

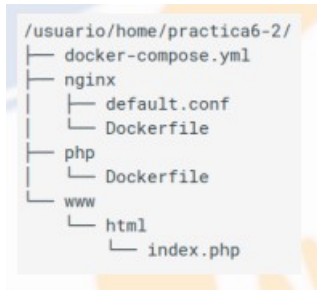
Tarea 2 Docker

INDICE

Lanzar una aplicación PHP con Nginx y MySQL usando Docker y docker-compose.....	2
Estructura de escritorios.....	2
Crear contenedor Nginx.....	2
Creación de un contenedor en PHP.....	3
Creación de contenedor para los datos.....	5
Creación de un contenedor MySQL.....	6
Mostrar Tablas de SQL.....	7

Lanzar una aplicación PHP con Nginx y MySQL usando Docker y docker-compose.

Estructura de escritorios



Crear contenedor Nginx

Vamos a /home/practica6-2 y vamos a modificar el archivo docker-compose.yml para que quede así.

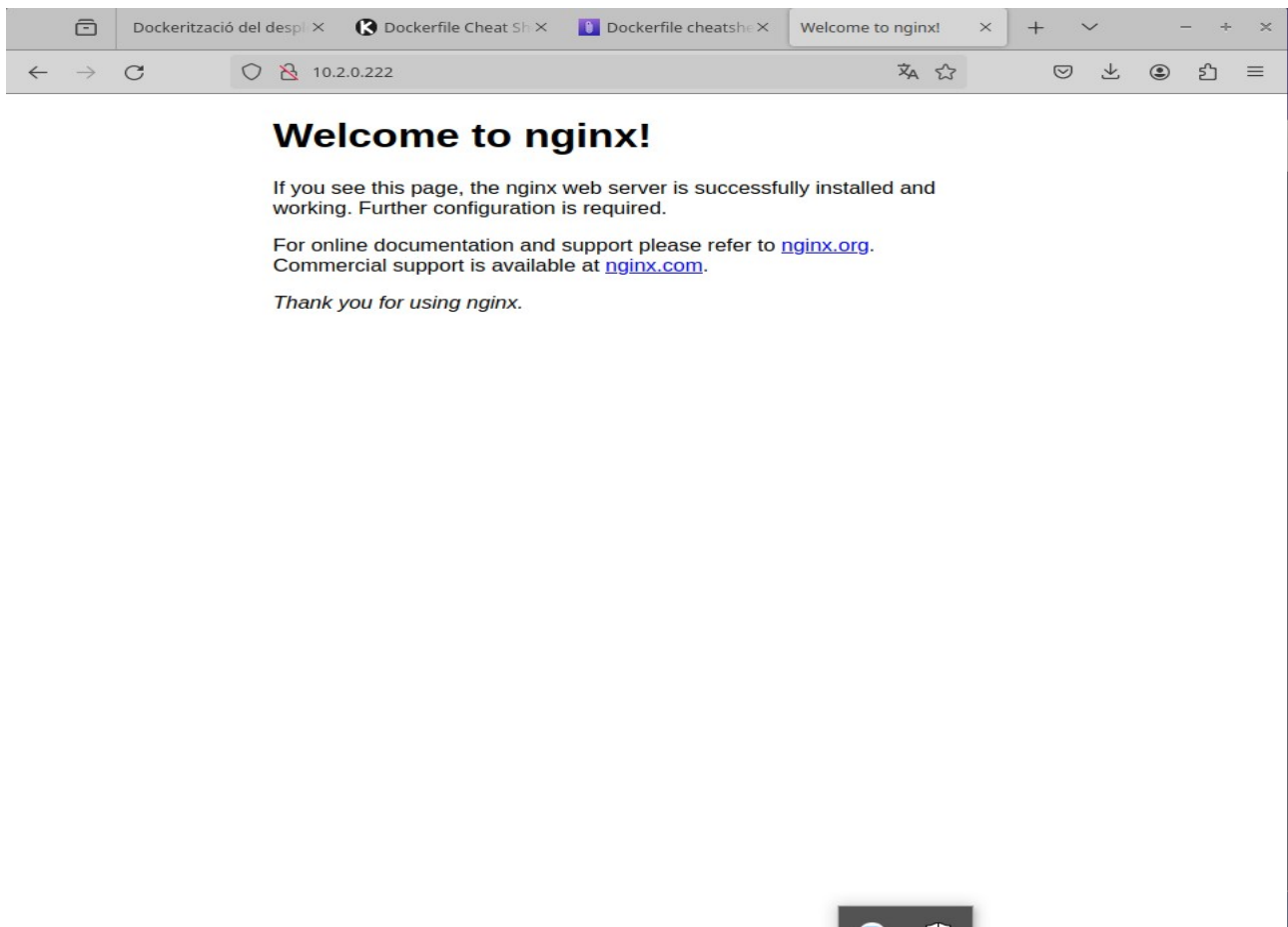
```
david@david-VirtualBox:/home/practica6-2$ cat docker-compose.yml
nginx:
  image: nginx:latest
  container_name: nginx-container
  ports:
    - 80:80
```

