

Este trabajo tiene licencia CC BY-NC-SA 4.0. Para ver una copia de esta licencia, visite <http://creativecommons.org/licenses/by-nc-sa/4.0/>

U8P01-Amazon RDS. Despliegue de un SGBD MariaDB

1. Descripción

Los **servicios administrados** se encargan de la mayoría (o todas) las tareas operativas necesarias para que el sistema gestor de bases de datos funcione correctamente. El proveedor de la nube maneja la infraestructura y la operación diaria.

Amazon RDS (Relational Database Service) es un servicio administrado de bases de datos relacionales proporcionado por AWS. Facilita la configuración, operación y escalado de bases de datos relacionales en la nube, eliminando gran parte de la carga operativa tradicional asociada con la administración de bases de datos.

En esta práctica se configurará un sistema gestor de bases de datos utilizando **Amazon RDS** con el motor de base de datos **MariaDB**, y se configurará para poder establecer conexiones desde **instancias EC2** y desde un **cliente externo**.

Nota: no es una buena práctica de seguridad exponer una base de datos a internet. Lo adecuado es crearla en una subred privada y que sea accesible solo desde las instancias/servicios que necesiten usarla. En esta práctica se realiza con fines didácticos.

2. Formato de entrega

- Realiza la opción que indique el profesor
 - MOSTRAR: muestra en la clase tu configuración al profesor.
 - ENTREGAR: Entrega un documento que incluya capturas de pantalla de los diferentes pasos realizados a lo largo de la práctica.

3. Trabajo a realizar

3.1. Desplegar de la bases de datos con Amazon RDS

Para crear una instancia de Amazon RDS, es necesario disponer previamente de un **grupo de subredes**, que definirá en qué subredes podrá **desplegarse la instancia de RDS**. En esta práctica vamos a usar un el grupo de subredes predeterminado que incluye a todas las subredes de la VPC por defecto.

1. Inicia su tu entorno *AWS Learner Lab* y accede a la consola de administración de AWS.
2. Accede al servicio **Aurora and RDS**.
3. En le menú de la izquierda accede a **Grupos de subredes** y observa que hay creado un grupo que engloba a todas las subredes de la VPC predeterminada.

Ahora se configurará y lanzará la instancia de BD con el motor **MariaDB**.

4. En el menú lateral, selecciona la opción **Base de Datos** y a continuación **Crear Base de datos**. Comenzará un asistente donde tienes que configurar las opciones que se proponen en los siguientes apartados dejando el resto a sus valores por defecto.

5. Selecciona la opción **Creación Estándar**.

- Dentro del apartado **Opciones de Motor** elige la opción **MariaDB** y deja el resto de las configuraciones por defecto. Observa la versión del motor que se va a utilizar, sería posible cambiarlo.
- En el apartado **Plantillas** selecciona **Capa Gratuita** (recuerda que esto hace referencia a que se usan los recursos de la capa gratuita que ofrece AWS durante un año para nuevas cuentas, pero no significa que no consuma créditos de tus 50 dólares, por lo tanto **si que tiene costes**).
- En el apartado **Configuración** indica los siguiente parámetros (el resto observalos pero deja sus valores por defecto).
 - **Identificador de instancias de bases de datos:** **dam/daw-mariadb-XX** (XX es tú numero en el aula).
 - Nombre de usuario maestro: **admin**.
 - Contraseña maestra: **vincivinci** (si pones otra recuérdala).
 - Confirmar la contraseña maestra: **vincivinci**
- En el apartado **Configuración de la instancia** elige dentro del campo **Clase de instancia de base de datos** la opción **Clases ampliables (incluye clases t)** y en el menú desplegable elige la opción **db.t4g.micro**.
- En el apartado de **Almacenamiento**, configura el volumen EBS asociado a nuestra instancia RDS. Elige dentro del menú desplegable **Tipo de almacenamiento** el valor **SSD de uso general (gp2)**, y en la opción **Almacenamiento asignado** indica **20 GiB**.
- Observa que con la configuración **de capa gratuita no se permite una configuración Multi AZ**.
- En el apartado **Conectividad** configura las siguientes opciones;:
 - Tipo de red: **IPv4**
 - Nube privada virtual (VPC): **VPC Predeterminada**.
 - Acceso público: **SI**
 - RDS asigna una dirección IP pública a la base de datos.
 - Las instancias de Amazon EC2 y otros recursos fuera de la VPC pueden conectarse a la base de datos.
 - **Crea un nuevo grupo de seguridad: sg_RDSMariaDBXX**. Se creará un nuevo grupo de seguridad asociado a la BD que permita el tráfico entrante entrante hacia el puerto 3306/TCP que es donde escucha peticiones María DB.
 - Zona de disponibilidad: **us-east1-a**.

