

**FACULTAD DE INGENIERIA Y CIENCIAS APLICADAS**



**Asignatura:** Administración de Base de Datos

**Autores:** Erick Acosta, Xavier Logroño, Jonathan Paredes

**Tema:** Proyecto Integrador

**Fecha:** 28/05/2025

## 1. Contexto del Problema

El Broker de Seguros Regional opera en varios países de Latinoamérica, ofreciendo servicios que conectan a clientes con aseguradoras. Debido a que sus operaciones están distribuidas geográficamente, enfrentan el desafío de integrar datos críticos que provienen de diferentes sistemas especializados:

- Un sistema financiero basado en Oracle DB para gestionar pólizas y pagos.
- Un sistema CRM en SQL Server para la atención y administración de los contactos con los clientes.
- Un sistema de atención de siniestros que utiliza MariaDB para procesar reclamos.

Actualmente, estos datos están dispersos en sistemas que no se comunican entre sí, lo que genera duplicidad, errores y dificulta obtener un análisis consolidado y estratégico. Además, la elaboración de reportes gerenciales y operativos es lenta, lo que afecta la toma de decisiones y la calidad del servicio al cliente. El reto consiste en diseñar una arquitectura que permita a cada área trabajar con su sistema especializado, pero que al mismo tiempo logre una integración fluida entre ellos. Esta solución debe garantizar que los datos estén disponibles, sean consistentes y trazables, todo dentro de una plataforma moderna, escalable y capaz de manejar esta diversidad tecnológica.

La falta de integración ha generado cuellos de botella en la atención al cliente, demoras en la validación de siniestros y dificultades para auditar transacciones entre departamentos. Esto no solo limita la eficiencia operativa, sino que también pone en riesgo la confianza de los clientes y la competitividad del negocio. Ante esta realidad, se vuelve imprescindible implementar una plataforma de base de datos heterogénea interconectada que permita a los distintos sistemas colaborar de manera coordinada, automatizando flujos de información críticos y reduciendo el margen de error humano.

## 2. Solución Implementada

Se implementa una plataforma de base de datos heterogénea interconectada, compuesta por tres SGBD distintos desplegados en una máquina virtual con Linux. Esta plataforma permitirá realizar integraciones entre bases de datos mediante procesos automáticos desarrollados en Python.

### Tecnologías Utilizadas:

- Oracle Database 21c
- Microsoft SQL Server 2019
- MariaDB 10.5
- Python 3.10 + librerías cx\_Oracle, pyodbc, mysql-connector-python

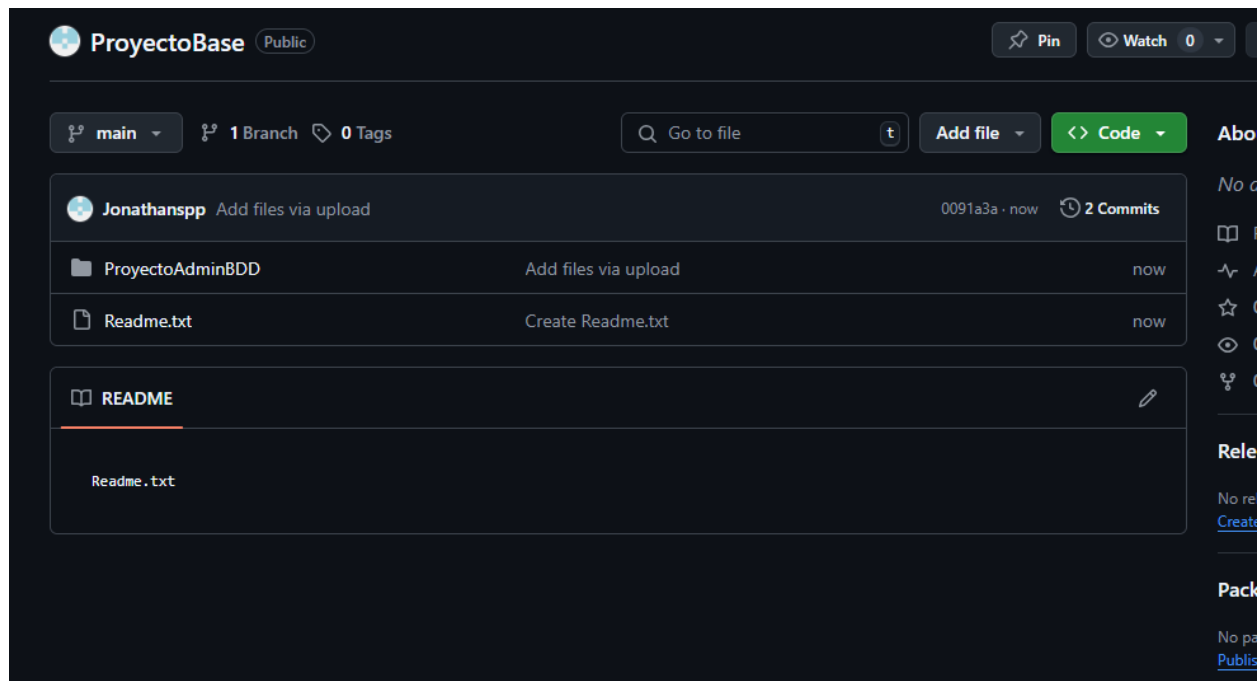
## Buenas Prácticas implementadas:

- Uso de puertos no estándar por seguridad
- Creación de usuarios/esquemas separados para cada SGDB
- Ubicación de binarios conforme a documentación oficial
- Creación de Jobs programados para sincronización

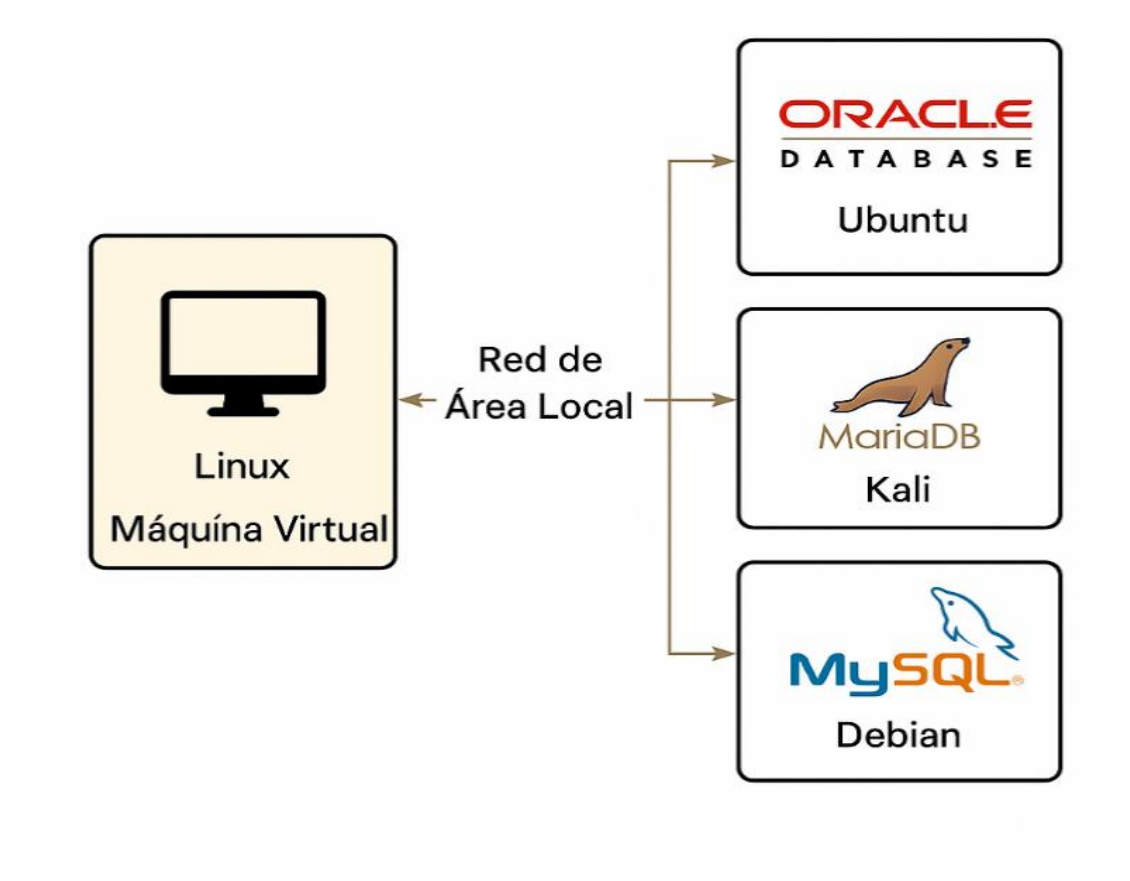
### 3. Instrucciones para acceder al repositorio GitHub

