

Instalar Apache Web Server en WSL Debian

Asegure de tener actualizado el sistema antes de instalar:

`sudo apt update && sudo apt upgrade -y`

Instale Apache usando el comando:

`sudo apt install apache2 -y`

```
rodrigo@RodrigoTR:/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb$ sudo apt install apache2 -y
Reading package lists... Done
Building dependency tree... Done
```

Apache no se inicia en WSL de manera automática al iniciar la sesión habrá que hacerlo cada vez que iniciemos sesión.

`sudo service apache2 start`

```
rodrigo@RodrigoTR:/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb$ sudo service apache2 start
rodrigo@RodrigoTR:/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb$
```

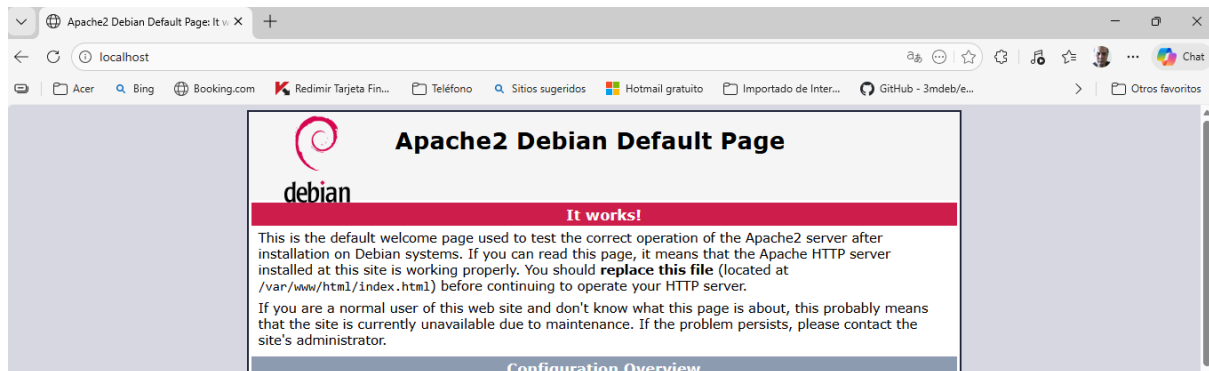
Ahora puede verificar que la instalación se haya realizado correctamente:

`sudo service apache2 status`

```
rodrigo@RodrigoTR:/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb$ sudo service apache2 status
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; preset: enabled)
   ➔ Active: active (running) since Thu 2026-01-08 23:21:37 CST; 4min 32s ago
     Docs: https://httpd.apache.org/docs/2.4/
    Main PID: 7591 (apache2)
      Tasks: 55 (limit: 4597)
     Memory: 21.1M
    CGroup: /system.slice/apache2.service
           └─7591 /usr/sbin/apache2 -k start
             └─7592 /usr/sbin/apache2 -k start
               └─7593 /usr/sbin/apache2 -k start

Jan 08 23:21:37 RodrigoTR systemd[1]: Starting apache2.service - The Apache HTTP Server...
Jan 08 23:21:37 RodrigoTR systemd[1]: Started apache2.service - The Apache HTTP Server.
```

En el Navegador en la dirección <http://localhost>, verifique que aparezca la página de bienvenida:



En la carpeta `/var/www/html/`, es donde se guardaran los archivos de nuestra página web, ahí el punto de entrada será un archivo llamado `index.html` o `index.php` (nota si al intentar escribir archivos en la carpeta recibe un mensaje de error tendrá que modificar los permisos usando `chmod`).

```
rodrigo@RodrigoTR:/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb$ ls /var/www/html
index.html
```

Algunos comandos útiles para administrar el servicio de Apache se listan a continuación:

Iniciar `sudo service apache2 start`

Detener `sudo service apache2 stop`

Reiniciar `sudo service apache2 restart`

Instalar MariaDB en WSL Debian

Asegure de tener actualizado su sistema WSL Debian y proceda a instalar el servicio de MariaDB:

```
sudo apt update
```

```
sudo apt install mariadb-server -y
```

```
rodrigo@RodrigoTR:/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb$ sudo apt update
sudo apt install mariadb-server -y
Hit:1 http://deb.debian.org/debian bookworm InRelease
Hit:2 http://security.debian.org/debian-security bookworm-security InRelease
Hit:3 http://ftp.debian.org/debian bookworm-backports InRelease
Hit:4 http://deb.debian.org/debian bookworm-updates InRelease
Hit:5 https://download.docker.com/linux/debian bookworm InRelease
```

