

Instalar docker windows 10:

<https://www.youtube.com/watch?v=BK-C2RofmTE>

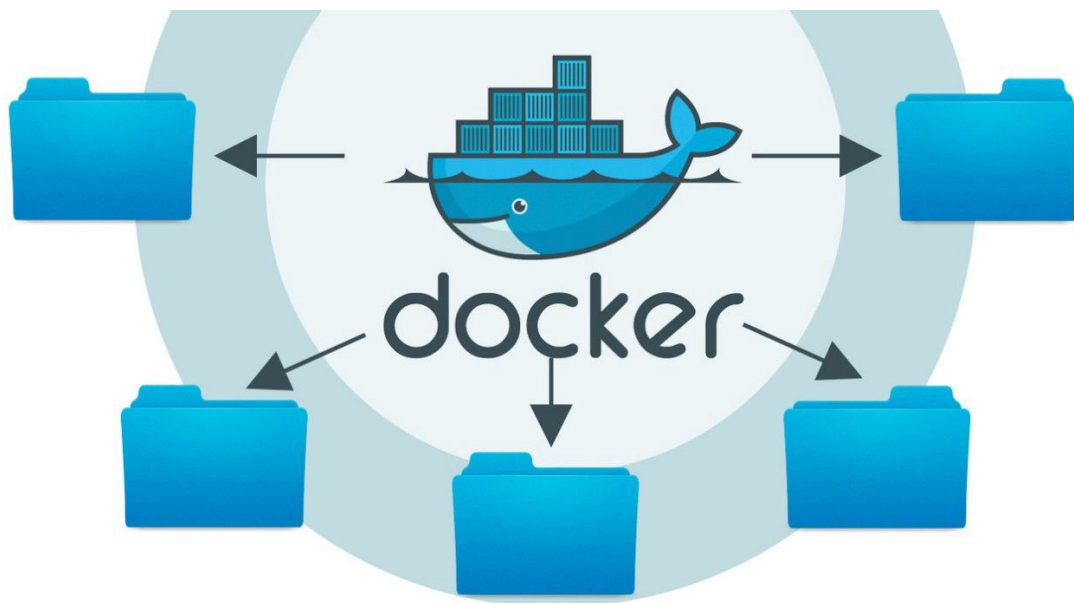
Instalar docker compose windows 10:

<https://www.youtube.com/watch?v=6cm49WRH5lk>

Instalar docker compose Ubuntu:

[`sudo apt install docker-compose`](#)

Volúmenes de docker.



Cuando un contenedor es borrado, toda la información contenida en él, desaparece. Para tener almacenamiento persistente en nuestros contenedores, que no se elimine al borrar el contenedor, es necesario utilizar volúmenes de datos (data volume). Un volumen es un directorio o un fichero en el docker engine que se monta directamente en el contenedor. Podemos montar varios volúmenes en un contenedor y en varios contenedores podemos montar un mismo volumen.

Tenemos dos alternativas para gestionar el almacenamiento en docker:

- Usando volúmenes de datos
- Usando contenedores de volúmenes de datos

Volúmenes de datos [Permalink](#)

Los volúmenes de datos tienen las siguientes características:

- Son utilizados para guardar e intercambiar información de forma independientemente a la vida de un contenedor.

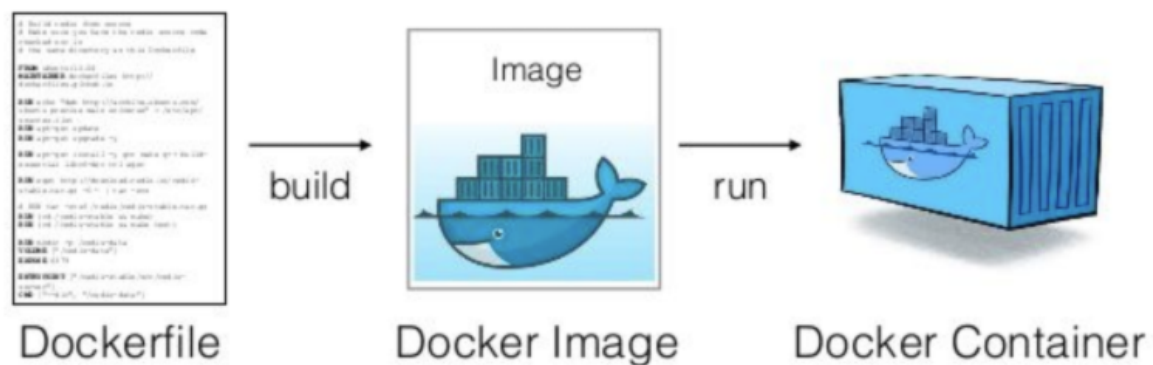
- Nos permiten guardar e intercambiar información entre contenedores.
- Cuando borramos el contenedor, no se elimina el volumen asociado.
- Los volúmenes de datos son directorios del host montados en un directorio del contenedor, aunque también se pueden montar ficheros.
- En el caso de montar en un directorio ya existente de un contenedor un volumen de datos , su contenido no será eliminado.

<https://www.josedomingo.org/pledin/2016/05/gestion-del-almacenamiento-en-docker/>

Dockerfile

Docker can build images automatically by reading the instructions from a Dockerfile. A Dockerfile is a text document that contains all the commands a user could call on the command line to assemble an image. Using docker build users can create an automated build that executes several command-line instructions in succession.

Definir como creo la imagen



<https://docs.docker.com/engine/reference/builder/>

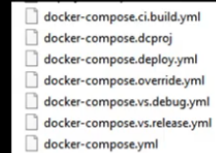
Docker Compose

Saber cómo configurar un contenedor a partir de una imagen o comandos que se basen en una imagen

Comunicar contenedores para usar una sola aplicación.

Compose

- docker-compose.yml
- Es equivalente al dockerfile pero para lanzar contenedores
- Permite definir
 - Grupos de contenedores para ser lanzados juntos
 - Volúmenes
 - Dependencias
 - dockerfile a usar
 - Imágenes a usar
- Agrega flexibilidad al despliegue
- Permite hacer transformaciones basado en entornos



- ☐ docker-compose.ci.build.yml
- ☐ docker-compose.dcpj
- ☐ docker-compose.deploy.yml
- ☐ docker-compose.override.yml
- ☐ docker-compose.vs.debug.yml
- ☐ docker-compose.vs.release.yml
- ☐ docker-compose.yml

Ejercicio: Mysql

1. `sudo docker pull mysql/mysql-server:latest`
2. `sudo docker images`
3. `sudo docker run --name=[container_name] -d [image_tag_name]`

Replace **[container_name]** with the name of your choice. If you do not provide a name, Docker generates a random one.

