



Ivana Sánchez Pérez

2º ASIR

Contenido

1.- Introducción	2
2.- Ejercicio 1. Configuración conexión en MySQL Workbench	2
2.1.- Insertar fila	3
2.2.- Consulta listado clientes	3
3.- Ejercicio 2. Crear un usuario	4
4.- Ejercicio 3. Creación y ejecución del script de una base de datos	5
4.1.- Descargar el archivo <i>distribuidora.sql</i>	5
4.2.- Concesión de permiso	6
4.3.- Verificación del permiso	6
5.- Ejercicio 4. Crear tarea2vista	7
5.1.- Concesión de permisos para crear vistas	7
5.2.- Crear la vista	7
6.- Ejercicio 5. Insertar una fila en tarea2vista	9
7.- Ejercicio 6. Creación de un rol con sus permisos	9
7.1.- Crear un rol	9
7.2.- Asignación del rol	9
7.3.- Verificación del rol	10
8.- Ejercicio 7. Activación de rol y borrado de filas.	10
8.1.- Activación del rol	10
8.2.- Borrado de filas	10
9.- Ejercicio 8. Alteración de usuario	11
10.- Ejercicio 9. Eliminar el rol de un usuario	12
11.- Ejercicio 10. Bloquear una tabla para escritura	12



1.- Introducción

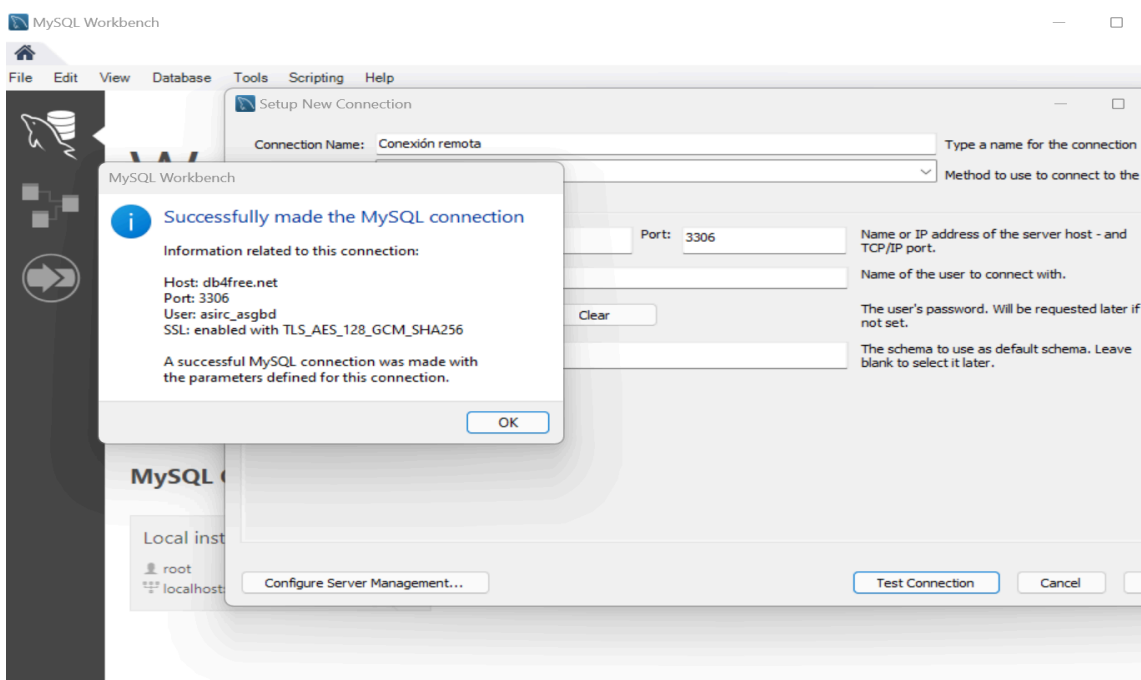
En el presente trabajo se muestra la tarea 2 de la asignatura ASGB. Se trata de un caso práctico en BK Programación para empezar a ver cómo es la administración de un SGBD.