- En el apartado **Autenticación de bases de datos** selecciona **Autenticación con contraseña**.
- En el apartado **Supervisión** dejan la opciones por defecto. Observa que no habilitamos las opciones de monitorización para no incurrir en gastos pero es una base de datos en producción serían necesarios para monitorizar y supervisar todo lo que ocurre.
- Despliega el apartado **Configuración adicional** y **deshabilita la Copia de seguridad** con el objeto de no incurrir en costes adicionales. La primera copia de respaldo de Amazon RDS está incluida en el precio de la instancia. En un entorno de producción es muy importante tener una buena política de copias de seguridad.
- En **Costos mensuales estimados** lee la información disponible. Como no estamos usando la capa gratuita la base de datos que vamos a crear tendrá un pago por uso asociado que podemos consultar en <https://aws.amazon.com/es/rds/pricing/> (haz una estimación de lo que costará lo que vamos a crear).

6. Una vez terminada la configuración, pincha en **Crear base de datos** para comenzar con el despliegue de la instancia de Amazon RDS. No usos los complementos sugeridos.

El lanzamiento de la instancia puede demorar unos 5-10 minutos.

7. Cuando haya terminado el despliegue consulta las propiedades de la instancia de base de datos creada y navega por las diferentes pestañas para ver sus propiedades.

• Conectividad y seguridad

- Observa el **punto de acceso (nombre DNS para por conectarte)** a la base de datos y el puerto a la escucha.

dam-maraidb-xx

Resumen
Identificador de base de datos
dam-maraidb-xx
CPU
2.36%

Estado
Disponible
Clase
db.t4g.micro

Rol
Instancia
Actividad actual
0 Conexiones

Motor
MariaDB
Región y AZ
us-east-1a

Recomendaciones

Conectividad y seguridad | Supervisión | Registros y eventos | Configuración | Mantenimiento y copias de seguridad | Migraciones de datos - nuevo | Etiquetas | Recomendaciones

Conectividad y seguridad
Punto de enlace y puerto
Punto de enlace
dam-maraidb-xx.cetvt2pxrjy.us-east-1.rds.amazonaws.com
Puerto
3306

Redes
Zona de disponibilidad
us-east-1a
VPC
vpc-047cf6f7d794a318d
Grupo de subredes
default-vpc-047cf6f7d794a318d
Subredes
subnet-0124ef33002d40eec
subnet-0721c05e43814147d
subnet-076541c612fdb02e3
subnet-0b43b389f86000f84
subnet-0ef1cd07d4f0eb913
subnet-05be40f1259ded41c
subnet-0b578944dc7a22310
subnet-08855ef6bba24387

Seguridad
Grupos de seguridad de la VPC
sg-RDSMariaDBXX (sg-001e4ec72734a4421)
Activo
rds-ec2-1 (sg-0bf809494f5f691c9)
Activo
Accesible públicamente
Sí
Entidad de certificación
rds-ca-rsa2048-g1
Fecha de la entidad de certificación
May 26, 2061, 01:34 (UTC+02:00)
Fecha de expiración del certificado de instancia de base de datos
March 28, 2026, 17:15 (UTC+01:00)

- La **VPC y subredes** en las que atiende peticiones.

- Si consultas el **grupo de seguridad** que se ha creado verás que tiene las siguientes reglas.
 - Permite todo el tráfico de salida.
 - Solo permite el tráfico de entrada hacia el puerto 3306/TCP desde una IP (que es la IP desde donde te estás conectado, en este caso la IP pública el instituto). Esto lo ha creado por seguridad, como se ha comentado al inicio de la práctica no es seguro exponer una base de datos a Internet.
- **Supervisión.**
- **Registros y eventos.**
- **Configuración.**
- **Mantenimiento y copias de seguridad.**
- **Migraciones de datos.**
- **Etiquetas.**
- **Recomendaciones.**

3.2. Establecer conexiones desde clientes

3.2.1. Habilitar el tráfico en el grupo de seguridad

Para que clientes externos se puedan conectar al servidor MariaDB es necesario habilitar en el grupo de seguridad al que pertenece la base de datos el tráfico de entrada hacia el puerto 3306/TCP. Como hemos comprobado ante el asistente ha creado un grupo que solo permite el tráfico de entrada desde la IP desde la que nos estamos conectando.

