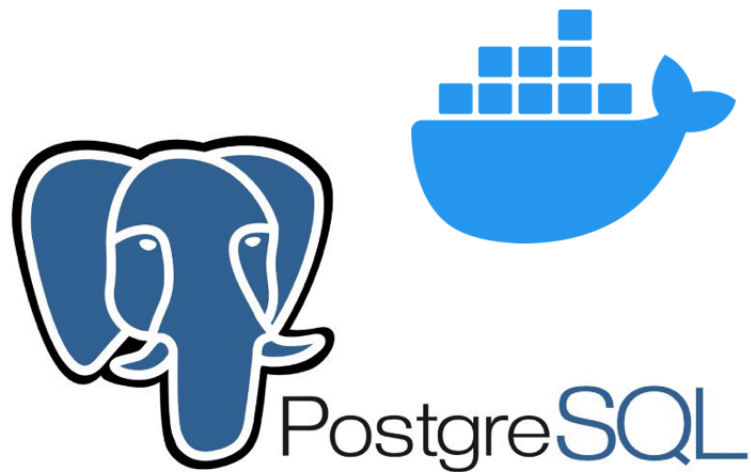


UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE CIENCIAS  
FUNDAMENTOS DE BASES DE DATOS 2026-II



---

**Práctica 01:**

*Bitácora Tapia*

---

PROFESOR

- Gerardo Avilés Rosas

AYUDANTE DE TEORÍA:

- Luis Enrique García Gómez

AYUDANTE DE LABORATORIO:

- Ricardo Badillo Macías

# 1. Sistema Operativo y Versión

- **Sistema Operativo:** Ubuntu 24.04.4 LTS
- **Kernel:** 6.17.0-14-generic
- **Arquitectura:** 64-bit
- **Entorno de escritorio:** GNOME 46
- **Procesador (CPU):** 12th Gen Intel Core i7-1255U × 12
- **Memoria RAM:** 16 GB

# 2. Distribución (Solo aplica a Linux)

- **Distribución:** Ubuntu 24.04.4 LTS (basada en Debian)

# 3. Versión de la instalación

En esta práctica se instalaron Docker y PostgreSQL en el sistema, y se utilizó pgAdmin4 como herramienta gráfica para administrar PostgreSQL.

- **Docker:** 29.2.1, build a5c7197
- **Docker Compose (plugin):** v5.0.2
- **PostgreSQL:** PostgreSQL 16 (paquete postgresql-16), versión **16.11-0ubuntu0.24.04.1**
- **pgAdmin4:** 9.12

# 4. Tiempo requerido

- **Tiempo total:** 40 minutos.
- **Observación:** La mayor parte del tiempo correspondió a descargas de paquetes e imágenes.

# 5. Explicación del paso a paso que realicé (con capturas)

## 5.1. Instalación y verificación de Docker

Inicialmente intenté utilizar los comandos del documento de la práctica; sin embargo, el sistema mostró advertencias relacionadas con `apt-key` (*deprecated*). Entonces seguí el procedimiento recomendado en la documentación oficial de Docker para Ubuntu, así pude completar la instalación sin errores.