Link para acceder al repositorio de github

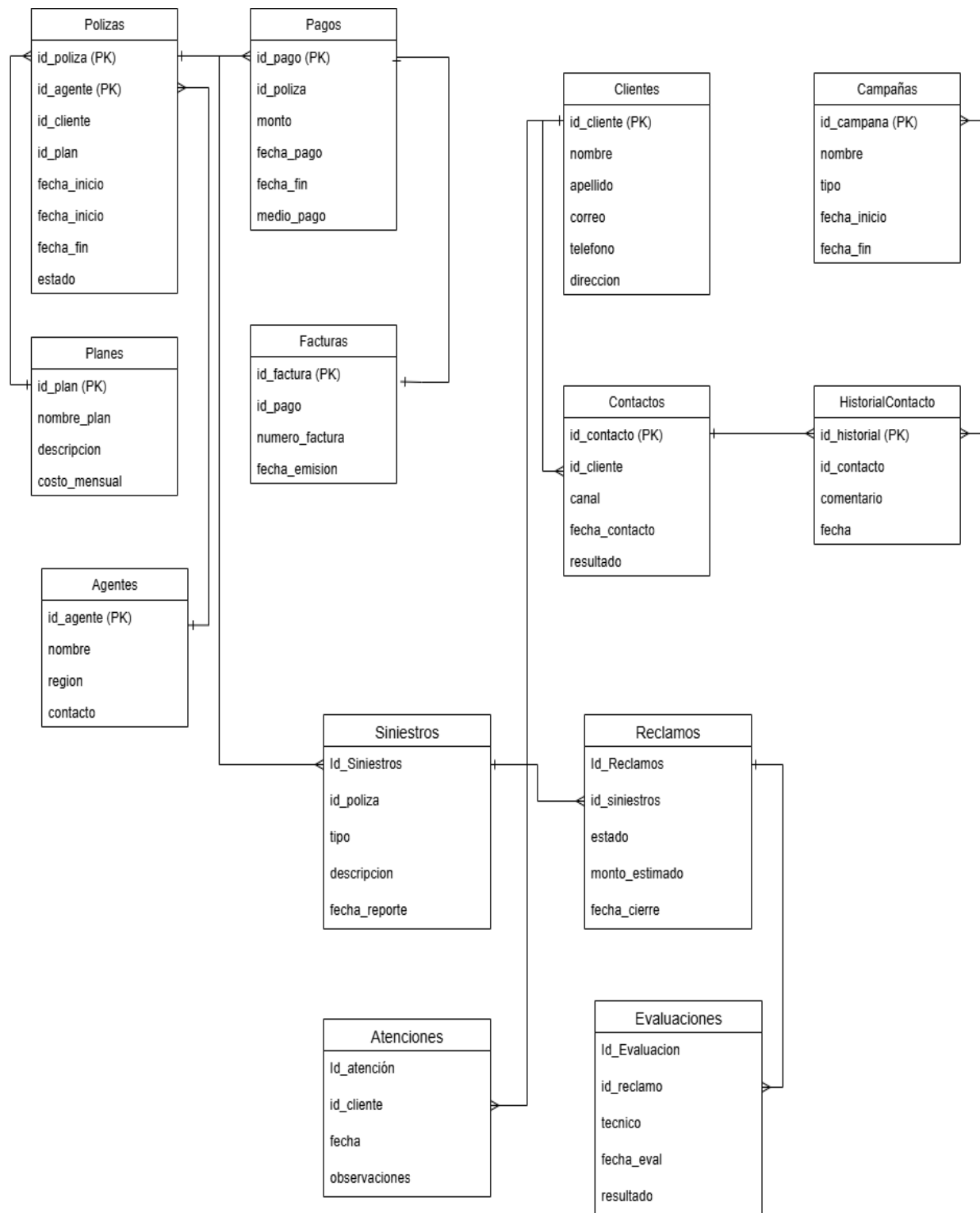
<https://github.com/Jonathanspp/ProyectoBase.git>



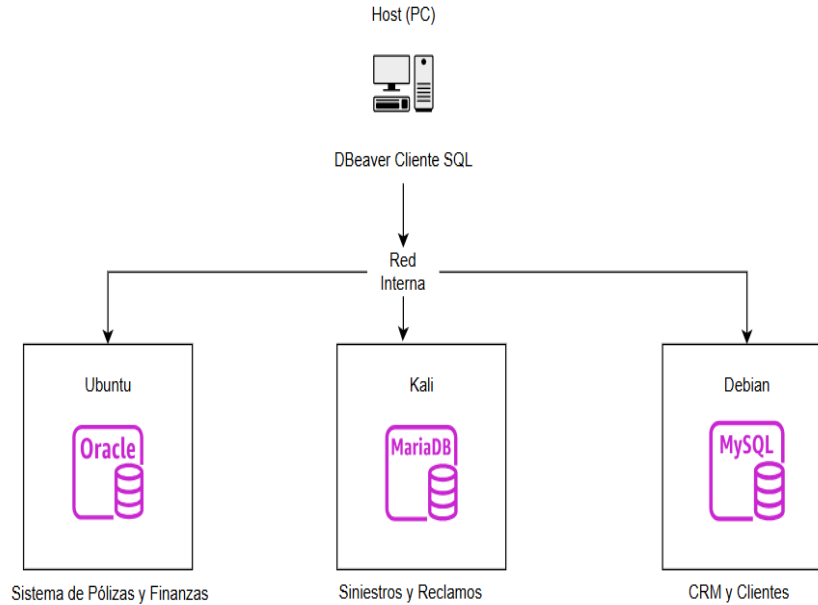
#### 4. Diagrama de Despliegue de la Plataforma