Deberá iniciar el servicio cada que entre en la sesión de WSL, y verificar que está ejecutando correctamente:

```
sudo service mariadb start
```

```
sudo service mariadb status
```

```
rodrigo@RodrigoTR:/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb$ sudo service mariadb start
rodrigo@RodrigoTR:/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb$ sudo service mariadb status
● mariadb.service - MariaDB 10.11.14 database server
   Loaded: loaded (/lib/systemd/system/mariadb.service; enabled; preset: enabled)
   Active: active (running) since Thu 2026-01-08 23:38:49 CST; 2min 29s ago
     Docs: man:mariadb(8)
           https://mariadb.com/kb/en/library/systemd/
   Main PID: 9390 (mariabdd)
   Status: "Taking your SQL requests now..."
     Tasks: 9 (limit: 30342)
    Memory: 114.2M
   CGroup: /system.slice/mariadb.service
           └─9390 /usr/sbin/mariabdd

Jan 08 23:38:48 RodrigoTR mariabdd[9390]: 2026-01-08 23:38:48 0 [Note] Plugin 'FEEDBACK' is disabled.
Jan 08 23:38:48 RodrigoTR mariabdd[9390]: 2026-01-08 23:38:48 0 [Note] InnoDB: Buffer pool(s) load completed at 260108 23:38:48
Jan 08 23:38:49 RodrigoTR mariabdd[9390]: 2026-01-08 23:38:49 0 [Warning] You need to use --log-bin to make --expire-logs-days or --binlog-expire-logs-seconds work.
Jan 08 23:38:49 RodrigoTR mariabdd[9390]: 2026-01-08 23:38:49 0 [Note] Server socket created on IP: '127.0.0.1', port: '3306'.
Jan 08 23:38:49 RodrigoTR mariabdd[9390]: 2026-01-08 23:38:49 0 [Note] /usr/sbin/mariabdd: ready for connections.
Jan 08 23:38:49 RodrigoTR mariabdd[9390]: Version: '10.11.14-MariaDB-0+deb12u2' socket: '/run/mysqld/mysqld.sock' port: 3306 Debian 12
Jan 08 23:38:49 RodrigoTR systemd[1]: Started mariadb.service - MariaDB 10.11.14 database server.
Jan 08 23:38:49 RodrigoTR /etc/mysql/debian-start[9406]: Upgrading MySQL tables if necessary.
Jan 08 23:38:49 RodrigoTR /etc/mysql/debian-start[9417]: Checking for insecure root accounts.
Jan 08 23:38:49 RodrigoTR /etc/mysql/debian-start[9421]: Triggering myisam-recover for all MyISAM tables and aria-recover for all Aria tables
```

Proceda a configurar la seguridad (muy importante):

```
sudo mysql_secure_installation
```

```
rodrigo@RodrigoTR:/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb$ sudo mysql_secure_installation

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!
```

Se sugiere responder lo siguiente a las preguntas que se realicen durante la configuración:

Enter current password for root: Presiona *Enter* (está vacío por defecto). para este ejemplo usaremos 1234

```
In order to log into MariaDB to secure it, we'll need the current
password for the root user. If you've just installed MariaDB, and
haven't set the root password yet, you should just press enter here.
```

```
Enter current password for root (enter for none): 
```

Switch to unix_socket authentication? Responde *n* (para usar contraseñas tradicionales en este ejercicio).

```
Switch to unix_socket authentication [Y/n] n
... skipping.
```

Change the root password? Responde **y** y define una contraseña que recuerdes bien.

```
Change the root password? [Y/n] y
New password:
Re-enter new password:
Warning: World-writable config file '/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb/.my.cnf.9783' is ignored
Password updated successfully!
Reloading privilege tables..
Warning: World-writable config file '/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb/.my.cnf.9783' is ignored
... Success!
```

Remove anonymous users? Responde **y**.

```
Remove anonymous users? [Y/n] y
Warning: World-writable config file '/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb/.my.cnf.9783' is ignored
... Success!
```

Normally, root should only be allowed to connect from 'localhost'. This ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? Responde **y**.

```
Disallow root login remotely? [Y/n] y
Warning: World-writable config file '/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb/.my.cnf.9783' is ignored
... Success!
```

By default, MariaDB comes with a database named 'test' that anyone can access. This is also intended only for testing, and should be removed before moving into a production environment.

