

**BASE DE DATOS AVANZADA**

**PARTICIPANTES**

**LILIANA ANDREA RINCON C.C 1.033.764.222**

**DIEGO FERNANDO RAMÍREZ CASTELLANO C.C 7186178**

**RIGOBERTO COY C.C 6759899**

**GRUPO**

**301125\_26**

**TUTOR**

**MARCO ANTONIO LOPEZ OSPINA**

**UNIVERSIDAD NACIONAL ABIERTA Y A DISTANCIA**

**OCTUBRE 2016**

## INTRODUCCION

En el presente trabajo hablaremos sobre la seguridad en el gestor de bases de datos Mysql, el cual cuenta con muchas operaciones tanto en modo consola como en modo gráfico, que nos permite implementar seguridad con el fin de mantener nuestros datos a salvo.

Por otro lado mencionaremos procesos técnicos que se deben seguir para una buena implementación de la seguridad, ya que si no tenemos claros los conceptos podríamos cometer errores donde estaría en riesgo información confidencial.

## RESPUESTAS A LAS PREGUNTAS ORIENTADORAS

✚ Porque es importante la seguridad en una base de datos

Una buena configuración de seguridad en las bases de datos nos asegura privacidad y confidencialidad de la información, hay información que es sensible y que solo debe ser consultada por cierto grupo de personas con privilegios, si no se establecen estos niveles de protección esta información puede ser obtenida por personas con fines dañinos, además de la seguridad en el acceso a la información, también es necesario asegurar la integridad de los datos, que no se alterados de forma irregular por los usuarios o por el mismo sistema.

También es importante la disponibilidad de los datos, una buena práctica es realizar copias redundantes en otros discos para en caso de una pérdida de información, los daños sean mínimos.

✚ A que niveles se puede implementar seguridad en una base de datos.

Física: controlar el acceso al equipo o servidores donde se tenga almacenada nuestra información. Actualmente existen edificios enteros en los cuales se encuentran servidores exclusivos para grandes empresas donde el acceso a estas infraestructuras es muy compleja y se debe tener la autorización correspondiente.

Lógica: control de acceso basado en la autenticación y la identificación de los usuarios que quieran acceder a la base de datos, eso se logra a través de cuentas de usuario las cuales son esenciales para la asignación de roles en la base de datos que pueden ser desde el administrador, dueño de la BD, usuario administrativo, usuario de aplicación (solo lectura), auditor y operador. Cada uno de estos tienen ciertos privilegios dependiendo de su rol, no todos pueden acceder y modificar o eliminar información, no todos tienen acceso a toda la información sino a información parcial, etc.

Correctivo: la información siempre debe ser reconstruirse ya que existen amenazas de todo tipo, no solamente personas que quieran acceder a la información con fines ilícitos, sino también desastres como incendios, terremotos e inundaciones.

✚ Enuncie 3 estrategias de seguridad que recomendaría en una base de datos.

- copias de Seguridad: Se debe realizar periódicamente copias de seguridad de todas las bases de datos activas para impedir que se pierdan datos y para proteger su inversión en el diseño de la base de datos. Si dispone de una copia de seguridad, podrá restaurar fácilmente una base de datos completa u objetos de la base de datos seleccionados.

Si el número de registros de la base de datos aumenta con frecuencia, puede considerar la posibilidad de archivar los datos antiguos. Archivar es un proceso que consiste en mover periódicamente registros antiguos de una tabla de una base de datos activa a una base de datos de archivo. En este artículo no se explica cómo archivar datos antiguos.

- **Identificación y autenticación:** El sistema debe poder identificar y autenticar a los usuarios, utilizando mecanismos como: código y contraseña, identificación por hardware y por conocimientos, aptitudes o hábitos del usuario. Además se deben especificar los privilegios que pose sobre los diversos objetos del sistema como crear nuevas tablas, consultar ciertos datos o tablas, actualizar datos, ejecutar o crear procedimientos almacenados, eliminar información, columnas de tablas o hasta la misma tabla.
- **Monitoreo en tiempo real:** La implementación de agentes inteligentes de monitoreo y detección de intrusiones, puede ser un mecanismo clave y esencial para detectar anomalías e intrusiones no debidas en la base de datos. Por ejemplo, alertas sobre patrones inusuales de acceso, que podrían indicar la presencia de un ataque de inyección SQL, cambios no autorizados a los datos, cambios en privilegios de las cuentas, y los cambios de configuración que se ejecutan a mediante de comandos de SQL.

El monitoreo es un elemento esencial en la evaluación de vulnerabilidad, le permite ir más allá de evaluaciones estáticas o forenses. Por ejemplo se puede evidenciar cuando múltiples usuarios comparten credenciales con privilegios o un número excesivo de inicios de

🚦 Qué tipo de sentencias se utilizan para implementar seguridad en una base de datos.

## Usuarios

Un Usuario es un elemento de la base de datos que contiene la información necesaria para que un usuario de la base de datos (administrador, programador, usuario final, aplicación,...), se autentique contra el SGBD para que éste le asigne los permisos y/o controle las restricciones correspondientes al usuario.

Un usuario se compone básicamente de Nombre y Contraseña. Para crear un usuario en una base de datos se utiliza el siguiente código SQL:

```
CREATE USER <nombre_usuario> IDENTIFIED BY <contraseña>
```

Para modificar la contraseña de acceso de un usuario se utiliza el siguiente comando:

- `ALTER USER <nombre_usuario> IDENTIFIED BY <nueva_contraseña>`

Y finalmente para borrar un usuario se puede utilizar el siguiente comando:

```
DROP USER <nombre_usuario>
```

(En la versión 5.0.2 de **MySQL** existe una sentencia para crear usuarios, CREATE USER, en versiones anteriores se usa exclusivamente la sentencia GRANT para crearlos.

En general es preferible usar GRANT, ya que si se crea un usuario mediante CREATE USER, posteriormente hay que usar una sentencia GRANT para concederle privilegios.

Usando GRANT podemos crear un usuario y al mismo tiempo concederle también los privilegios que tendrá.)

## Roles

Un Rol (papel), es una abstracción que, entre otras cosas, facilita la gestión de privilegios y restricciones sobre los objetos de una base de datos. Al igual que a los usuarios, a los roles se les definen permisos y restricciones. A un usuario se le pueden asignar uno o más roles, y un rol puede ser asignado a uno o muchos usuarios. Cuando un usuario tiene asignado un rol tiene los mismos privilegios y restricciones del rol.

Para definir un nuevo rol se utiliza el siguiente comando:

```
CREATE ROLE <nombre_rol>
```

Dado que el rol solamente tiene un nombre, no hay acciones de modificación sobre estos, existe solamente la acción de borrado que tiene la siguiente sintaxis:

```
DROP ROLE <nombre_rol>.
```

Para asignar un rol a un usuario se puede utilizar el siguiente constructo SQL:

```
GRANT <nombre_rol> TO <nombre_usuario>
```

Para remover un rol a un usuario se puede utilizar el siguiente comando:

```
REVOKE <nombre_rol> FROM <nombre_usuario>
```

## Privilegios

Los privilegios que se pueden conceder a roles o a usuarios y los constructos SQL necesarios para asignarlos. Es de resaltar que estos privilegios pueden establecerse a tablas, vistas o a partes de ambas.

- **Selección (SELECT):** proporciona los permisos de consulta sobre los objetos relacionados.
- **Insertión (INSERT):** permite insertar nuevos registros en los objetos relacionados al privilegio.
- **Actualización (UPDATE):** permite modificar la información contenida en los objetos relacionados.
- **Borrado (DELETE):** permite eliminar registros o tuplas contenidas en los objetos relacionados.
- **Referencia (REFERENCES):** permite crear “constraints” que referencia a la tabla relacionada al privilegio.
- **Alteración (ALTER):** permite realizar modificaciones a la tabla relacionada

- **Indexación (INDEX):** permite crear índices en la tabla relacionada

### Asignar y Revocar Privilegios

En SQL, la sintaxis para la asignación de privilegios a roles o a usuarios para un objeto de la base de datos es la siguiente:

```
GRANT <privilegio> ON <objeto_bd> TO {<nombre_usuario> | <nombre_rol> }
```

De otro lado, para remover los privilegios previamente asignados a usuarios o roles se puede utilizar la siguiente sintaxis:

```
REVOKE <privilegio> ON <objeto_bd> FROM {<nombre_usuario> | <nombre_rol> }
```

Existe un privilegio especial que le permite conceder dichos privilegios a otros roles o usuarios, para ello es necesario adicionar la sentencia “WITH GRANT OPTION” al final del SQL utilizado para otorgar privilegios. Para remover este privilegio especial es necesario agregar la cláusula “CASCADE” al final del comando para revocar privilegios. Existe además un constructo que permite establecer todos los privilegios posibles mediante una sola sentencia, este es “ALL PRIVILEGES”.

Incluyendo las tres últimas cláusulas, la sintaxis para asignación de privilegios queda de la siguiente forma:

```
GRANT {ALL PRIVILEGES | <privilegio> } ON <objeto_bd> TO  
{<nombre_usuario> | <nombre_rol> } [WITH GRANT OPTION]
```

Y la sintaxis para remover privilegios así:

```
REVOKE {ALL PRIVILEGES | <privilegio> } ON <objeto_bd> FROM  
{<nombre_usuario> | <nombre_rol> } [CASCADE]
```

Para que se puedan ejecutar una acción de conceder uno o más privilegios sobre un objeto de la base de datos, quien está ejecutando el procedimiento debe tener, además de los privilegios a otorgar, los permisos para otorgarlos a otros usuarios o roles.

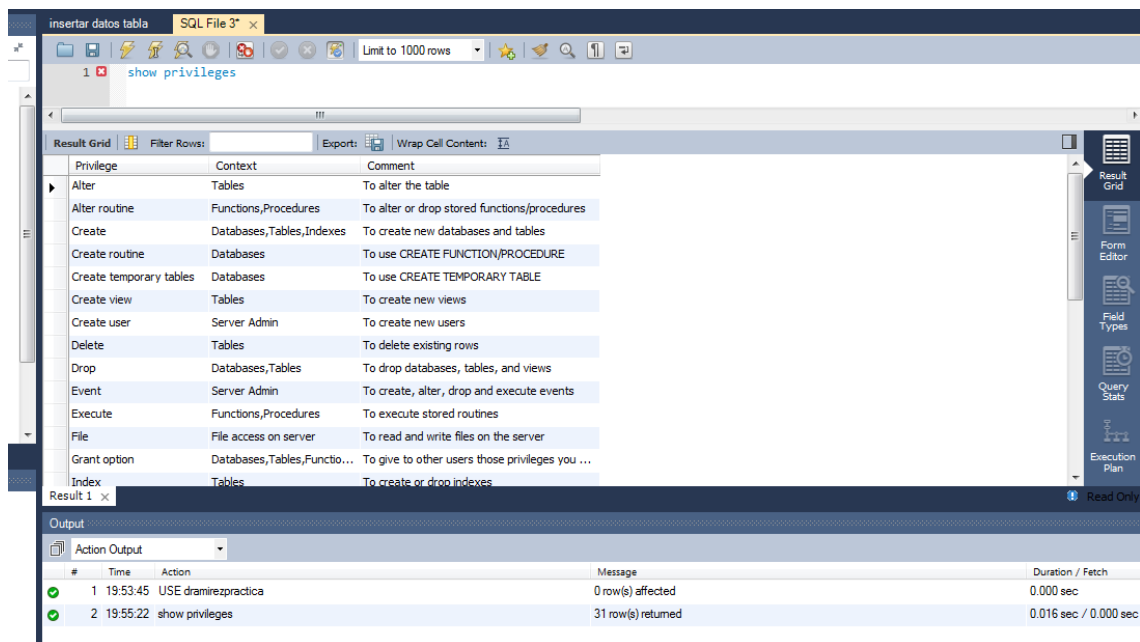
### Que es un Trigger y para que se utiliza

El trigger o disparador no es más que una pequeña instrucción o rutina que se dispara o ejecuta ante alguna operación (insert, update, o delete) en algún momento (before, after) sobre una tabla. Cada vez que se inserte, modifique, o elimine algún elemento en una tabla, se ejecutarán las rutinas empleadas. Esto nos lleva a que hay 6 tipos de triggers: after insert, before insert, after update, before update, after delete, before delete. Dependiendo de las necesidades, y la rutina se debe optar por uno u otro contexto.

