Microsoft Windows [Versión 10.0.26100.3476]

(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Alumno\Documents\test>mkdir proyecto-php

C:\Users\Alumno\Documents\test>cd proyecto-php

C:\Users\Alumno\Documents\test\proyecto-php>cd..

C:\Users\Alumno\Documents\test>mkdir PHP

C:\Users\Alumno\Documents\test>CD PHP

C:\Users\Alumno\Documents\test\PHP>mkdir proyect01

C:\Users\Alumno\Documents\test\PHP>cd proyect01

C:\Users\Alumno\Documents\test\PHP\proyect01>mkdir nginx php src

C:\Users\Alumno\Documents\test\PHP\proyect01>touch docker-compose.yml nginx/default.conf php/Dockerfile src/index.php

"touch" no se reconoce como un comando interno o externo,

programa o archivo por lotes ejecutable.

C:\Users\Alumno\Documents\test\PHP\proyect01>copy nul docker-compose.yml

1 archivo(s) copiado(s).

C:\Users\Alumno\Documents\test\PHP\proyect01>copy nul nginx/default.conf

1 archivo(s) copiado(s).

C:\Users\Alumno\Documents\test\PHP\proyect01>copy nul php/Dockerfile

1 archivo(s) copiado(s).

C:\Users\Alumno\Documents\test\PHP\proyect01>copy nul src/index.php

La sintaxis del comando no es correcta.

C:\Users\Alumno\Documents\test\PHP\proyect01>cd src

C:\Users\Alumno\Documents\test\PHP\proyect01\src>copy nul index.php

(podemos crear toda la estructura desde visual studio si se prefiere)

**docker-compose.yml**

services:

php:

image: php:8.2-fpm

build: ./php ***(dónde está el archivo Dockerfile)***

container\_name: docker\_php

volumes:

- ./src:/var/www/html

environment:

- MYSQL\_HOST=host.docker.internal ***(en lugar de localhost)***

- MYSQL\_USER=alumno

- MYSQL\_PASSWORD=alumno

- MYSQL\_DB=test

- MONGODB\_HOST=host.docker.internal

- MONGODB\_USER=alumno

- MONGODB\_PASSWORD=alumno

- MONGODB\_DB=test

- MONGODB\_PORT=27017

nginx: ***(Otro contenedor)***

image: nginx:latest

ports:

- "8080:80"

volumes:

- ./src:/var/www/html

- ./nginx/default.conf:/etc/nginx/conf.d/default.conf ***(corresponde al fichero creado en la carpeta nginx)***

depends\_on:

- php

Configuración de Nginx

1. **Abre el archivo nginx/default.conf**.
2. **Pega el siguiente contenido** en el archivo default.conf:

**Configuración nginx**

server {

listen 80;

server\_name localhost;

root /var/www/html;

index index.php index.html;

location / {

try\_files $uri $uri/ =404;

}

location ~ \.php$ {

include fastcgi\_params;

fastcgi\_pass php:9000;

fastcgi\_index index.php;

fastcgi\_param SCRIPT\_FILENAME /var/www/html$fastcgi\_script\_name;

}

}

Crear el Dockerfile para PHP (si lo necesitas)

1. **Abre el archivo php/Dockerfile**.
2. **Pega el siguiente contenido** para crear el contenedor de PHP con las extensiones necesarias, de momento las libreriás PDO y MySQL:

Sería recomendado compartir bind el directorio usr/local/etc/php/conf.d para que puedas modificar el archivo php.ini desde tu máquina host.

FROM php:8.2-fpm

# Instalar extensiones de PHP necesarias (como PDO y MySQL)

RUN docker-php-ext-install pdo pdo\_mysql

# Instalar las dependencias necesarias para MongoDB

RUN apt-get update && apt-get install -y \

libmongoc-1.0-0 \

libjemalloc2 \

&& rm -rf /var/lib/apt/lists/\*

# Configuración de PHP, initializar el archivo php.ini

# Copiar el archivo php.ini-development a php.ini

RUN mv /usr/local/etc/php/php.ini-development /usr/local/etc/php/php.ini

# Instalar las extensiones de PHP necesarias

RUN pecl install mongodb \

&& docker-php-ext-enable mongodb

# Establecer la ruta de trabajo

WORKDIR /var/www/html

Crear un archivo PHP de prueba en el archivo src/index.php.