Remove test database? Responde **y**.

```
Remove test database and access to it? [Y/n] y
- Dropping test database...
Warning: World-writable config file '/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb/.my.cnf.9783' is ignored
... Success!
- Removing privileges on test database...
Warning: World-writable config file '/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb/.my.cnf.9783' is ignored
... Success!
```

Reloading the privilege tables will ensure that all changes made so far will take effect immediately.

Reload privilege tables now? Responde **y**.

```
Reload privilege tables now? [Y/n] y
Warning: World-writable config file '/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb/.my.cnf.9783' is ignored
... Success!
```

Cleaning up...

All done! If you've completed all of the above steps, your MariaDB installation should now be secure.

Thanks for using MariaDB!

Verifique el acceso a la base de datos:

sudo mariadb -u root -p

```
rodrigo@RodrigoTR:/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb$ sudo mariadb -u root -p
Enter password:
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 39
Server version: 10.11.14-MariaDB-0+deb12u2 Debian 12

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

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

MariaDB [(none)]> █
```

Para salir teclee *quit*.

Notas para tu documentación

En una **arquitectura monolítica**, MariaDB guarda físicamente los datos en el sistema de archivos de Debian. Es vital conocer estas rutas:

- Directorio de datos: `/var/lib/mysql` (Aquí es donde "viven" tus bases de datos).
- Archivo de configuración: `/etc/mysql/mariadb.conf.d/50-server.cnf`.
- Log de errores: `/var/log/mysql/error.log`.

En este entorno WSL, si borras Debian, **pierdes tus bases de datos**.

Instalación de PHP en WSL Debian

Actualice su sistema, para instalar utilice el siguiente comando:

```
sudo apt install php libapache2-mod-php php-mysql -y
```

```
rodrigo@RodrigoTR:/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb$ sudo apt install php libapache2-mod-php php-mysql -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libapache2-mod-php8.2 php-common php8.2 php8.2-cli php8.2-common php8.2-mysql php8.2-opcache php8.2-readline
```

Reinicie PHP para terminar la instalación:

```
sudo service apache2 restart
```

```
rodrigo@RodrigoTR:/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb$ sudo service apache2 restart
```

Para verificar la instalación se ha realizado correctamente, cree el siguiente archivo de prueba (el cual se enviará a la carpeta del servicios Web de Apache):

```
echo "<?php phpinfo(); ?>" | sudo tee /var/www/html/info.php
```

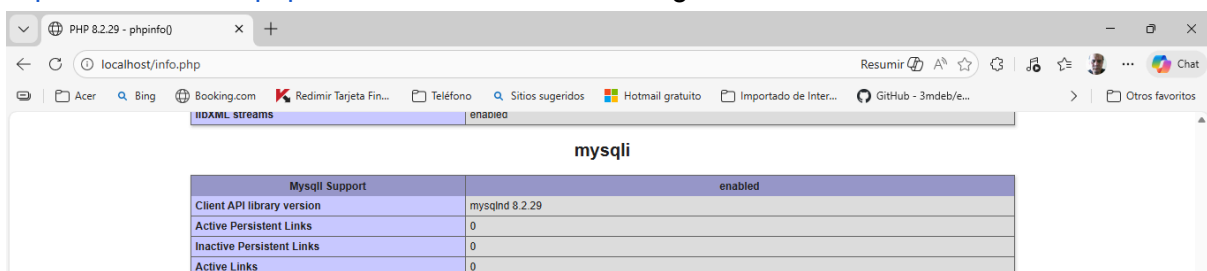
```
rodrigo@RodrigoTR:/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb$ echo "<?php phpinfo(); ?>" | sudo tee /var/www/html/info.php
rodrigo@RodrigoTR:/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb$ ls /var/www/html
index.html  info.php
```

Si tiene problemas para escribir en la carpeta utilice lo siguiente:

```
sudo chown -R $USER:$USER /var/www/html
```

Adicionalmente en el navegador verifique la instalación en la página web

<http://localhost/info.php> deberá ver la información siguiente:



mysql	
Mysql Support	enabled
Client API library version	mysqlnd 8.2.29
Active Persistent Links	0
Inactive Persistent Links	0
Active Links	0

PHP 8.2.29 - phpinfo()

localhost/info.php

Resumir

A

Indicando que PHP está instalado correctamente y además puede conectar con MariaDB.