Su función es permitir la implementación de reglas corporativas y permanentes, y su uso más típico ha sido el de proteger la integridad referencial de la base de datos.

## INSTRUCTIVO SOBRE DE GESTIÓN DE SEGURIDAD EN BASES DE DATOS.

- Ejecutamos el comando show privileges y nos mostrara una tabla con todos los privilegios que se le pueden asignar a un usuario

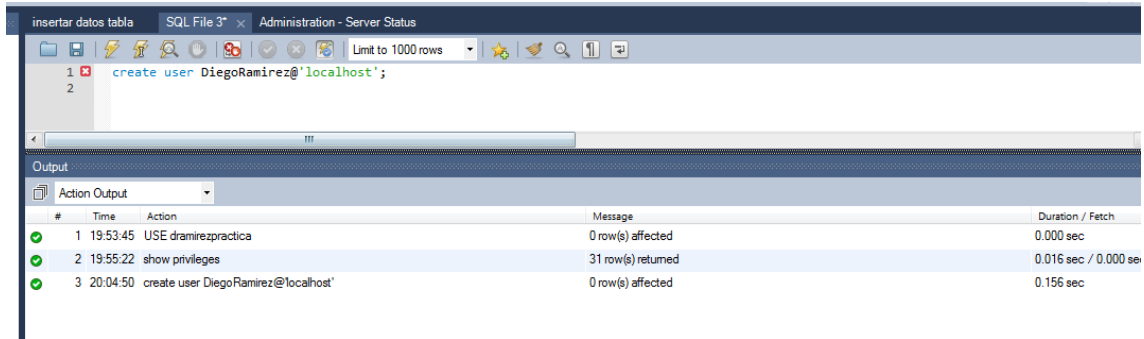


Privilege	Context	Comment
Alter	Tables	To alter the table
Alter routine	Functions,Procedures	To alter or drop stored functions/procedures
Create	Databases,Tables,Indexes	To create new databases and tables
Create routine	Databases	To use CREATE FUNCTION/PROCEDURE
Create temporary tables	Databases	To use CREATE TEMPORARY TABLE
Create view	Tables	To create new views
Create user	Server Admin	To create new users
Delete	Tables	To delete existing rows
Drop	Databases,Tables	To drop databases, tables, and views
Event	Server Admin	To create, alter, drop and execute events
Execute	Functions,Procedures	To execute stored routines
File	File access on server	To read and write files on the server
Grant option	Databases,Tables,Function...	To give to other users those privileges you ...
Index	Tables	To create or drop indexes

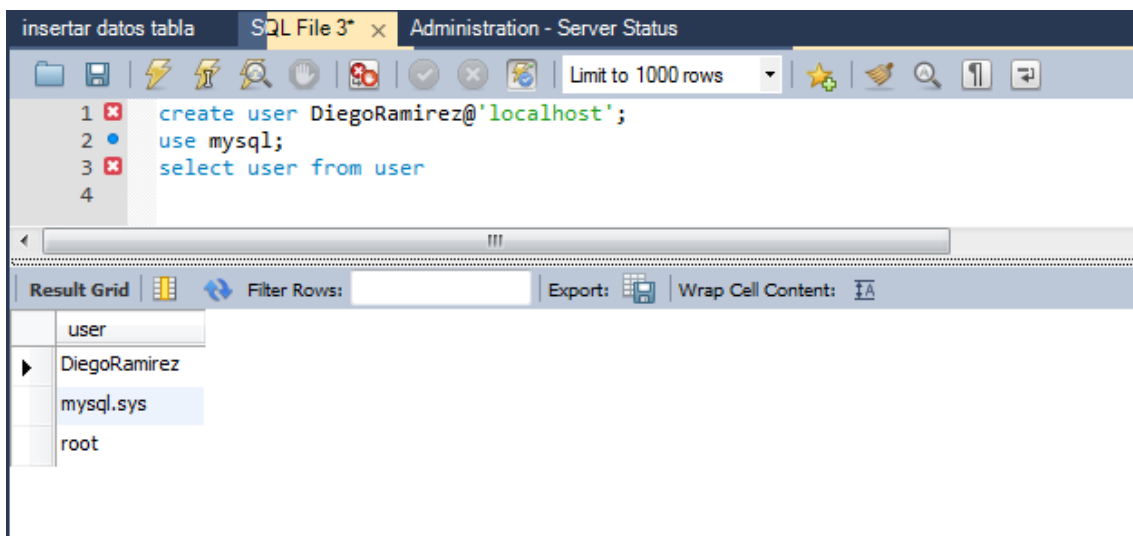
  

#	Time	Action	Message	Duration / Fetch
1	19:53:45	USE dramirezpractica	0 row(s) affected	0.000 sec
2	19:55:22	show privileges	31 row(s) returned	0.016 sec / 0.000 sec

- Crear usuarios para cada uno de los compañeros del grupo colaborativo y asignarles diferentes niveles de acceso a la base de datos. Ingresar con las credenciales de los diferentes usuarios y verificar la seguridad.



Se crea el usuario DiegoRamirez con el comando `create user DiegoRamirez@'localhost'`



Comprobamos que el usuario se haya creado bien.

Creamos las cuentas de los compañeros.



The screenshot shows the MySQL Workbench interface with the 'Administration - Users and Privileges' window open. The SQL editor contains the following queries:

```

1 create user Anap@'localhost';
2 use mysql;
3 select user from user;
4 select user, host from user;
5 select user, host , db from mysql.db;
6
7

```

The 'Result Grid' shows the output of the queries:

user
Anap
DiegoRamirez
RigoCoy
mysql.sys
root

The 'Output' window shows the 'Action Output' for the executed queries:

#	Time	Action	Message	Duration / Fetch
1	20:54:03	create user RigoCoy@'localhost'	0 row(s) affected	0.047 sec
2	20:54:19	create user Anap@'localhost'	0 row(s) affected	0.000 sec
3	20:54:22	use mysql	0 row(s) affected	0.000 sec
4	20:54:26	select user from user LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec

Escogemos la base de datos a la que tendrá acceso este usuario.

The screenshot shows the MySQL Workbench 'Users and Privileges' window for 'Local instance MySQL57'. The 'User Accounts' table lists the following users:

User	From Host
DiegoRamirez	localhost
mysql.sys	localhost
root	localhost

The 'Details for account DiegoRamirez@localhost' window is open, showing the 'Schema Privileges' tab. The 'Schema' is 'dramirezpractica' and the 'Privileges' are:

Schema	Privileges
dramirezpractica	ALTER, ALTER ROUTINE, CREATE, CREATE ROUTINE, CREATE TEMPORARY TABLES, CREATE VIEW, DELETE, DROP, E

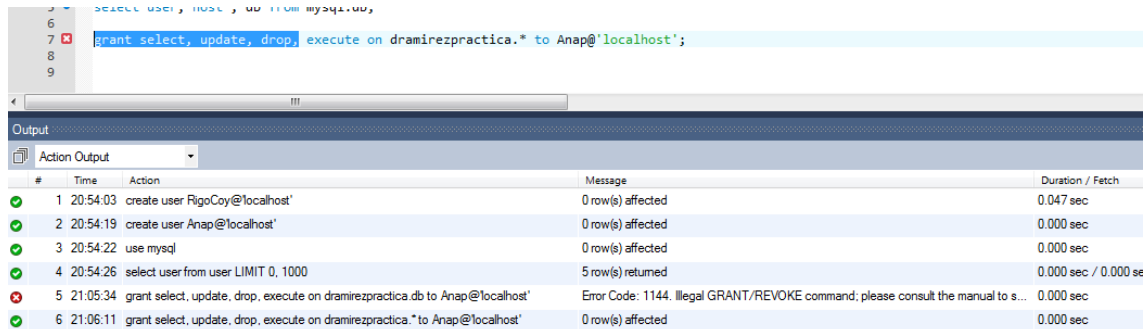
The 'Object Rights' tab is also visible, showing the following rights for the user 'DiegoRamirez@localhost' on the schema 'dramirezpractica':

Object Rights	DDL Rights	Other Rights
<input checked="" type="checkbox"/> SELECT	<input checked="" type="checkbox"/> CREATE	<input checked="" type="checkbox"/> GRANT OPTION
<input checked="" type="checkbox"/> INSERT	<input checked="" type="checkbox"/> ALTER	<input checked="" type="checkbox"/> CREATE TEMPORARY TABLES
<input checked="" type="checkbox"/> UPDATE	<input checked="" type="checkbox"/> REFERENCES	<input checked="" type="checkbox"/> LOCK TABLES
<input checked="" type="checkbox"/> DELETE	<input checked="" type="checkbox"/> INDEX	
<input checked="" type="checkbox"/> EXECUTE	<input checked="" type="checkbox"/> CREATE VIEW	
<input checked="" type="checkbox"/> SHOW VIEW	<input checked="" type="checkbox"/> CREATE ROUTINE	
	<input checked="" type="checkbox"/> ALTER ROUTINE	
	<input checked="" type="checkbox"/> EVENT	
	<input checked="" type="checkbox"/> DROP	
	<input checked="" type="checkbox"/> TRIGGER	

Activamos los privilegios que le queremos asignar a nuestro usuario en este caso el mío DiegoRamirez lo activo y lo dejo con acceso administrativo.

Los privilegios de las cuentas de mis compañeros los activo por código.