con esto le estamos diciendo que coja la ultima imagen del de nginx, como queremos que se llame el contenedor cuando hagamos docker ps o docker ps -a y el puerto que querremos que escuche nuestro ordenador y el puerto que escucha nginx en el contenedor.

Vamos a iniciar este proceso utilizando docker-compose up -d (-d es que se ejecute en 2 plano).

```
david@david-VirtualBox:/home/practica6-2$ docker-compose up -d
Pulling nginx (nginx:latest)...
latest: Pulling from library/nginx
57ee5000d61: Already exists
9b0163235c08: Pull complete
24a6f652770: Pull complete
9f3589a5fc50: Pull complete
9bd99a47d4a: Pull complete
998157bc5c51: Pull complete
ef1c1a36ec2: Pull complete
Digest: sha256:5f44022eab9198d75939d9eaa5341bc077eca16fa51d4ef32d33f1bd4c8cbe7d
Status: Downloaded newer image for nginx:latest
Creating nginx-container ... done
```

Al hacer docker ps tendria que aparecernos el contenedor y si queremos saber si funciona el servidor(que si nos aparece el contenedor en docker ps tendria que funcionar) ponemos en nuestro navegador http://ip_maquina_virtual.



Creación de un contenedor en PHP

Creamos dentro de practica6-2 los siguientes directorios `www/html`.

Te metes dentro del directorio `www/html` y creamos un `index.php` con el siguiente código.

```
david@david-VirtualBox: /home/practica6-2/www/html$ sudo nano index.php
david@david-VirtualBox: /home/practica6-2/www/html$ cat index.php
<!DOCTYPE html>
<html>
<head>
<title>Hola mundo!</title>
</head>
<body>
<h1>Hola mundo!</h1>
<p><?php echo 'Estamos corriendo PHP, version: ' . phpversion(); ?></p>
</body>
```

Ahora vamos a crear el directorio `nginx`: dentro de `practica6-2` ponemos `mkdir nginx` y creamos un archivo llamado `default.conf` con el siguiente código.

```

david@david-VirtualBox:/home/practica6-2/nginx$ cat default.conf
server {

    listen 80 default_server;
    root /var/www/html;
    index index.html index.php;

    charset utf-8;

    location / {
        try_files $uri $uri/ /index.php?$query_string;
    }

    location = /favicon.ico { access_log off; log_not_found off; }
    location = /robots.txt { access_log off; log_not_found off; }

    access_log off;
    error_log /var/log/nginx/error.log error;

    sendfile off;

    client_max_body_size 100m;

    location ~ .php$ {
        fastcgi_split_path_info ^(.+\.php)(/.+)$;
        fastcgi_pass php:9000;
        fastcgi_index index.php
        include fastcgi_params;
        fastcgi_param SCRIPT_FILENAME
        $document_root$fastcgi_script_name;
        fastcgi_intercept_errors off;
        fastcgi_buffer_size 16k;
        fastcgi_buffers 4 16k;
    }
    location ~ /\.ht {
        deny all;
    }
}

```

Ahora creamos el archivo Dockerfile que este archivo guardara la configuración de nginx al contenedor que le asignemos.