Podríamos crear las reglas que consideremos para permitir conexiones desde otros orígenes.

3.2.2. Conexión desde otra instancia

Vamos a establecer una conexión desde otra instancia de EC2.

1. Despliega una instancia en **AWS EC2** con las siguientes propiedades.

- Nombre: **ganonXX-A** (XX es tú número en el aula).
- Imágenes de aplicaciones y sistemas operativos (Imagen de máquina de Amazon): **Ubuntu Server 24.04 LTS para arquitectura x86 (64 bits)**. Observa que es **Apta para la capa gratuita**.
- Tipo de instancia: **tipo t2.micro (1 CPU virtual y 1 GiB de memoria)** , que también es **apta para la capa gratuita**.
- Par de claves: **vockey**.
- Configuraciones de red:
 - VPC: **VPC Predeterminada**
 - Subred : la **subred** asignada a la **zona de disponibilidad us-east-1a**
 - Asignar automáticamente la IP pública: **Habilitar**.

- **Firewall (grupos de seguridad):**
 - Nombre del grupo de seguridad: **sg_SSH-XX**
 - Descripción: **Acceso SSH**
 - **Regla para permitir el acceso desde cualquier lugar al puerto 22/TCP.**
- **Configurar almacenamiento:** volumen de 8 GB.

2. Accede por SSH a la instancia.

3. Instala un cliente de **mariadb**.

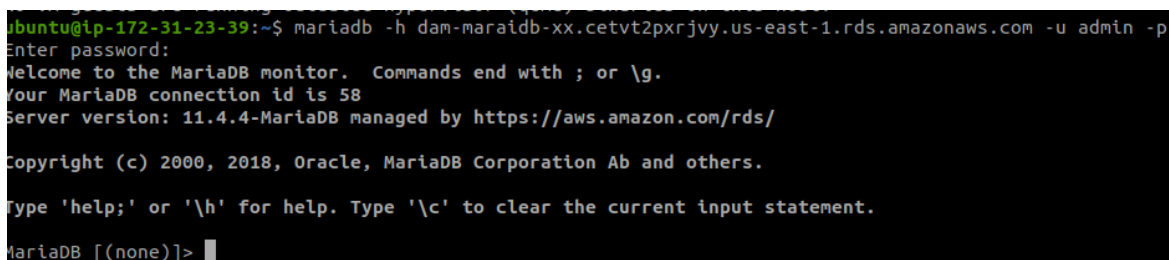
```
sudo apt update
sudo apt install mariadb-client
```

4. Para poder establecer conexiones a la base de datos desde esta instancia hay que configurar las reglas adecuada en los grupos de seguridad. En lugar de hacerlo de forma manual **vamos a utilizar un asistente que ofrece Amazon RDS.**

- Accede a la consola de RDS.
- Selecciona la base de datos.
- En Acciones selecciona **Configurar la conexión de EC2.**
- Selecciona **ganonXX-A** como instancia y pincha en **Continuar.**
- Observa que el asistente va a crear dos nuevos grupos de seguridad con la regla adecuadas y los va a añadir a la instancia y a la base de datos.
- Pincha en **Configurar.**
- Puedes consultar los nuevos grupos de seguridad para ver las reglas que se han creado.
-

5. Desde la instancia establece una conexión contra el servidor de bases de datos con el usuario admin.

```
mariadb -h <Punto de enlace de la BD> -u admin -p
```



```
ubuntu@ip-172-31-23-39:~$ mariadb -h dam-mariadb-xx.cetvt2pxrjvy.us-east-1.rds.amazonaws.com -u admin -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 58
Server version: 11.4.4-MariaDB managed by https://aws.amazon.com/rds/
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)]>
```

6. Ejecuta la siguiente sentencia SQL para crear la base de datos (esquema) juegos.

```
MariaDB [(none)]>create database juegos;
```

7. Ejecuta las siguientes sentencias SQL para crear el usuario **juegos** con password **juegos** tanto para conexiones locales como conexiones desde cualquier origen.

```
MariaDB [(none)]>create user juegos@'localhost' identified by 'juegos';
MariaDB [(none)]>create user juegos@'%' identified by 'juegos';
```