The **-d** option instructs Docker to run the container as a service in the background.

Replace **[image_tag_name]** with the name of the image downloaded in Step 1.

4. `sudo docker run --name=[container_name] -d mysql/mysql-server:latest`
5. `docker ps`
6. `apt-get install mysql-client`
7. `sudo docker logs [container_name]`
8. `sudo docker exec -it [container_name] bash`
9. `mysql -u root -p`
10. `mysql> ALTER USER 'root'@'localhost' IDENTIFIED BY '[newpassword]';`
11. Comandos en sql:
 - `show databases;`
 - `create database miprueba;`
 - `use miprueba;`
 - `create table persona (id_persona int, nombrepersona varchar(50), correopersona varchar(50));`
 - `show tables;`
 - `create table estudiante (id_estudiante int, nombreestudiante varchar(50), correoestudiante varchar(50));`

- show tables;
- describe persona;
- insert into persona (id_persona, nombrepersona, correopersona) values (1,'persona1', 'correo1');
- insert into persona (id_persona, nombrepersona, correopersona) values (2,'persona2', 'correo2');
- describe estudiante;
- insert into estudiante (id_estudiante, nombreestudiante, correoestudiante) values (1,'estudiante1', 'correo1');
- insert into estudiante (id_estudiante, nombreestudiante, correoestudiante) values (2,'estudiante2', 'correo2');
- select * from persona;
- select * from estudiante;
- exit

Imágenes:

```

harvey@harvey-HP-Pavilion-Laptop-15-cw005la: ~$ docker images
REPOSITORY          TAG             IMAGE ID        CREATED         SIZE
nginx                latest          c919045c4c2b   13 days ago    142MB
mysql/mysql-server   latest         434c35b82b08   7 weeks ago    417MB
harvey@harvey-HP-Pavilion-Laptop-15-cw005la: ~$ sudo docker run --name mysqldb -d mysql/mysql-server
[sudo] contraseña para harvey:
[sudo] contraseña para harvey:
iac3dfec5ba0c1c0246824c5acf036d3d9b98e8339883960e447a9dbacce
harvey@harvey-HP-Pavilion-Laptop-15-cw005la: ~$ docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                               NAMES
iac3dfec5b     mysql/mysql-server  /entrypoint.sh mysq-  5 seconds ago  Up 3 seconds (health: starting)  3306/tcp, 33060-33061/tcp  mysqldb
harvey@harvey-HP-Pavilion-Laptop-15-cw005la: ~$ docker logs iac
[Entrypoint] MySQL Docker Image 8.0.28-1.2.7-server
[Entrypoint] No password option specified for new database.
[Entrypoint] A random onetime password will be generated.
[Entrypoint] Initializing database
2022-03-14T23:36:31.029027Z 0 [System] [MY-013109] [Server] /usr/sbin/mysqld (mysqld 8.0.28) initializing of server in progress as process 17
2022-03-14T23:36:31.740227Z 1 [System] [MY-013576] [InnoDB] InnoDB initialization has started.
2022-03-14T23:36:36.346241Z 1 [System] [MY-013577] [InnoDB] InnoDB initialization has ended.
2022-03-14T23:36:49.322060Z 0 [Warning] [MY-010453] [Server] root@localhost is created with an empty password ! Please consider switching off the --initialize-insecure option.
[Entrypoint] Database initialized
2022-03-14T23:37:16.158206Z 0 [System] [MY-010116] [Server] /usr/sbin/mysqld (mysqld 8.0.28) starting as process 70
2022-03-14T23:37:16.243769Z 1 [System] [MY-013576] [InnoDB] InnoDB initialization has started.
2022-03-14T23:37:16.768953Z 1 [System] [MY-013577] [InnoDB] InnoDB initialization has ended.
2022-03-14T23:37:17.441899Z 0 [Warning] [MY-010806] [Server] CA certificate ca.pem is self signed.
2022-03-14T23:37:17.441159Z 0 [System] [MY-013602] [Server] Channel mysql_main configured to support TLS. Encrypted connections are now supported for this channel.
2022-03-14T23:37:17.487267Z 0 [System] [MY-011323] [Server] X Plugin ready for connections. Socket: /var/run/mysqld/mysqld.sock
2022-03-14T23:37:17.487308Z 0 [System] [MY-010931] [Server] /usr/sbin/mysqld: ready for connections. Version: '8.0.28' socket: '/var/lib/mysql/mysql.sock' port: 0 MySQL Community Server - GPL.
Warning: Unable to load '/usr/share/zoneinfo/iso3166.tab' as time zone, skipping it.
Warning: Unable to load '/usr/share/zoneinfo/leapseconds' as time zone, skipping it.
Warning: Unable to load '/usr/share/zoneinfo/tzdata.zi' as time zone, skipping it.
Warning: Unable to load '/usr/share/zoneinfo/zone.tab' as time zone, skipping it.
Warning: Unable to load '/usr/share/zoneinfo/zone1970.tab' as time zone, skipping it.
[Entrypoint] GENERATED ROOT PASSWORD: 360WRTQ9v83F2Xn7rA&+i6:JTYiy&u
[Entrypoint] Ignoring /docker-entrypoint-initdb.d/*
2022-03-14T23:37:33.419916Z 11 [System] [MY-013172] [Server] Received SHUTDOWN from user root. Shutting down mysqld (Version: 8.0.28).
2022-03-14T23:37:40.146299Z 0 [System] [MY-010910] [Server] /usr/sbin/mysqld: Shutdown complete (mysqld 8.0.28) MySQL Community Server - GPL.
[Entrypoint] Server shut down
[Entrypoint] Setting root user as expired. Password will need to be changed before database can be used.
[Entrypoint] MySQL init process done. Ready for start up.
[Entrypoint] Starting MySQL 8.0.28-1.2.7-server
2022-03-14T23:37:40.826019Z 0 [System] [MY-010116] [Server] /usr/sbin/mysqld (mysqld 8.0.28) starting as process 1
2022-03-14T23:37:40.878133Z 1 [System] [MY-013576] [InnoDB] InnoDB initialization has started.
2022-03-14T23:37:41.441838Z 1 [System] [MY-013577] [InnoDB] InnoDB initialization has ended.
2022-03-14T23:37:41.954989Z 0 [Warning] [MY-010806] [Server] CA certificate ca.pem is self signed.
2022-03-14T23:37:41.955052Z 0 [System] [MY-013602] [Server] Channel mysql_main configured to support TLS. Encrypted connections are now supported for this channel.
2022-03-14T23:37:42.045541Z 0 [System] [MY-011323] [Server] X Plugin ready for connections. Bind-address: '::' port: 33060, socket: /var/run/mysqld/mysqld.sock
2022-03-14T23:37:42.045688Z 0 [System] [MY-010931] [Server] /usr/sbin/mysqld: ready for connections. Version: '8.0.28' socket: '/var/lib/mysql/mysql.sock' port: 3306 MySQL Community Server - GPL.