#### 5. Diagramas Físicos de cada Base de Datos.

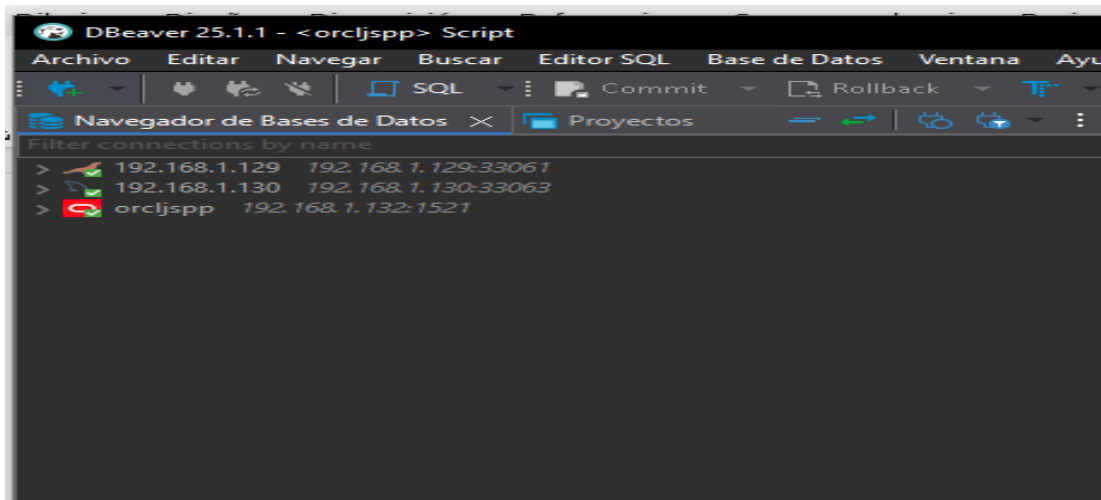


## 6. Diagrama de Despliegue de la Capa de Interconexión

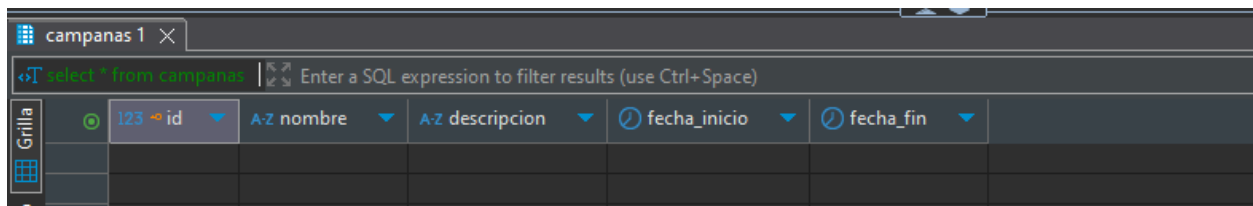
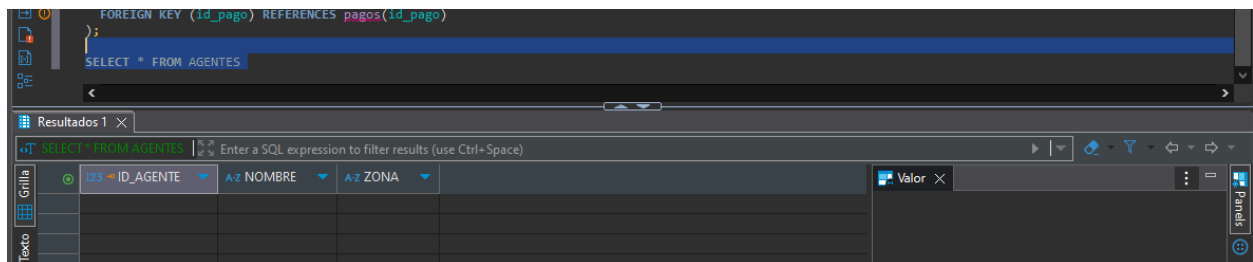


El diagrama muestra una plataforma de base de datos heterogénea interconectada compuesta por tres sistemas operativos distintos (Ubuntu, Kali y Debian), cada uno ejecutando un motor de base de datos diferente (Oracle, MariaDB y MySQL respectivamente), los cuales están conectados mediante una red interna. Estas bases de datos representan distintos módulos de un sistema empresarial: pólizas y finanzas, siniestros y reclamos, y CRM con clientes. La gestión centralizada se realiza desde un equipo host que utiliza DBeaver como cliente SQL gráfico, permitiendo administrar y consultar las distintas bases de datos desde un solo punto. Esta arquitectura facilita la interoperabilidad entre plataformas distintas, manteniendo una organización modular y eficiente.

## 7. Evidencias del correcto funcionamiento de la plataforma de base de datos heterogénea interconectada.



## Comprobación de la creación correcta de la tabla en Oracle, MariaDB, MySQL



## 8. Informe de cumplimiento de los siguientes criterios de desempeño

| <b>Criterio</b>                               | <b>% Cumplimiento</b> | <b>Evidencia(s)</b>   |
|---|-----------------------|---|
| <b>Definición de los Objetivos de Trabajo</b> | 100%                  | Se planteó como objetivo implementar una plataforma de base de datos heterogénea interconectada compuesta por Oracle (pólizas y finanzas), SQL Server (CRM) y MariaDB (siniestros), integrados mediante scripts automáticos en Python. Los objetivos específicos incluyen el despliegue correcto de los SGBD, sincronización entre sistemas, generación de datos sintéticos y documentación formal. |
| <b>Definición de Cronograma</b>               | 100%                  | Se elaboró un cronograma de cinco semanas con actividades definidas: instalación de la VM, configuración de SGBD, carga de datos, desarrollo de la interconexión y documentación.   |
| <b>Definición de Roles</b>                    | 100%                  | Cada miembro asumió un rol específico: Coordinador General, DBA Oracle, DBA SQL Server, y DBA MariaDB.  |
| <b>Asignación de Roles</b>                    | 100%                  | Los roles fueron asignados por consenso y todos los integrantes aceptaron sus funciones, comprometiéndose a cumplir con las tareas asignadas.   |
| <b>Asignación de Responsabilidades</b>        | 100%                  | Se definieron tareas específicas por rol: creación de esquemas y tablas, configuración de usuarios, generación de datos, desarrollo de scripts en Python y documentación del proyecto.  |



|   |      |   |
|---|------|---|
| <b>Cronograma de Reuniones de Trabajo</b>       | 100% | Se realizaron reuniones semanales por Discord y seguimiento diario por WhatsApp.  |
| <b>Cumplimiento de los Objetivos de Trabajo</b> | 100% | Todos los objetivos fueron alcanzados. Se instalaron los 3 SGBD correctamente, se generaron datos con Faker, se desarrollaron procesos automáticos de integración con Python y se crearon diagramas y documentación técnica. El repositorio en GitHub y el documento PDF consolidan la entrega. |

## 9. ANEXOS EVIDENCIAS

### Cronograma de Evidencias

| Semana   | Fecha         | Actividad   |
|----------|---------------|---|
| Semana 1 | 02 – 6junio   | Instalación de la máquina virtual con Linux y configuración inicial.    |
| Semana 2 | 10 – 14 junio | Instalación de Oracle, SQL Server y MariaDB en puertos no estándar.     |
| Semana 3 | 17 – 21 junio | Creación de esquemas, tablas y relaciones físicas.                      |
| Semana 4 | 24 – 28 junio | Desarrollo de scripts de integración en Python y configuración de Jobs. |
| Semana 5 | 29 – 01 julio | Pruebas, documentación, generación del PDF y carga en GitHub.           |

