

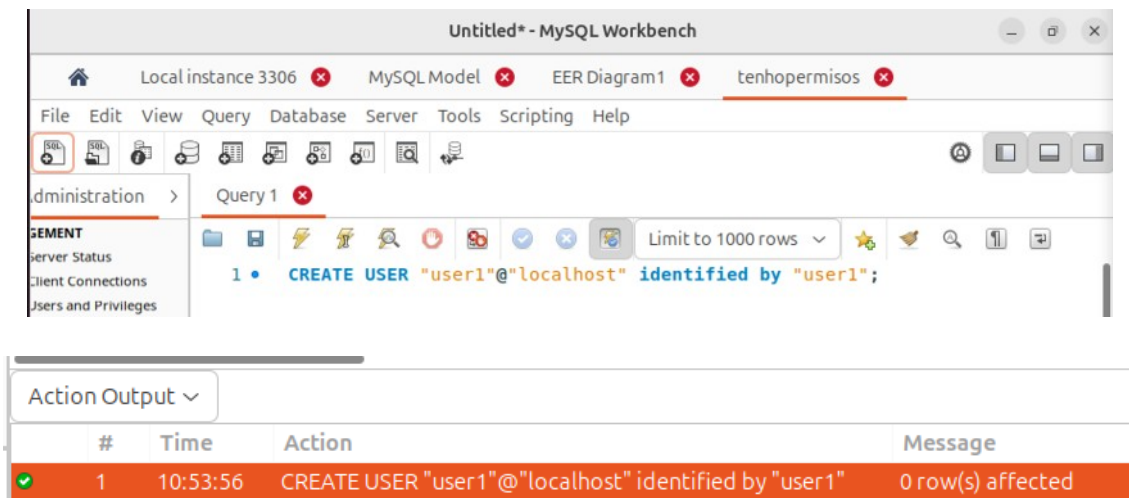
Exercicios Xestión de permisos

Nome e apelidos: Rubén Rey Feal

1. Crea un usuario 'tenhopermisos' e dale permisos para que poida crear usuarios.

```
79 • CREATE USER "tenhopermisos"@"localhost" identified by "tenhopermisos";  
80 • GRANT CREATE USER on *.* TO "tenhopermisos"@"localhost";
```

2. Conectado como 'tenhopermisos' crea un novo usuario 'user1'.



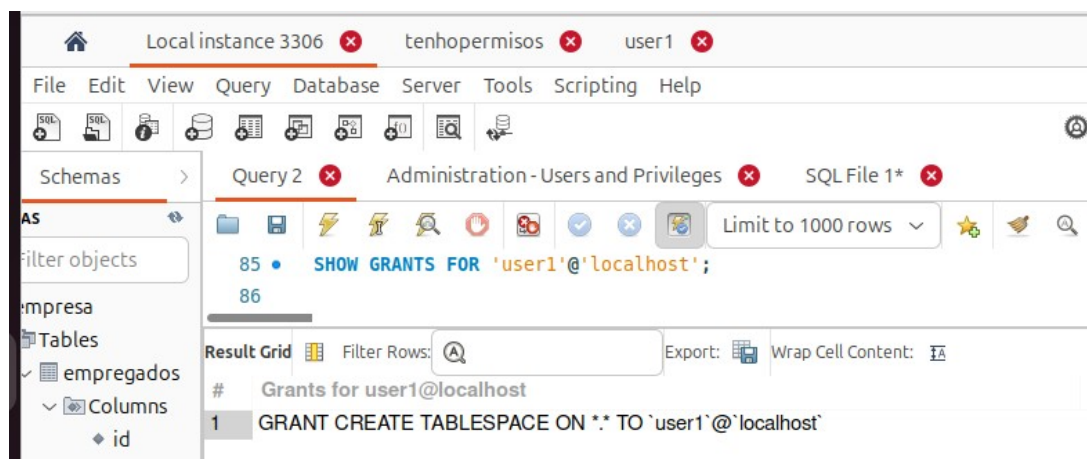
3. Conectado como 'root', dale permisos ao usuario 'user1' para que poida crear tablespaces.

```
82 • GRANT CREATE tablespace on *.* TO "user1"@"localhost";
```

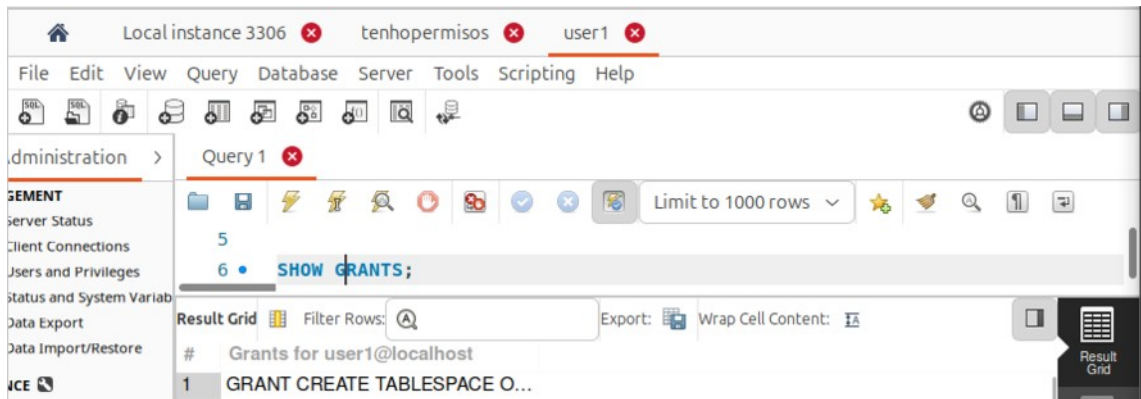
4. Conectado como 'user1' comproba que dispón de estos permisos executando sentenzas SQL que necesiten ter o permiso outorgado.

```
1 • CREATE TABLESPACE novo_tablespace  
2 ADD DATAFILE '/var/lib/mysql/novo_tablespace.ibd'  
3 ENGINE = InnoDB;  
4
```

