Controlling and Managing MySQL Database

Session 16

Objectives

- *Describe the creation of user accounts in MySQL*
- *Identify the privileges in MySQL*
- *Explain the privileges present in MySQL*
- *Explain the commands for setting up of restrictions in MySQL*

User Accounts

- In MySQL, a user is a record present in the user table of the MySQL server
- After the installation of MySQL server is complete, the only existing user account is root user account
- MySQL uses the root user account to execute administrative commands
- MySQL server does not require a password for the root user, if you have not set a password at configuration
- MySQL allows the root user to set a password using the `mysqladmin` command
- In MySQL, you can create a new user account by using the `CREATE USER` or `GRANT` command

Table lists the commands for user accounts in MySQL

MySQL Command	Description
CREATE USER	Allows the creation of a new account
SET PASSWORD	Allows to assign a password
GRANT	Allows to assign privileges
RENAME USER	Allows to change the name of the account
REVOKE	Allows to cancel or remove privileges
DROP USER	Allows to delete an account

User Accounts

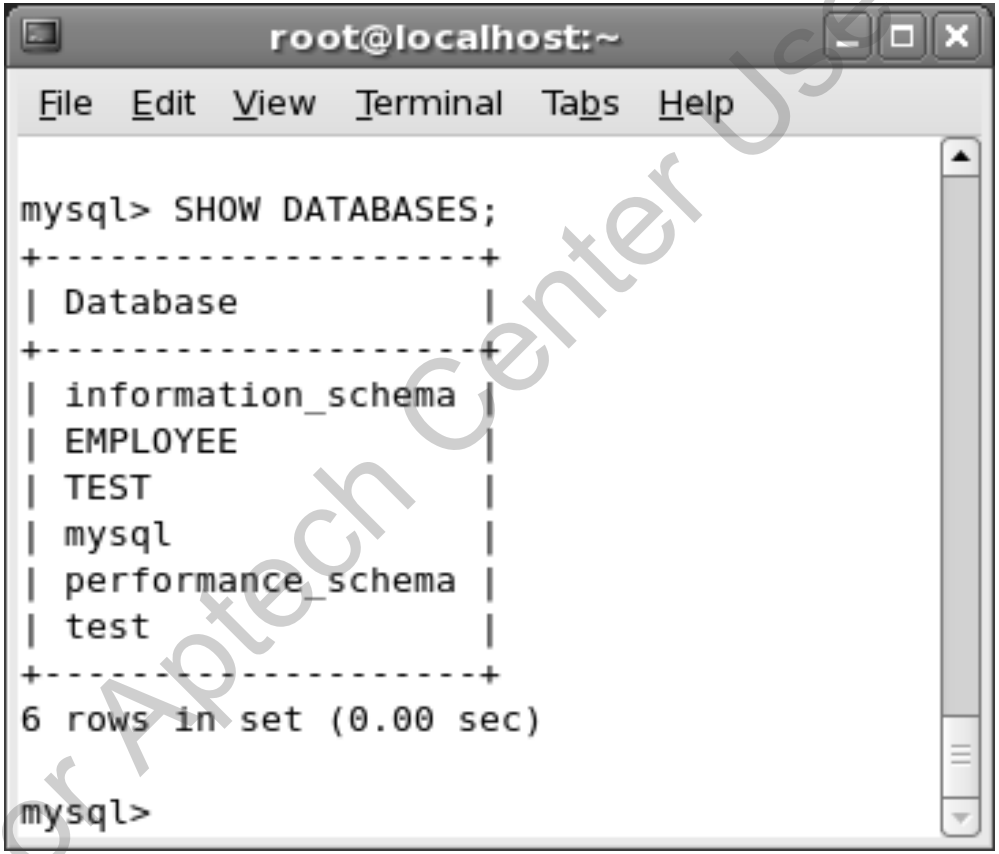
- The MySQL Grant tables contain information about the privileges to user accounts
- These tables are stored in the mysql database and their names are as follows:
  - `columns_priv` – contains access details for columns in tables
  - `db` – contains information of user accounts that can access the database
  - `host` – contains list of user accounts that have access to the database
  - `tables_priv` – contains access details for tables in a database
  - `user` – contains access details for a user account

