

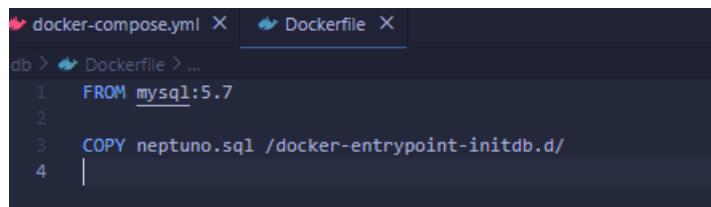
Contenedores Docker. Desplegando una aplicación PHP

1. Usando los ficheros que se entregan en la práctica se pide crear mediante docker-compose un despliegue de la aplicación clientesPedidos que permite mostrar los pedidos de un cliente seleccionado a parie de los datos almacenados en la base de datos Neptuno que se entrega en formato SQL.

Los requisitos son los siguientes:

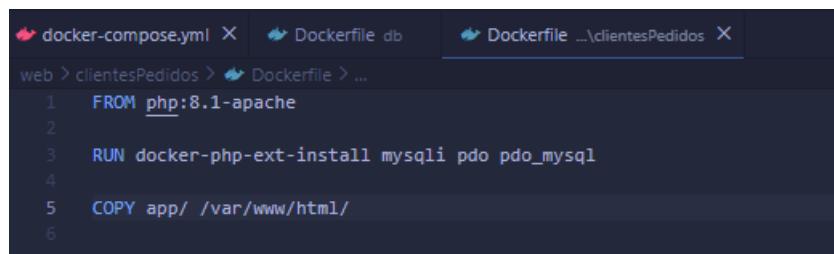
- a) Como base se emplearán los contenedores php:7.4-apache y mysql
- b) Se usarán ficheros Dockerfile para customizar las imágenes de base anteriores. En el caso de php se deben instalar las extensiones php necesarias para la conexión con la base de datos y para copiar los archivos de la aplicación, y en el caso de mysql para agregar el script neptuno.sql de tal modo que se ejecute al lanzar el servidor.
- c) Se crearán los servicios web, db y phpmyadmin
- d) Se mapearán los puertos web tanto del contenedor php como de phpadmin
- e) Se crearán volúmenes para los servicios web y db
- f) Los servicios web y phpmyadmin dependen del servicio db

1.- Dockerfile bd:



```
db > Dockerfile > ...
1 FROM mysql:5.7
2
3 COPY neptuno.sql /docker-entrypoint-initdb.d/
4 |
```

2.- Dockerfile back:



```
web > clientesPedidos > Dockerfile > ...
1 FROM php:8.1-apache
2
3 RUN docker-php-ext-install mysqli pdo pdo_mysql
4
5 COPY app/ /var/www/html/
6 |
```

3.- Docker-compose.yml:

```
docker-compose.yml X
docker-compose.yml
  Run All Services
services:
  db:
    build: ./db
    container_name: clientes_db
    environment:
      MYSQL_ROOT_PASSWORD: rootpass
      MYSQL_DATABASE: neptuno
      MYSQL_USER: neptuno_user
      MYSQL_PASSWORD: neptuno_pass
    volumes:
      - db_datos:/var/lib/mysql
    networks:
      - neptunoN
  web:
    build: ./web/clientesPedidos
    container_name: clientes_web
    depends_on:
      - db
    ports:
      - "8000:80"
    volumes:
      - ./web/clientesPedidos/app:/var/www/html
    networks:
      - neptunoN
  phpmyadmin:
    image: phpmyadmin/phpmyadmin
    container_name: clientes_phpmyadmin
    depends_on:
      - db
    environment:
      PMA_HOST: db
      PMA_USER: root
      PMA_PASSWORD: rootpass
    ports:
      - "8081:80"
    networks:
      - neptunoN
volumes:
  db_datos:
networks:
  neptunoN:
```