Primero vamos a configurar una conexión en nuestro cliente MySQL Workbench para trabajar con un servidor remoto y no en el mismo ordenador, y posteriormente vamos a trabajar la administración del SGBD.

2.- Ejercicio 1. Configuración conexión en MySQL Workbench

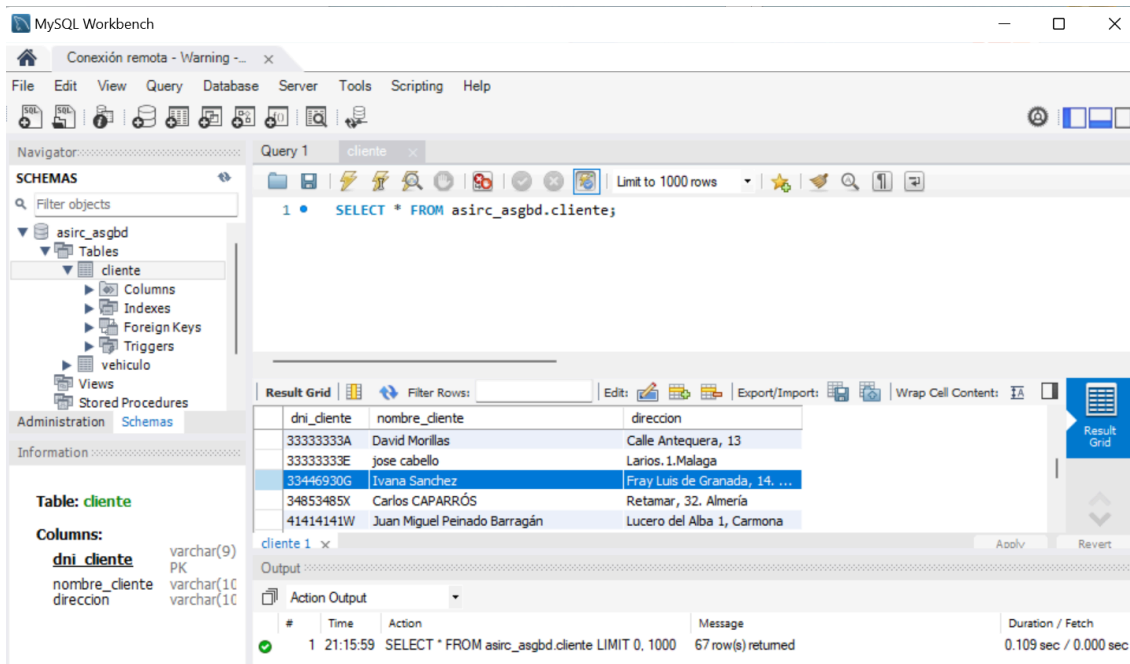
En primer lugar, abrimos el MySQL para comenzar con las instrucciones que pide el ejercicio.

Al lado de donde pone MySQL Connections hay un símbolo , que es el que vamos a pinchar para crear la conexión con los datos que se nos ha facilitado. Una vez introducidos en la ventana que se ha abierto y, que el test de conexión es favorable, le daremos al botón  para continuar.