Adding User Accounts Using the CREATE USER Command 1-8

- MySQL server provides the `CREATE USER` command to add new accounts
- You must have the global `CREATE USER` privilege for the mysql database to execute the command
- You must login as the root user and connect to the `mysql` database to create user accounts
- To view the list of available databases, enter the following command at the command prompt:

```
SHOW DATABASES;
```

Figure displays a list of available databases, including `mysql`, the default database

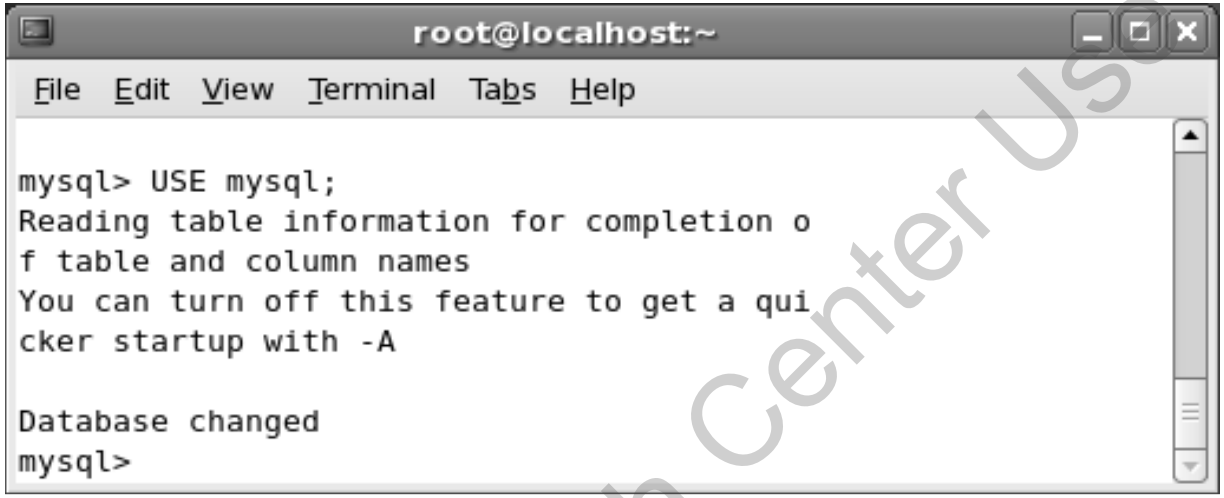


Adding User Accounts Using the CREATE USER Command 3-8

- To load the default mysql database, enter the following command at the command prompt:

```
USE mysql;
```

Figure displays the output of the command



- To list the tables present in the default database mysql, enter the following command at the command prompt:

```
SHOW TABLES;
```