5. Conectado como 'root', amosa os permisos que ten o usuario 'user1'.

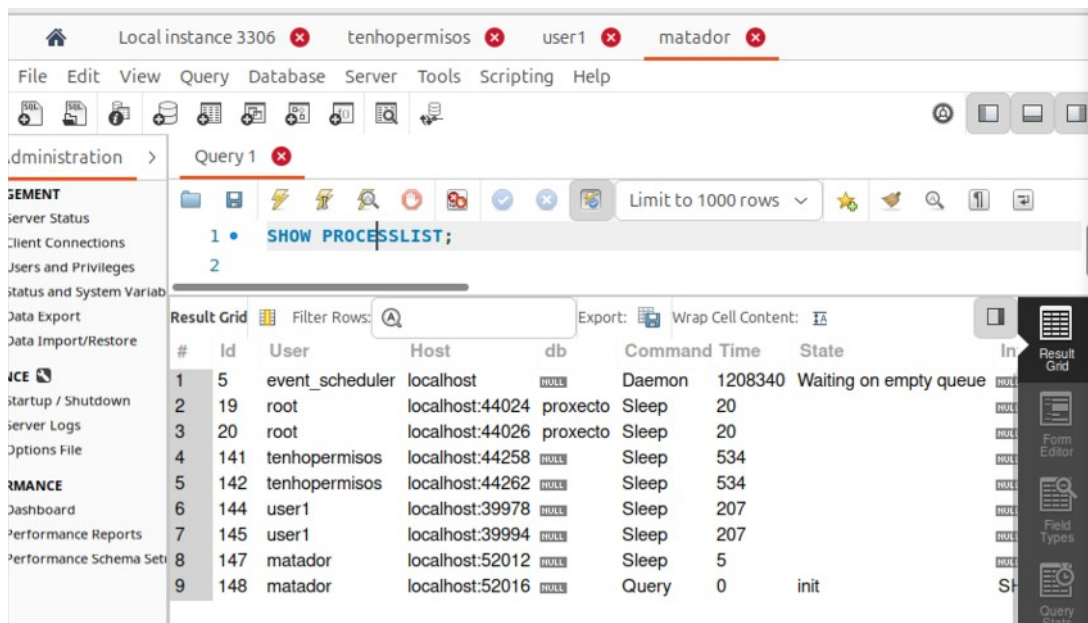


6. Conectado como 'user1' amosa os permisos que posúe e comproba que son os da orde anterior.



7. Conectado como 'root' crea un usuario 'matador' que teña permisos para poder eliminar conexións activas e poida ver as conexións activas de todos os usuarios co servidor.

- **CREATE USER "matador"@"localhost" identified by "matador";**
- **GRANT PROCESS, SUPER ON *.* TO 'matador'@'localhost';**
- **FLUSH PRIVILEGES;**



8. Conectado como 'root' outorga permiso de creación e borrado de procedementos, así como de execución ao usuario 'user1' sobre unha base de datos creada previamente.

- ```
105 • GRANT CREATE ROUTINE, ALTER ROUTINE, DROP , EXECUTE ON rubenrf.* TO 'user1'@'localh
106 • FLUSH PRIVILEGES;
107
```

9. Conectado como 'user1' crea un procedimiento almacenado na base de datos indicada no paso anterior, coas seguintes ordes SQL:

```
1 DELIMITER $$
2 DROP PROCEDURE IF EXISTS test_mysql_while_loop$$
3 CREATE PROCEDURE test_mysql_while_loop()
4 BEGIN
5 DECLARE x INT;
6
7 SET x = 1;
8
9 WHILE x >= 0 DO
10 set x = x+1;
11 END WHILE;
12
13 END$$
14 DELIMITER ;
```

Este procedimiento crea un bucle infinito.