Al usuario anap le asignamos los privilegios grant select, update, drop



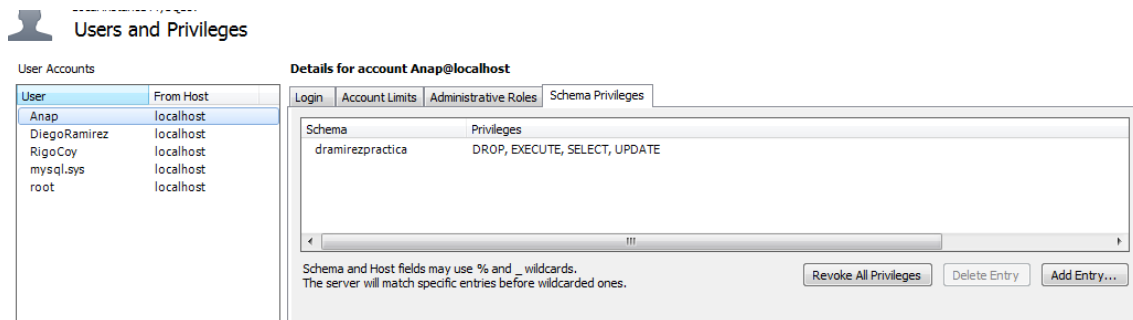
```

5 select user, host, db from mysql.db;
6
7 grant select, update, drop, execute on dramirezpractica.* to Anap@'localhost';
8
9

```

#	Time	Action	Message	Duration / Fetch
✓ 1	20:54:03	create user RigoCoy@'localhost'	0 row(s) affected	0.047 sec
✓ 2	20:54:19	create user Anap@'localhost'	0 row(s) affected	0.000 sec
✓ 3	20:54:22	use mysql	0 row(s) affected	0.000 sec
✓ 4	20:54:26	select user from user LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 se
✗ 5	21:05:34	grant select, update, drop, execute on dramirezpractica.db to Anap@'localhost'	Error Code: 1144. Illegal GRANT/REVOKE command; please consult the manual to s...	0.000 sec
✓ 6	21:06:11	grant select, update, drop, execute on dramirezpractica.* to Anap@'localhost'	0 row(s) affected	0.000 sec

Comprobamos los privilegios asignados



**Users and Privileges**

User	From Host
Anap	localhost
DiegoRamirez	localhost
RigoCoy	localhost
mysql.sys	localhost
root	localhost

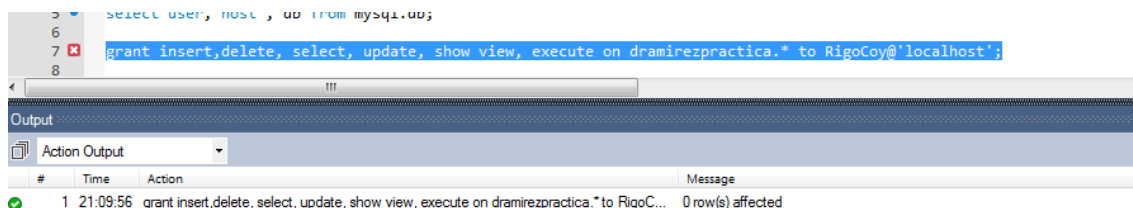
**Details for account Anap@localhost**

Schema	Privileges
dramirezpractica	DROP, EXECUTE, SELECT, UPDATE

Schema and Host fields may use % and \_ wildcards.  
The server will match specific entries before wildcarded ones.

Buttons: Revoke All Privileges, Delete Entry, Add Entry...

Al usuario RigoCoy le asignamos insert,delete, select, update, show view,



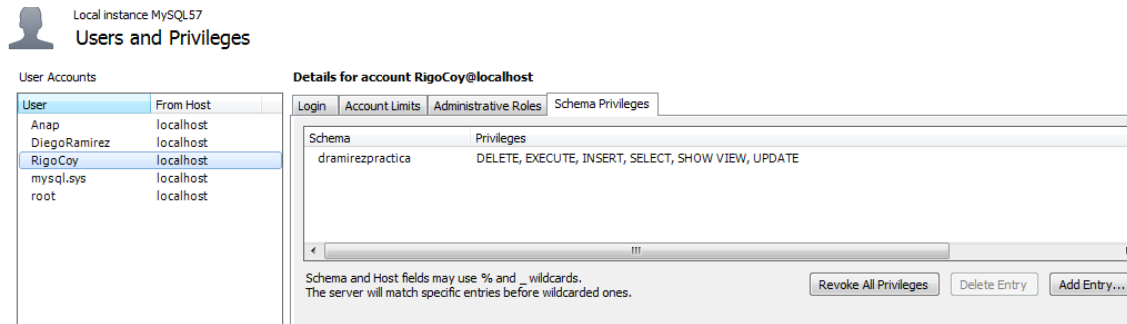
```

5 select user, host, db from mysql.db;
6
7 grant insert,delete, select, update, show view, execute on dramirezpractica.* to RigoCoy@'localhost';
8

```

#	Time	Action	Message
✓ 1	21:09:56	grant insert,delete, select, update, show view, execute on dramirezpractica.* to RigoCoy@'localhost';	0 row(s) affected

Comprobamos los privilegios asignados



Local instance MySQL57  
Users and Privileges

User Accounts

User	From Host
Anap	localhost
DiegoRamirez	localhost
RigoCoy	localhost
mysql.sys	localhost
root	localhost

Details for account RigoCoy@localhost

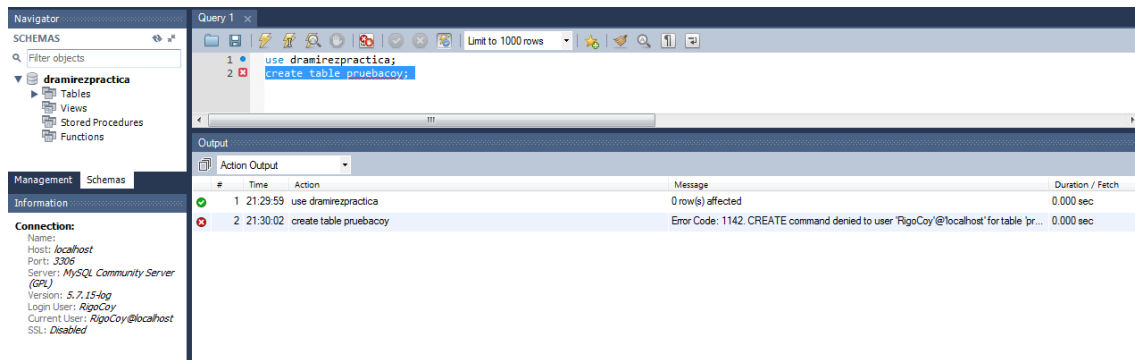
Login Account Limits Administrative Roles Schema Privileges

Schema	Privileges
dramirezpractica	DELETE, EXECUTE, INSERT, SELECT, SHOW VIEW, UPDATE

Schema and Host fields may use % and \_ wildcards.  
The server will match specific entries before wildcarded ones.

Revoke All Privileges Delete Entry Add Entry...

Ahora accedemos con la cuenta de usuario RigoCoy e intentamos crear una tabla, para lo cual no tiene privilegios, comprobamos en la columna izquierda que solo tenemos acceso a la base de datos dramirezpractica y abajo la conexión esta con el usuario ya mencionado.



Navigator

SCHEMAS

Filter objects

dramirezpractica

Tables

Views

Stored Procedures

Functions

Management Schemas

Information

Connection:

Name: localhost

Host: localhost

Port: 3306

Server: MySQL Community Server (GPL)

Version: 5.7.15-log

Login User: RigoCoy

Current User: RigoCoy@localhost

SSL: Disabled

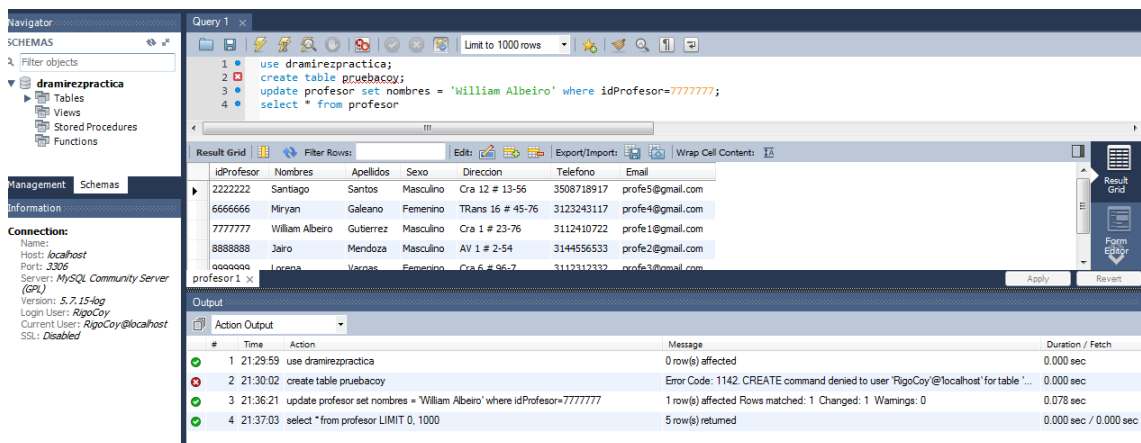
Query 1

```
1 use dramirezpractica;
2 create table pruebaCoy;
```

Output

#	Time	Action	Message	Duration / Fetch
1	21:29:59	use dramirezpractica	0 row(s) affected	0.000 sec
2	21:30:02	create table pruebaCoy	Error Code: 1142. CREATE command denied to user 'RigoCoy'@'localhost' for table 'pr...	0.000 sec

Ahora ejecutamos un comando para el cual si tiene permiso, vamos a actualizar la tabla profesores, cambiamos el nombre de “William Alberto” a “William Albeiro”



Navigator

SCHEMAS

Filter objects

dramirezpractica

Tables

Views

Stored Procedures

Functions

Management Schemas

Information

Connection:

Name: localhost

Host: localhost

Port: 3306

Server: MySQL Community Server (GPL)

Version: 5.7.15-log

Login User: RigoCoy

Current User: RigoCoy@localhost

SSL: Disabled

Query 1

```
1 use dramirezpractica;
2 create table pruebaCoy;
3 update profesor set nombres = 'William Albeiro' where idProfesor=7777777;
4 select * from profesor
```