Ahora modificamos nuestro archivo docker-compose.yml.

```

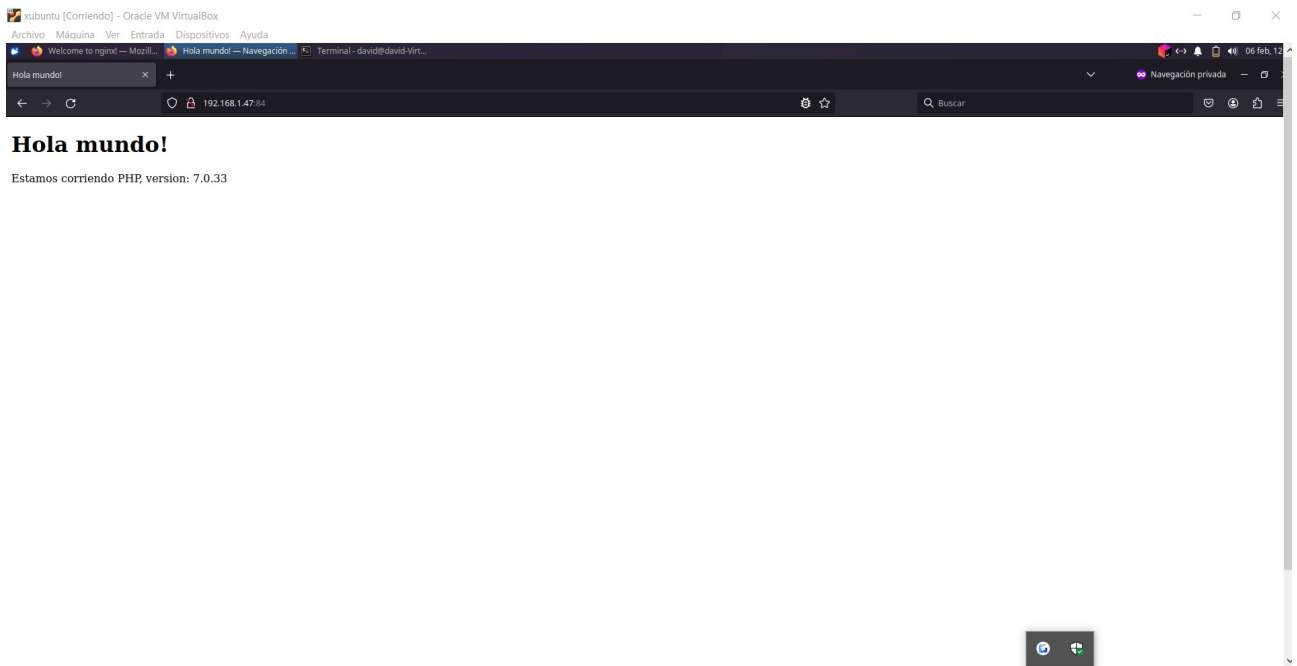
david@david-VirtualBox:/home/practica6-2$ cat docker-compose.yml
services:
  nginx:
    build: ./nginx/
    container_name: nginx-container
    ports:
      - 80:80
    links:
      - php
    volumes:
      - ./www/html:/var/www/html/

  php:
    image: php:7.0-fpm
    container_name: php-container
    expose:
      - 9000
    volumes:
      - ./www/html:/var/www/html/

```

Creara un nuevo contenedor PHP con puerto 9000 que se enlace con el contenedor nginx y el de php que creara un volumen y que estara en var/www/html.

Haremos docker-compose up -d y veremos como estan los contenedores activos utilizando docker ps y si utilizamos nuestra http://ip_nuestra veremos la pagina Hola mon.



Creación de contenedor para los datos.

Modificamos nuestro docker-compose.yml con el siguiente código.

```
david@david-VirtualBox:/home/practica6-2$ cat docker-compose.yml
services:
  nginx:
    build: ./nginx/
    container_name: nginx-container
    ports:
      - 84:80
    links:
      - php
    volumes_from:
      - app-data

  php:
    image: php:7.0-fpm
    container_name: php-container
    expose:
      - 9000
    volumes_from:
      - app-data

  app-data:
    image: php:7.0-fpm
    container_name: app-data-container
    volumes:
      - ./www/html:/var/www/html/
    command: "true"
```

vamos hacer un docker-compose up -d y ver los contenedores.

```
david@david-VirtualBox:/home/practica6-2$ sudo docker ps -a
CONTAINER ID   IMAGE                  COMMAND                  CREATED        STATUS        PORTS                               NAMES
5d3429d7b1b9   practica6-2_nginx     "/docker-entrypoint..." 11 seconds ago Up 10 seconds 0.0.0.0:84->80/tcp, :::84->80/tcp   nginx-container
a05f36c36f4e   php:7.0-fpm          "docker-php-entrypoi..." 12 seconds ago Up 10 seconds 9000/tcp                               php-container
ad7a044973d5   php:7.0-fpm          "docker-php-entrypoi..." 12 seconds ago Exited (0) 11 seconds ago                               app-data-contain
er
```