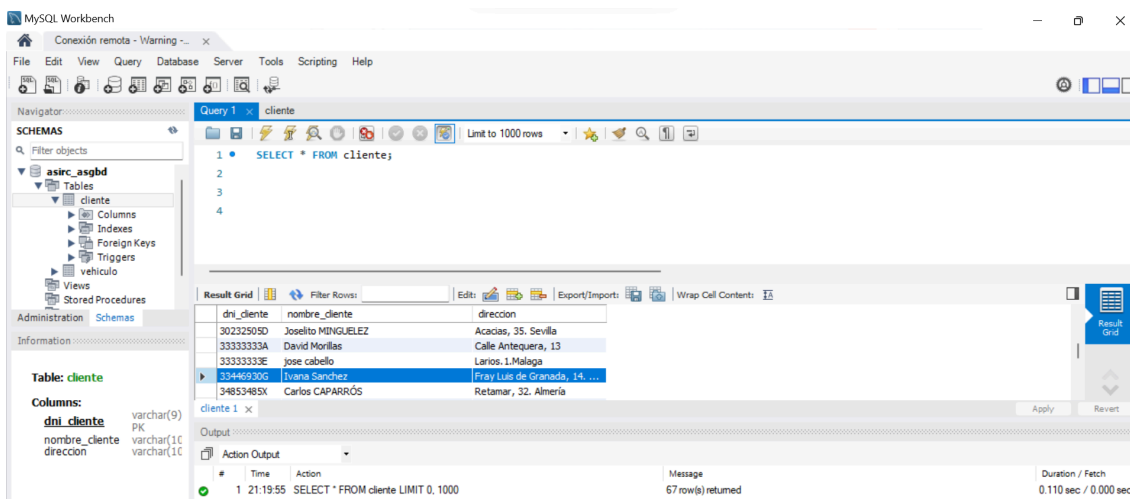
2.1.- Insertar fila

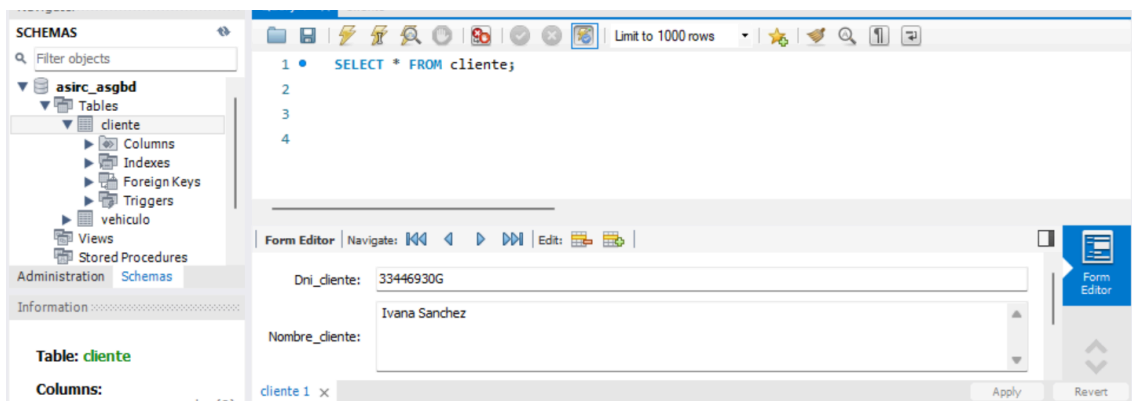
Insertamos una fila en la tabla cliente. Para ello, en el desplegable de la izquierda, pincho en **asirc_asgbd** **Tables** **cliente** (con el botón derecho del ratón) **Select Rows** **Limit-1000** y debajo de la última línea del listado que ha aparecido, escribimos nuestra línea y **Apply** **Apply** (en la ventana que se abre).



2.2.- Consulta listado clientes

Para consultar dónde aparece la fila añadida: **SELECT * FROM cliente;**





3.- Ejercicio 2. Crear un usuario

Abrimos el terminal de MySQL y procedemos a crear el usuario **tarea2user** cuya contraseña debe cambiarse la primera vez que se acceda a él.

```

MySQL 8.0 Command Line Cli
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 147
Server version: 8.0.39 MySQL Community Server - GPL

Copyright (c) 2000, 2024, 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> CREATE USER 'tarea2user'@'%' IDENTIFIED BY '111111';
Query OK, 0 rows affected (0.02 sec)

mysql> ALTER USER 'tarea2user'@'%' PASSWORD EXPIRE;
Query OK, 0 rows affected (0.01 sec)

mysql>