## Definición de Roles

| Integrante       | Rol Asignado        |
|------------------|---------------------|
| Erick Acosta     | Coordinador General |
| Xavier Logroño   | DBA Oracle          |
| Jonathan Paredes | DBA SQL Server      |

## Asignación de Roles

| Rol                 | Descripción del Rol  |
|---------------------|--|
| Coordinador General | Gestiona el cronograma, organiza reuniones, coordina entregables                                 |
| DBA Oracle          | Instala Oracle, diseña el esquema de pólizas y pagos, y realiza pruebas de conexión con DBeaver. |
| DBA SQL Server      | Configura SQL Server, implementa el CRM y valida consultas desde DBeaver.                        |

## Cronograma de Reuniones

| Fecha | Hora | Plataforma | Asunto Principal |
|-------|------|------------|------------------|
|-------|------|------------|------------------|

|        |       |         |  |
|--------|-------|---------|--|
| 5-jun  | 19h00 | Discord | Revisión del cronograma y asignación de roles            |
| 12-jun | 19h00 | Discord | Progreso de instalación de SGBD y conexión desde DBeaver |
| 19-jun | 19h00 | Discord | Pruebas de datos y definición de interconexión           |
| 26-jun | 19h00 | Discord | Verificación de Jobs automáticos y evidencias            |
| 3-jul  | 19h00 | Discord | Cierre del proyecto, revisión del PDF y carga en GitHub  |

**Debian (MySQL)**

```
May 31 11:53:56 debian12 systemd[1]: Starting mysql.service - MySQL Community Server...
May 31 11:53:56 debian12 systemd[1]: Started mysql.service - MySQL Community Server.
debian@debian12:~$ mysql
ERROR 1045 (28000): Access denied for user 'debian'@'localhost' (using password: NO)
debian@debian12:~$ sudo mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 11
Server version: 8.0.42 MySQL Community Server - GPL

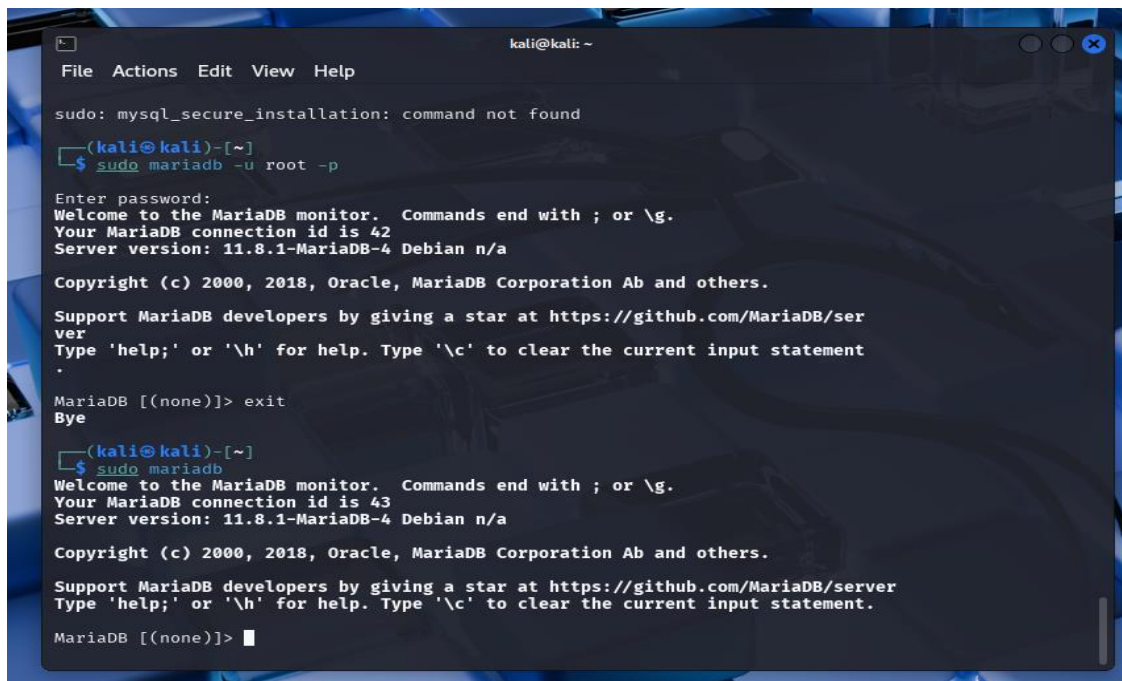
Copyright (c) 2000, 2025, Oracle and/or its affiliates.

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> █
```

## Kali( Mariadb)



```
kali@kali: ~
File Actions Edit View Help

sudo: mysql_secure_installation: command not found

(kali@kali)-[~]
$ sudo mariadb -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 42
Server version: 11.8.1-MariaDB-4 Debian n/a

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

Support MariaDB developers by giving a star at https://github.com/MariaDB/server
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement
.

MariaDB [(none)]> exit
Bye

(kali@kali)-[~]
$ sudo mariadb
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 43
Server version: 11.8.1-MariaDB-4 Debian n/a

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

Support MariaDB developers by giving a star at https://github.com/MariaDB/server
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> █
```