```

```

harvey@harvey-HP-Pavilion-Laptop-15-cw0005la:~$ sudo docker exec -it iac bash
bash-4.4# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 15
Server version: 8.0.28

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> ALTER USER 'root'@'localhost' IDENTIFIED BY '0000';
Query OK, 0 rows affected (0.16 sec)

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
4 rows in set (0.01 sec)

mysql> create database mprueba;
Query OK, 1 row affected (0.20 sec)

mysql> use mprueba;
Database changed
mysql> show tables;
Empty set (0.00 sec)

mysql> create table persona (id_persona int, nonbrepersona varchar(50), correopersona varchar(50))
-> ;
Query OK, 0 rows affected (0.34 sec)

mysql> show tables;
+-----+
| Tables_in_mprueba |
+-----+
| persona |
+-----+
1 row in set (0.00 sec)

mysql> create table estudiante (id_estudiante int, nonbreestudiante varchar(50),
correoestudiante varchar(50));
Query OK, 0 rows affected (0.35 sec)

```

```

harvey@harvey-HP-Pavilion-Laptop-15-cw0005la:~$ mysql -u root -p mprueba
Enter password:
mysql> show tables;
+-----+
| Tables_in_mprueba |
+-----+
| estudiante |
| persona |
+-----+
2 rows in set (0.00 sec)

mysql> describe persona;
+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+
| id_persona | int | YES | | NULL | |
| nonbrepersona | varchar(50) | YES | | NULL | |
| correopersona | varchar(50) | YES | | NULL | |
+-----+
3 rows in set (0.01 sec)

mysql> insert into persona (id_persona, nonbrepersona, correopersona) values (1,'persona1', 'correo1');
Query OK, 1 row affected (0.07 sec)

mysql> insert into persona (id_persona, nonbrepersona, correopersona) values (2,'persona2', 'correo2');
Query OK, 1 row affected (0.05 sec)

mysql> describe estudiante;
+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+
| id_estudiante | int | YES | | NULL | |
| nonbreestudiante | varchar(50) | YES | | NULL | |
| correoestudiante | varchar(50) | YES | | NULL | |
+-----+
3 rows in set (0.00 sec)

mysql> insert into estudiante (id_estudiante, nonbreestudiante, correoestudiante) values (1,'estudiante1', 'correo1');
Query OK, 1 row affected (0.07 sec)

mysql> insert into estudiante (id_estudiante, nonbreestudiante, correoestudiante) values (2,'estudiante2', 'correo2');
Query OK, 1 row affected (0.08 sec)

mysql> select * from persona;
+-----+
| id_persona | nonbrepersona | correopersona |
+-----+
| 1 | persona1 | correo1 |
| 2 | persona2 | correo2 |
+-----+
2 rows in set (0.00 sec)

```

```

mysql> select * from estudiante;
+-----+
| id_estudiante | nonbreestudiante | correoestudiante |
+-----+
| 1 | estudiante1 | correo1 |
| 2 | estudiante2 | correo2 |
+-----+
2 rows in set (0.00 sec)

mysql> exit
Bye
bash-4.4# exit

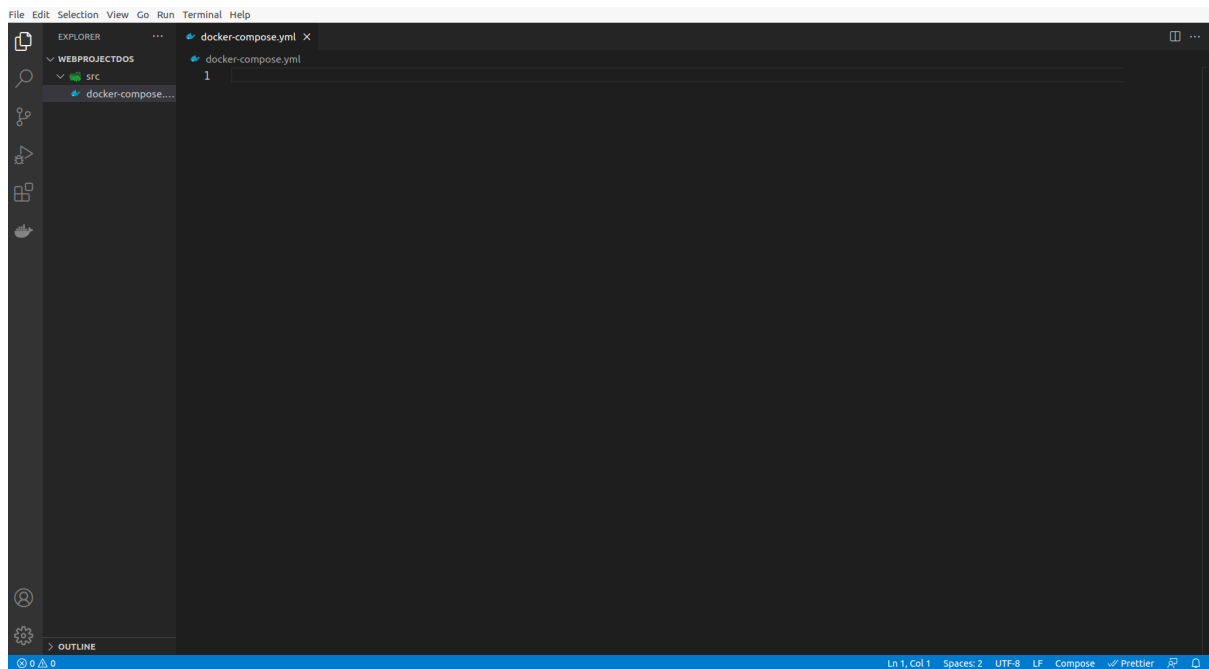
```