1. Se verificó que el servicio de Docker estuviera activo mediante:

```
sudo systemctl status docker
```

2. Se ejecutó la imagen de prueba `hello-world` para comprobar el funcionamiento del motor:

```
sudo docker run hello-world
```

```
root@axel-Aspire16:~# sudo systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/usr/lib/systemd/system/docker.service; enabled; preset: enabled)
   Active: active (running) since Sat 2026-02-21 17:41:06 CST; 16s ago
     TriggeredBy: ● docker.socket
       Docs: https://docs.docker.com
      Main PID: 9673 (dockerd)
         Tasks: 16
        Memory: 28.8M (peak: 31.4M)
           CPU: 467ms
          CGroup: /system.slice/docker.service
                  └─9673 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

Feb 21 17:41:05 axel-Aspire16 dockerd[9673]: time="2026-02-21T17:41:05.647356755-06:00" level=info msg="Restoring containers: start."
Feb 21 17:41:05 axel-Aspire16 dockerd[9673]: time="2026-02-21T17:41:05.710497399-06:00" level=info msg="Deleting nftables IPv4 rules" error="exit status 1"
Feb 21 17:41:05 axel-Aspire16 dockerd[9673]: time="2026-02-21T17:41:05.722258105-06:00" level=info msg="Deleting nftables IPv6 rules" error="exit status 1"
Feb 21 17:41:06 axel-Aspire16 dockerd[9673]: time="2026-02-21T17:41:06.196738105-06:00" level=info msg="Loading containers: done."
Feb 21 17:41:06 axel-Aspire16 dockerd[9673]: time="2026-02-21T17:41:06.207457989-06:00" level=info msg="Docker daemon" commit=6bc6209 containerd-snapshotter=true storage-driver=overlays ve
Feb 21 17:41:06 axel-Aspire16 dockerd[9673]: time="2026-02-21T17:41:06.207786856-06:00" level=info msg="Initializing buildkit"
Feb 21 17:41:06 axel-Aspire16 dockerd[9673]: time="2026-02-21T17:41:06.229156084-06:00" level=info msg="Completed buildkit initialization"
Feb 21 17:41:06 axel-Aspire16 dockerd[9673]: time="2026-02-21T17:41:06.237465330-06:00" level=info msg="Daemon has completed initialization"
Feb 21 17:41:06 axel-Aspire16 dockerd[9673]: time="2026-02-21T17:41:06.237609640-06:00" level=info msg="API listen on /run/docker.sock"
Feb 21 17:41:06 axel-Aspire16 systemd[1]: Started docker.service - Docker Application Container Engine.

lines 1-22/22 (END)
```

Figura 1: Servicio docker activo (systemctl status).

```
Memory: 28.8M (peak: 31.4M)
CPU: 467ms
CGroup: /system.slice/docker.service
        └─9673 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

Feb 21 17:41:05 axel-Aspire16 dockerd[9673]: time="2026-02-21T17:41:05.647356755-06:00" level=info msg="Restoring containers: start."
Feb 21 17:41:05 axel-Aspire16 dockerd[9673]: time="2026-02-21T17:41:05.710497399-06:00" level=info msg="Deleting nftables IPv4 rules" error="exit status 1"
Feb 21 17:41:05 axel-Aspire16 dockerd[9673]: time="2026-02-21T17:41:05.722258105-06:00" level=info msg="Deleting nftables IPv6 rules" error="exit status 1"
Feb 21 17:41:06 axel-Aspire16 dockerd[9673]: time="2026-02-21T17:41:06.196738105-06:00" level=info msg="Loading containers: done."
Feb 21 17:41:06 axel-Aspire16 dockerd[9673]: time="2026-02-21T17:41:06.207457989-06:00" level=info msg="Docker daemon" commit=6bc6209 containerd-snapshotter=true storage-driver=overlays ve
Feb 21 17:41:06 axel-Aspire16 dockerd[9673]: time="2026-02-21T17:41:06.207786856-06:00" level=info msg="Initializing buildkit"
Feb 21 17:41:06 axel-Aspire16 dockerd[9673]: time="2026-02-21T17:41:06.229156084-06:00" level=info msg="Completed buildkit initialization"
Feb 21 17:41:06 axel-Aspire16 dockerd[9673]: time="2026-02-21T17:41:06.237465330-06:00" level=info msg="Daemon has completed initialization"
Feb 21 17:41:06 axel-Aspire16 dockerd[9673]: time="2026-02-21T17:41:06.237609640-06:00" level=info msg="API listen on /run/docker.sock"
Feb 21 17:41:06 axel-Aspire16 systemd[1]: Started docker.service - Docker Application Container Engine.

root@axel-Aspire16:~# sudo docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
17eec7bbc9d7: Pull complete
ea52d2000f90: Download complete
Digest: sha256:ef54e839ef541993b4e87f25e752f7cf4238fa55f017957c2eb44077083d7a6a
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
   (amd64)
3. The Docker daemon created a new container from that image which runs the
   executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it
   to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/

root@axel-Aspire16:~#
```