## Centos7(Oracle)

```
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=svrjspp.lpmb.com)(PORT=1521)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=EXTPROC1521)))
The listener supports no services
The command completed successfully
[oracle@svrjspp ~]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Tue Jun 3 17:09:50 2025
Version 19.3.0.0.0

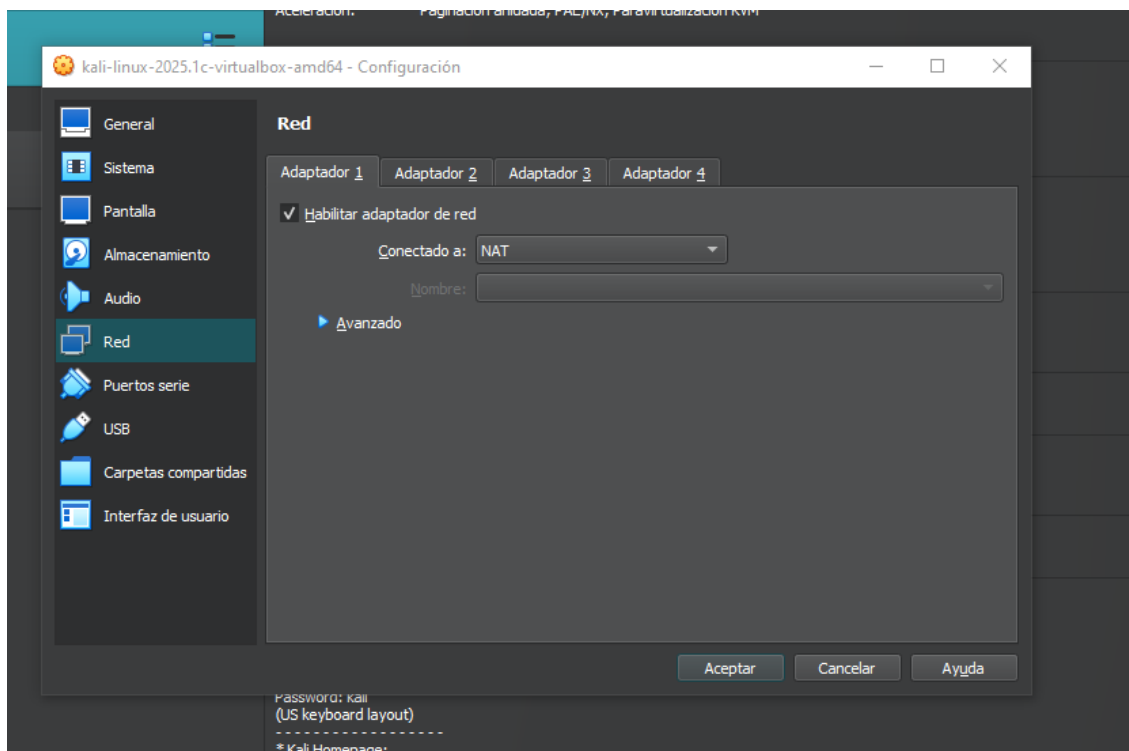
Copyright (c) 1982, 2019, Oracle. All rights reserved.

Connected to an idle instance.

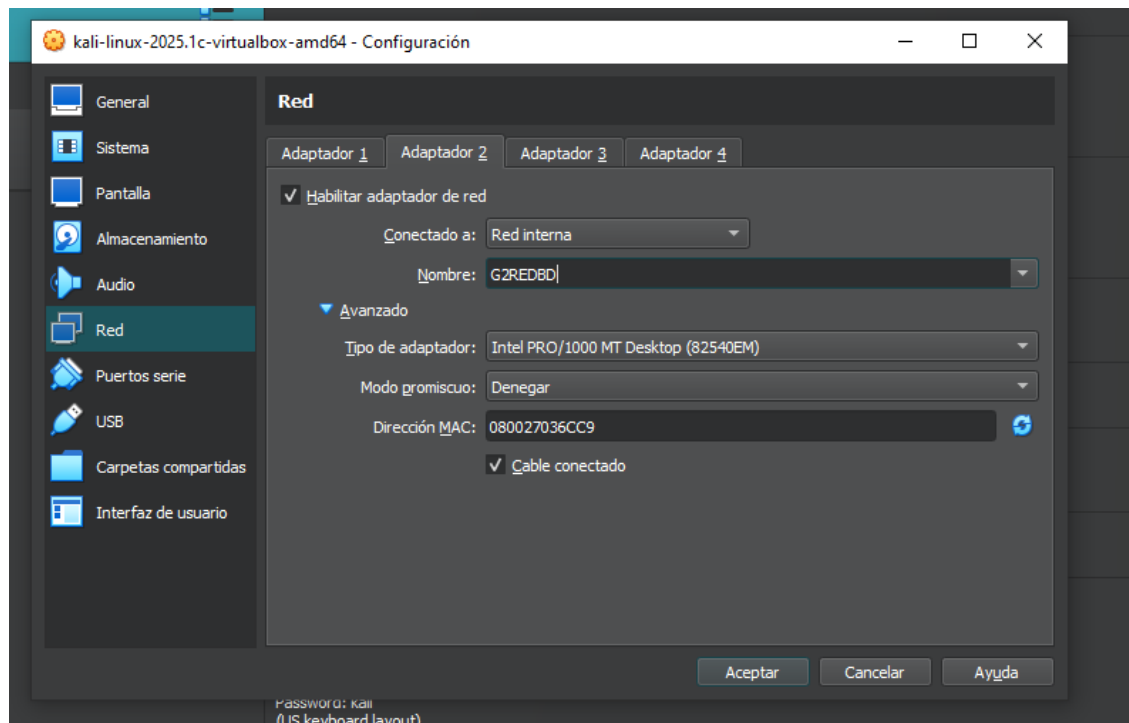
SQL>
```

## Conexión de las 3 máquinas

### Creación de red interna



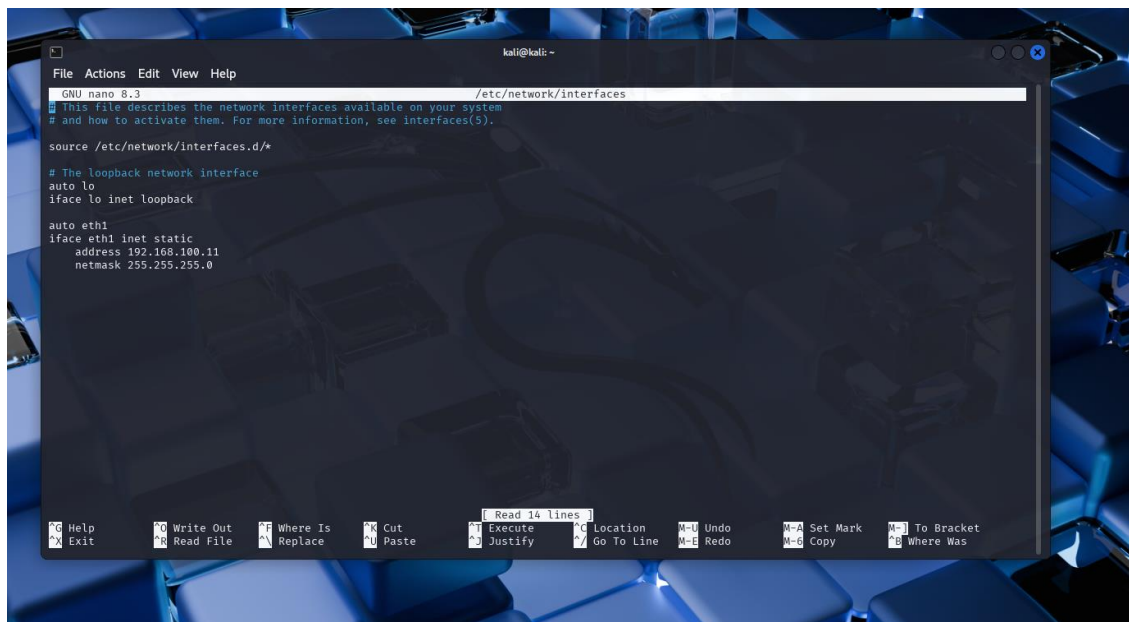
### Asignación de nombre a la Red Interna (G2REDBD)



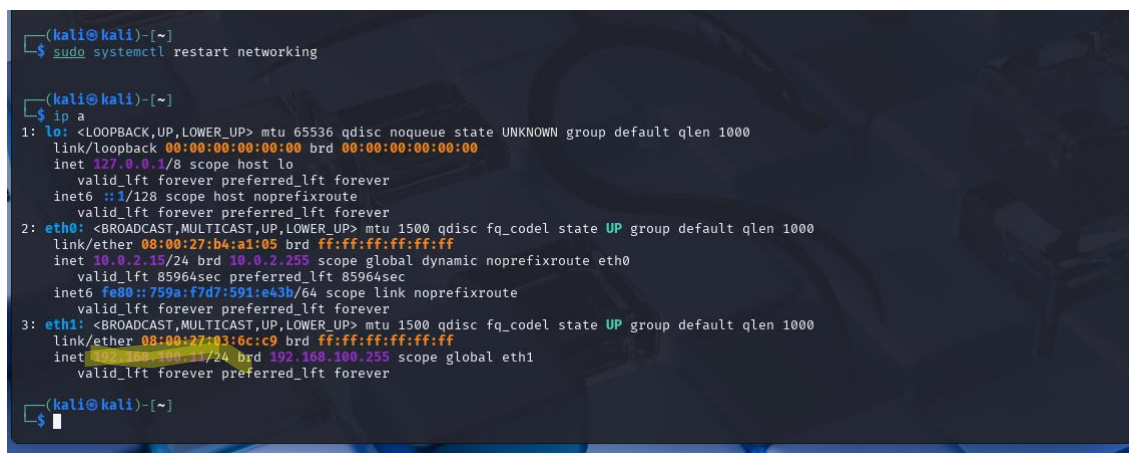
## Configuración de IP a cada máquina

```
(kali@kali)-[~]
$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
   link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
   inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
   inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
   link/ether 08:00:27:b4:a1:05 brd ff:ff:ff:ff:ff:ff
   inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute eth0
       valid_lft 86268sec preferred_lft 86268sec
   inet6 fe80::759a:f7d7:591:e43b/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
   link/ether 08:00:27:03:6c:c9 brd ff:ff:ff:ff:ff:ff
   inet6 fe80::da7b:435e:a370:10ac/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
(kali@kali)-[~]
$
```