<https://phoenixnap.com/kb/mysql-docker-container>

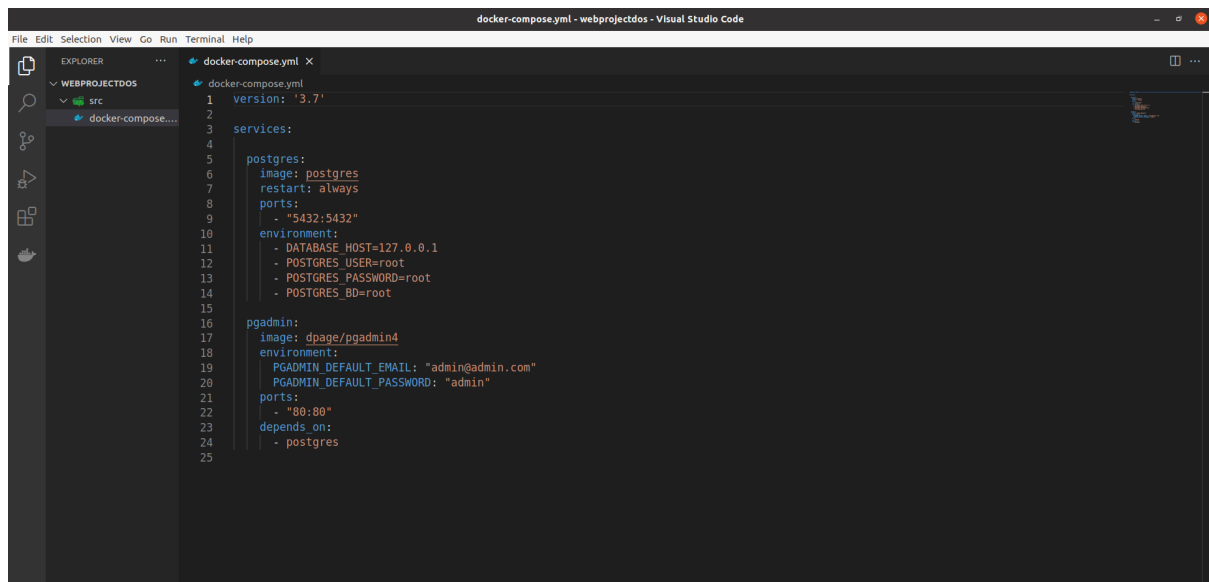
Taller: PgAdmin y postgres

1. Crear una nueva carpeta llamada webprojectdos.

2. Abrir dicha carpeta con visualstudio
3. Dentro de la carpeta principal crear una nueva carpeta llamada src y dentro de ella un archivo llamado docker-compose.yml



4. Visitar la siguiente página:
<https://docs.docker.com/compose/compose-file/compose-versioning/>, para verificar la última versión de docker compose.
5. En el archivo docker-compose.yml agregar lo siguiente:



<https://gist.github.com/FaztTech/ecf7ccf20d04e090f8896ad9f74810e4>

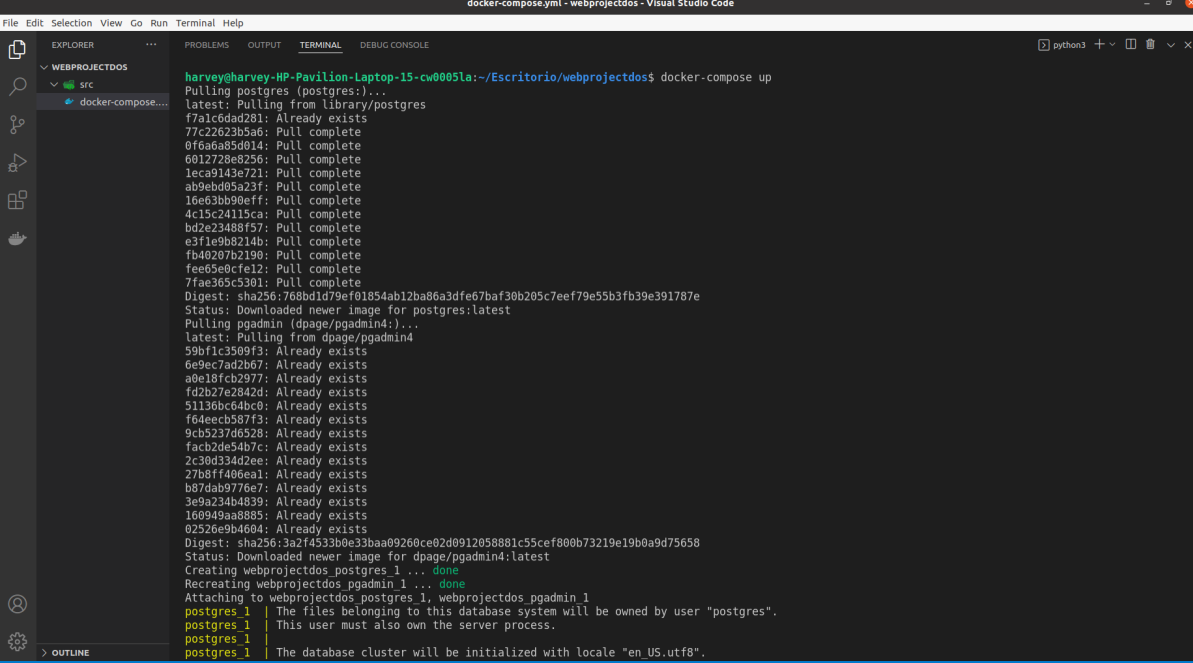
6. Los enlaces de referencia son:
https://www.pgadmin.org/docs/pgadmin4/latest/container_deployment.html