Result Grid

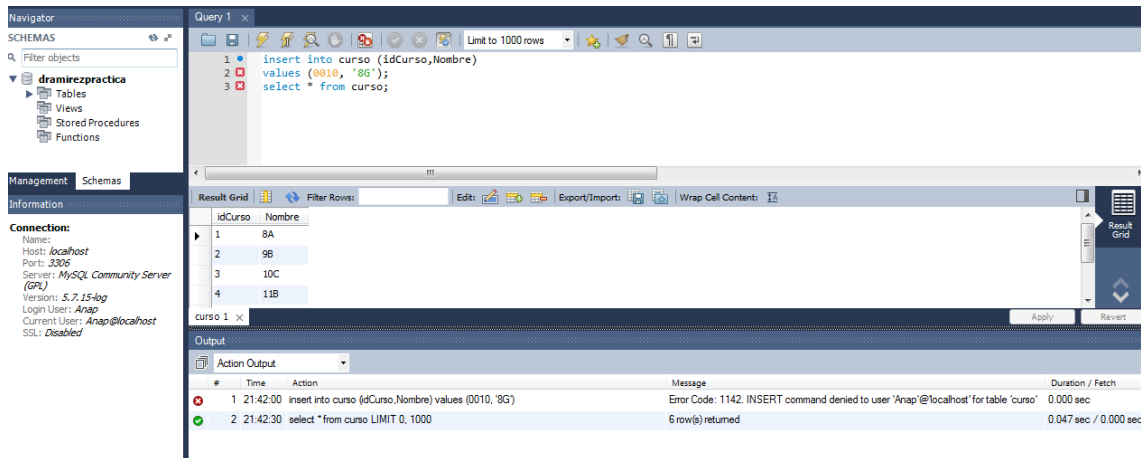
idProfesor	Nombres	Apellidos	Sexo	Direccion	Telefono	Email
2222222	Santiago	Santos	Masculino	Cra 12 # 13-56	3508718917	prof5@gmail.com
6666666	Miryan	Galeano	Femenino	Tlano 16 # 45-76	3123243117	prof4@gmail.com
7777777	William Albeiro	Gutierrez	Masculino	Cra 1 # 23-76	3112410722	prof1@gmail.com
8888888	Jairo	Mendoza	Masculino	AV 1 # 2-54	3144556533	prof2@gmail.com
9999999	Lorena	Vargas	Femenino	Cra 6 # 96-7	3112312332	prof3@gmail.com

profesor1

Output

#	Time	Action	Message	Duration / Fetch
1	21:29:59	use dramirezpractica	0 row(s) affected	0.000 sec
2	21:30:02	create table pruebaCoy	Error Code: 1142. CREATE command denied to user 'RigoCoy'@'localhost' for table 'pr...	0.000 sec
3	21:36:21	update profesor set nombres = 'William Albeiro' where idProfesor=7777777	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.078 sec
4	21:37:03	select * from profesor LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec

Ahora hacemos lo mismo pero con el usuario Anap, ingresamos con este usuario e intentamos insertar un dato a la tabla curso, lo cual no nos deja hacer por no tener este privilegio, hacemos la consulta de esta misma tabla curso, lo cual si nos deja hacer.



- Hacer copia de seguridad de la base de datos.

Se ejecuta el siguiente comando para realizar una copia de seguridad completa de la base de datos llamada dramirezpractica, poner el usuario, este caso es DiegoRamirez y la contraseña del usuario.

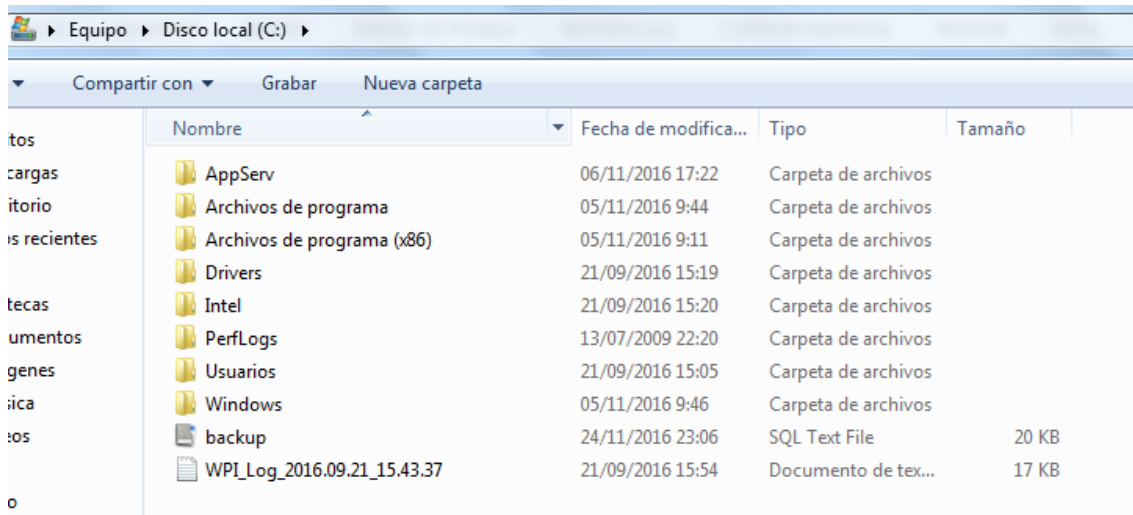
Si por motivos de seguridad no queremos escribir la contraseña como parte del comando, reemplazamos la opción --password=XX por -p. Al hacerlo, MySQL pedirá que escribir la contraseña a mano cada vez que realices una copia de seguridad.

```
C:\Program Files\MySQL\MySQL Server 5.7\bin>mysqldump
Usage: mysqldump [OPTIONS] database [tables]
OR mysqldump [OPTIONS] --databases [OPTIONS] DB1 [DB2 DB3...]
OR mysqldump [OPTIONS] --all-databases [OPTIONS]
For more options, use mysqldump --help

C:\Program Files\MySQL\MySQL Server 5.7\bin>mysqldump -uDiegoRamirez -pdramirezpractica > "c:\backup.sql"
mysqldump: [Warning] Using a password on the command line interface can be insecure.

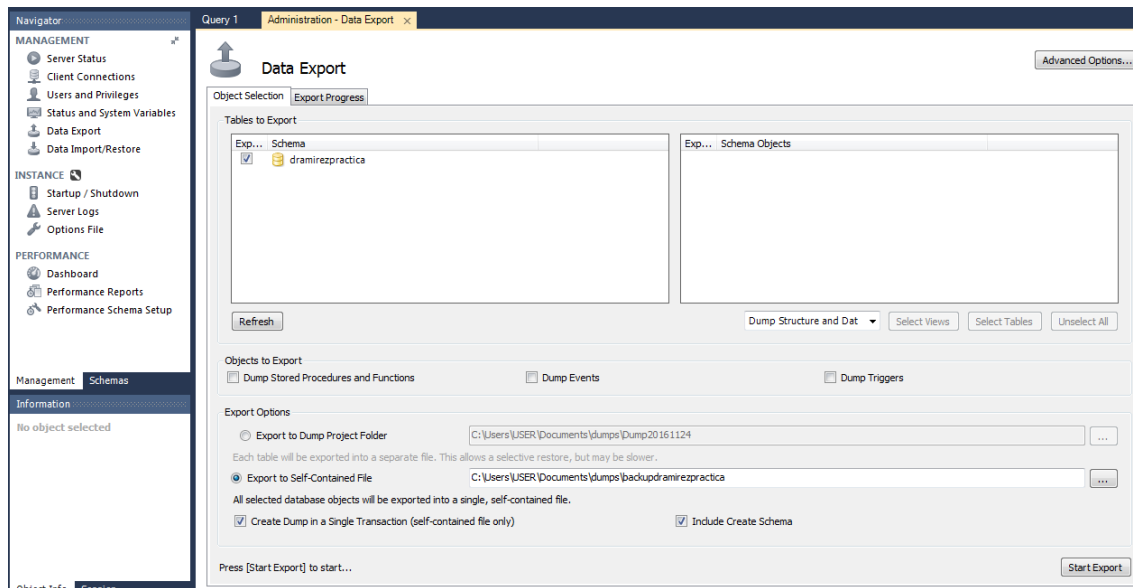
C:\Program Files\MySQL\MySQL Server 5.7\bin>mysqldump -uDiegoRamirez -p dramirezpractica > "c:\backup.sql"
Enter password: *****

C:\Program Files\MySQL\MySQL Server 5.7\bin>
```



Nombre	Fecha de modifica...	Tipo	Tamaño
AppServ	06/11/2016 17:22	Carpeta de archivos	
Archivos de programa	05/11/2016 9:44	Carpeta de archivos	
Archivos de programa (x86)	05/11/2016 9:11	Carpeta de archivos	
Drivers	21/09/2016 15:19	Carpeta de archivos	
Intel	21/09/2016 15:20	Carpeta de archivos	
PerfLogs	13/07/2009 22:20	Carpeta de archivos	
Usuarios	21/09/2016 15:05	Carpeta de archivos	
Windows	05/11/2016 9:46	Carpeta de archivos	
backup	24/11/2016 23:06	SQL Text File	20 KB
WPI_Log_2016.09.21_15.43.37	21/09/2016 15:54	Documento de tex...	17 KB

La otra forma de hacerlo es



**Data Export**

Object Selection | Export Progress

Tables to Export

Exp...	Schema
<input checked="" type="checkbox"/>	dramirezpractica

Refresh

Dump Structure and Dat | Select Views | Select Tables | Unselect All

Objects to Export

☐ Dump Stored Procedures and Functions ☐ Dump Events ☐ Dump Triggers

Export Options

☐ Export to Dump Project Folder C:\Users\USER\Documents\dumps\Dump20161124

Each table will be exported into a separate file. This allows a selective restore, but may be slower.

☒ Export to Self-Contained File C:\Users\USER\Documents\dumps\backup\dramirezpractica

All selected database objects will be exported into a single, self-contained file.

☒ Create Dump in a Single Transaction (self-contained file only) ☒ Include Create Schema

Press [Start Export] to start...

Start Export

Query 1 Administration - Data Export x

**Data Export** Advanced Options...

Object Selection Export Progress

Export Completed

Status:  
16 of 16 exported.

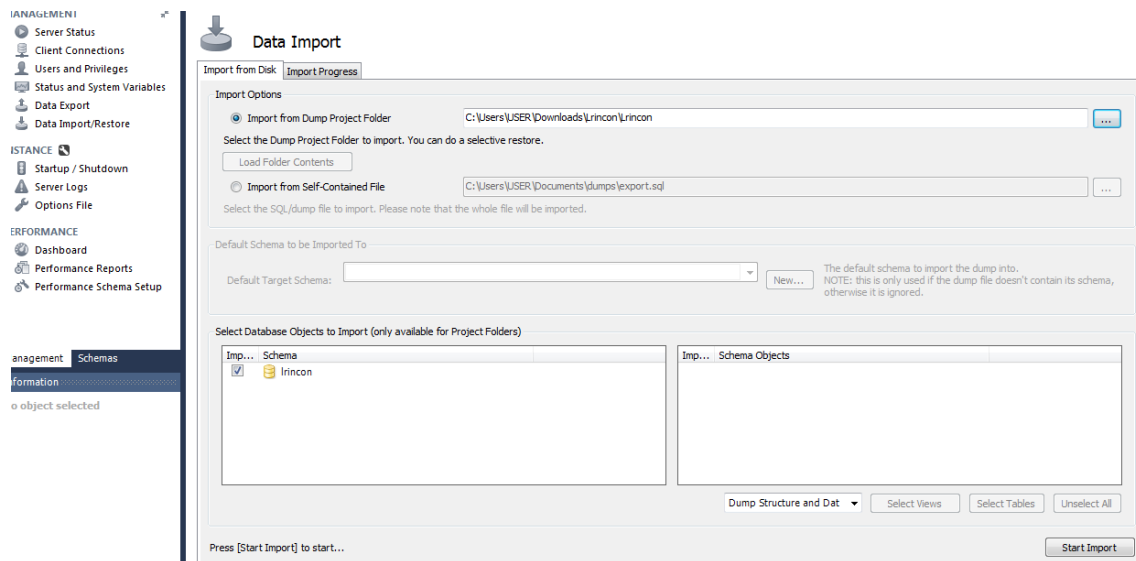
Log:

```
mysqldump.exe is version 5.7.12, but the MySQL Server to be dumped has version 5.7.15.
Because the version of mysqldump is older than the server, some features may not be backed up properly.
It is recommended you upgrade your local MySQL client programs, including mysqldump, to a version equal to or newer than that of the target server.
The path to the dump tool must then be set in Preferences -> Administrator -> Path to mysqldump Tool:
23:11:25 Dumping dramirezpractica (all tables)
Running: mysqldump.exe --defaults-file="c:\users\user\appdata\local\temp\tmpx6m7dk.cnf" --user=DiegoRamirez --host=localhost --protocol=tcp --port=3306 --default-character-set=utf8 --single-transaction=TRUE --skip-triggers "dramirezpractica"
23:11:26 Export of C:\Users\USER\Documents\dumps\backupdramirezpractica has finished
```