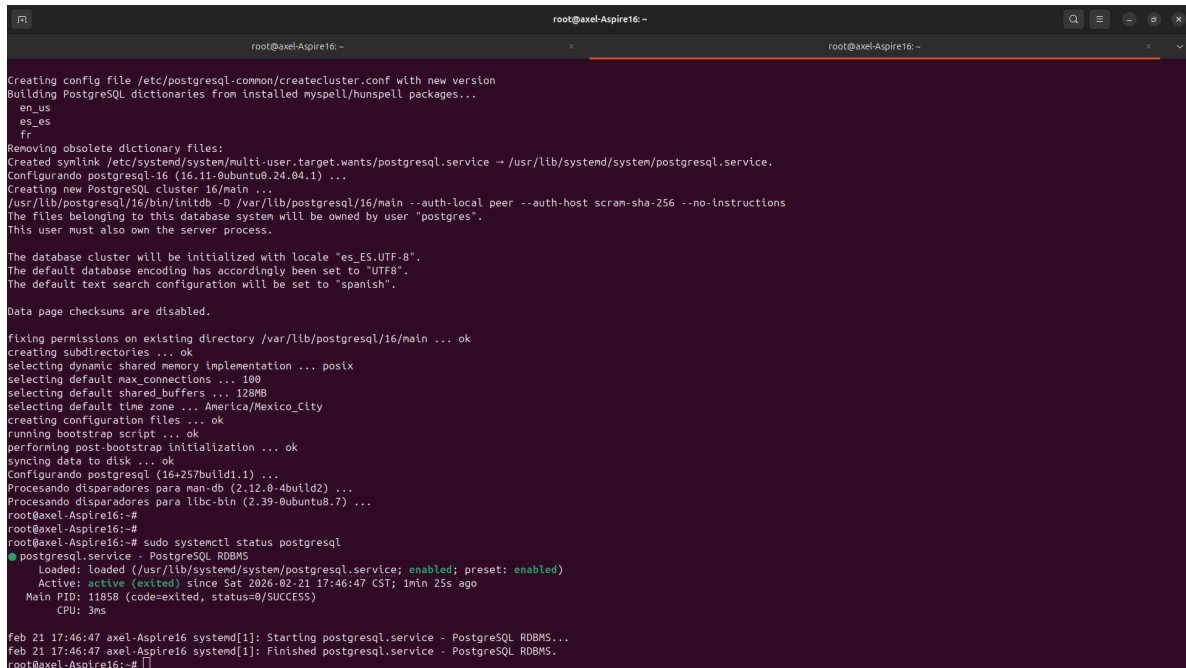
Figura 2: Ejecución exitosa de docker run hello-world.

## 5.2. Instalación y verificación de PostgreSQL

Para PostgreSQL seguí la documentación oficial de PostgreSQL para Ubuntu, logrando instalar la versión provista por los repositorios de Ubuntu 24.04.