- This command, lists all the available tables in the default database

Figure displays the output of the command

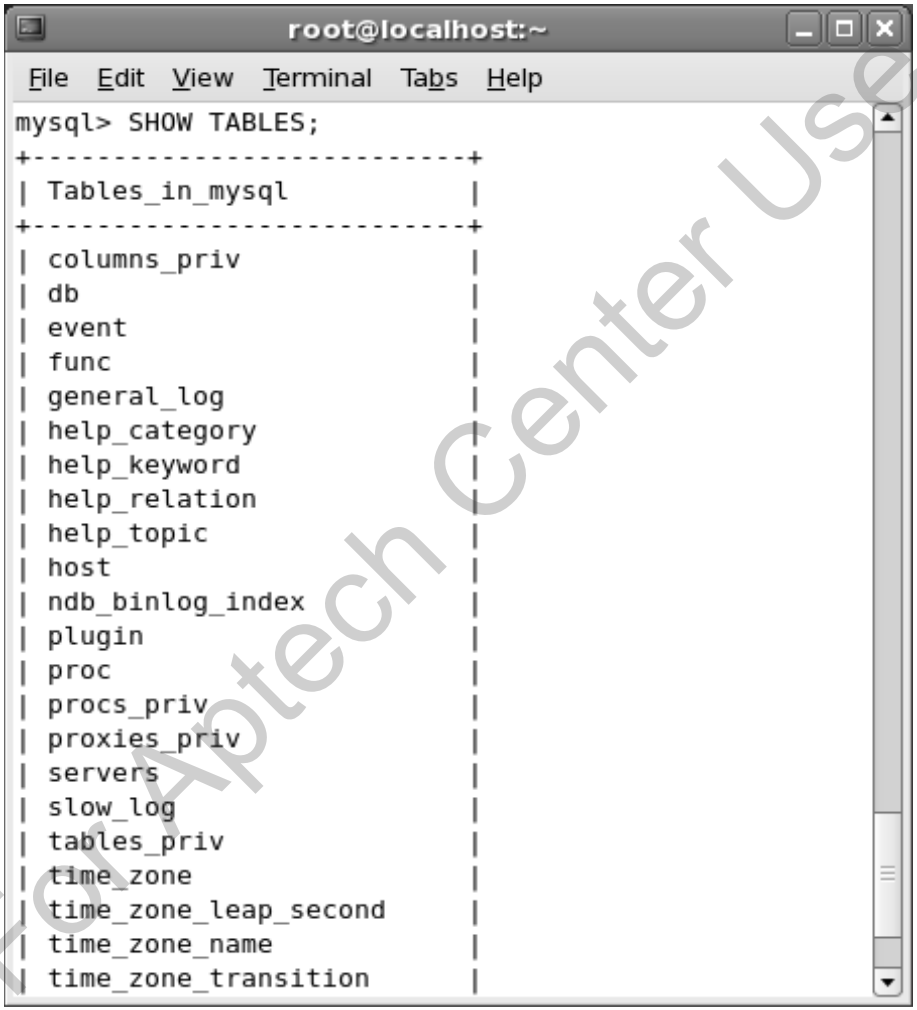


Table lists the functions of some of the tables in the mysql database

Table Name	Description
columns_priv	Contains access details for columns in tables
db	Contains information of user accounts that can access the database
func	Contains list of functions and is updated at MySQL server restart
host	Contains list of user accounts that have access to the database
proc	Contains information about database and table modifications
tables_priv	Contains access details for tables in a database
user	Contains access details for a user account

Adding User Accounts Using the CREATE USER Command 8-8

- The `CREATE USER` command allows you to add a user account
- To add a user account using the `CREATE USER` command, enter the following command at the command prompt:

```
CREATE USER 'temp_user'@'localhost';
```

- You can also create a new user account using the `GRANT` command
- You can grant and revoke rights to a MySQL user account at four levels that are as follows:
  - **Global Level**
    - Grants and revokes privileges that apply to all the databases present on the server
    - These privileges are stored in the `user` table of the `mysql` database

- **Database Level**
  - Grants and revokes privileges that apply to all the tables of a given database
  - These privileges are stored in the `db` and host tables of the `mysql` database
- **Table Level**
  - Grants and revokes privileges that apply to all the columns of a given table
  - These privileges are stored in the `tables_priv` table of the `mysql` database

- **Column Level**
  - Grants and revokes privileges that apply to a single column in a given table
  - These privileges are stored in the `columns_priv` table of the `mysql` database

Creating User Account Using the GRANT Command

Table lists the privilege types that can be specified in the GRANT and REVOKE statements