8. Ejecuta las siguientes sentencias SQL para asignarle todos los privilegios al usuario juegos sobre la base de datos juegos.

```
MariaDB [(none)]>grant all privileges on juegos.* to juegos@'localhost';
MariaDB [(none)]>grant all privileges on juegos.* to juegos@'%';
```

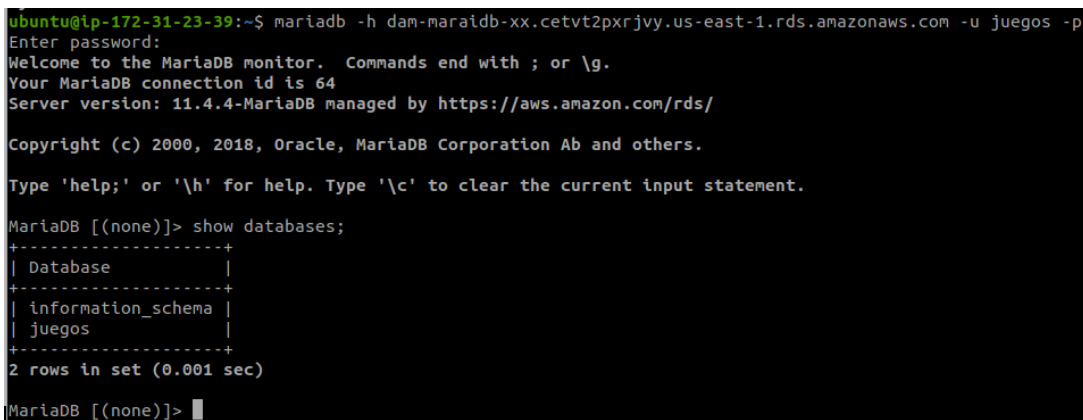
9. Cierra la conexión con `exit`.

10. Establece una conexión con el usuario **juegos**.

```
mariadb -h <Punto de enlace de la BD> -u juegos -p
```

11. Consulta las bases de datos a las que tiene acceso.

```
MariaDB [(none)]>show databases;
```



```
ubuntu@ip-172-31-23-39:~$ mariadb -h dam-mariadb-xx.cetvt2pxrjvy.us-east-1.rds.amazonaws.com -u juegos -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 64
Server version: 11.4.4-MariaDB managed by https://aws.amazon.com/rds/
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)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| juegos |
+-----+
2 rows in set (0.001 sec)

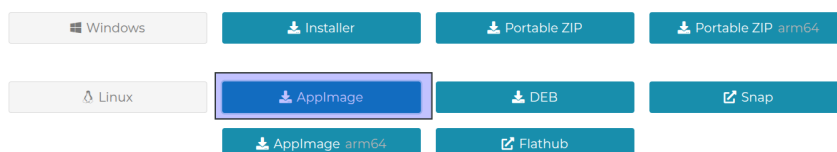
MariaDB [(none)]>
```

3.2.3. Conexión desde tu máquina real

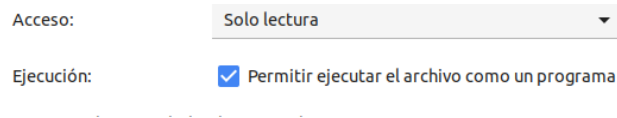
Vamos a establecer una conexión desde tu máquina real. Par ello puedes usar el cliente que consideres, existen muchas opciones que puedes instalar en Windows, Linux, y Mac, así como extensiones para IDEs como Visual Studio Code.

En esta práctica te propongo usar **Dbgate** (<https://dbgate.org/>) un cliente sencillo. Dispone de versiones para instalar y de versiones portables tanto para Windows como para Linux.

1. En tu máquina real accede a <https://dbgate.org/download/> y descarga la versión portable (en Linux, es la versión Appliance).