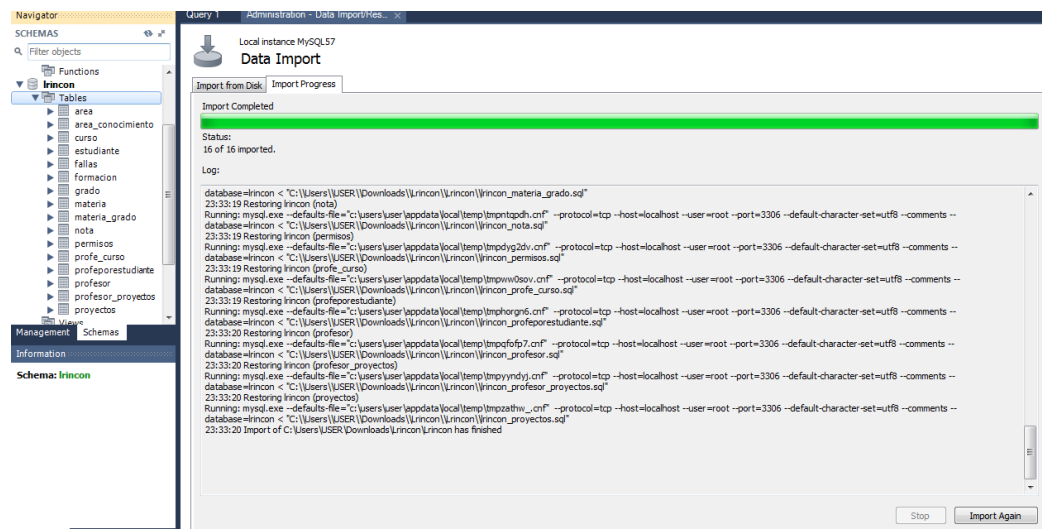
Nombre	Fecha de modifica...	Tipo	Tamaño
dramirezpractica_area	24/11/2016 23:14	SQL Text File	2 KB
dramirezpractica_area_pro	24/11/2016 23:14	SQL Text File	3 KB
dramirezpractica_calificaciones	24/11/2016 23:14	SQL Text File	4 KB
dramirezpractica_curso	24/11/2016 23:14	SQL Text File	2 KB
dramirezpractica_estudiante	24/11/2016 23:14	SQL Text File	4 KB
dramirezpractica_fallas	24/11/2016 23:14	SQL Text File	3 KB
dramirezpractica_formacion	24/11/2016 23:14	SQL Text File	3 KB
dramirezpractica_grado	24/11/2016 23:14	SQL Text File	2 KB
dramirezpractica_gradomateria	24/11/2016 23:14	SQL Text File	3 KB
dramirezpractica_materia	24/11/2016 23:14	SQL Text File	3 KB
dramirezpractica_permisos	24/11/2016 23:14	SQL Text File	3 KB
dramirezpractica_profe_est	24/11/2016 23:14	SQL Text File	3 KB
dramirezpractica_profe_proyectos	24/11/2016 23:14	SQL Text File	3 KB
dramirezpractica_profesor	24/11/2016 23:14	SQL Text File	3 KB
dramirezpractica_proyectos	24/11/2016 23:14	SQL Text File	3 KB
dramirezpractica_prueba	24/11/2016 23:14	SQL Text File	2 KB

- Descargar las copias de seguridad subidas por los compañeros del grupo colaborativo y restaurarlas en su equipo. Consultar el listado de bases de datos y tomar un pantallazo para presentarlo como prueba de que se desarrolló exitosamente la recuperación de todas las copias.

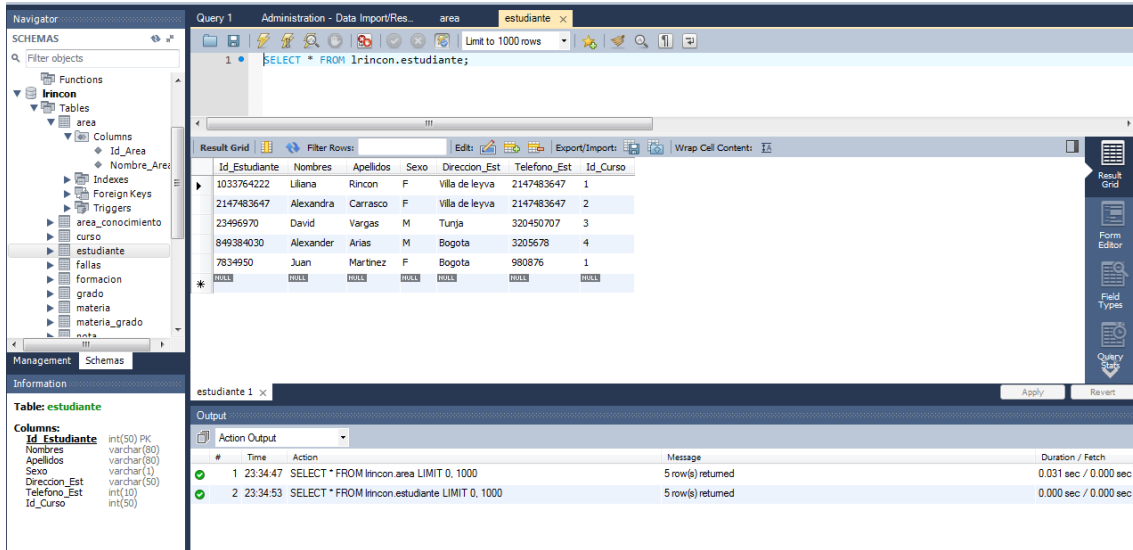
Vamos a la opción Data Import/restore y allí ubicamos la carpeta donde está la base de datos de la compañera ana.



En esta imagen podemos ver la base de datos con todas sus tablas ya importadas



La información de lavase de datos también está completa



Query 1 Administration - Data Import/Res... area estudiante

1 • `SELECT * FROM Irincon.estudiante;`

Result Grid

Id_Estudiante	Nombres	Apellidos	Sexo	Direccion_Est	Telefono_Est	Id_Curso
1033764222	Liliana	Rincon	F	Villa de leyva	2147483647	1
2147483647	Alexandra	Carrasco	F	Villa de leyva	2147483647	2
23496970	David	Vargas	M	Tunja	320450707	3
849384030	Alexander	Arias	M	Bogota	3205678	4
7834950	Juan	Martinez	F	Bogota	980876	1
NULL	NULL	NULL	NULL	NULL	NULL	NULL

estudiante 1

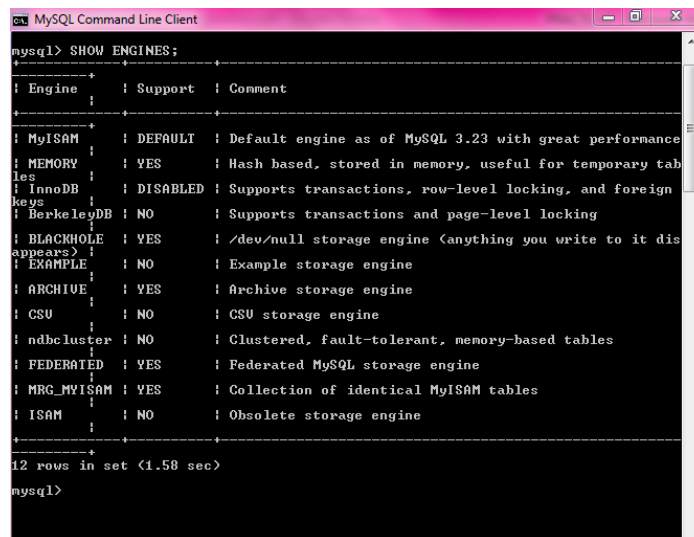
Output

#	Time	Action	Message	Duration / Fetch
1	23:34:47	SELECT * FROM Irincon.area LIMIT 0, 1000	5 row(s) returned	0.031 sec / 0.000 sec
2	23:34:53	SELECT * FROM Irincon.estudiante LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec

Se toma la base de datos trabajada en las practicas anteriores

### 1. Aplicar estrategias de seguridad en la configuración de la base de datos.

Motores de almacenamiento que soporta su servidor



MySQL Command Line Client

```
mysql> SHOW ENGINES;
```

Engine	Support	Comment
MyISAM	DEFAULT	Default engine as of MySQL 3.23 with great performance
MEMORY	YES	Hash based, stored in memory, useful for temporary tables
InnoDB	DISABLED	Supports transactions, row-level locking, and foreign keys
BerkeleyDB	NO	Supports transactions and page-level locking
BLACKHOLE	YES	/dev/null storage engine (anything you write to it disappears)
EXAMPLE	NO	Example storage engine
ARCHIVE	YES	Archive storage engine
CSV	NO	CSV storage engine
ndbcluster	NO	Clustered, fault-tolerant, memory-based tables
FEDERATED	YES	Federated MySQL storage engine
MRG_MYISAM	YES	Collection of identical MyISAM tables
ISAM	NO	Obsolete storage engine

12 rows in set (1.58 sec)

mysql>

Comprobar que el valor de la variable para el motor de almacenamiento en que está Interesado.



```
mysql> show variables LIKE'havex%';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| have_archive  | YES   |
| have_bdb      | NO    |
| have_blackhole_engine | YES   |
| have_compress | YES   |
| have_crypt    | NO    |
| have_csv      | NO    |
| have_dynamic_loading | YES   |
| have_example_engine | NO    |
| have_federated_engine | YES   |
| have_geometry | YES   |
| have_innodb   | DISABLED |
| have_isam     | NO    |
| have_merge_engine | YES   |
| have_ndbcluster | NO    |
| have_openssl  | DISABLED |
| have_ssl      | DISABLED |
| have_query_cache | YES   |
| have_raid     | NO    |
| have_rtree_keys | YES   |
| have_symlink  | YES   |
+-----+-----+
20 rows in set (0.96 sec)

mysql>
```

Lo primero que se debe hacer es colocar contraseña al usuario root, para esto buscamos: xampp > phpMyAdmin > config.inc.php

```
17
18 /* Authentication type and info */
19 $cfg['Servers'][$i]['auth_type'] = 'config';
20 $cfg['Servers'][$i]['user'] = 'root';
21 $cfg['Servers'][$i]['password'] = '██████████';
22 $cfg['Servers'][$i]['extension'] = 'mysqli';
23 $cfg['Servers'][$i]['AllowNoPassword'] = true;
24 $cfg['Lang'] = '';
```

- Luego debemos crear un usuario admin para manejar la base de datos y darle todos los privilegios. Esto se realiza en el phpmyadmin en cuentas de usuario.