Creación de un contenedor MySQL

Dentro de practica6-2 creamos el directorio php(mkdir php) y dentro de el añadimos un archivo Dockerfile.

```
david@david-VirtualBox:/home/practica6-2$ tree
.
├── docker-compose.yml
├── nginx
│   ├── default.conf
│   └── Dockerfile
├── php
├── www
│   └── html
│       └── index.php
└── 4 directories, 4 files
david@david-VirtualBox:/home/practica6-2$ cd php/
david@david-VirtualBox:/home/practica6-2/php$ mkdir Dockerfile
mkdir: no se puede crear el directorio «Dockerfile»: Permiso denegado
david@david-VirtualBox:/home/practica6-2/php$ sudo mkdir Dockerfile
david@david-VirtualBox:/home/practica6-2/php$ sudo rm -rf Dockerfile/
david@david-VirtualBox:/home/practica6-2/php$ ls
david@david-VirtualBox:/home/practica6-2/php$ sudo nano Dockerfile
david@david-VirtualBox:/home/practica6-2/php$ cat
FROM php:7.0-fpm
RUN docker-php-ext-install pdo_mysql
david@david-VirtualBox:/home/practica6-2/php$
```

Ahora modificamos el docker-compose.yml

```
david@david-VirtualBox:/home/practica6-2$ sudo nano docker-compose.yml
david@david-VirtualBox:/home/practica6-2$ cat docker-compose.yml
services:
  nginx:
    build: ./nginx/
    container_name: nginx-container
    ports:
      - 84:80
    links:
      - php
    volumes from:
      - app-data

  php:
    image: ./php/
    container_name: php-container
    expose:
      - 9000
    links:
      - mysql
    volumes from:
      - app-data

  app-data:
    image: php:7.0-fpm
    container_name: app-data-container
    volumes:
      - ./www/html:/var/www/html/
    command: "true"

  mysql:
    image: mysql:5.7
    container_name: mysql-container
    volumes from:
      - mysql-data
    environment:
      MYSQL_ROOT_PASSWORD: secret
      MYSQL_DATABASE: mydb
      MYSQL_USER: myuser
      MYSQL_PASSWORD: password

  mysql-data:
    image: mysql:5.7
    container_name: mysql-data-container
    volumes:
      - /var/lib/mysql
    command: "true"
```

Ahora hacemos unos cambios en el index.php que esta en www/html para comprobar al conexión a la base de datos.

```

david@david-VirtualBox:/home/practica6-2/www/html$ sudo nano index.php
david@david-VirtualBox:/home/practica6-2/www/html$ cat index.php
<!DOCTYPE html>
<html>
<head>
<title>Hola mundo!</title>
</head>

<body>
<h1>Hola mundo!</h1>
<p><?php echo 'Estamos corriendo PHP, version: ' . phpversion(); ?></p>
<?php
$dbname = "mydb";
$user = "myuser";
$password = "password";
$host = "mysql";

$con = new PDO("mysql:host=$host;dbname=$dbname");
charset=utf8", $user, $password);
$query = $con->query("SELECT TABLE_NAME FROM
information_schema.TABLES WHERE TABLE_TYPE='BASE
TABLE'");
$tables = $query->fetchAll(PDO::FETCH_COLUMN);

if (empty($tables)) {
echo "<p>No hay tablas en la base de datos \"{$dbname}
}\"</p>";
} else {
echo "<p>La base de datos \"{$dbname}\" tiene las
siguientes tablas:</p>";
echo "<ul>";
foreach ($tables as $table) {
echo "<li>{$table}</li>";
}
echo "</ul>";
}
?>
</body>
</html>