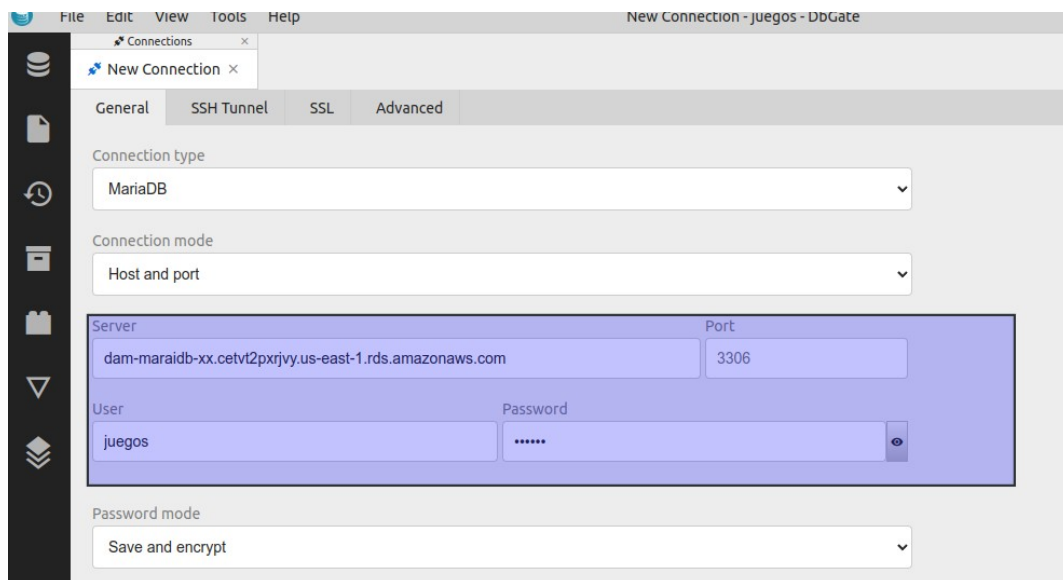


2. Ejecuta el cliente. En los equipos de clase previamente tienes que darle permisos de ejecución, botón derecho del ratón, Permisos y marcar la opción **“Permitir ejecutar el archivo como un programa”**.



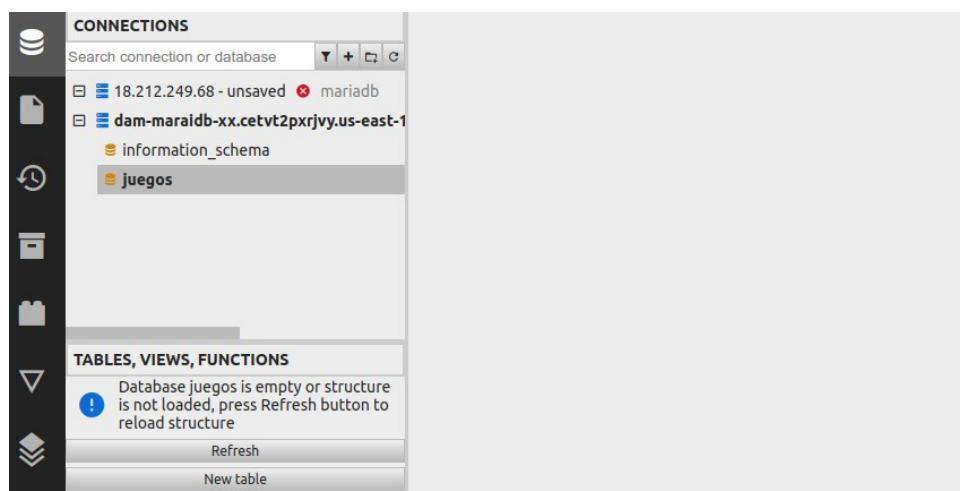
3. Añade una nueva conexión para **MariaDB** y configura los datos del servidor al que te vas a conectar.

- Server: **<Punto de enlace de la BD>**
- Usuario: **juegos**
- Password: **juegos**