<https://hub.docker.com/r/dpage/pgadmin4/>

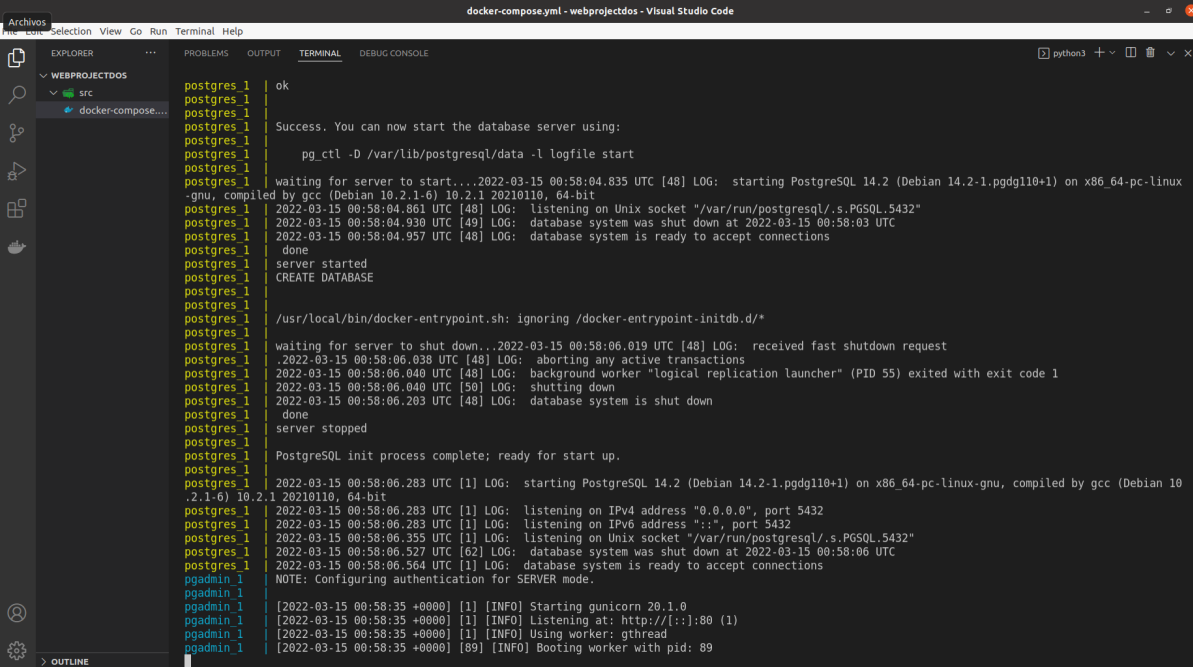
https://hub.docker.com/_/postgres

<https://docs.docker.com/compose/compose-file/compose-versioning/>

7. Abrir una terminal y ejecutar el siguiente comando: `docker-compose up`



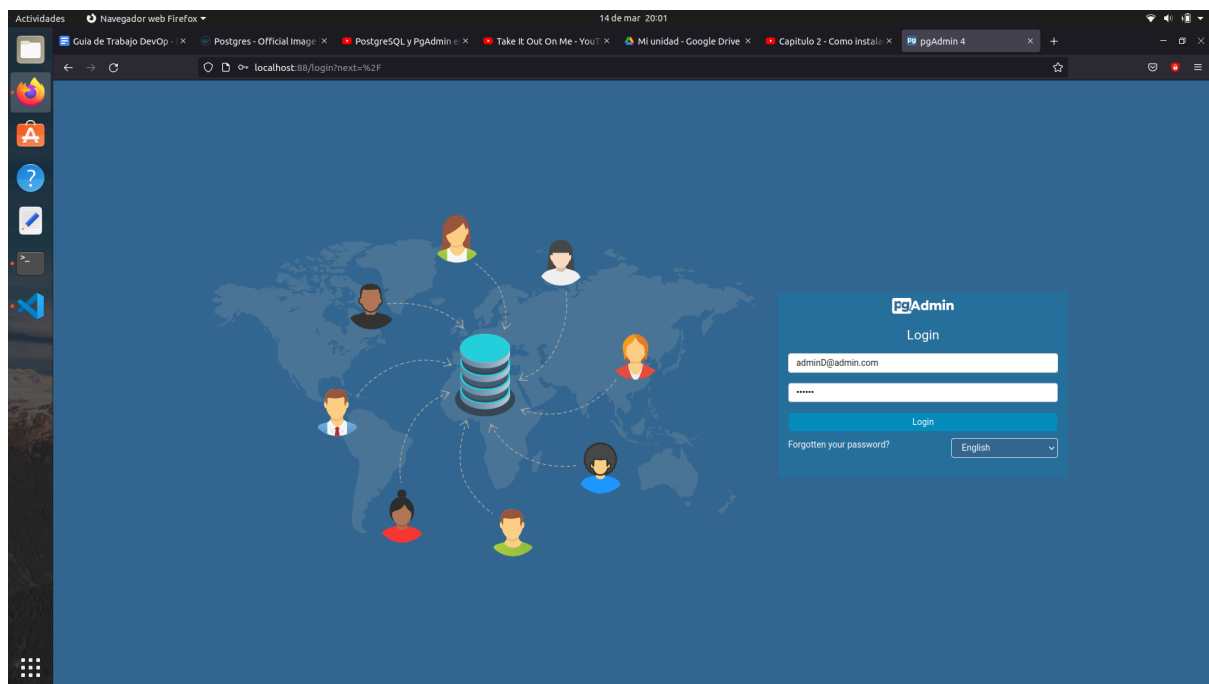
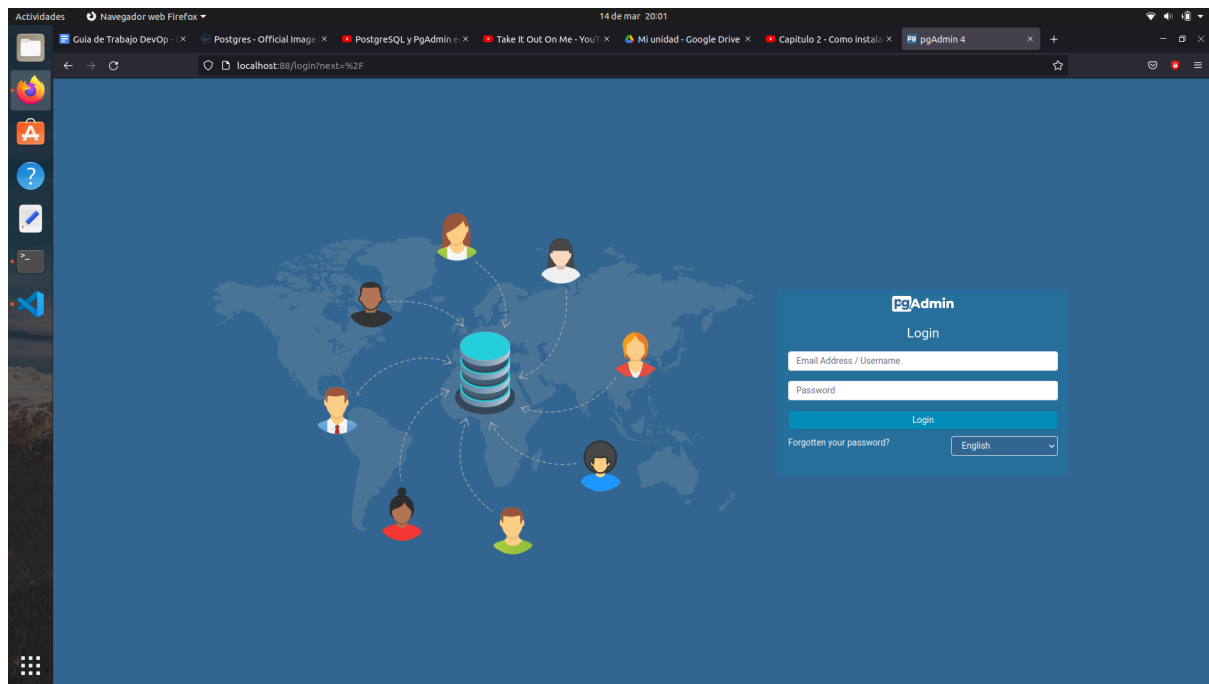
```
harvey@harvey-HP-Pavilion-Laptop-15-cw0005la:~/Escritorio/webprojectdos$ docker-compose up
Pulling postgres (postgres)...
latest: Pulling from library/postgres
f7alcddad281: Already exists
77c22623b5a6: Pull complete
0f6a6a85d014: Pull complete
6012728e8256: Pull complete
1eca9143e721: Pull complete
ab9eb085a23f: Pull complete
16e63bb90eff: Pull complete
4c15c24115ca: Pull complete
bd2e23488f57: Pull complete
e3f1e9b8214b: Pull complete
fb40207b2190: Pull complete
fee650ecf12: Pull complete
7f9e355c5901: Pull complete
Digest: sha256:768bd1d79ef01854ab12ba86a3dfe67baf30b205c7eef79e55b3fb39e391787e
Status: Downloaded newer image for postgres:latest
Pulling pgadmin (dpage/pgadmin4)...
latest: Pulling from dpage/pgadmin4
59b1c3509f3: Already exists
6e9ec7ad2b67: Already exists
a0e18fcb2977: Already exists
fd2b7e2842d: Already exists
51136bc64bc0: Already exists
f64e0cb587f3: Already exists
9cb5237d6528: Already exists
facb2de54b7c: Already exists
2c30d334d2ee: Already exists
27b8f406eal: Already exists
b87dab9776e7: Already exists
3e9a234b4839: Already exists
160949aa8885: Already exists
02526e9b4604: Already exists
Digest: sha256:3a2f4533b0e33baa09260ce02d0912058881c55cef800b73219e19b0a9d75658
Status: Downloaded newer image for dpage/pgadmin4:latest
Creating webprojectdos_postgres_1 ... done
Recreating webprojectdos_pgadmin_1 ... done
Attaching to webprojectdos_postgres_1, webprojectdos_pgadmin_1
postgres_1 | The files belonging to this database system will be owned by user "postgres".
postgres_1 | This user must also own the server process.
postgres_1 |
postgres_1 | The database cluster will be initialized with locale "en_US.utf8".
```



```
postgres_1 | ok
postgres_1 |
postgres_1 | Success. You can now start the database server using:
postgres_1 |
postgres_1 | pg_ctl -D /var/lib/postgresql/data -l logfile start
postgres_1 |
postgres_1 | waiting for server to start...2022-03-15 00:58:04.835 UTC [48] LOG: starting PostgreSQL 14.2 (Debian 14.2-1.pgdg110+1) on x86_64-pc-linux