Bases de datos

SQL

Estado actual

Cuentas de usuarios

Exportar

Importar

Configuración

Replicación

Más

Vista global de las cuentas de usuario

Grupos de usuario

Vista global de las cuentas de usuario

Existe una cuenta de usuario que permite a cualquier usuario de localhost conectarse. Esto evitará conectarse a otros usuarios, si la parte del host de su cuenta permite una conexión desde cualquier host (%).

	Nombre de usuario	Nombre del servidor	Contraseña	Privilegios globales	Grupo de usuario	Conceder	Acción
<input type="checkbox"/>	cualquiera	%	No	USAGE		No	<div>Editar privilegios</div> <div>Exportar</div>
<input type="checkbox"/>	cualquiera	localhost	No	USAGE		No	<div>Editar privilegios</div> <div>Exportar</div>
<input type="checkbox"/>	pma	localhost	No	USAGE		No	<div>Editar privilegios</div> <div>Exportar</div>
<input type="checkbox"/>	root	127.0.0.1	No	ALL PRIVILEGES		Sí	<div>Editar privilegios</div> <div>Exportar</div>
<input type="checkbox"/>	root	::1	No	ALL PRIVILEGES		Sí	<div>Editar privilegios</div> <div>Exportar</div>
<input type="checkbox"/>	root	localhost	No	ALL PRIVILEGES		Sí	<div>Editar privilegios</div> <div>Exportar</div>

Seleccionar todo

Para los elementos que están marcados:

Exportar

**Privilegios globales** ☒ [Seleccionar todo](#)

*Nota: Los nombres de los privilegios de MySQL están expresados en inglés.*

<input checked="" type="checkbox"/> Datos	<input checked="" type="checkbox"/> Estructura	<input checked="" type="checkbox"/> Administración
<input checked="" type="checkbox"/> SELECT	<input checked="" type="checkbox"/> CREATE	<input checked="" type="checkbox"/> GRANT
<input checked="" type="checkbox"/> INSERT	<input checked="" type="checkbox"/> ALTER	<input checked="" type="checkbox"/> SUPER
<input checked="" type="checkbox"/> UPDATE	<input checked="" type="checkbox"/> INDEX	<input checked="" type="checkbox"/> PROCESS
<input checked="" type="checkbox"/> DELETE	<input checked="" type="checkbox"/> DROP	<input checked="" type="checkbox"/> RELOAD
<input checked="" type="checkbox"/> FILE	<input checked="" type="checkbox"/> CREATE TEMPORARY TABLES	<input checked="" type="checkbox"/> SHUTDOWN
	<input checked="" type="checkbox"/> SHOW VIEW	<input checked="" type="checkbox"/> SHOW DATABASES
	<input checked="" type="checkbox"/> CREATE ROUTINE	<input checked="" type="checkbox"/> LOCK TABLES
	<input checked="" type="checkbox"/> ALTER ROUTINE	<input checked="" type="checkbox"/> REFERENCES
	<input checked="" type="checkbox"/> EXECUTE	<input checked="" type="checkbox"/> REPLICATION CLIENT
	<input checked="" type="checkbox"/> CREATE VIEW	<input checked="" type="checkbox"/> REPLICATION SLAVE
	<input checked="" type="checkbox"/> EVENT	<input checked="" type="checkbox"/> CREATE USER
	<input checked="" type="checkbox"/> TRIGGER	

Para esto se debe determinar que privilegios o acciones pueden realizar los usuarios en la base de datos.

1. Crear usuarios para cada uno de los compañeros del grupo colaborativo y asignarles diferentes niveles de acceso a la base de datos. Ingresar con las credenciales de los diferentes usuarios y verificar la seguridad.

**Bases de datos** **SQL** **Estado actual** **Cuentas de usuarios** **Exportar** **Importar** **Configuración** **Replicación** **Más**

**Vista global de las cuentas de usuario**

**!** Existe una cuenta de usuario que permite a cualquier usuario de localhost conectarse. Esto evitará conectarse a otros usuarios, si la parte del host de su cuenta permite una conexión desde cualquier host (%).

	Nombre de usuario	Nombre del servidor	Contraseña	Privilegios globales	Grupo de usuario	Conceder	Acción
<input type="checkbox"/>	cualquiera	%	No	USAGE		No	<a href="#">Editar privilegios</a> <a href="#">Exportar</a>
<input type="checkbox"/>	cualquiera	localhost	No	USAGE		No	<a href="#">Editar privilegios</a> <a href="#">Exportar</a>
<input checked="" type="checkbox"/>	clisander	%	Sí	CREATE, DROP, INDEX, ALTER, CREATE TEMPORARY TABLES, CREATE VIEW, EVENT, TRIGGER, SHOW VIEW, CREATE ROUTINE, ALTER ROUTINE, EXECUTE		No	<a href="#">Editar privilegios</a> <a href="#">Exportar</a>
<input checked="" type="checkbox"/>	rcoy	%	Sí	ALL PRIVILEGES		Sí	<a href="#">Editar privilegios</a> <a href="#">Exportar</a>
<input checked="" type="checkbox"/>	dramirez	%	Sí	SELECT, INSERT, UPDATE, DELETE, FILE		No	<a href="#">Editar privilegios</a> <a href="#">Exportar</a>
<input type="checkbox"/>	pma	localhost	No	USAGE		No	<a href="#">Editar privilegios</a> <a href="#">Exportar</a>
<input type="checkbox"/>	root	127.0.0.1	No	ALL PRIVILEGES		Sí	<a href="#">Editar privilegios</a> <a href="#">Exportar</a>
<input type="checkbox"/>	root	:::1	No	ALL PRIVILEGES		Sí	<a href="#">Editar privilegios</a> <a href="#">Exportar</a>
<input type="checkbox"/>	root	localhost	No	ALL PRIVILEGES		Sí	<a href="#">Editar privilegios</a> <a href="#">Exportar</a>
<input checked="" type="checkbox"/>	Irincon	%	Sí	CREATE, DROP, RELOAD, SHUTDOWN, PROCESS, REFERENCES, INDEX, ALTER, SHOW DATABASES, SUPER, CREATE TEMPORARY TABLES, LOCK TABLES, REPLICATION SLAVE, REPLICATION CLIENT, CREATE VIEW, EVENT, TRIGGER, SHOW VIEW, CREATE ROUTINE, ALTER ROUTINE, CREATE USER, EXECUTE		Sí	<a href="#">Editar privilegios</a> <a href="#">Exportar</a>

Primer paso es acceder al panel de configuración de mysql, podemos usar para este fin, si el servidor el local se puede acceder de la siguiente forma <http://127.0.0.1/phpmyadmin/>

De

Vista global de las cuentas de usuario

Existe una cuenta de usuario que permite a cualquier usuario de localhost conectarse. Esto evitará conectarse a otros usuarios, si la parte del host de su cuenta permite una conexión desde cualquier host (%).

Nombre de usuario	Nombre del servidor	Contraseña	Privilegios globales	Grupo de usuario	Conceder	Acción
<input type="checkbox"/> cualquiera	%	No	USAGE	No		Editar privilegios  Exportar
<input type="checkbox"/> cualquiera	localhost	No	USAGE	No		Editar privilegios  Exportar
<input type="checkbox"/> pma	localhost	No	USAGE	No		Editar privilegios  Exportar
<input type="checkbox"/> root	127.0.0.1	No	ALL PRIVILEGES	Sí		Editar privilegios  Exportar
<input type="checkbox"/> root	:::1	No	ALL PRIVILEGES	Sí		Editar privilegios  Exportar
<input type="checkbox"/> root	localhost	No	ALL PRIVILEGES	Sí		Editar privilegios  Exportar

Seleccionar todo Para los elementos que están marcados: Exportar

Nuevo

Agregar cuenta de usuario

forma general, en phpmyadmin podemos crear un usuario en mysql por la pestaña de cuentas de usuario. Es de aclarar que al crear un usuario por esta pestaña, el usuario seria creado y asignado a todas las bases de datos existentes. Como podemos ver en la imagen, se aprecian los usuarios creados actualmente en el sistema, el nombre de servidor, si tienen establecida una contraseña, el tipo de privilegios, también podemos editar sus privilegios.

En la pestaña de agregar nueva cuenta, será donde crearemos el usuario. Para ello nos pedirán una serie de datos, estos son el nombre del usuario, el nombre el host (por defecto se establece como %, que significa que este usuario puede acceder desde cualquier host, si queremos que el usuario solo se pueda acceder desde el pc que tiene instalado el mysql colocaríamos Localhost), la contraseña y el tipo de autenticación, también nos da la posibilidad de generar una contraseña aleatoria.

The screenshot shows the 'Agregar cuenta de usuario' (Add user account) form in the MySQL User Management tool. The form is titled 'Información de la cuenta' (Account information). It contains the following fields and options:

- Nombre de usuario:** A text input field with the value 'rooy'.
- Nombre de Host:** A dropdown menu set to 'Cualquier servidor' (Any server) and a text input field with the value '%'. A blue information icon is next to it.
- Contraseña:** A text input field with masked characters (dots).
- Debe volver a escribir:** A text input field with masked characters (dots).
- Complemento de autenticación:** A dropdown menu set to 'Native MySQL authentication'.
- Generar contraseña:** A button labeled 'Generar' and a text input field.

Luego podemos seleccionar los privilegios que posee el usuario, estos privilegios pueden ser sobre los datos, la estructura o administración de la base de datos.

The screenshot shows the 'Privilegios globales' (Global privileges) section in the MySQL User Management tool. It includes a 'Selecionar todo' (Select all) button and a note: 'Nota: Los nombres de los privilegios de MySQL están expresados en inglés.' (Note: The names of the MySQL privileges are expressed in English.)

The privileges are organized into four categories, each with a 'Seleccionar todo' (Select all) button:

- Datos:** SELECT, INSERT, UPDATE, DELETE, FILE.
- Estructura:** CREATE, ALTER, INDEX, DROP, CREATE TEMPORARY TABLES, SHOW VIEW, CREATE ROUTINE, ALTER ROUTINE, EXECUTE, CREATE VIEW, EVENT, TRIGGER.
- Administración:** GRANT, SUPER, PROCESS, RELOAD, SHUTDOWN, SHOW DATABASES, LOCK TABLES, REFERENCES, REPLICATION CLIENT, REPLICATION SLAVE, CREATE USER.
- Límites de recursos:** A section with a note: 'Nota: si cambia los parámetros de estas opciones a 0 (cero), remueve el límite.' (Note: if you change the parameters of these options to 0 (zero), remove the limit.) and four input fields: MAX QUERIES PER HOUR, MAX UPDATES PER HOUR, MAX CONNECTIONS PER HOUR, and MAX USER\_CONNECTIONS, all set to 0.

Por ultimo podemos indicar si se quiere establecer autenticación por SSL (Secure Sockets Layer) que nos asegura una conexión cifrada y segura.

☐ Requiere SSL

☐ SPECIFIED

REQUIRE\_CIPHER

REQUIRE\_ISSUER

REQUIRE\_SUBJECT

☐ REQUIRE\_X509

☒ REQUIRE\_SSL










Cuando damos en crear phpmyadmin nos muestra que el usuario ha sido creado correctamente y además nos muestra cómo sería la sentencia para crearlo por consola.