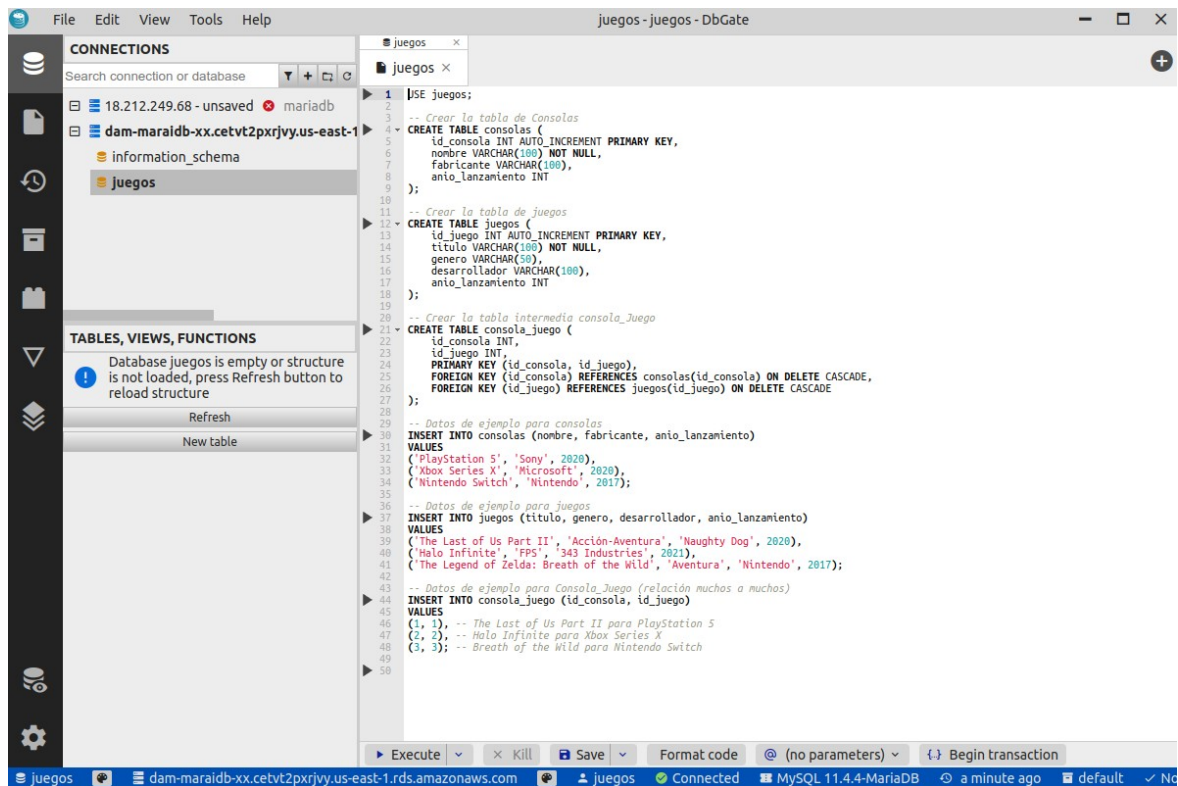


Establece la conexión pinchando en **Connect**.