The screenshot shows the MySQL Workbench interface. The top toolbar includes icons for file operations, query execution, and server management. The left sidebar displays the 'Administration' tree with categories like 'Server Status', 'Users and Privileges', and 'Performance'. The main query editor window, titled 'Query 1', contains the following SQL code:

```
1 • use rubenrf;
2 DELIMITER $$
3 • DROP PROCEDURE IF EXISTS test_mysql_while_loop$$
4 • CREATE PROCEDURE test_mysql_while_loop()
5 BEGIN
6 DECLARE x INT;
7
8 SET x = 1;
9
10 • WHILE x >= 0 DO
11 set x = x+1;
12 END WHILE;
13
14 END$$
15 DELIMITER ;
```

Below the query editor, the 'Action Output' tab shows the execution results:

| #  | Time     | Action                                              | Message                  |
|----|----------|-----------------------------------------------------|--------------------------|
| 19 | 11:21:05 | DROP PROCEDURE IF EXISTS test_mysql_while_loop      | 1305 PROCEDURE rubenrf.t |
| 20 | 11:21:05 | CREATE PROCEDURE test_mysql_while_loop() BEGIN D... | 0 row(s) affected        |

10. Conéctate como 'user1' e executa o procedemento coa orde SQL: **call test\_mysql\_while\_loop** Nota: Se che dá algún tipo de erro, proba a pechar a conexión e volver a conectarte.

```
mysql> use rubenrf
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> call test_mysql_while_loop;
```

11. Conectado como 'matador' identifica o proceso e mátaló.

**SHOW PROCESSLIST;**

| #  | Id  | User            | Host            | db       | Command | Time  | State           | Info            |
|----|-----|-----------------|-----------------|----------|---------|-------|-----------------|-----------------|
| 1  | 5   | event_scheduler | localhost       | NULL     | Daemon  | 12... | Waiting on e... | NULL            |
| 2  | 19  | root            | localhost:44024 | proxecto | Sleep   | 260   |                 | NULL            |
| 3  | 20  | root            | localhost:44026 | proxecto | Sleep   | 260   |                 | NULL            |
| 4  | 141 | tenhopermisos   | localhost:44258 | NULL     | Sleep   | 174   |                 | NULL            |
| 5  | 142 | tenhopermisos   | localhost:44262 | NULL     | Sleep   | 174   |                 | NULL            |
| 6  | 145 | user1           | localhost:39994 | rubenrf  | Query   | 0     | NULL            | call test_mysql |
| 7  | 147 | matador         | localhost:52012 | NULL     | Sleep   | 92    |                 | NULL            |
| 8  | 148 | matador         | localhost:52016 | NULL     | Query   | 0     | init            | SHOW PROCES     |
| 9  | 150 | user1           | localhost:42838 | NULL     | Sleep   | 229   |                 | NULL            |
| 10 | 151 | user1           | localhost:42844 | rubenrf  | Query   | 0     | Opening tables  | call test_mysql |
| 11 | 154 | user1           | localhost       | rubenrf  | Query   | 0     | Opening tables  | call test_mysql |

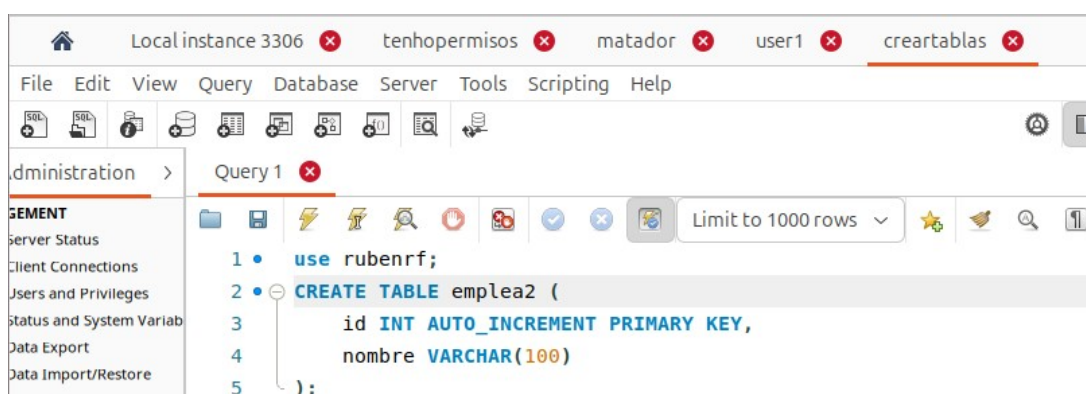
**KILL 154;**

12. Como farías para determinar cal é a consulta que está consumindo os recursos de Mysql ? Faino.

13. Conectado como 'root' crea un usuario de nome 'creartablas' que teña permisos para crear, borrar e modificar táboas dunha base de datos creada previamente.

```
108 • CREATE USER 'creartablas'@'localhost' IDENTIFIED BY 'creartablas';
109 • GRANT CREATE, ALTER, DROP ON rubenrf.* TO 'creartablas'@'localhost';
```