1. Tras finalizar la instalación, se verificó el servicio:

```
sudo systemctl status postgresql
```

A terminal window on a system named 'axel-Aspire16' showing the output of the 'sudo systemctl status postgresql' command. The output details the creation of configuration files, building of dictionaries, removal of obsolete files, creation of a new PostgreSQL cluster '16/main', setting of locale to 'es\_ES.UTF-8', and the successful start of the 'postgresql.service'. The service is shown as 'active (exited)' since February 21, 2026, at 17:46:47 CST.

```
root@axel-Aspire16:~# sudo systemctl status postgresql
Creating config file /etc/postgresql-common/createcluster.conf with new version
Building PostgreSQL dictionaries from installed myspell/hunspell packages...
  en_us
  es_es
  fr-
Removing obsolete dictionary files:
Created symlink /etc/systemd/system/multi-user.target.wants/postgresql.service → /usr/lib/systemd/system/postgresql.service.
Configurando postgresql-16 (16.11-0ubuntu0.24.04.1) ...
Creating new PostgreSQL cluster 16/main ...
/usr/lib/postgresql/16/bin/initdb -D /var/lib/postgresql/16/main --auth-local peer --auth-host scram-sha-256 --no-instructions
The files belonging to this database system will be owned by user "postgres".
This user must also own the server process.

The database cluster will be initialized with locale "es_ES.UTF-8".
The default database encoding has accordingly been set to "UTF8".
The default text search configuration will be set to "spanish".

Data page checksums are disabled.

fixing permissions on existing directory /var/lib/postgresql/16/main ... ok
creating subdirectories ... ok
selecting dynamic shared memory implementation ... posix
selecting default max_connections ... 100
selecting default shared_buffers ... 128MB
selecting default time zone ... America/Mexico_City
creating configuration files ... ok
running bootstrap script ... ok
performing post-bootstrap initialization ... ok
syncing data to disk ... ok
Configurando postgresql (16.257build1) ...
Procesando disparadores para man-db (2.12.0-4build2) ...
Procesando disparadores para libc-bin (2.39-0ubuntu0.7) ...
root@axel-Aspire16:~#
root@axel-Aspire16:~#
root@axel-Aspire16:~# sudo systemctl status postgresql
● postgresql.service - PostgreSQL RDBMS
   Loaded: loaded (/usr/lib/systemd/system/postgresql.service; enabled; preset: enabled)
   Active: active (exited) since Sat 2026-02-21 17:46:47 CST; 1min 25s ago
     Main PID: 11858 (code=exited, status=0/SUCCESS)
        CPU: 3ms

Feb 21 17:46:47 axel-Aspire16 systemd[1]: Starting postgresql.service - PostgreSQL RDBMS...
Feb 21 17:46:47 axel-Aspire16 systemd[1]: Finished postgresql.service - PostgreSQL RDBMS.
root@axel-Aspire16:~#
```

Figura 3: Servicio postgresql verificado mediante systemctl.

## 5.3. Instalación y ejecución de pgAdmin4

Para administrar PostgreSQL mediante interfaz gráfica se instaló y ejecutó pgAdmin4 (documentación oficial).

1. Se completó la instalación de pgAdmin4 (paquetes).
2. Se ejecutó pgAdmin4 y se verificó su apertura correcta.

```
root@axel-Aspire16 ~  
Configurando libaprutil1-ldap:amd64 (1.6.3-1.1ubuntu7) ...  
Configurando libaprutil1-dbd-sqlite3:amd64 (1.6.3-1.1ubuntu7) ...  
Configurando apache2-utils (2.4.58-1ubuntu8.10) ...  
Configurando apache2-bin (2.4.58-1ubuntu8.10) ...  
Configurando apache2 (2.4.58-1ubuntu8.10) ...  
Enabling module mpm_event.  
Enabling module authz_core.  
Enabling module authz_host.  
Enabling module authn_core.  
Enabling module auth_basic.  
Enabling module access_compat.  
Enabling module authn_file.  
Enabling module authz_user.  
Enabling module alias.  
Enabling module dir.  
Enabling module autoindex.  
Enabling module env.  
Enabling module mime.  
Enabling module negotiation.  
Enabling module setenvif.  
Enabling module filter.  
Enabling module deflate.  
Enabling module status.  
Enabling module reqtimeout.  
Enabling conf charset.  
Enabling conf localized-error-pages.  
Enabling conf other-vhosts-access-log.  
Enabling conf security.  
Enabling conf serve-cgi-bin.  
Enabling site 000-default.  
Created symlink /etc/systemd/system/multi-user.target.wants/apache2.service → /usr/lib/systemd/system/apache2.service.  
Created symlink /etc/systemd/system/multi-user.target.wants/apache-htcacheclean.service → /usr/lib/systemd/system/apache-htcacheclean.service.  
Configurando libapache2-mod-wsgi-py3 (5.0.0-1build2) ...  
apache2_invoke: Enable module wsgi  
Configurando pgadmin4-web (9.12) ...  
Configurando pgadmin4 (9.12) ...  
Procesando disparadores para desktop-file-utils (0.27-2build1) ...  
Procesando disparadores para hicolor-icon-theme (0.17-2) ...  
Procesando disparadores para gnome-menus (3.36.0-1.1ubuntu3) ...  
Procesando disparadores para libc-bin (2.39-0ubuntu8.7) ...  
Procesando disparadores para ufw (0.36.2-6) ...  
Procesando disparadores para man-db (2.12.0-4build2) ...  
root@axel-Aspire16:~#
```

Figura 4: pgAdmin4 ejecutándose correctamente.