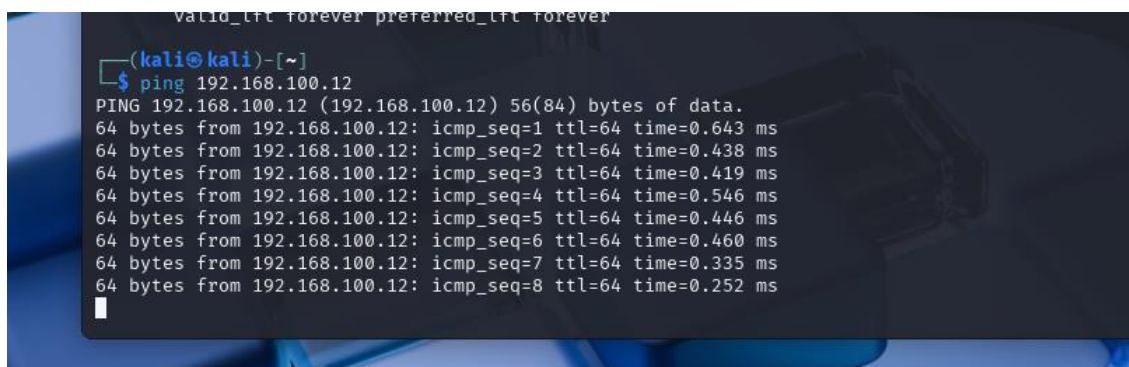
## Asignación de Ip

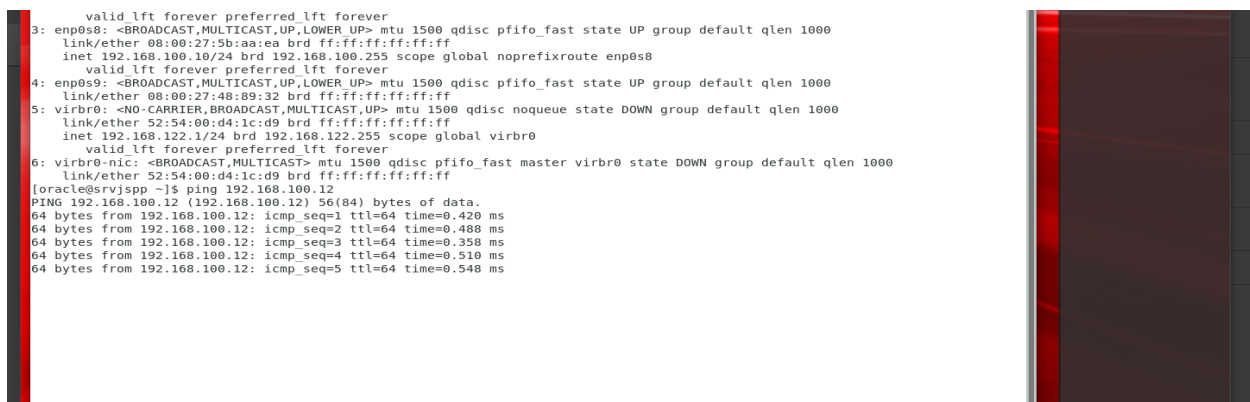
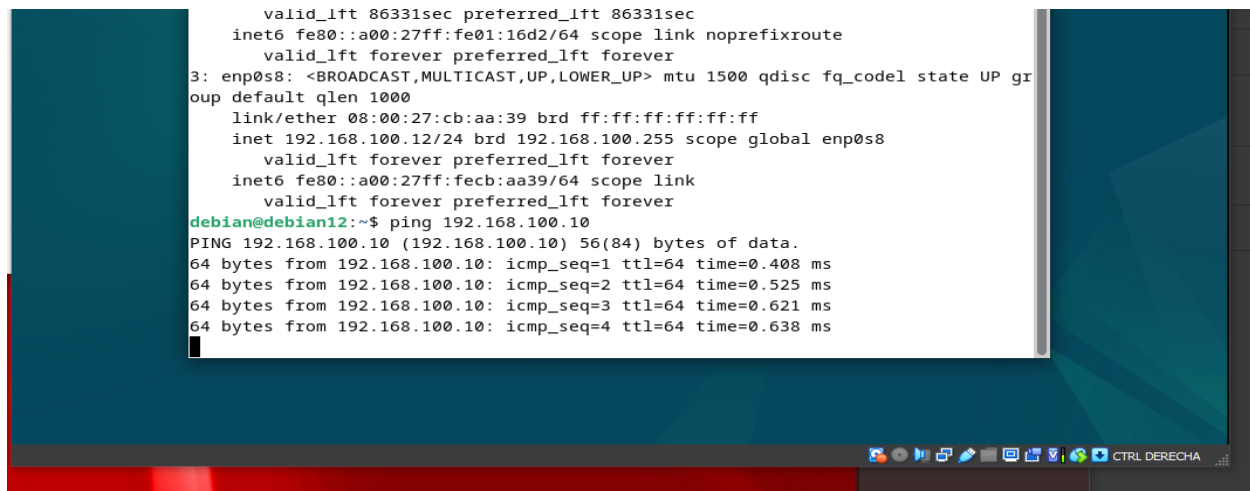