```

En el terminal de Windows, para entrar en sql, lo hacemos con el comando **"C:\Program Files\MySQL\MySQL Server 8.0\bin\mysql.exe" "--defaults-file=C:\ProgramData\MySQL\MySQL Server 8.0\my.ini" "-unombre_usuario" "-p"**


```
Símbolo del sistema
Microsoft Windows [Versión 10.0.22631.4317]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\IVANA>

C:\Users\IVANA> "C:\Program Files\MySQL\MySQL Server 8.0\bin\mysql.exe" "--defaults-file=C:\Program
er 8.0\my.ini" "-utarea2user" "-p"
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 156
Server version: 8.0.39

Copyright (c) 2000, 2024, 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> exit
Bye
C:\Users\IVANA>
```

A la hora de ejecutar un comando nos avisa del cambio de contraseña el cual, procedemos a realizarlo con el comando ***ALTER USER 'tarea2user'@'%' IDENTIFIED BY 'nueva_contraseña';***

```
mysql> SELECT * FROM distribuidora.articulos;
ERROR 1820 (HY000): You must reset your password using ALTER USER statement
fore executing this statement.

mysql> ALTER USER 'tarea2user'@'%' IDENTIFIED BY 'usuario';
Query OK, 0 rows affected (0.03 sec)

mysql> |
```

4.- Ejercicio 3. Creación y ejecución del script de una base de datos

4.1.- Descargar el archivo ***distribuidora.sql***

Abierto el MySQL y, como usuario **root** en la instalación local, cargamos el archivo ***distribuidora.sql***. Para ello lo primero que debemos hacer es descargar en nuestro equipo dicho archivo cambiando la extensión a sólo **.sql** y, seguidamente, en el terminal, utilizar el comando con la ruta correspondiente: ***SOURCE C:/Users/IVANA/Desktop/distribuidora.sql;*** para ejecutar la descarga.


```
MySQL 8.0 Command Line Cli  x  +  v

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> SOURCE C:/Users/IVANA/Desktop/distribuidora.sql;
Query OK, 1 row affected (0.01 sec)

Database changed
Query OK, 0 rows affected (0.02 sec)

Query OK, 0 rows affected (0.02 sec)

Query OK, 0 rows affected (0.04 sec)

Query OK, 0 rows affected (0.04 sec)

Query OK, 3 rows affected (0.00 sec)
Records: 3  Duplicates: 0  Warnings: 0

Query OK, 3 rows affected (0.00 sec)
Records: 3  Duplicates: 0  Warnings: 0

Query OK, 5 rows affected (0.00 sec)
Records: 5  Duplicates: 0  Warnings: 0

Query OK, 9 rows affected (0.00 sec)
Records: 9  Duplicates: 0  Warnings: 0

mysql> |
```

4.2.- Concesión de permiso

Otorgamos el permiso de consulta sobre la tabla artículos de la base de datos **distribuidora** al usuario **tarea2user**, y para que los cambios surtan efecto de inmediato ejecutaremos la sentencia **FLUSH PRIVILEGES**;

```
mysql> GRANT SELECT ON distribuidora.articulos TO 'tarea2user'@'%';
Query OK, 0 rows affected (0.01 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.01 sec)

mysql> |
```

4.3.- Verificación del permiso

Para verificar si tenemos acceso a la tabla artículos, realizaremos la siguiente consulta: **SELECT * FROM distribuidora.articulos**; que nos devolverá la tabla correspondiente


```
mysql> SELECT * FROM distribuidora.articulos;
```