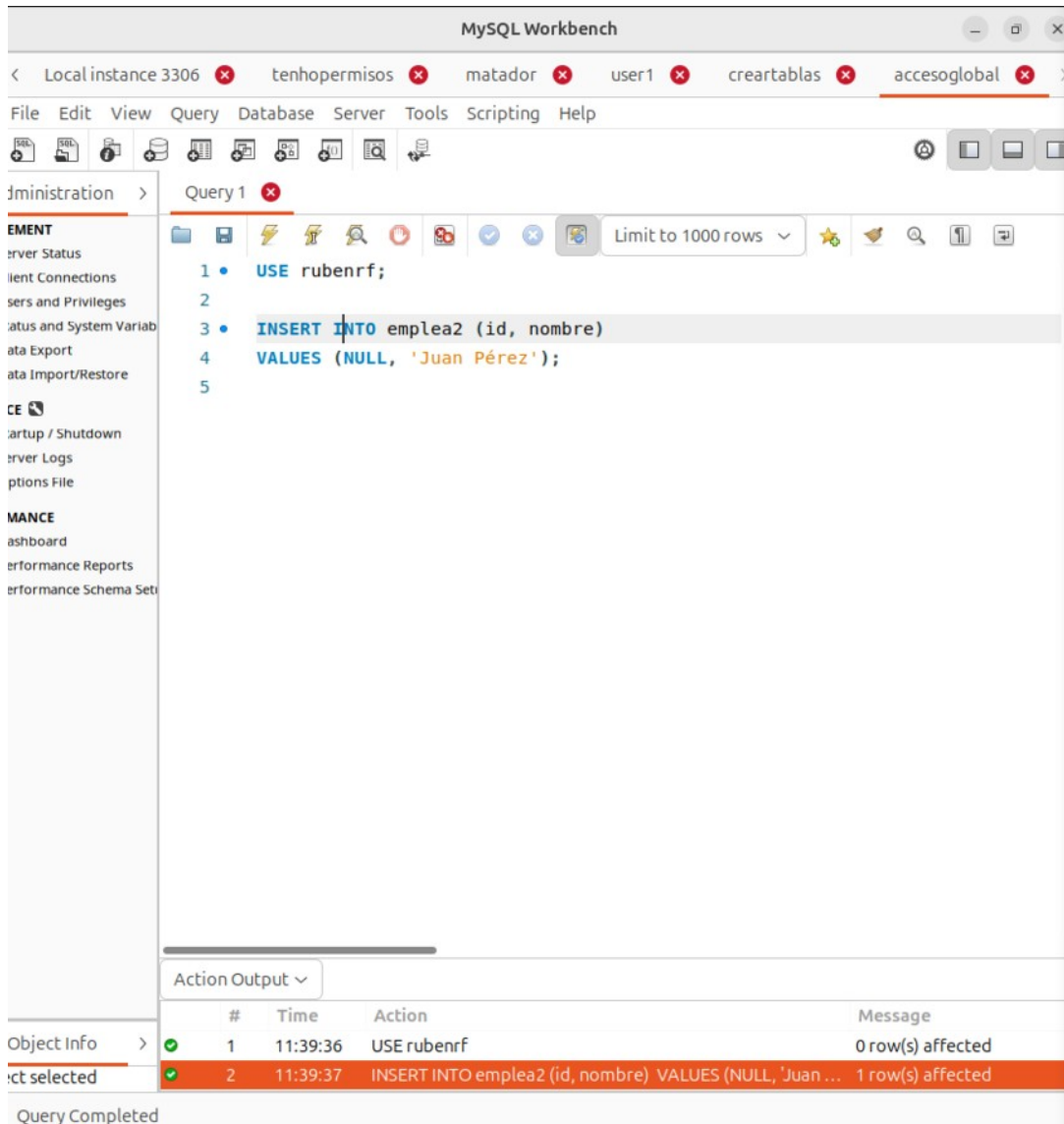
14. Conectado como 'creartaboas' crea un taboa sinxela de, polo menos, dúas columnas.



15. Conectado como 'root' crea un usuario de nome 'accesoglobal' que poida realizar operacións de selección e inserción sobre todas as táboas de todas as bases de datos.

```
115 • CREATE USER 'accesoglobal'@'localhost' IDENTIFIED BY 'accesoglobal';
116 • GRANT SELECT, INSERT ON *.* TO 'accesoglobal'@'localhost';
```