Privilege	Description
ALL [PRIVILEGES]	Assigns all rights except GRANT OPTION. The user cannot set access controls
ALTER	Allows the use of the ALTER TABLE command
CREATE	Allows the use of the CREATE TABLE command
CREATE TEMPORARY TABLES	Allows the use of the CREATE TEMPORARY TABLE command
DELETE	Allows the use of the DELETE command
DROP	Allows the use of the DROP TABLE command
EXECUTE	Allows the user to run stored procedures
INDEX	Allows the use of the CREATE INDEX and DROP INDEX commands
INSERT	Allows the use of the INSERT command
SELECT	Allows the use of the SELECT command
SHOW DATABASES	Allows the use of the SHOW DATABASES command
SHUTDOWN	Allows the use of the mysqladmin shutdown privilege
UPDATE	Allows the use of the UPDATE command
USAGE	Allows the use of the 'no privileges' user account
GRANT OPTION	Allows privileges to be granted or revoked from other accounts

- The syntax for creating a new user with all privileges for all the tables in the database and who can connect to the `localhost` or any other host is as follows:

```
GRANT ALL [PRIVILEGES] ON *.* TO
user_name[IDENTIFIED BY [PASSWORD] 'password' [WITH
GRANT OPTION]
```

where,

`user_name` – specifies the name for the new account

`IDENTIFIED BY [PASSWORD]` – sets the password for the new account

`password` – specifies the password for the account

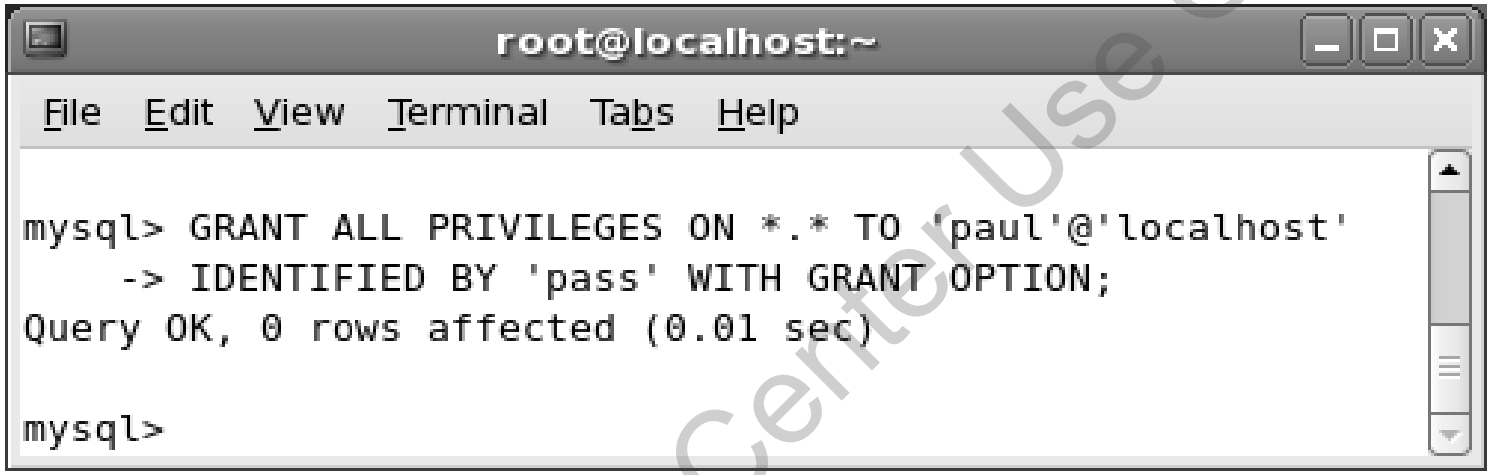
`[WITH GRANT OPTION]` – allows the user to set privileges

- To create a new user with all the privileges and access to MySQL from `localhost`, enter the following command at the command prompt:

```
GRANT ALL PRIVILEGES ON *.* TO 'paul'@'localhost'
IDENTIFIED BY 'pass' WITH GRANT OPTION;
```

- The command will assign all privileges to the user paul present in the instance, localhost.
- The clause WITH `GRANT OPTION` will allow the user to set privileges