IDArticulo	NombreArticulo	Descripcion	Precio	IDProveedor
101	Lámpara LED	Lámpara LED de 60 vatios, luz blanca	19.99	1
102	Teléfono Inteligente	Teléfono Android de última generación	299.99	2
103	Caja de Herramientas	Caja de herramientas de metal con 150 piezas	79.99	1
104	Cafetera Automática	Cafetera programable de acero inoxidable	49.99	2
105	Botas de Senderismo	Botas de senderismo impermeables, tallas 7-12	89.99	3

```
5 rows in set (0.00 sec)

mysql>
```

5.- Ejercicio 4. Crear tarea2vista

5.1.- Concesión de permisos para crear vistas

Conectamos como **root** en MySQL y otorgamos los permisos necesarios a **tarea2user** con la sentencia **GRANT CREATE VIEW ON distribuidora.* TO 'tarea2user'@'%'**;-. Aplicamos los cambios con **FLUSH PRIVILEGES**;

```
mysql> GRANT CREATE VIEW ON distribuidora.* TO 'tarea2user'@'%';
Query OK, 0 rows affected (0.01 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.01 sec)
```

También le vamos a dar permisos sobre todas las tablas, pues nos va a hacer falta para la creación de la vista.

```
mysql> GRANT CREATE VIEW ON distribuidora.proveedores TO 'tarea2user'@'%';
Query OK, 0 rows affected (0.01 sec)

mysql> GRANT CREATE VIEW ON distribuidora.articulos TO 'tarea2user'@'%';
Query OK, 0 rows affected (0.01 sec)

mysql> GRANT CREATE VIEW ON distribuidora.articuloscentrosdistribucion TO 'tarea2user'@'%';
Query OK, 0 rows affected (0.01 sec)

mysql> GRANT CREATE VIEW ON distribuidora.centrosdistribucion TO 'tarea2user'@'%';
Query OK, 0 rows affected (0.02 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.03 sec)

mysql>
```

5.2.- Crear la vista

Una vez que el usuario **tarea2user** tiene los permisos necesarios, puede conectarse y crear la vista: **tarea2vista**.


```
mysql> USE distribuidora;
Database changed
mysql> CREATE VIEW tarea2vista AS
-> SELECT
->     a.NombreArticulo AS articulos,
->     p.NombreProveedor AS proveedores,
->     acd.CantidadEnCentro AS cantidadEnExistencias
-> FROM
->     distribuidora.articulos a
-> JOIN
->     distribuidora.proveedores p ON a.IDProveedor = p.IDProveedor
-> JOIN
->     distribuidora.articuloscentrosdistribucion acd ON a.IDArticulo = acd.IDArticulo;
Query OK, 0 rows affected (0.04 sec)

mysql>
```

Hemos asignado alias a las tablas (a, p, acd) para facilitar su escritura y su lectura a la hora de seleccionar los datos que se nos piden y, luego, hemos creado las relaciones entre las tablas, algo necesario para unir los datos de distintas tablas basadas en una clave común, en este caso la clave primaria o foránea (ID).

Una vez que la vista **tarea2vista** se ha creado correctamente, la ejecutamos simplemente con: **SELECT * FROM tarea2vista;**

```
mysql> SELECT * FROM tarea2vista;
+-----+-----+-----+
| articulos | proveedores | cantidadEnExistencias |
+-----+-----+-----+
| Lámpara LED | Suministros Iberia | 50 |
| Lámpara LED | Suministros Iberia | 50 |
| Caja de Herramientas | Suministros Iberia | 100 |
| Caja de Herramientas | Suministros Iberia | 100 |
| Teléfono Inteligente | Distribuidora Hispania | 20 |
| Teléfono Inteligente | Distribuidora Hispania | 30 |
| Cafetera Automática | Distribuidora Hispania | 40 |
| Botas de Senderismo | Proveedora Andaluza | 60 |
| Botas de Senderismo | Proveedora Andaluza | 60 |
+-----+-----+-----+
9 rows in set (0.01 sec)
```