4. Selecciona la base de datos **juegos**.

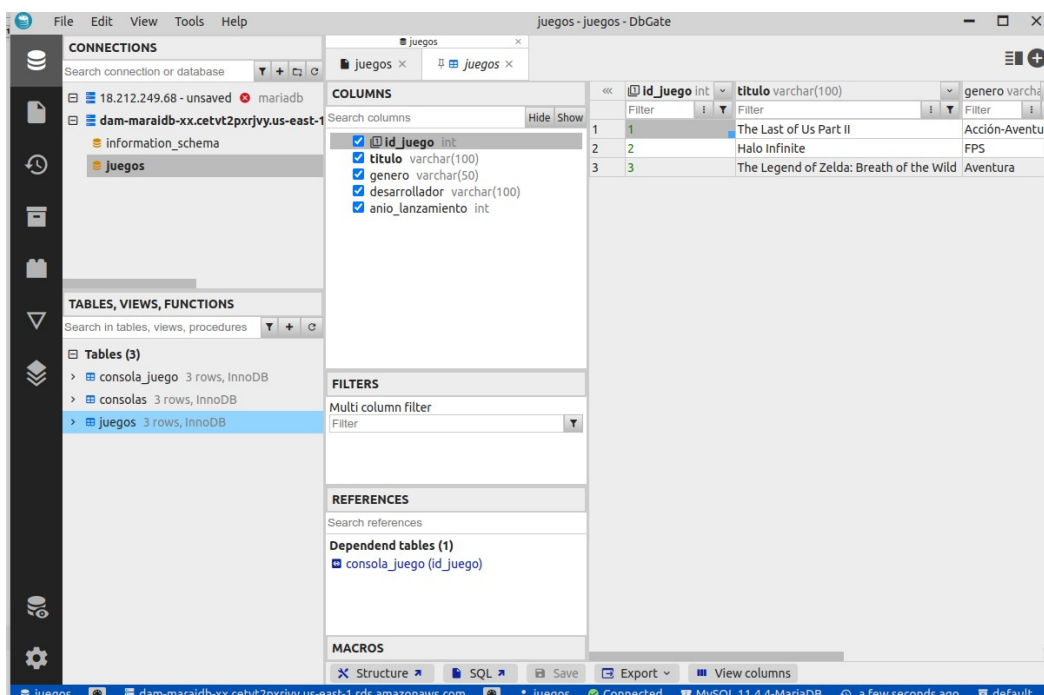


5. Abre el archivo **juegos.sql** que te proporcione el profesor con sentencias sql para crear tablas y datos de prueba.

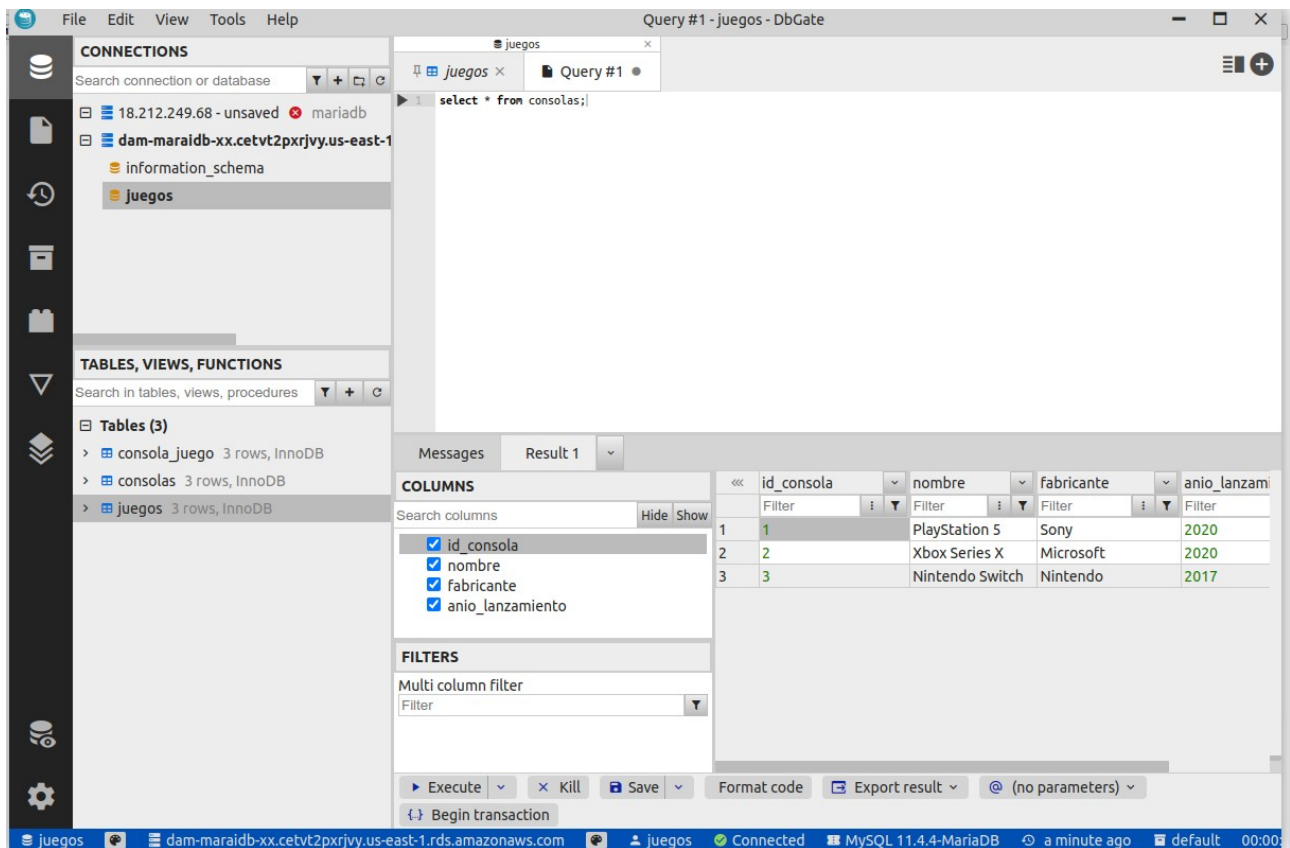


6. Ejecuta las sentencias con el botón **Execute**.

7. Refresca la vista para ver como se han creado las tablas y sus datos.



8. Prueba a hacer una consulta.



Puedes probar a usar alguna de las extensiones de **Visual Studio Code** que permiten actuar como cliente SQL, por ejemplo, **Database Client**.