```

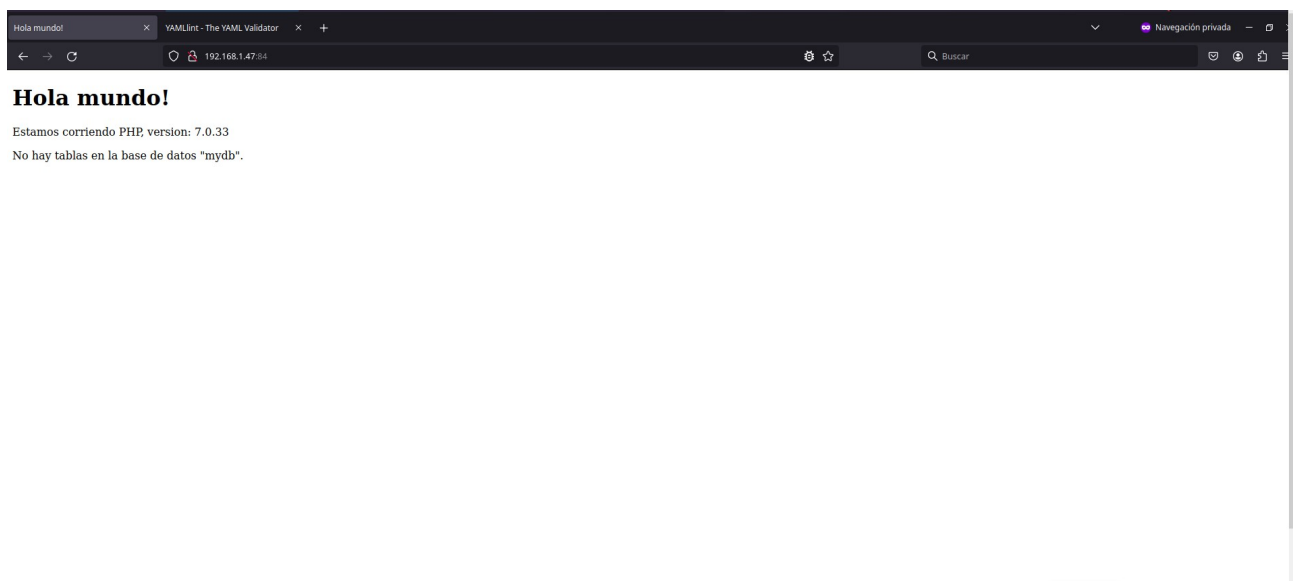
Hacemos docker-compose up -d y miramos los contenedores creados con docker ps -a

```

david@david-VirtualBox:/home/practica6-2/www/html$ sudo docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                               NAMES
080c54d32c25   practica6-2_nginx   "/docker-entrypoint..." 15 minutes ago Up 15 minutes 0.0.0.0:84->80/tcp, :::84->80/tcp   nginx-container
04a2be37161b   practica6-2_php     "/docker-php-entrypoi..." 15 minutes ago Up 15 minutes 9000/tcp                          php-container
b1128bae712    mysql:5.7          "/docker-entrypoint.s..." 15 minutes ago Up 15 minutes 3306/tcp, 33060/tcp               mysql-container
0a102313e710   mysql:5.7          "/docker-entrypoint.s..." 33 minutes ago Exited (0) 15 minutes ago        mysql-data-container
ad7a044973d5   php:7.0-fpm        "/docker-php-entrypoi..." About an hour ago Exited (0) 15 minutes ago        app-data-container

```

Si accedemos a la http://ip_nuestra.



Mostrar Tablas de SQL

Ahora para que aparezcan las tablas que estan en sql(si estan pero no son visibles) para ello vamos a modificar el index.php y donde aparece \$user y \$password hay que poner \$user = “root”; y en \$password = “secret”;

```
david@david-VirtualBox:/home/practica6-2/www/html$ cat index.php
<!DOCTYPE html>
<head>
<title>Hola mundo!</title>
</head>
<body>
<h1>Hola mundo!</h1>
<p><?php echo 'Estamos corriendo PHP, version: ' . phpversion(); ?></p>
<?php
$database = "mydb";
$user = "root";
$password = "secret";
$host = "mysql";

$conection = new PDO("mysql:host={$host};dbname={$database}");
charset=utf8", $user, $password);
$query = $conection->query("SELECT TABLE_NAME FROM
information_schema.TABLES WHERE TABLE_TYPE='BASE
TABLE'");
$tables = $query->fetchAll(PDO::FETCH_COLUMN);

if (empty($tables)) {
echo "<p>No hay tablas en la base de datos \"{$database}
}\"</p>";
} else {
echo "<p>La base de datos \"{$database}\" tiene las
siguientes tablas:</p>";
echo "<ul>";
foreach ($tables as $table) {
echo "<li>{$table}</li>";
}
echo "</ul>";
}
?>
</body>
</html>
```

Sí hacemos F5 a la pagina debería de salir todas las tabasl de sql.