6.- Ejercicio 5. Insertar una fila en tarea2vista

Intentamos insertar una fila en **tarea2vista** y para ello utilizamos la sentencia con **INSERT INTO** (antes le daremos permiso para insertar).

```
mysql> GRANT INSERT ON distribuidora.tarea2vista TO 'tarea2user'@'%';
Query OK, 0 rows affected (0.02 sec)

mysql> |
```

```
mysql> use distribuidora;
Database changed
mysql> INSERT INTO tarea2vista (a, p, acd)
-> VALUES ('Enchufe Wifi', 'Electrica Gala', 100);
ERROR 1356 (HY000): View 'distribuidora.tarea2vista' references invalid table(s) or column(s) or function(s) or definer/invoke
r of view lack rights to use them
mysql> |
```

No es posible insertar en una vista, ya que las vistas suelen ser sólo para lectura, por lo que solamente podemos modificar los datos directamente en las tablas base, no en las vistas derivadas.

7.- Ejercicio 6. Creación de un rol con sus permisos

7.1.- Crear un rol

Para lograr la creación del **rol**, su asignación y posterior verificación, iniciamos una sesión con la cuenta **root** y creamos el rol que incluya los permisos necesarios sobre la tabla proveedores (selección, modificación, inserción y borrado).

```
mysql> CREATE ROLE 'actualiza_proveedor';
Query OK, 0 rows affected (0.02 sec)

mysql> GRANT SELECT, INSERT, UPDATE, DELETE ON distribuidora.proveedores TO 'actualiza_proveedor';
Query OK, 0 rows affected (0.01 sec)
```

7.2.- Asignación del rol

Asignamos el **rol** al usuario **tarea2user** y lo activamos para que esté disponible cuando el usuario inicie la sesión:

```
mysql> GRANT 'actualiza_proveedor' TO 'tarea2user'@'%';
Query OK, 0 rows affected (0.01 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.02 sec)

mysql>
```


7.3.- Verificación del rol

Cambiamos de cuenta y verificamos que el rol ha sido asignado correctamente con la sentencia **SHOW GRANTS FOR CURRENT_USER;**

```
mysql> SHOW GRANTS FOR CURRENT_USER;
+-----+
| Grants for tarea2user@% |
+-----+
| GRANT USAGE ON *.* TO `tarea2user`@`%` |
| GRANT CREATE VIEW ON `distribuidora`.* TO `tarea2user`@`%` |
| GRANT SELECT, CREATE VIEW ON `distribuidora`.`articulos` TO `tarea2user`@`%` |
| GRANT SELECT, CREATE VIEW ON `distribuidora`.`articuloscentrosdistribucion` TO `tarea2user`@`%` |
| GRANT CREATE VIEW ON `distribuidora`.`centrosdistribucion` TO `tarea2user`@`%` |
| GRANT SELECT, CREATE VIEW ON `distribuidora`.`proveedores` TO `tarea2user`@`%` |
| GRANT SELECT, INSERT ON `distribuidora`.`tarea2vista` TO `tarea2user`@`%` |
| GRANT| `actualiza_proveedor`@`%` TO `tarea2user`@`%` |
+-----+
8 rows in set (0.00 sec)
```

8.- Ejercicio 7. Activación de rol y borrado de filas.

8.1.- Activación del rol

Iniciamos la sesión como tare2user y activamos el rol actualiza-proveedor con el comando **SET ROLE 'actualiza_proveedor';**

```
mysql> SET ROLE 'actualiza_proveedor';
Query OK, 0 rows affected (0.01 sec)

mysql>
```

8.2.- Borrado de filas

Probamos a borrar una fila de la tabla.

```
mysql> use distribuidora;
Database changed
mysql> DELETE FROM distribuidora.proveedores WHERE IDProveedor = 1;
ERROR 1451 (23000): Cannot delete or update a parent row: a foreign key constraint fails (`distribuidora`.`articulos`, CONSTRAINT `articulos_ibfk_1` FOREIGN KEY (`IDProveedor`) REFERENCES `proveedores` (`IDProveedor`))
mysql>
```

Observamos que da un mensaje de error por el que nos dice que no se puede eliminar la fila que le hemos pedido que borre.