Instalación de PHPMyAdmin en WSL Debian

Tener un gestor de base de datos con interfaz gráfica es de mucha utilidad antes de instalar asegure tener el sistema actualizado:

```
sudo apt update
```

```
sudo apt install phpmyadmin -y
```

```
rodrigo@RodrigoTR:/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb$ sudo apt install phpmyadmin -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
```

En las siguientes opciones de configuración se recomienda lo siguiente:

1. Web server to reconfigure automatically: Selecciona **apache2** (usa la tecla **Espacio** para marcar el asterisco **[*]** y luego **Enter**).

Configuring phpmyadmin

Please choose the web server that should be automatically configured to run phpMyAdmin.

Web server to reconfigure automatically:

[*] apache2

[] lighttpd

<Ok>

2. Configure database for phpmyadmin with dbconfig-common? Selecciona **Yes**.

Configuring phpmyadmin

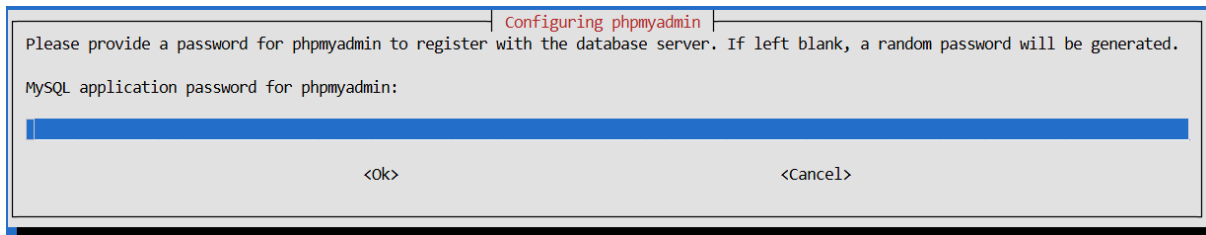
The phpmyadmin package must have a database installed and configured before it can be used. This can be optionally handled with dbconfig-common.
If you are an advanced database administrator and know that you want to perform this configuration manually, or if your database has already been installed and configured, you should refuse this option. Details on what needs to be done should most likely be provided in /usr/share/doc/phpmyadmin.
Otherwise, you should probably choose this option.

Configure database for phpmyadmin with dbconfig-common?

<Yes>

<No>

3. MySQL application password for phpmyadmin: Define una contraseña para el uso interno de la herramienta (o déjala en blanco para que genere una aleatoria, ya que no la usarás frecuentemente).



Habilite el acceso de phpmyadmin con los siguientes comandos:

```
sudo phpenmod mysqli
```

```
sudo ln -s /etc/phpmyadmin/apache.conf /etc/apache2/conf-available/phpmyadmin.conf
```

```
sudo a2enconf phpmyadmin
```

```
sudo service apache2 restart
```

```
rodrigo@RodrigoTR:/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb$ sudo phpenmod mysqli
rodrigo@RodrigoTR:/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb$ sudo ln -s /etc/phpmyadmin/apache.conf /etc/apache2/conf-available/phpmyadmin.conf
ln: failed to create symbolic link '/etc/apache2/conf-available/phpmyadmin.conf': File exists
rodrigo@RodrigoTR:/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb$ sudo a2enconf phpmyadmin
Conf phpmyadmin already enabled
rodrigo@RodrigoTR:/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb$ sudo service apache2 restart
rodrigo@RodrigoTR:/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb$
```

Configurar el acceso a MariaDB

```
sudo mariadb -u root -p
```

Y ejecute los siguientes comandos:

```
ALTER USER 'root'@'localhost' IDENTIFIED VIA mysql_native_password USING
PASSWORD('tu_password');
```

```
MariaDB [(none)]> ALTER USER 'root'@'localhost' IDENTIFIED VIA mysql_native_password USING PASSWORD('tu_password');
```

No olvide agregar su password que usarás para ingresar.