16. Conectado como 'accesoglobal' engade unha fila á táboa creada anteriormente. Intenta borrar a fila creada. Podes?

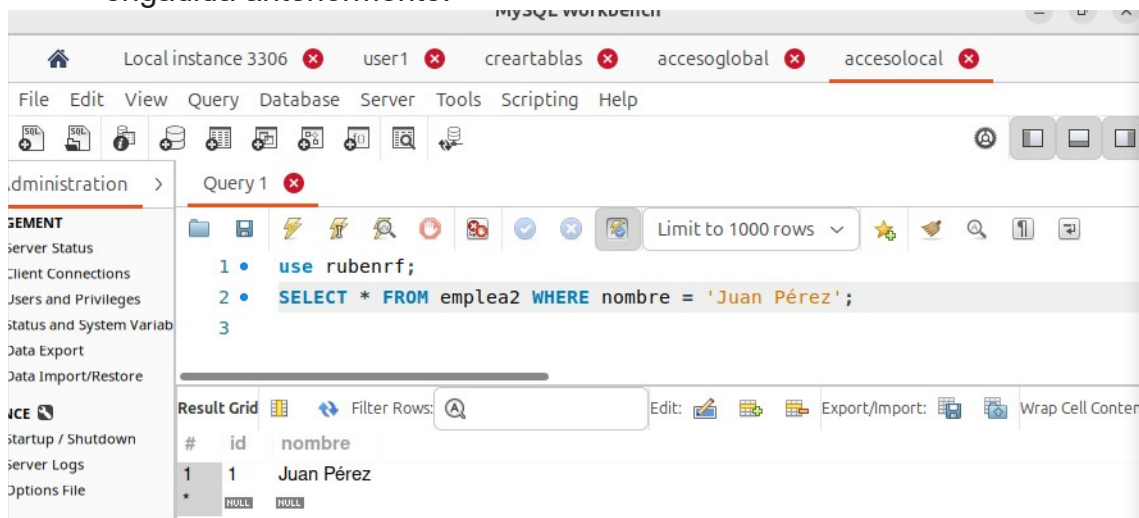


17. Conectado como 'root' crea un usuario de nome 'accesolocal' que poida seleccionar todas as táboas da base de datos anterior.

```
118 • CREATE USER 'accesolocal'@'localhost' IDENTIFIED BY 'accesolocal';
119 • GRANT SELECT ON rubenrf.* TO 'accesolocal'@'localhost';
```



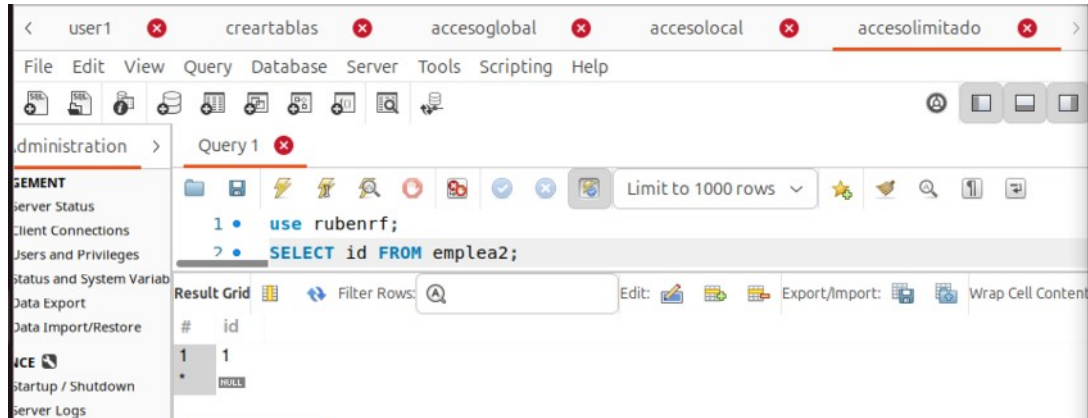
18. Conéctate como 'accesolocal' e comproba que podes seleccionar a fila engadida anteriormente.



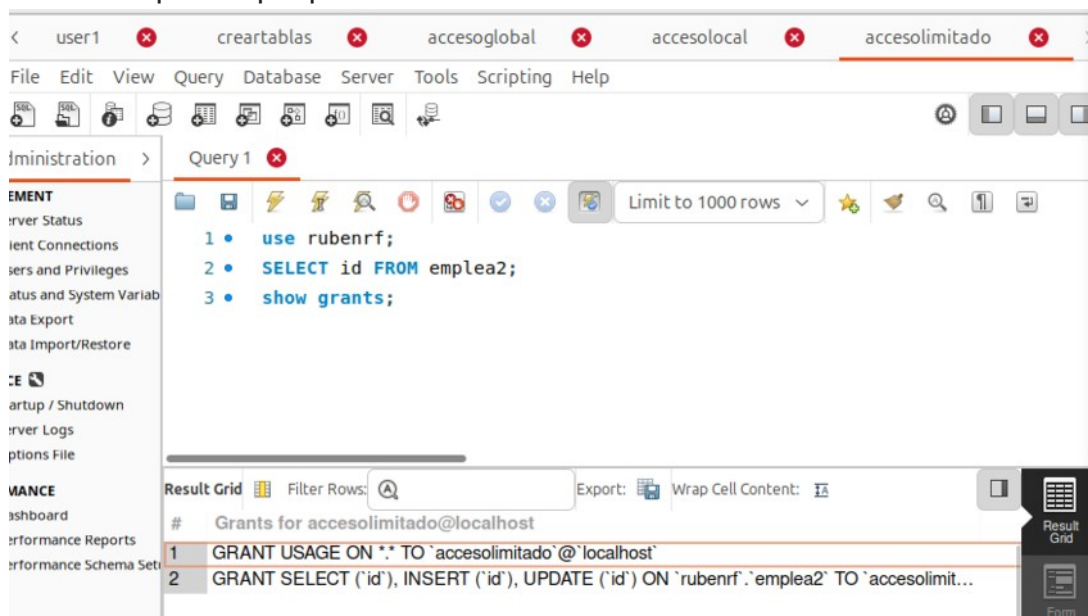
19. Conectado como 'root' crea un usuario de nome 'accesolimitado' que poida realizar operacións de inserción, actualización e selección sobre a primeira columna da táboa creada previamente.

```
1 • CREATE USER 'accesolimitado'@'localhost' IDENTIFIED BY 'accesolimitado';
2 • GRANT SELECT(id), INSERT(id), UPDATE(id) ON rubenrf.emplea2 TO 'accesolimitado'@'lo
3
```