## Verificación de IP asignada en el adaptador de red

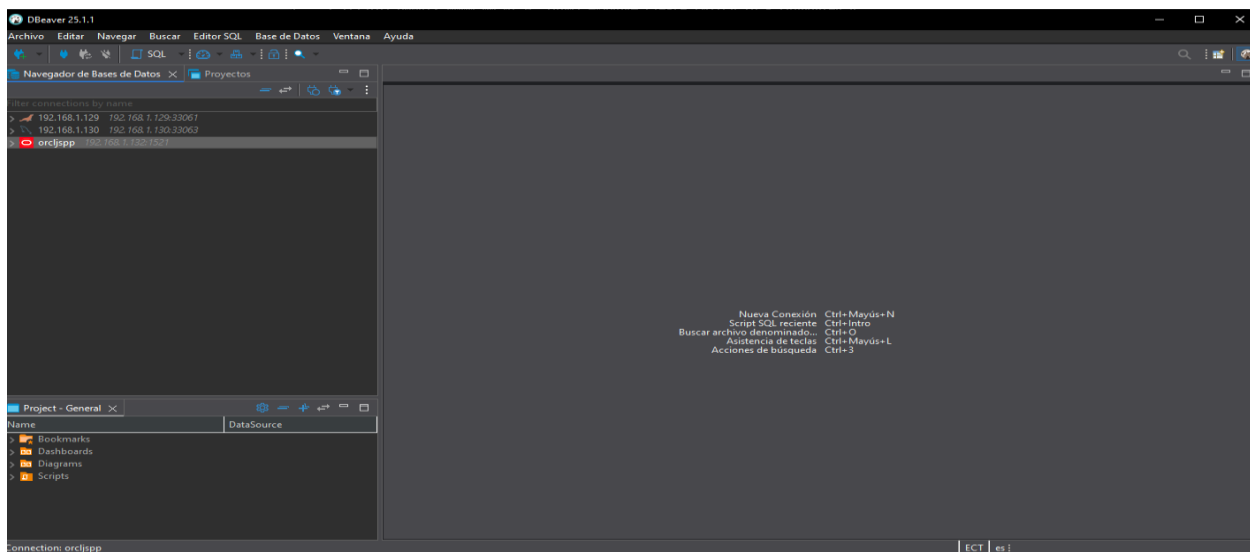


## Validación con Ping hacia la otra máquina

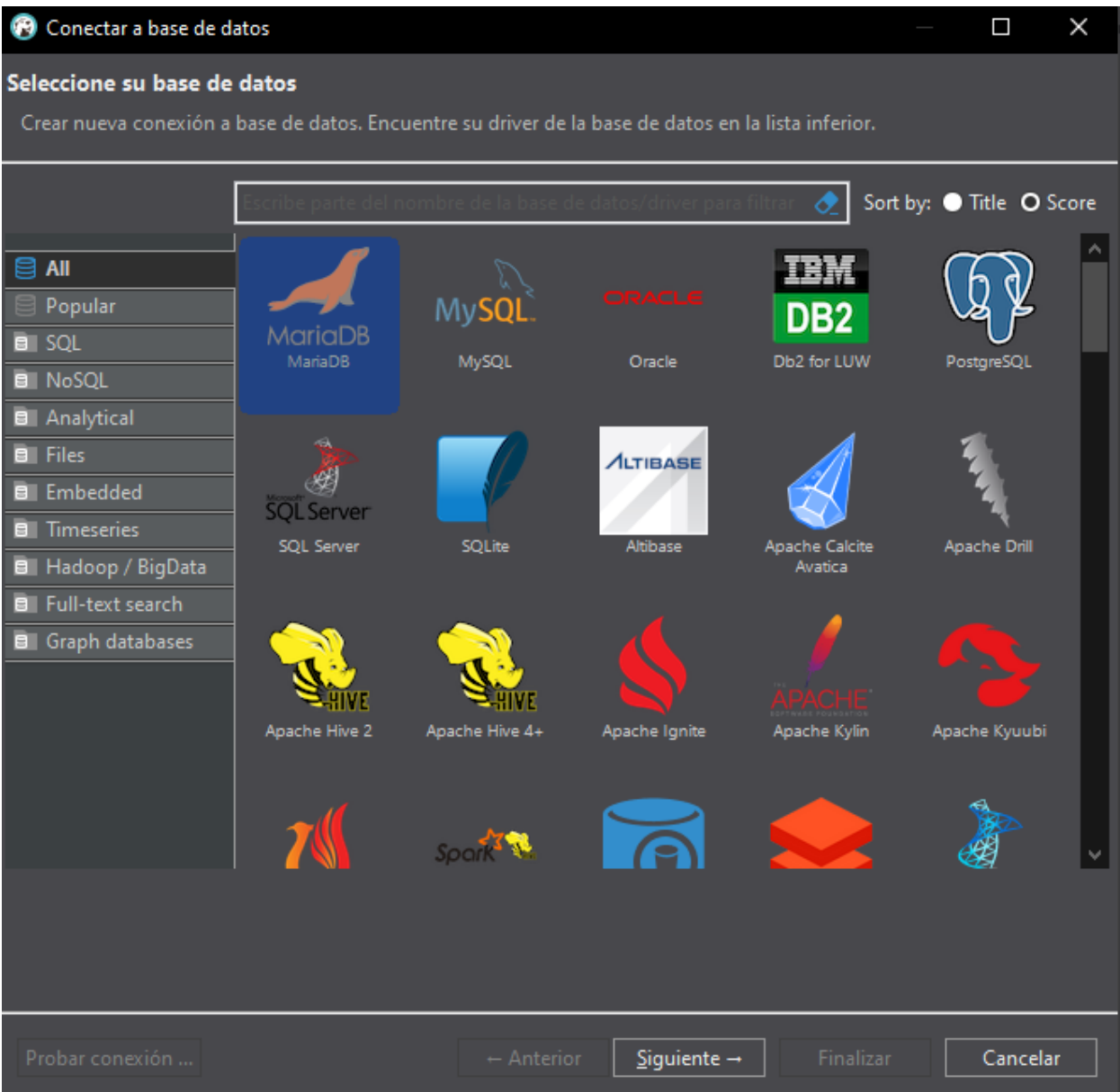




## Conexión con Dbeaver








Conectar a base de datos

### Connection Settings

MariaDB ajustes de conexión



General Driver properties + SSH, SSL, ... No profile

Server

Connect by: ☐ Host ☒ URL

URL: jdbc:mariadb://192.168.1.129:33061/

Server Host: 192.168.1.129 Port: 33061

Database:

Authentication (Database Native)

Nombre de usuario: admin

Contraseña: ..... ☒ Save password

Advanced

Server Time Zone: Auto-detect

Local Client: MariaDB Binaries

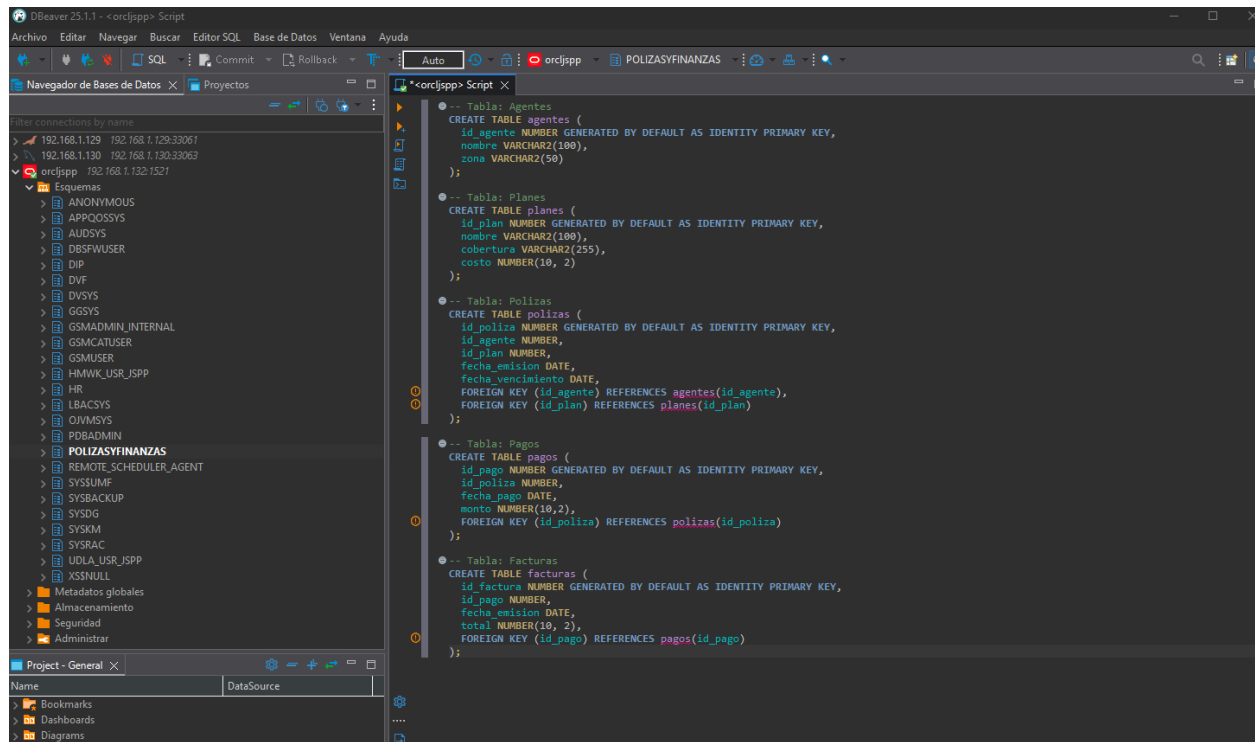
[Connection variables information](#) Connection details (name, type, ...)