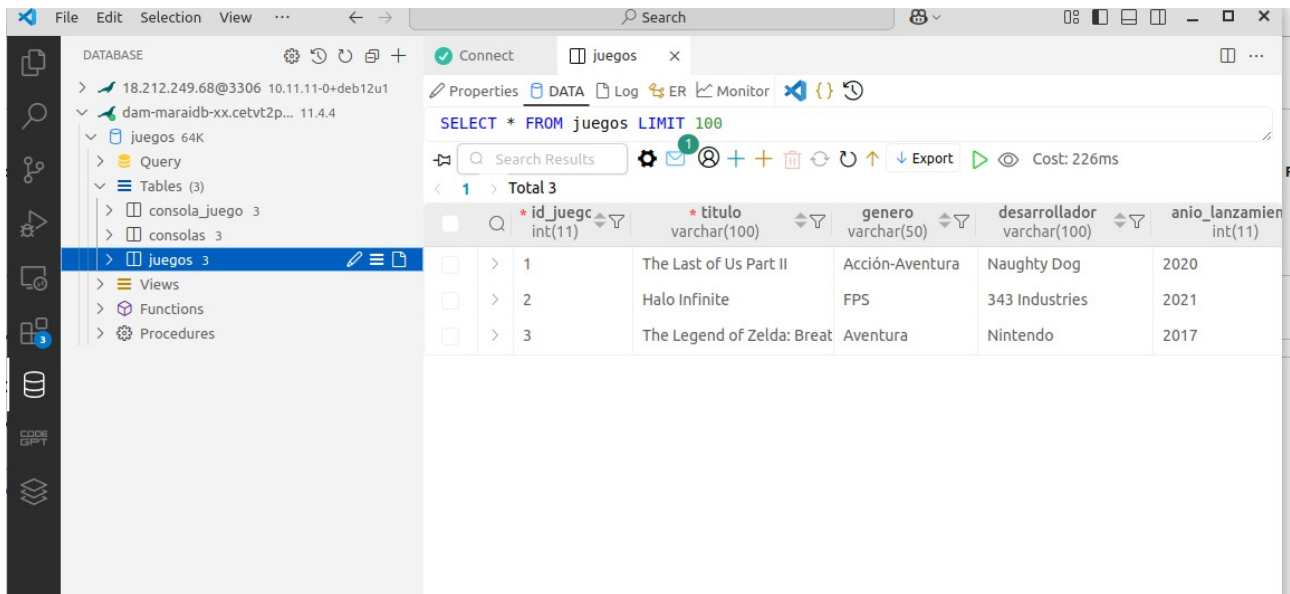
Note: Requires a [Premium License](#) to unlock all features. (Limited to 3 connections) Database 1/3, NoSQL 0/3

Server Type

MySQL MariaDB PostgreSQL SQLite SQL Server Db2 Oracle Kingbase
ClickHouse JDBC
SSH Docker Redis Elasticsearch MongoDB S3 FTP Kafka RabbitMQ
Cassandra More

Config Main SSH Tunnel Socks Proxy HTTP Proxy

* Host dam-maraiddb-xx.cetv * Port 3306
* Username juegos * Password *****
Database Database Advanced
Socket Path e.g. /var/run/mysqld/ Use
Features ☐ Event ☐ Trigger
Use Connection String ☐ SSL ☐



The screenshot shows a database management interface. On the left, a sidebar lists the database structure: 'DATABASE' (18.212.249.68@3306 10.11.11-0+deb12u1), 'dam-maraidb-xx.cetvt2p...' (11.4.4), 'juegos 64K', 'Query', 'Tables (3)' (consola_juego 3, consolas 3, juegos 3), 'Views', 'Functions', and 'Procedures'. The 'juegos 3' table is selected. The main window displays a SQL query: `SELECT * FROM juegos LIMIT 100`. Below the query, the results are shown in a table with columns: `* id_juegc` (int(11)), `* titulo` (varchar(100)), `genero` (varchar(50)), `desarrollador` (varchar(100)), and `anio_lanzamien` (int(11)). The results show 3 rows.

	* id_juegc int(11)	* titulo varchar(100)	genero varchar(50)	desarrollador varchar(100)	anio_lanzamien int(11)
1		The Last of Us Part II	Acción-Aventura	Naughty Dog	2020
2		Halo Infinite	FPS	343 Industries	2021
3		The Legend of Zelda: Breat	Aventura	Nintendo	2017

NOTA: no elimines los recursos creados ya que se usarán en las próximas prácticas.