Este error se produce debido a una **restricción de integridad referencial** en tu base de datos. Esto quiere decir que hay otra tabla, llamada **"artículos"** en este caso, que tiene una columna **"IDProveedor"** que hace

referencia a la columna **"IDProveedor"** de la tabla **"proveedores"**. Esto crea una relación entre las dos tablas, donde los valores de **"IDProveedor"** en **"artículos"** deben existir también en **"proveedores"**.

En el root desactivamos el rol, le damos permisos de borrar al usuario tarea2user y volvemos a intentar la eliminación de la fila de proveedores sin tener el rol activo.

```
mysql> SET ROLE NONE;
Query OK, 0 rows affected (0.00 sec)

mysql> GRANT DELETE ON distribuidora.proveedores TO 'tarea2user'@'%';
Query OK, 0 rows affected (0.01 sec)
```

```
mysql>
mysql> DELETE FROM distribuidora.proveedores WHERE IDProveedor = 1;
ERROR 1451 (23000): Cannot delete or update a parent row: a foreign key constraint fails ('distribuidora`.`articulos`, CONSTRAINT `articulos_ibfk_1` FOREIGN KEY (`IDProveedor`) REFERENCES `proveedores` (`IDProveedor`))
mysql>
```

Y Ocurre lo mismo. Nos da el mismo error a la hora de intentar borrarlo.

9.- Ejercicio 8. Alteración de usuario

Entrando en MySQL como **root**, alteramos el usuario tarea2user para que su rol por defecto sea **actualiza_proveedor**.

```
mysql> ALTER USER 'tarea2user'@'%' DEFAULT ROLE 'actualiza_proveedor';
Query OK, 0 rows affected (0.02 sec)

mysql>
```

Cerramos y accedemos de nuevo con **tarea2user** para observar ejecutando el comando **SHOW GRANTS FOR 'tarea2user'@'%'**; el cual nos mostrará todos los roles que tiene el usuario, además de los privilegios que tiene otorgados.

```
mysql> SHOW GRANTS FOR 'tarea2user'@'%';
+-----+
| Grants for tarea2user@% |
+-----+
| GRANT USAGE ON *.* TO `tarea2user`@`%` |
| GRANT CREATE VIEW ON `distribuidora`.* TO `tarea2user`@`%` |
| GRANT SELECT, CREATE VIEW ON `distribuidora`.`articulos` TO `tarea2user`@`%` |
| GRANT SELECT, CREATE VIEW ON `distribuidora`.`articuloscentrosdistribucion` TO `tarea2user`@`%` |
| GRANT CREATE VIEW ON `distribuidora`.`centrosdistribucion` TO `tarea2user`@`%` |
| GRANT SELECT, DELETE, CREATE VIEW ON `distribuidora`.`proveedores` TO `tarea2user`@`%` |
| GRANT SELECT, INSERT ON `distribuidora`.`tarea2vista` TO `tarea2user`@`%` |
| GRANT `actualiza_proveedor`@`%` TO `tarea2user`@`%` |
+-----+
8 rows in set (0.01 sec)
```


Este tipo de motor no gestiona los bloqueos de tablas, sino que lo hace a nivel de filas, por lo que el comando LOCK TABLES en InnoDB es más limitado, y aunque se ejecute, aún se pueden leer las tablas bloqueadas, es decir, que el bloqueo sólo afecta a las operaciones de escritura.