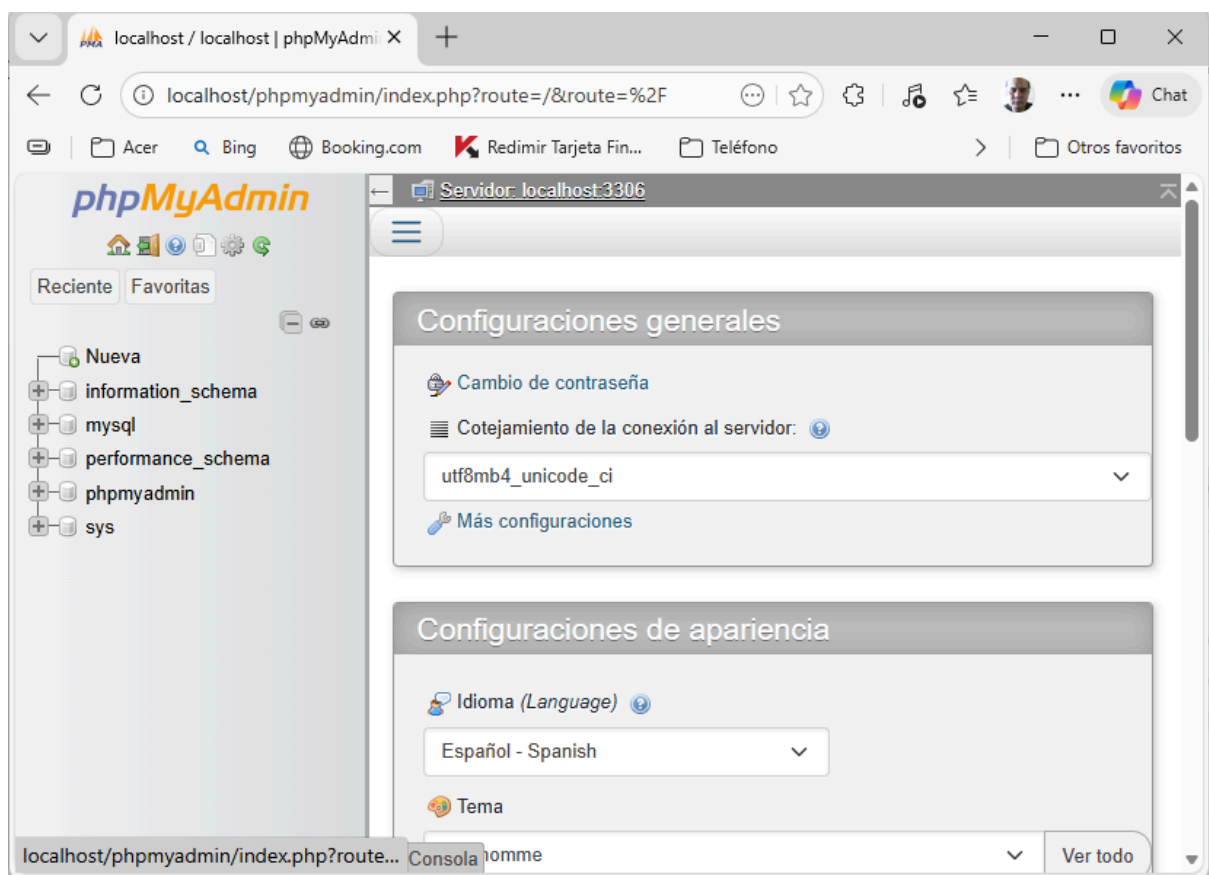
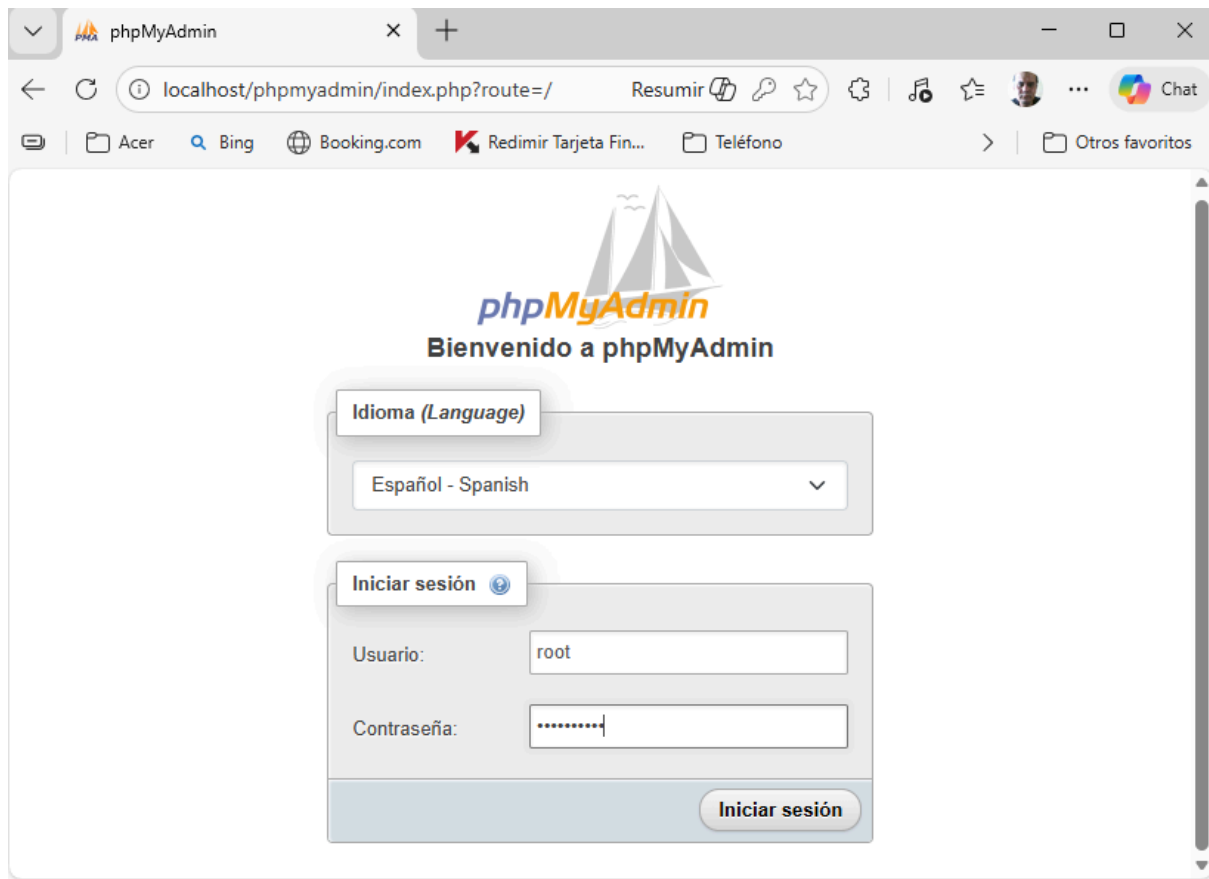
```
FLUSH PRIVILEGES;
```

```
MariaDB [(none)]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.049 sec)
```