postgres_1 | -gnu, compiled by gcc (Debian 10.2.1-6) 10.2.1 20210110, 64-bit
postgres_1 | 2022-03-15 00:58:04.861 UTC [48] LOG: listening on Unix socket "/var/run/postgresql/.s.PGSQL.5432"
postgres_1 | 2022-03-15 00:58:04.930 UTC [49] LOG: database system was shut down at 2022-03-15 00:58:03 UTC
postgres_1 | 2022-03-15 00:58:04.957 UTC [48] LOG: database system is ready to accept connections
postgres_1 | done
postgres_1 | server started
postgres_1 | CREATE DATABASE
postgres_1 |
postgres_1 | /usr/local/bin/docker-entrypoint.sh: ignoring /docker-entrypoint-initdb.d/*
postgres_1 |
postgres_1 | waiting for server to shut down...2022-03-15 00:58:06.019 UTC [48] LOG: received fast shutdown request
postgres_1 | 2022-03-15 00:58:06.038 UTC [48] LOG: aborting any active transactions
postgres_1 | 2022-03-15 00:58:06.040 UTC [48] LOG: background worker "logical replication launcher" (PID 55) exited with exit code 1
postgres_1 | 2022-03-15 00:58:06.040 UTC [50] LOG: shutting down
postgres_1 | 2022-03-15 00:58:06.203 UTC [48] LOG: database system is shut down
postgres_1 | done
postgres_1 | server stopped
postgres_1 |
postgres_1 | PostgreSQL init process complete; ready for start up.
postgres_1 |
postgres_1 | 2022-03-15 00:58:06.283 UTC [1] LOG: starting PostgreSQL 14.2 (Debian 14.2-1.pgdg110+1) on x86_64-pc-linux-gnu, compiled by gcc (Debian 10
postgres_1 | 2.1-6) 10.2.1 20210110, 64-bit
postgres_1 | 2022-03-15 00:58:06.283 UTC [1] LOG: listening on IPv4 address "0.0.0.0", port 5432
postgres_1 | 2022-03-15 00:58:06.283 UTC [1] LOG: listening on IPv6 address "::", port 5432
postgres_1 | 2022-03-15 00:58:06.355 UTC [1] LOG: listening on Unix socket "/var/run/postgresql/.s.PGSQL.5432"
postgres_1 | 2022-03-15 00:58:06.527 UTC [62] LOG: database system was shut down at 2022-03-15 00:58:06 UTC
postgres_1 | 2022-03-15 00:58:06.564 UTC [1] LOG: database system is ready to accept connections
pgadmin_1 | NOTE: Configuring authentication for SERVER mode.
pgadmin_1 |
pgadmin_1 | [2022-03-15 00:58:35 +0000] [1] [INFO] Starting unicorn 20.1.0
pgadmin_1 | [2022-03-15 00:58:35 +0000] [1] [INFO] Listening at: http://[::]:80 (1)
pgadmin_1 | [2022-03-15 00:58:35 +0000] [1] [INFO] Using worker: gthread
pgadmin_1 | [2022-03-15 00:58:35 +0000] [89] [INFO] Booting worker with pid: 89
```