20. Conéctate como 'accesolimitado' e comproba que ten os permisos executando as ordes SQL SELECT, UPDATE e INSERT.



21. Comproba que permisos ten.

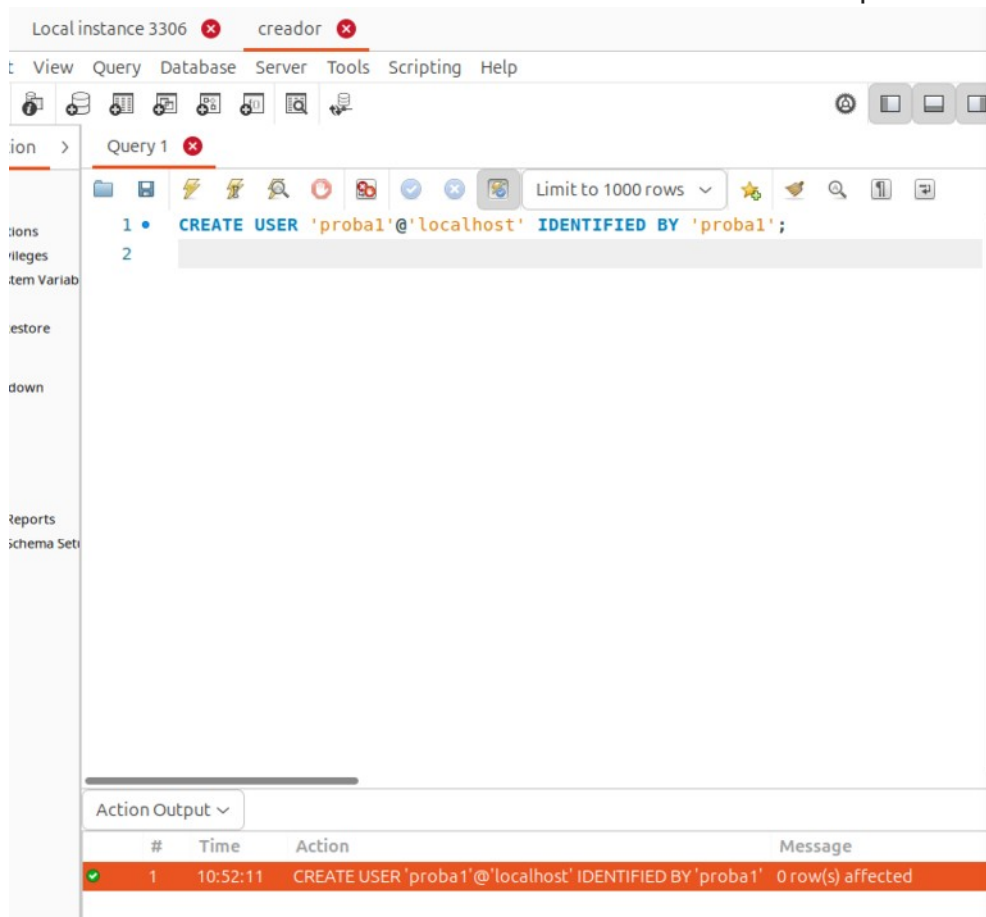


## Exercicios (grant / revoke / with grant option)

1. Conectado como root crea un usuario de nome 'creador' que teña permisos para crear usuarios.

```
137 • CREATE USER 'creador'@'localhost' IDENTIFIED BY 'password_creador';
138 • GRANT CREATE USER ON *.* TO 'creador'@'localhost' WITH GRANT OPTION;
```

2. Conectado como 'creador' crea un novo usuario de nome 'proba1'.



3. Conectado como root fai que teña permisos de selección e borrado a nivel global e todos os permisos sobre unha base de datos de exemplo creada previamente. Estes permisos poderán ser xestionados polo usuario.

```
139
140 • GRANT SELECT, DELETE ON *.* TO 'proba1'@'localhost';
141 • GRANT ALL PRIVILEGES ON rubenrf.* TO 'proba1'@'localhost' WITH GRANT OPTION;
142
```

4. Conectado como 'root' crea un usuario de nome 'proba' que teña permiso para actualizar unha táboa dunha base de datos creada previamente. Poderá xestionar este permiso.

```
142
143 • CREATE USER 'proba'@'localhost' IDENTIFIED BY 'proba';
144 • GRANT UPDATE ON rubenrf.usuarios TO 'proba'@'localhost' WITH GRANT OPTION;
```

5. Conectado como 'creador' crea un usuario 'proba' e 'proba4'

- 2 • `CREATE USER 'proba4'@'localhost' IDENTIFIED BY 'proba4';`
- 3 • `CREATE USER 'proba3'@'localhost' IDENTIFIED BY 'proba';`

6. Conectado como 'proba' outorga permiso de selección, actualización dunha columna dunha táboa (creada previamente na base de datos de exemplo) e execución de procedementos ao usuario 'proba' na base de datos de exemplo. Podes facelo? Por que?