Podemos ver los privilegios que posee los usuarios sobre una base de datos, si seleccionamos la base de datos y vamos a la pestaña de privilegios. Podemos apreciar los usuarios asignados a nuestra base de datos en este caso tenemos el root y cguateque que son usuarios que poseen todos los privilegios y tenemos los usuarios de los compañeros que están limitados a ciertas funciones entro de la base de datos.

Servidor: 127.0.0.1 • Base de datos: cguateque

Estructura SQL Buscar Generar una consulta Exportar Importar Operaciones Privilegios Rutinas Más

Usuarios con acceso a "cguateque"

Nombre de usuario	Nombre del servidor	Tipo	Privilegios	Conceder	Acción
lrincon	%	global	SELECT	No	 Editar privilegios
		especifico para la base de datos	SELECT	No	 Editar privilegios
root	%	global	ALL PRIVILEGES	Si	 Editar privilegios
danirez	%	global	INSERT	No	 Editar privilegios
gcarmelo	%	global	SELECT	No	 Editar privilegios
root	127.0.0.1	global	ALL PRIVILEGES	Si	 Editar privilegios
root	:1	global	ALL PRIVILEGES	Si	 Editar privilegios
root	localhost	global	ALL PRIVILEGES	Si	 Editar privilegios
diegocera	%	global	DELETE	No	 Editar privilegios

☐ Seleccionar todo Para los elementos que están marcados:  Exportar

Una vez tenemos los privilegios establecidos podemos probar por ejemplo.

Se ha definido un usuario lrincon que solo puede hacer consultas a las base de datos si probamos hacer una consulta con este usuario efectivamente podemos evidenciar que se pueden realizar consultas.

Mostrar ventana de consultas SQL

✓ Mostrando filas 0 - 8 (total de 9, La consulta tardó 0.0000 segundos.)

```
SELECT * FROM `mecanicos` WHERE 1
```

[ Editar en línea ] [ Editar ] [ Explicar SQL ] [ Crear código PHP ] [ Actualizar ]

Obtendremos un error donde nos indica que el usuario no posee privilegios para realizar dicha operación.

**Error**

consulta SQL:

```
update `mecanicos` set `nombres_mec` = 'Andres' WHERE `idmecanicos` = 1
```

MySQL ha dicho: 

Pero si realizamos la misma operación con el usuario rcory que tiene privilegios de consulta y de actualización, podemos observar que pudo realizar los cambios en la tabla.

Mostrar ventana de consultas SQL

✓ 1 fila afectada. (La consulta tardó 0.0640 segundos.)


```
update `mecanicos` set `nombres_mec` = 'Andres' WHERE `idmecanicos` = 1
```

[ Editar en línea ] [ Editar ] [ Crear código PHP ]

En esta parte ingresamos con un usuario en partículas que solo puede borrar:

- Servidor: localhost via TCP/IP
- Tipo de servidor: MySQL
- Versión del servidor: 5.6.26 - MySQL Community Server (GPL)
- Versión del protocolo: 10
- Usuario: angelam@localhost
- Conjunto de caracteres del servidor: UTF-8 Unicode (utf8)

Como observamos en la siguiente imagen al intentar hacer una consulta nos sale error, porque el usuario no tiene permisos.

 #1142 - SELECT command denied to user dramirez@'localhost' for table 'areaatencion'

En esta parte ingresamos con el usuario rcory que tiene permisos para hacer select.

- Servidor: localhost via TCP/IP
- Tipo de servidor: MySQL
- Versión del servidor: 5.6.26 - MySQL Community Server (GPL)
- Versión del protocolo: 10
- Usuario: henryg@localhost
- Conjunto de caracteres del servidor: UTF-8 Unicode (utf8)

Como podemos observar los selects se pueden hacer sin ningún problema:

+ Opciones

				Categoria_Id	Nombre_Cat
<input type="checkbox"/>	Editar	Copiar	Borrar	1	deportivo
<input type="checkbox"/>	Editar	Copiar	Borrar	2	camioneta
<input type="checkbox"/>	Editar	Copiar	Borrar	3	camion
<input type="checkbox"/>	Editar	Copiar	Borrar	4	tractor

Pero si intentamos hacer delete, nos aparece un error ya que el usuario no tiene estos permisos.

### Error

consulta SQL:

```
DELETE FROM rcoy.`categorias` WHERE `categorias`.
```

MySQL ha dicho: ?

```
#1142 - DELETE command denied to user  
Irincon@'localhost' for table 'categorias'
```

Por ultimo ingresamos con otro usuario que tiene privilegios de delete.



- Servidor: localhost via TCP/IP
- Tipo de servidor: MySQL
- Versión del servidor: 5.6.26 - MySQL Community Server (GPL)
- Versión del protocolo: 10
- Usuario: christianl@localhost
- Conjunto de caracteres del servidor: UTF-8 Unicode (utf8)

A continuación observamos que realiza el proceso correctamente y elimina la fila.

✓ 1 fila eliminada. (La consulta tardó 0.0695 segundos.)

Con el tutorial anterior observamos que cada usuario puede contar con privilegios diferentes lo que asegura la confiabilidad y la seguridad de la base de datos. Esto es muy usado en grandes empresas para que no todos los usuarios puedan hacer lo que quieran.

Normalmente en las bases de datos se quitan los permisos de borrar, ya que con un registro borrado pueden alterar drásticamente toda la información.



### **1. Hacer copia de seguridad de la base de datos.**

Para la realización de las copias de seguridad se realiza a través del siguiente comando.

```
mysqldump --user=root -p --databases acme > copia_seguridad.sql
```

En el cual se inicia con el comando **mysqldump**

Sobre el nombre del usuario administrador **--user=nombre usuario**.

Luego se solicita el password de ingreso del usuario por seguridad con **-p**.

Seguido de la instrucción **--databases** que incluye la línea de comando CREATE DATABASE /\*!32312 IF NOT EXISTS en la base de datos si no se incluye el **--databases** se debe tener creada la base de datos previamente para la reconstrucción de las tablas.

Luego se asigna el nombre de la base de datos a respaldar **acme**.

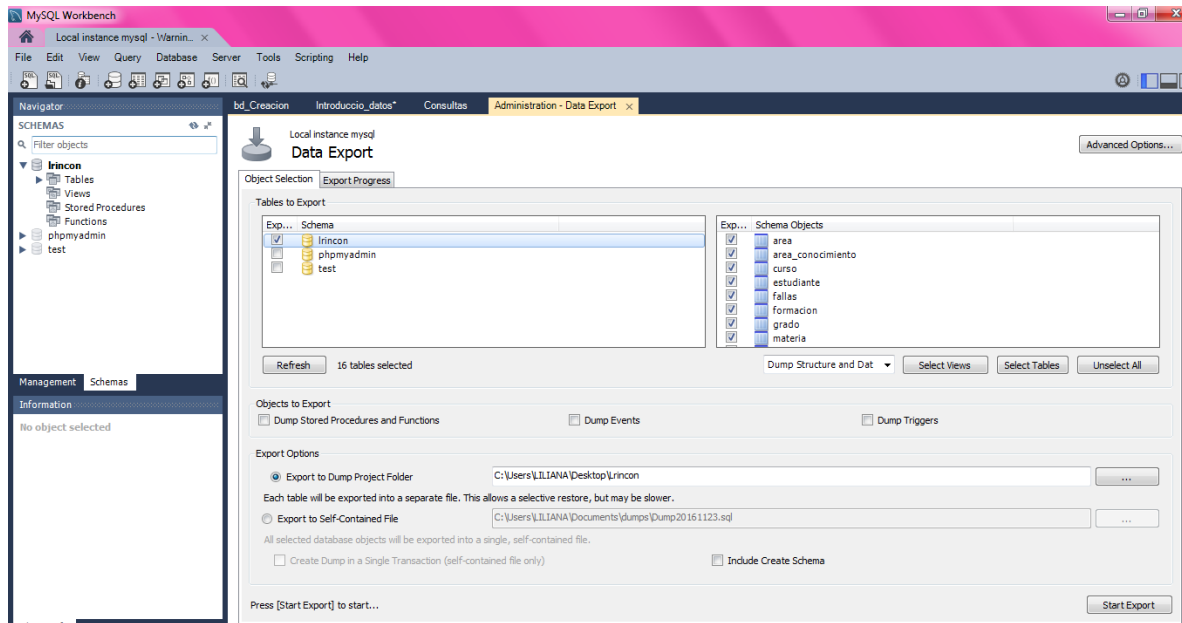
Seguido del símbolo **>** que indica exportar.

Por último, el nombre de la copia de seguridad que se desea para este caso **copia\_seguridad.sql**.

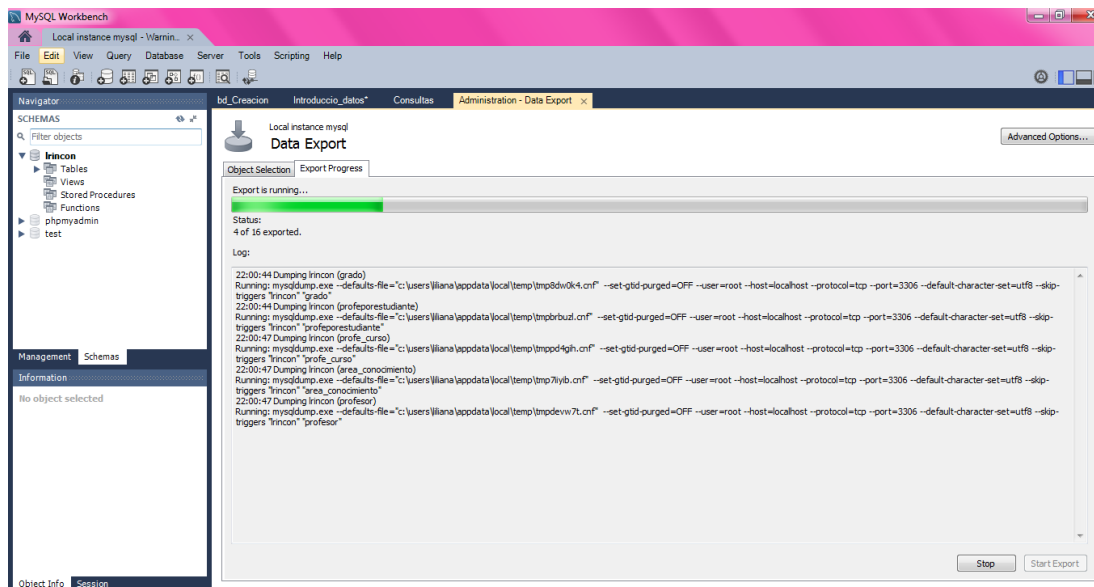
```
mysqldump --user=TU_USUARIO --password=TU_CONTRASEÑA  
NOMBRE_BASE_DE_DATOS > copia_seguridad.sql  
mysql> --user=Coy --password=Coy46 lrincon > copia_seguridad.sql
```

Por otra parte se debe exportar la base de datos de la siguiente manera:

