORES ABIERTOS
  Bienvenido
  index.php src
  X docker-compo...
MYSQL-DB
  > app
  ✓ src
  index.php
  docker-compos...
  Dockerfile
docker-compose.yml
1  version: '3.8'
2
3  services:
4    app:
5      build:
6        context: .
7        dockerfile: Dockerfile
8      ports:
9        - "80:80"
10     depends_on:
11       - db
12
13     db:
14       image: mysql:5.7
15       environment:
16         MYSQL_DATABASE: banco
17         MYSQL_ROOT_PASSWORD: root
18       volumes:
19         - db_data:/var/lib/mysql
20
21     volumes:
22       db_data:
23
```

Archivo para la conexión con la base de datos con php.



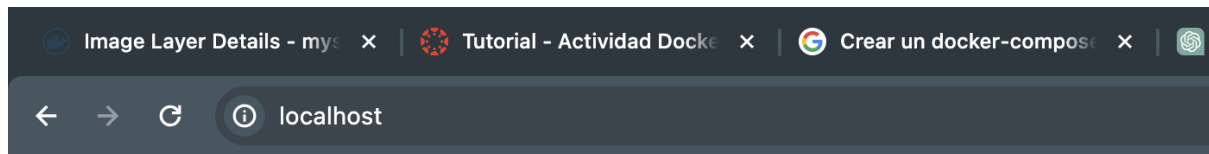
The screenshot shows the Visual Studio Code editor interface. The Explorer sidebar on the left displays the project structure: 'Mysql-DB' folder containing 'app' and 'src' subfolders. 'src' contains 'index.php', 'docker-compose.yml', and 'Dockerfile'. The Editor pane shows 'index.php' with the following PHP code:

```
src > index.php
1  <?php
2  $host = 'db'; // El nombre del servicio db en docker-compose.yml
3  $db   = 'banco';
4  $user = 'root';
5  $pass = 'root';
6
7  try {
8      $pdo = new PDO("mysql:host=$host;dbname=$db", $user, $pass);
9      echo "Conexión exitosa a la base de datos MySQL!";
10 } catch (PDOException $e) {
11     die("No se pudo conectar a la base de datos: " . $e->getMessage());
12 }
13 ?>
14
```

Estructura de mi proyecto

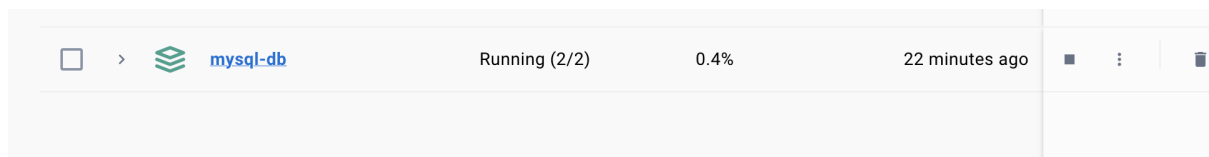
mysql-db	hoy, 14:36	--	Carpeta
app	hoy, 14:35	--	Carpeta
package.json	10/11/23, 10:03	270 bytes	PlainTextType
server.js	10/11/23, 10:04	552 bytes	JavaScript
docker-compose.yml	hoy, 14:30	314 bytes	YAML
Dockerfile	hoy, 14:35	257 bytes	Documento
src	hoy, 14:20	--	Carpeta
index.php	hoy, 14:49	353 bytes	Script PHP

Base de datos conectada exitosamente.



Conexión exitosa a la base de datos MySQL!

```
mysql-db-app-1 | 192.168.65.1 - - [25/Nov/2023:13:37:22 +0000] "GET / HTTP/1.1" 200 273 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 Safari/537.36"
```



Un paso importante que no capturé es el siguiente:

Abrir la Terminal, navegar al Directorio del Proyecto, utilizando el comando `cd` para navegar al directorio donde se encuentran mis archivos. Por ejemplo:

```
bash
```

```
cd Applications/MAMP/htdocs/DAW/mysql-db
```

Ejecutar Docker Compose: Una vez en el directorio, ejecuté el siguiente comando:

```
docker-compose up
```

Este comando construirá las imágenes necesarias si no existen y luego iniciará los contenedores definidos en el `docker-compose.yml`.