Non podo por que non lle foi otorgado o permiso Grant a Proba

7. Conectado como 'root' fai que o usuario 'proba' poida parar o servizo mysql.



```
150
151 • GRANT SHUTDOWN ON *.* TO 'proba'@'localhost';
152
```

8. Impide que o usuario 'proba' poida xestionar os permisos outorgados a nivel global.

- `GRANT SHUTDOWN ON *.* TO proba @ localhost ,`
- `REVOKE GRANT OPTION ON *.* FROM 'proba'@'localhost';`

9. Comproba que aínda pode xestionar os permisos da base de datos.

```
154 • SHOW GRANTS FOR 'proba'@'localhost';
155
156
157
```

| result Grid |                                              | Filter Rows: <input type="text"/> | Export:  | Wrap Cell Content:  |
|-------------|----------------------------------------------|-----------------------------------|-----------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------|
| #           | Grants for proba@localhost                   |                                   |                                                                                               |                                                                                                          |
| 1           | GRANT SHUTDOWN ON *.* TO `proba`@`localhost` |                                   |                                                                                               |                                                                                                          |

10. Con qué usuario has realizado la operación ? Podes facelo conectado co usuario 'creador'? Por que?

Non, porque o usuario creador non ten privilexios para xestionar permisos globais.

11. Quita todos os permisos ao usuario 'proba'. Podes facelo conectado como 'creador'? Por que?

```
56 • REVOKE ALL PRIVILEGES, GRANT OPTION FROM 'proba'@'localhost';
57
```

Non podes facelo como creador porque non ten privilexios globais.

12. Quita os permisos 'específicos' outorgados a cada un dos usuarios anteriores, comprobando coa orde SQL **SHOW GRANTS** que realmente foron eliminados.



```

158 • REVOKE SELECT, DELETE ON *.* FROM 'proba'@'localhost';
159 • REVOKE UPDATE ON rubenrf.usuarios FROM 'proba'@'localhost';
160 • SHOW GRANTS FOR 'proba'@'localhost';
161
162

```

Result Grid | Filter Rows:  | Export: | Wrap Cell Content:

| # | Grants for proba@localhost       |
|---|----------------------------------|
| 1 | GRANT USAGE ON *.* TO `proba`... |

## Exercicios sobre roles

### 1. Crear un rol básico

- Crea un rol chamado backup\_manager que inclúa os permisos necesarios para realizar copias de seguridade de todas as bases de datos.
- Asigna este rol ao usuario backup\_user@localhost.
- Activa o rol por defecto para que backup\_user dispoña sempre dos permisos ao conectarse.

```

166 • CREATE USER 'backup_manager'@'localhost' IDENTIFIED BY 'backup_manager';
167 • CREATE ROLE 'backup_manager';
168 • GRANT SELECT, RELOAD, LOCK TABLES ON *.* TO 'backup_manager';
169 • GRANT 'backup_manager' TO 'backup_user'@'localhost';
170 • SET DEFAULT ROLE 'backup_manager' TO 'backup_user'@'localhost';

```

### 2. Xestión de roles e permisos

- Crea un rol chamado db\_maintenance que inclúa permisos para crear, modificar e borrar táboas na base de datos empresa.
- Asigna este rol ao usuario developer@localhost.
- Conéctate como developer e comproba que podes executar as accións indicadas sobre táboas da base de datos empresa.

```

172 • CREATE USER 'developer'@'localhost' IDENTIFIED BY 'developer';
173 • CREATE ROLE 'db_maintenance';
174 • GRANT CREATE, ALTER, DROP ON empresa.* TO 'db_maintenance';
175 • GRANT 'db_maintenance' TO 'developer'@'localhost';

```

Local instance 3306 x creador x developer x

File Edit View Query Database Server Tools Scripting Help

Administration > Query 1 x

1 • USE empresa;

2 • CREATE TABLE test (id INT);

3 • ALTER TABLE test ADD COLUMN name VARCHAR(50);

4 • DROP TABLE test;

Limit to 1000 rows v

|   |   |          |               |                                                           |             |
|---|---|----------|---------------|-----------------------------------------------------------|-------------|
| ✓ | 2 | 11:08:25 | USE empresa   | 0 row(s) affected                                         | 0,00077 sec |
| ✓ | 3 | 11:08:25 | CREATE TAB... | 0 row(s) affected                                         | 0,095 sec   |
| ✓ | 4 | 11:08:25 | ALTER TABL... | 0 row(s) affected<br>Records: 0 Duplicates: 0 Warnings: 0 | 0,150 sec   |
| ✓ | 5 | 11:08:25 | DROP TABLE... | 0 row(s) affected                                         | 0,094 sec   |

### 3. Roles combinados

- Define dous roles:
  - read\_access: que permita seleccionar datos de calquera táboa na base de datos ventas.
  - write\_access: que permita engadir e modificar filas na táboa clientes da base de datos ventas.
- Asigna ambos roles ao usuario sales\_user@%.
- Activa só o rol read\_access para o usuario sales\_user por defecto.
- Conéctate como sales\_user e comproba que inicialmente só podes seleccionar datos. Activa o rol write\_access e comproba que podes realizar insercións e actualizacións.

```

186 • CREATE USER 'sales_user2'@'localhost' IDENTIFIED BY 'sales_user2';
187
188 • CREATE ROLE 'read_access';
189 • GRANT SELECT ON empresa.* TO 'read_access';
190
191 • CREATE ROLE 'write_access';
192 • GRANT INSERT, UPDATE ON empresa.* TO 'write_access';
193
194 • GRANT 'read_access', 'write_access' TO 'sales_user2'@'localhost';
195 • SET DEFAULT ROLE 'read access' TO 'sales user2'@'localhost':