Driver name: MariaDB Driver Settings Licencia del driver

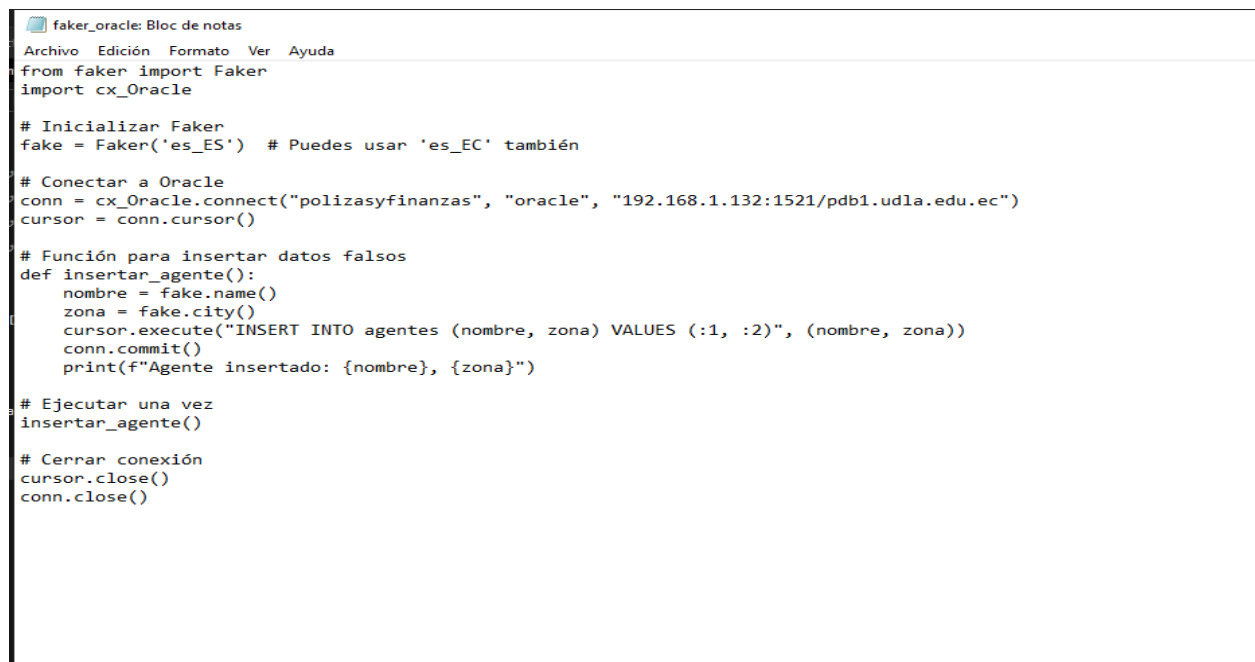
Probar conexión ... ← Anterior Siguiente → Finalizar Cancelar

## Creación de Base de Datos en Dbeaver

## Script de tablas Oracle



## Faker Python



## Ejecución del script de pyhton

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Versión 10.0.19045.5965]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Jonathan\Downloads\ProyectoAdminBDD>ls
"ls" no se reconoce como un comando interno o externo,
programa o archivo por lotes ejecutable.

C:\Users\Jonathan\Downloads\ProyectoAdminBDD>dir
El volumen de la unidad C no tiene etiqueta.
El número de serie del volumen es: BC52-1BF9

Directorio de C:\Users\Jonathan\Downloads\ProyectoAdminBDD

01/07/2025  14:40    <DIR>          .
01/07/2025  14:40    <DIR>          ..
01/07/2025  14:40                630 faker_oracle.py
01/07/2025  10:27                841 Script MariaDB.txt
01/07/2025  10:28                809 Script Mysql.txt
01/07/2025  10:09               1.130 Script Oracle.txt
                4 archivos              3.410 bytes
                2 dirs  706.271.629.312 bytes libres

C:\Users\Jonathan\Downloads\ProyectoAdminBDD>_
```

```
C:\Windows\System32\cmd.exe - python faker_mariadb.py

01/07/2025 15:36 <DIR> .
01/07/2025 15:36 <DIR> ..
01/07/2025 15:46 1.086 faker_mariadb.py
01/07/2025 15:46 1.015 faker_mysql.py
01/07/2025 15:45 878 faker_oracle.py
01/07/2025 10:27 841 Script MariaDB.txt
01/07/2025 10:28 809 Script Mysql.txt
01/07/2025 10:09 1.130 Script Oracle.txt
        6 archivos      5.759 bytes
        2 dirs 705.523.810.304 bytes libres

C:\Users\Jonathan\Downloads\ProyectoAdminBDD>python faker_mariadb.py
Traceback (most recent call last):
  File "C:\Users\Jonathan\Downloads\ProyectoAdminBDD\faker_mariadb.py", line 3, in <module>
    import schedule
ModuleNotFoundError: No module named 'schedule'

C:\Users\Jonathan\Downloads\ProyectoAdminBDD>pip install schedule
Collecting schedule
  Downloading schedule-1.2.2-py3-none-any.whl.metadata (3.8 kB)
  Downloading schedule-1.2.2-py3-none-any.whl (12 kB)
Installing collected packages: schedule
Successfully installed schedule-1.2.2

C:\Users\Jonathan\Downloads\ProyectoAdminBDD>python faker_mariadb.py
MariaDB: ejecutando cada minuto...
[MariaDB] Sinistro insertado: Corporis rem magnam. | 1998-03-09 | 2899.13
[MariaDB] Sinistro insertado: Provident nesciunt quis earum alias nobis. | 2001-02-08 | 4812.2
```

Grilla

clientes 1

select \* from clientes c

|   | id | nombre                | email                   | telefono         |
|---|----|-----------------------|-------------------------|------------------|
| 1 | 1  | Priscila Solano Carro | faustosalom@example.com | +34 955 87 65 59 |
| 2 | 2  | Teófila Parejo Amigó  | eugenia69@example.com   | +34 888 11 70 39 |
| 3 | 3  | Ciro Bertrán Rivas    | fcortes@example.net     | +34 730282406    |

Texto