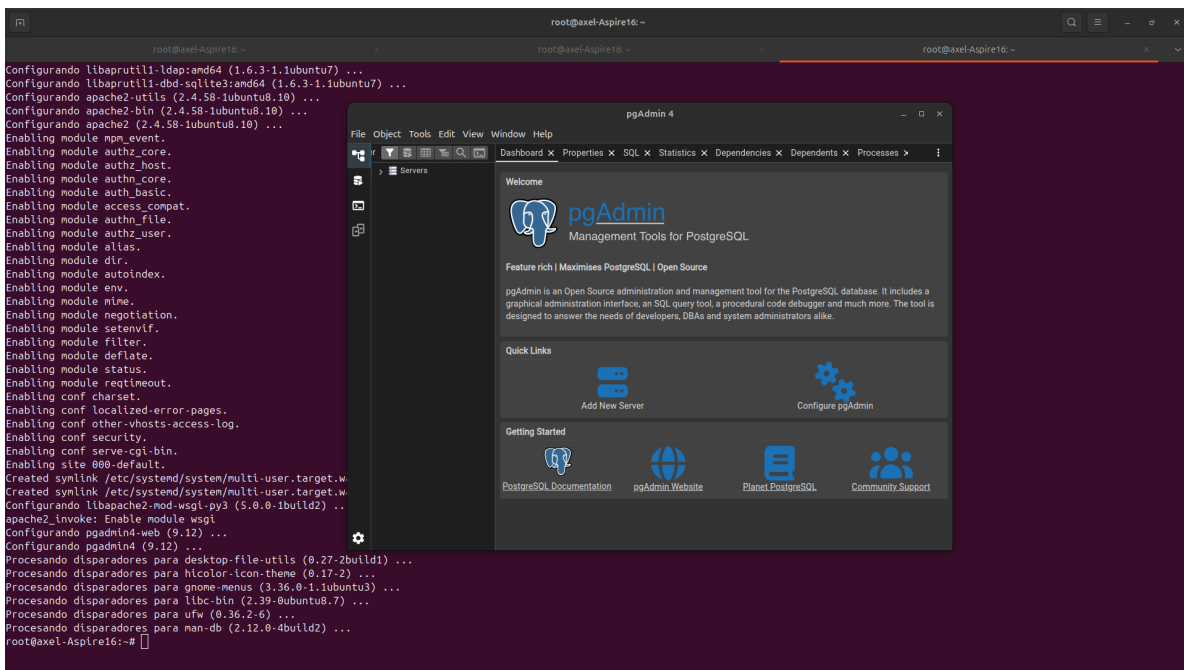


Figura 5: Interfaz de pgAdmin4 (pantalla de bienvenida).

## 6. Comentarios / problemas encontrados

- Al intentar seguir los comandos del PDF de la práctica, el sistema mostró advertencias relacionadas con `apt-key` (*deprecated*), por lo que consideré que el procedimiento del documento podía estar desactualizado para Ubuntu 24.04.
- Para evitar errores por comandos obsoletos y asegurar una instalación correcta, decidí utilizar las guías oficiales de **Docker**, **PostgreSQL** y **pgAdmin4**. Con ello, la instalación se completó sin problemas.
- La ejecución y verificación fueron correctas: Docker quedó activo y se validó con `hello-world`; PostgreSQL quedó instalado y su servicio se verificó con `systemctl`; pgAdmin4 abrió correctamente y fue posible visualizar la interfaz.
- El proceso tomó un tiempo relativamente largo (40 minutos) principalmente por las descargas, pero fuera de eso la instalación fue sencilla y no se presentaron errores importantes.