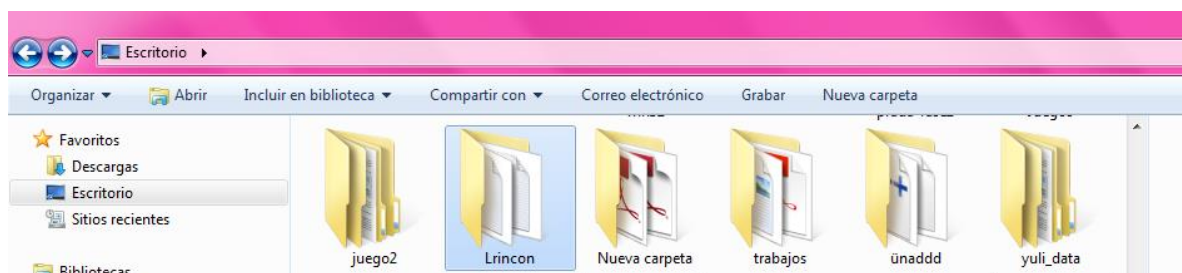
Se debe ingresar la base de datos a exportar



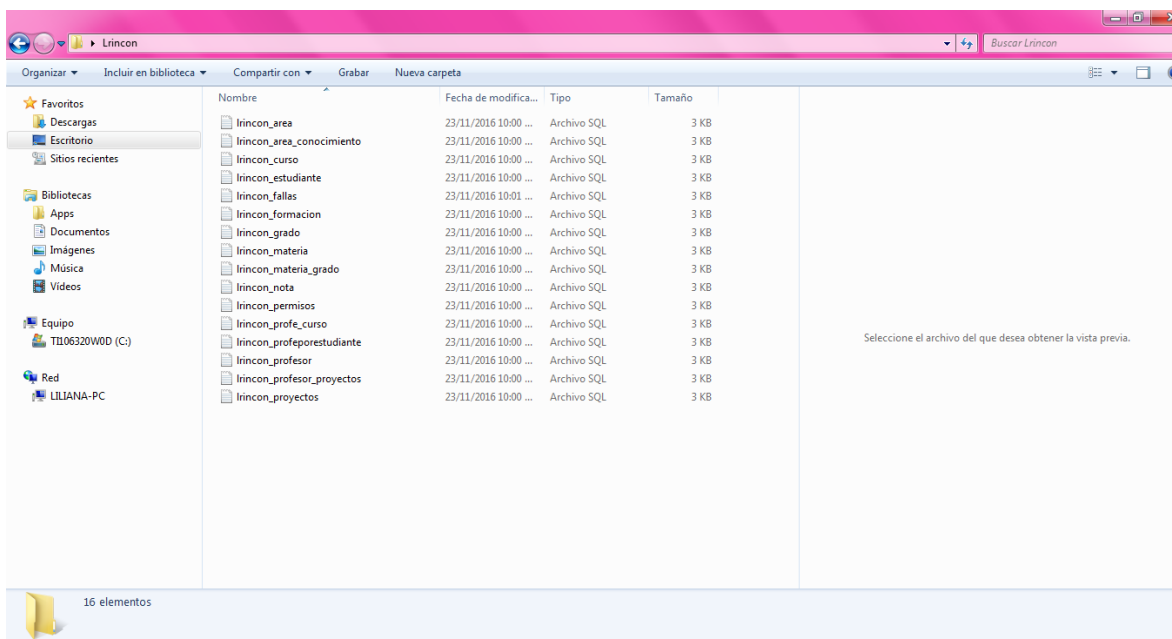
Aparecen los 16 tablas de la base de datos exportada



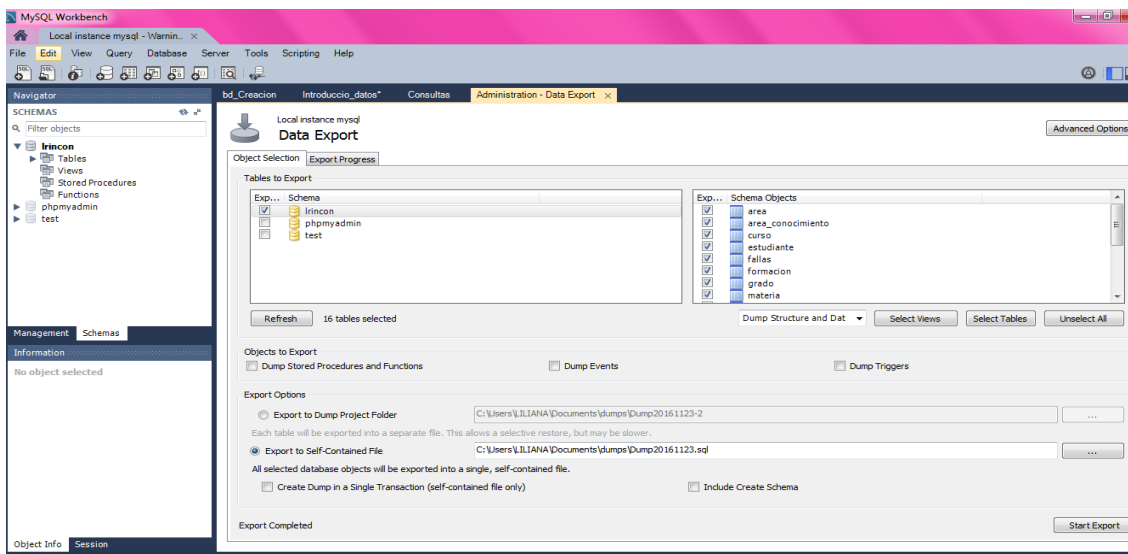
Se verifica que la base de datos esta guardada en el sitio



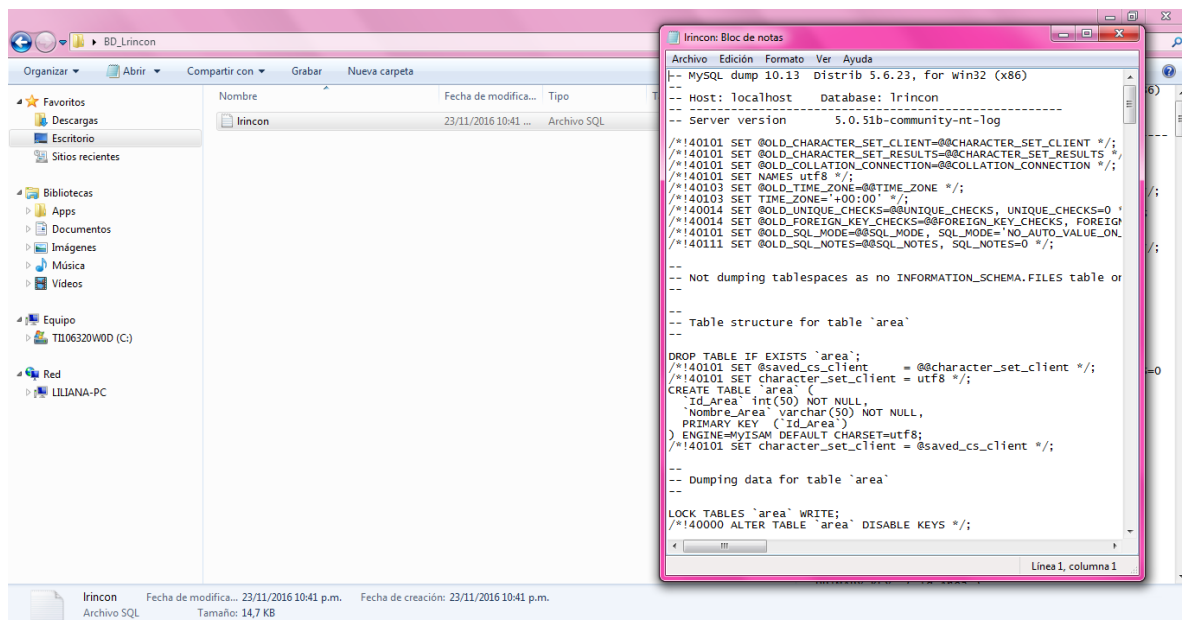
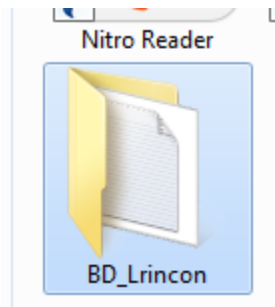
Se evidencia la base de datos con sus tablas



Para exportar la base de datos completa se puede realizar de la siguiente manera:



**4. Descargar las copias de seguridad subidas por los compañeros del grupo colaborativo y restaurarlas en su equipo. Consultar el listado de bases de datos y tomar un pantallazo para presentarlo como prueba de que se desarrolló exitosamente la recuperación de todas las copias**



## CONCLUSIONES

En el anterior trabajo podemos observar los conceptos básicos de sentencias de seguridad en MySQL. Por otro lado gestionamos la seguridad de nuestras bases de datos con la creación y manipulación de los privilegios de cada uno, garantizando la integralidad de la base de datos que estamos administrando.

Por ultimo podemos decir que MySQL es una robusta fuente para almacenar nuestros datos de manera segura y contiene muchas formas de realizar manejo de datos, desde la consola o en modo gráfico.

Se logra identificar la importancia de tener seguro la base de datos en cuanto a virus problemas técnicos entre otros. Esto hace que una base de datos sea aún más confiable. Por tal razon si hay pérdida se puede recuperar la información.

## REFERENCIAS BIBLIOGRAFICAS

- Pozo, Salvador (2005). Lenguaje SQL usuarios y privilegios. Recuperado de:  
<http://mysql.conclase.net/curso/?cap=013> ORACLE(2014)
- MySQL 5.0. Manual de referencia. Capítulo 5. Administración de la base de datos.  
Recuperado de: [http://datateca.unad.edu.co/contenidos/301125/2015-1/MySQL\\_Referencia\\_cap\\_5.pdf](http://datateca.unad.edu.co/contenidos/301125/2015-1/MySQL_Referencia_cap_5.pdf)
- Microsoft. *office.com*. Recuperado el 08 de 05 de 2016, de  
<https://support.office.com/es-es/article/Hacer-una-copia-de-seguridad-de-una-base-de-datos-483d3d0a-4786-4bff-9f70-b11baea520a7>
- 
- Drakonisl1. *slideshare.net*. Recuperado el 08 de 05 de 2016, de  
<http://es.slideshare.net/Drakonisl1/integridad-y-seguridad-en-las-bases-de-datos-presentation>
- 
- Unam (s.f.). *www.unam.mx* Recuperado el 08 de 05 de 2016, de  
<http://revista.seguridad.unam.mx/numero-12/principios-b%C3%A1sicos-de-seguridad-en-bases-de-datos>
- 
- Vegas, J. (s.f.). *www.infor.uva.es*. Recuperado el 08 de 05 de 2016, de  
<http://www.infor.uva.es/~jvegas/cursos/bd/oraseg/oraseg.html#2>
- Wikipedia. (s.f.). *wikipedia.org*. Recuperado el 07 de 05 de 2016, de  
[https://es.wikipedia.org/wiki/Trigger\\_\(base\\_de\\_datos\)](https://es.wikipedia.org/wiki/Trigger_(base_de_datos))