Descarga postgres y pgadmin

8. Abrir un navegador y acceder a localhost (se debe agregar el puerto en caso de ser necesario) e iniciar sesión con las respectivas credenciales.



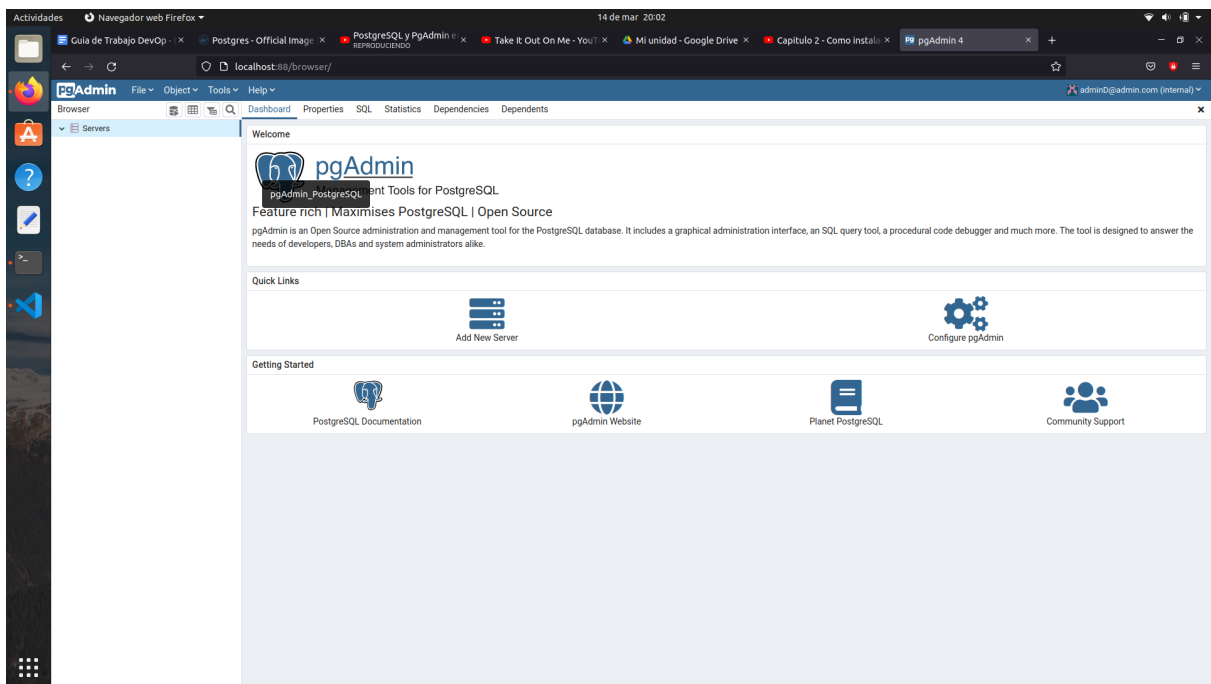
9. Verificamos que los contenedores estén corriendo con docker ps