```

The screenshot shows a MySQL IDE interface. The top toolbar includes icons for file operations, query execution, and schema management. The main window is titled 'Query 1' and contains the following SQL commands:

```

1 • USE empresa;
2 • SELECT * FROM empregados;
3

```

Below the query window, the 'Result Grid' displays the results of the query. The grid has columns for '#', 'id', 'nome', 'posicion', and 'salario'. The data is as follows:

| # | id | nome          | posicion       | salario |
|---|----|---------------|----------------|---------|
| 1 | 1  | Brais Pereira | Enxeñeira      | 3000.00 |
| 2 | 2  | Xoán Loureiro | Técnico        | 2000.00 |
| 3 | 3  | Uxía Cornide  | Administrativa | 1500.00 |
| * |    |               |                |         |

The left sidebar shows the 'SCHEMAS' tree with 'empresa' selected, and its sub-items: 'Tables', 'Columns', 'Indexes', 'Foreign Key', 'Triggers', 'Views', 'Stored Procedure', and 'Functions'.

```

1 • SET ROLE 'write_access';
2 • INSERT INTO empresa.empregados VALUES (1, 'John Doe');
3 • UPDATE empresa.empregados SET name = 'Jane Doe' WHERE id = 1;
4

```

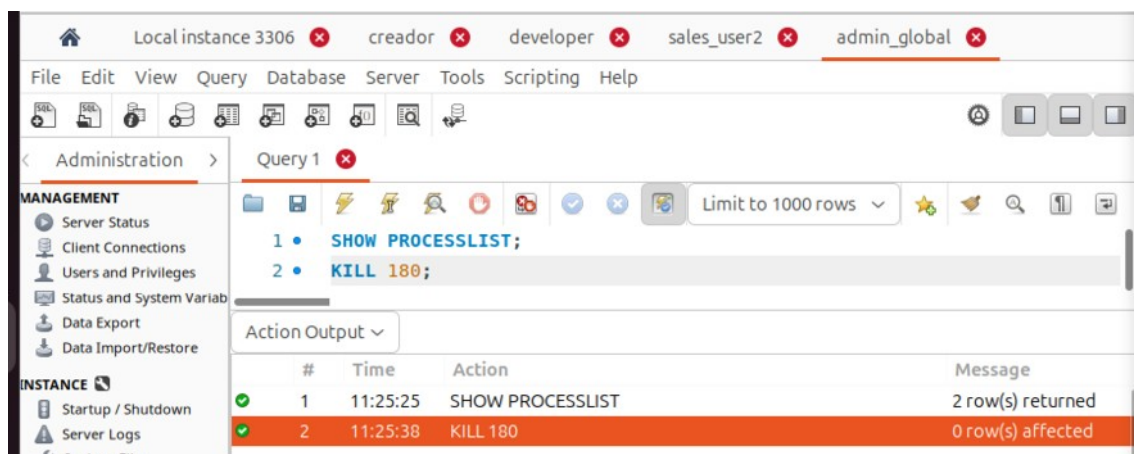
#### 4. Roles con permisos globais

- Crea un rol `admin_global` con permisos para xestionar todos os procesos no servidor (supervisión e terminación).
- Asigna este rol ao usuario `global_admin@%`.
- Conéctate como `global_admin` e executa accións para supervisar os procesos do servidor, como listar procesos activos e rematar un proceso.

```

207 • CREATE USER 'admin_global'@'localhost' IDENTIFIED BY 'admin_global';
208 • CREATE ROLE 'admin_global';
209 • GRANT PROCESS, SUPER ON *.* TO 'admin_global';
210 • GRANT 'admin_global' TO 'admin_global'@'localhost';

```



#### 5. Revogación de roles e permisos

- A un usuario `temp_user@%` con roles previamente asignados (`read_access` e `write_access`), realiza as seguintes accións:
  - Elimina o rol `write_access` e comproba que xa non pode realizar insercións.
  - Revoga manualmente os permisos que se outorgaron mediante o rol `read_access`.
  - Comproba os permisos restantes executando `SHOW GRANTS`.

```

228 • CREATE USER 'temp_user'@'%' IDENTIFIED BY 'temp_user';
229 • REVOKE 'write_access' FROM 'temp_user'@'%';
230 • REVOKE SELECT ON empresa.empregados FROM 'temp_user'@'%';
231 • SHOW GRANTS FOR 'temp_user'@'%';
232

```

|             |                                   |         |                                             |
|-------------|-----------------------------------|---------|---------------------------------------------|
| Result Grid | Filter Rows: <input type="text"/> | Export: | Wrap Cell Content: <input type="checkbox"/> |
| #           | Grants for temp_user@%            |         |                                             |
| 1           | GRANT USAGE ON *.* TO `temp_...   |         |                                             |