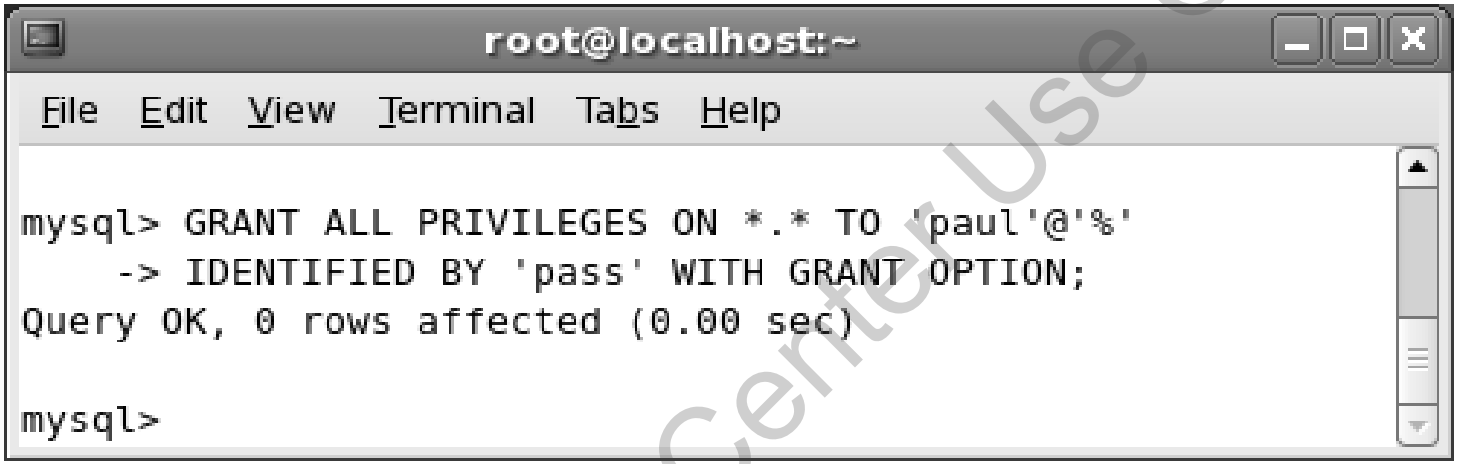
Figure displays the output of the GRANT command



- To create a new user with all the privileges without specifying the `localhost` or any other host, enter the following command at the command prompt:

```
GRANT ALL PRIVILEGES ON *.* TO 'paul'@'%'
IDENTIFIED BY 'pass' WITH GRANT OPTION;
```

Figure displays the output of the GRANT command



Create User Account Using the INSERT command

- The INSERT command adds a new entity to the user table of the mysql database
- This table contains information of all user accounts in the database with their hostnames, passwords, and privileges
- To create a user by the name martin using the INSERT command, enter the following command at the command prompt:

```
INSERT INTO user VALUES ('localhost','martin',
PASSWORD
('pass'),'Y','Y','Y','Y','Y','Y','Y','Y','Y','Y','Y'
,'Y','Y','Y','Y','Y','Y','Y','Y','Y','Y','Y','Y'
,'Y','Y','Y','Y','Y','Y','Y','Y','Y','Y','Y','Y'
,'Y','Y');
```

- The command will add a record into the user table with the values specified in the VALUES clause