Ante esto, para solucionar el ejercicio, tenemos que usar transacciones, que bloqueará todas las líneas de las tablas.

```
mysql> use distribuidora;
Database changed
mysql> START TRANSACTION;
Query OK, 0 rows affected (0.00 sec)

mysql> SELECT * FROM proveedores FOR UPDATE;
+-----+-----+-----+-----+-----+
| IDProveedor | NombreProveedor | Direccion | Telefono | Email |
+-----+-----+-----+-----+-----+
| 1 | Suministros Iberia | Calle Gran Vía, 123, Madrid, 28001 | +34 91 555 1234 | info@suministrosiberia.com |
| 2 | Distribuidora Hispania | Paseo del Roble, 456, Granada, 18005 | +34 958 987 678 | contacto@hispania.com |
| 3 | Proveedora Andaluza | Calle Real, 789, Sevilla, 41001 | +34 95 333 2222 | ventas@proveedoraandaluza.com |
+-----+-----+-----+-----+-----+
3 rows in set (0.02 sec)
```

Creada ya la transacción, cerramos, nos conectamos como **tarea2user** y verificamos que no podemos realizar UN UPDATE o un DELETE (escrituras) en las filas bloqueadas. Como se observa en la imagen, al intentar hacer un cambio de teléfono, no nos deja.

```
mysql> USE distribuidora;
Database changed
mysql> SELECT * FROM proveedores;
+-----+-----+-----+-----+-----+
| IDProveedor | NombreProveedor | Direccion | Telefono | Email |
+-----+-----+-----+-----+-----+
| 1 | Suministros Iberia | Calle Gran Vía, 123, Madrid, 28001 | +34 91 555 1234 | info@suministrosiberia.com |
| 2 | Distribuidora Hispania | Paseo del Roble, 456, Granada, 18005 | +34 958 987 678 | contacto@hispania.com |
| 3 | Proveedora Andaluza | Calle Real, 789, Sevilla, 41001 | +34 95 333 2222 | ventas@proveedoraandaluza.com |
+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> UPDATE proveedores SET Telefono = '+34 91 555 1234' WHERE IDProveedor = 1;
ERROR 1142 (42000): UPDATE command denied to user 'tarea2user'@'localhost' for table 'proveedores'
mysql> DELETE FROM proveedores WHERE IDProveedor = 3;
```

Volvemos al **root**, desbloqueamos la tabla con el comando COMMIT; , y al cambiar de usuario veremos que ya podemos escribir en la tabla.

```
mysql> USE distribuidora;
Database changed
mysql> COMMIT;
Query OK, 0 rows affected (0.00 sec)

mysql> |
```

```
mysql> USE distribuidora;
Database changed
mysql> SELECT * FROM proveedores;
+-----+-----+-----+-----+-----+
| IDProveedor | NombreProveedor | Direccion | Telefono | Email |
+-----+-----+-----+-----+-----+
| 1 | Suministros Iberia | Calle Gran Vía, 123, Madrid, 28001 | +34 91 555 1234 | info@suministrosiberia.com |
| 2 | Distribuidora Hispania | Paseo del Roble, 456, Granada, 18005 | +34 958 987 678 | contacto@hispania.com |
| 3 | Proveedora Andaluza | Calle Real, 789, Sevilla, 41001 | +34 95 333 2222 | ventas@proveedoraandaluza.com |
+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> UPDATE proveedores SET Telefono = '34 91 345 6785' WHERE IDProveedor = 1;
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0
mysql> SELECT * FROM proveedores;
+-----+-----+-----+-----+-----+
| IDProveedor | NombreProveedor | Direccion | Telefono | Email |
+-----+-----+-----+-----+-----+
| 1 | Suministros Iberia | Calle Gran Vía, 123, Madrid, 28001 | 34 91 345 6785 | info@suministrosiberia.com |
| 2 | Distribuidora Hispania | Paseo del Roble, 456, Granada, 18005 | +34 958 987 678 | contacto@hispania.com |
| 3 | Proveedora Andaluza | Calle Real, 789, Sevilla, 41001 | +34 95 333 2222 | ventas@proveedoraandaluza.com |
+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql>
```