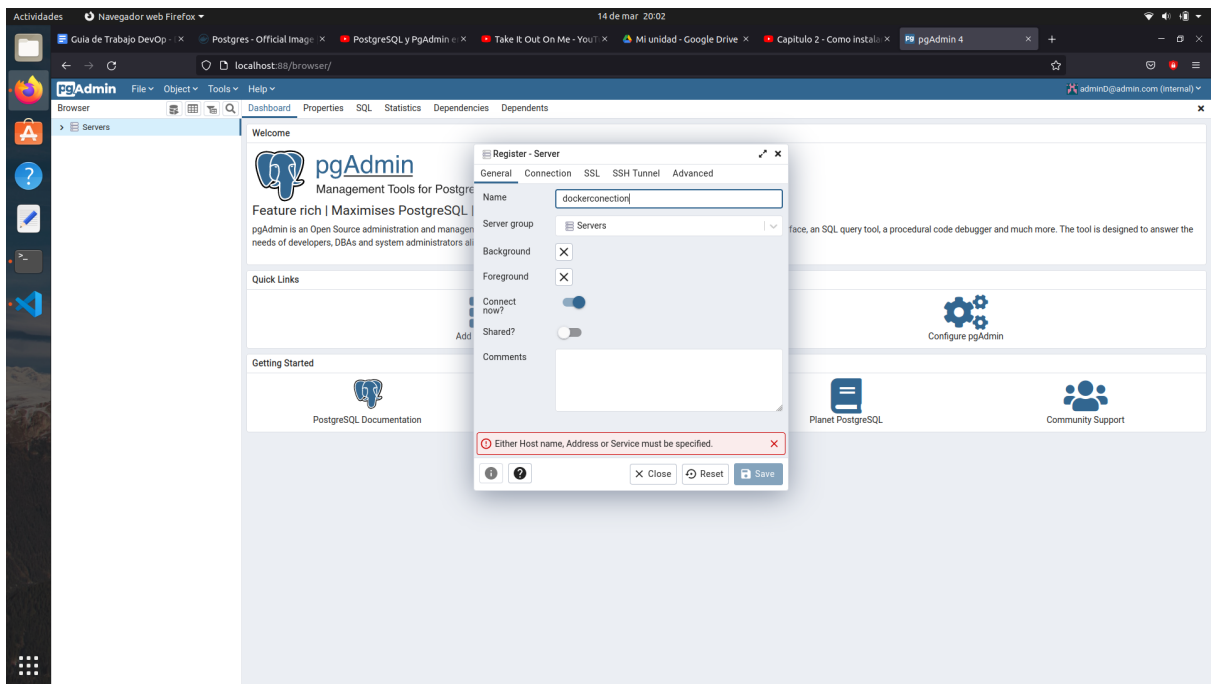
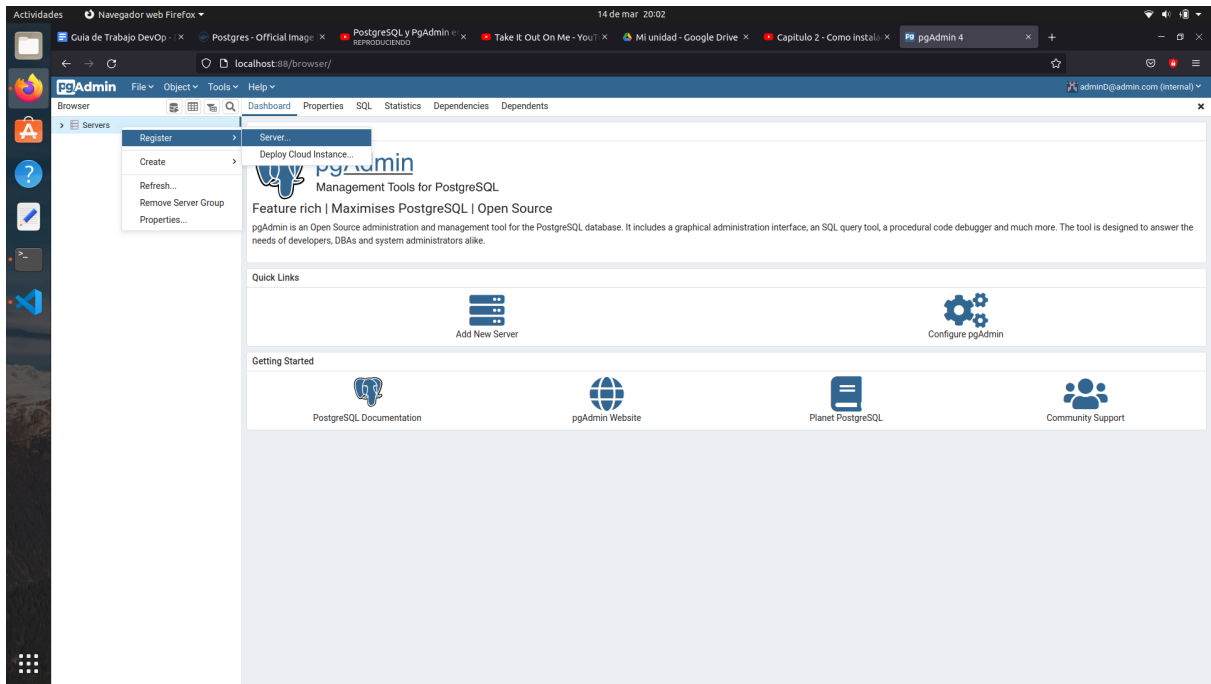
The screenshot shows the Visual Studio Code editor with a Docker Compose file named `docker-compose.yml` open. The file defines a `pgadmin` service that depends on a `postgres` service. The `pgadmin` service uses the `dpape/pgadmin4` image and has an environment variable `PGADMIN_DEFAULT_EMAIL` set to `adminD@admin.com` and `PGADMIN_DEFAULT_PASSWORD` set to `adminD`. The `postgres` service uses the `postgres` image and has environment variables for `DATABASE_HOST`, `POSTGRES_USER`, `POSTGRES_PASSWORD`, and `POSTGRES_DB`. The `pgadmin` service has a port mapping of `88:80`. The terminal shows the output of the `docker images` and `docker ps` commands, confirming that the `pgadmin` container is running.

```
7   environment:
8     - DATABASE_HOST=127.0.0.1
9     - POSTGRES_USER=root
10    - POSTGRES_PASSWORD=root
11    - POSTGRES_DB=root
12  ports:
13    - 5433:5432
14
15  pgadmin:
16    image: dpape/pgadmin4
17    environment:
18      PGADMIN_DEFAULT_EMAIL: "adminD@admin.com"
19      PGADMIN_DEFAULT_PASSWORD: "adminD"
20    ports:
21      - "88:80"
22    depends_on:
23      - postgres
```

```
harvey@harvey-HP-Pavilion-Laptop-15-cw0005la:~/Escritorio/webprojectdos$ docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
dpape/pgadmin4       latest             4b5bbdb3624        21 hours ago       340MB
postgres             latest             d7337c283715       6 days ago         376MB
nginx               latest             c919045c4c2b       13 days ago        142MB
mysql/mysql-server   latest             434c35b82b08       7 weeks ago        417MB
harvey@harvey-HP-Pavilion-Laptop-15-cw0005la:~/Escritorio/webprojectdos$ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED             STATUS              PORTS                               NAMES
609965f625d4   dpape/pgadmin4   "/entrypoint.sh"       3 minutes ago      Up 3 minutes        443/tcp, 0.0.0.0:88->80/tcp, :::88->80/tcp   webpro
e78d4aa750b8   postgres        "docker-entrypoint.s..." 3 minutes ago      Up 3 minutes        0.0.0.0:5433->5432/tcp, :::5433->5432/tcp   webpro
jctdos_pgadmin_1
jctdos_postgres_1
```

10. Configurar servidor en pgAdmin y validar que se encuentre la base de datos root.





Actividades Navegador web Firefox 14 de mar 22:03

pgAdmin

Browser Servers (1) dockerconection Databases Login/Group Roles Tablespaces

Dashboard Properties SQL Statistics Dependencies Dependents

Server sessions

Transactions per second

Transactions Commits Rollbacks

Tuples in

Fetches Returned

Block I/O

Reads Hits

Server activity

Sessions Locks Prepared Transactions Configur

dockerconection

General Connection SSL SSH Tunnel Advanced

Host name/address postgres

Port 5432

Maintenance database postgres

Username root

Kerberos authentication?

Role

Service

Close Reset Save

PID	Database	User	App	State	Wait event	Blocking PIDs
27					Activity: CheckpointerMain	
28					Activity: BgWriterMain	
29					Activity: WalWriterMain	
30					Activity: AutoVacuumMain	
32		root			Activity: LogicalLauncherMain	
54	postgres	root	pgAdmin 4 - DB:postgres	active		