EXIT;

```
MariaDB [(none)]> EXIT;
Bye
rodrigo@RodrigoTR:/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb$
```

Esto para habilitar el ingreso de phpmyadmin como root a MariaDB. Por último verifique en el navegador en la página <http://localhost/phpmyadmin> que se pueda ingresar con el usuario *root* y el password.



Listo, ya se ha configurado el gestor de base de datos para MariaDB.

Cuando se inicie la sesión de WSL no es requerido iniciar el servicio de PHPMyAdmin, ya que se inicia automáticamente con Apache, lo que si es requerido es asegurar que la base de datos está en ejecución, utilice:

```
sudo service apache2 start
```

```
sudo service mariadb start
```

Cada inicio de sesión de WSL para asegurar que todo esté en ejecución.

Desinstalar MariaDb de WSL Debian

Detener el servicio de MariaDB

```
sudo service mariadb stop
```

Desinstalar paquetes y dependencias con “purge” en vez de “remove”

```
sudo apt purge mariadb-server mariadb-client mariadb-common -y
```

```
rodrigo@RodrigoTR:/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb$ sudo service mariadb stop
[sudo] password for rodrigo:
rodrigo@RodrigoTR:/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb$ sudo apt purge mariadb-server mariadb-client mariadb-common -y
```

Aparecerá un mensaje que nos preguntará si deseamos mantener las bases de datos, en este caso elegimos que no ya que deseamos una instalación limpia.

Eliminamos las dependencias que ya no son necesarias:

```
sudo apt autoremove -y
```

```
sudo apt autoclean
```

```
rodrigo@RodrigoTR:/mnt/c/Users/sambo/Documents/Programacion/GitHub/DiseñoWeb$ sudo apt autoremove -y
sudo apt autoclean
Reading package lists... Done
Building dependency tree... Done
```

Limpieza manual de archivos residuales (importante limpiar):

```
sudo rm -rf /var/lib/mysql
```

```
sudo rm -rf /etc/mysql
```

Con esto habremos eliminado la instalación de MariaDb y estaremos listos para una instalación fresca y limpia.