Figure displays the output of the command

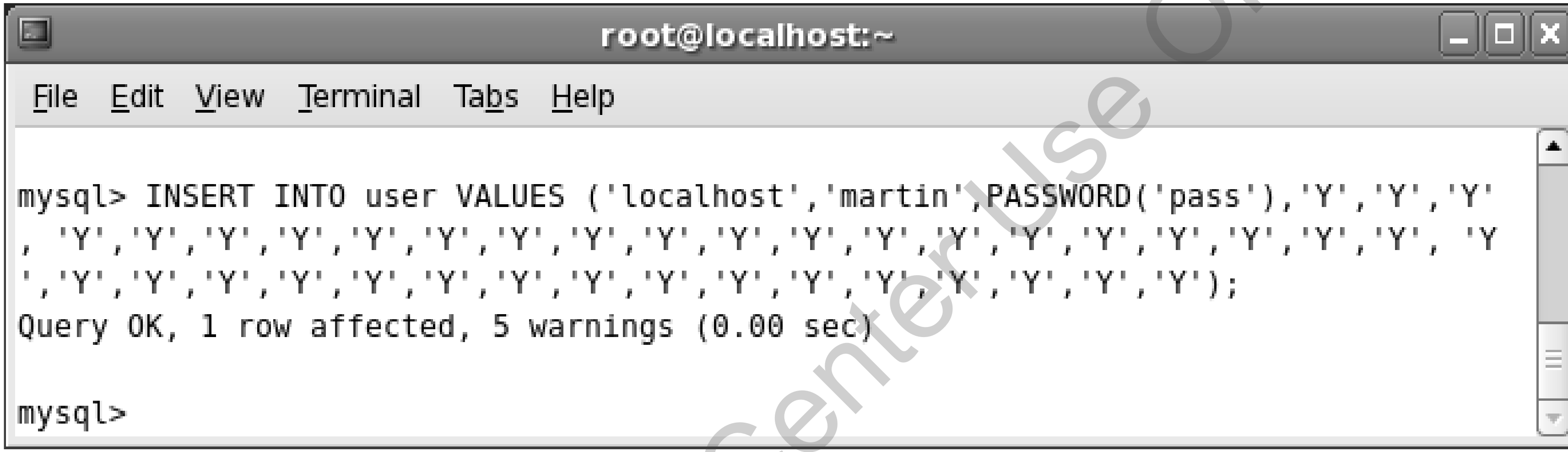


Table lists some of the column names in the user table of mysql database

Column Name
Select_priv
Insert_priv
Update_priv
Delete_priv
Index_priv
Alter_priv
Create_priv
Drop_priv
Grant_priv
Create_view_priv
Show_view_priv
Create_routine_priv
Alter_routine_priv
Execute_priv

Table lists some of the column names in the user table of mysql database

Column Name
Create_tmp_table_priv
Lock_tables_priv
References_priv
Shutdown_priv
Process_priv
File_priv
Show_db_priv
Super_priv
Repl_slave_priv
Repl_client_priv
Create_user_priv

- After creating a user with the `INSERT` command, you must run the `FLUSH PRIVILEGES` command
- This command reloads all the grant tables and assigns the new privileges
- The syntax for the `FLUSH PRIVILEGES` command is:

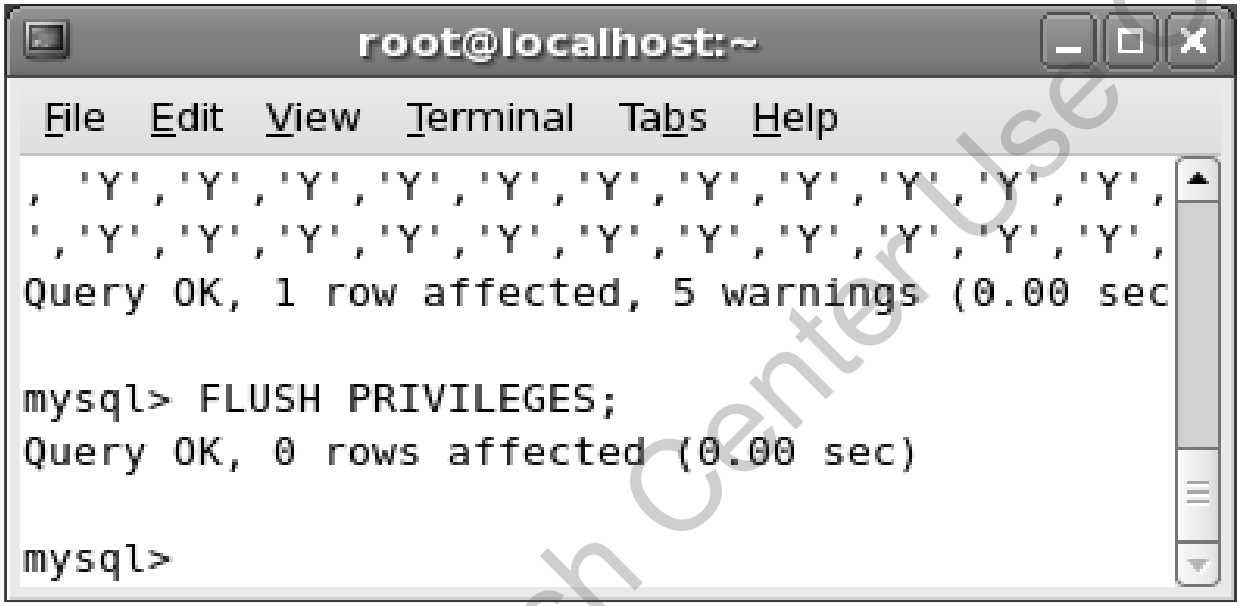
```
FLUSH PRIVILEGES;
```

where,

`FLUSH` – specifies the reloading of the object

`PRIVILEGES` – specifies to refresh and reload the privileges

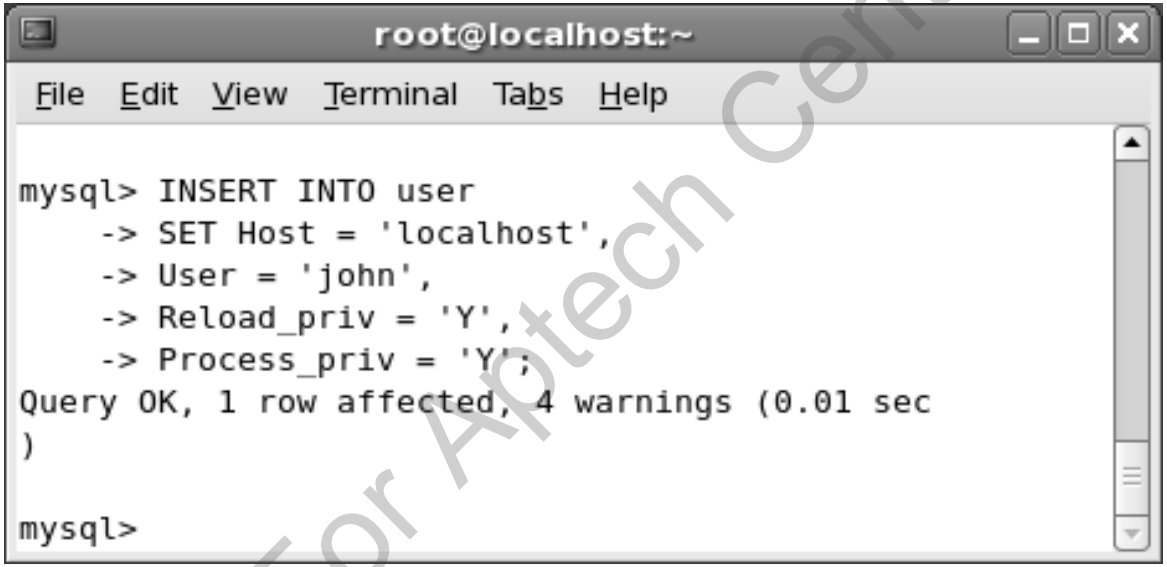
Figure displays the output of the command



- To create a user having only `RELOAD` and `PROCESS` privileges, enter the following command at the command prompt:

```
INSERT INTO user SET Host='localhost', User='john',
Reload_priv='Y', Process_priv='Y';
```

Figure displays the output of the command

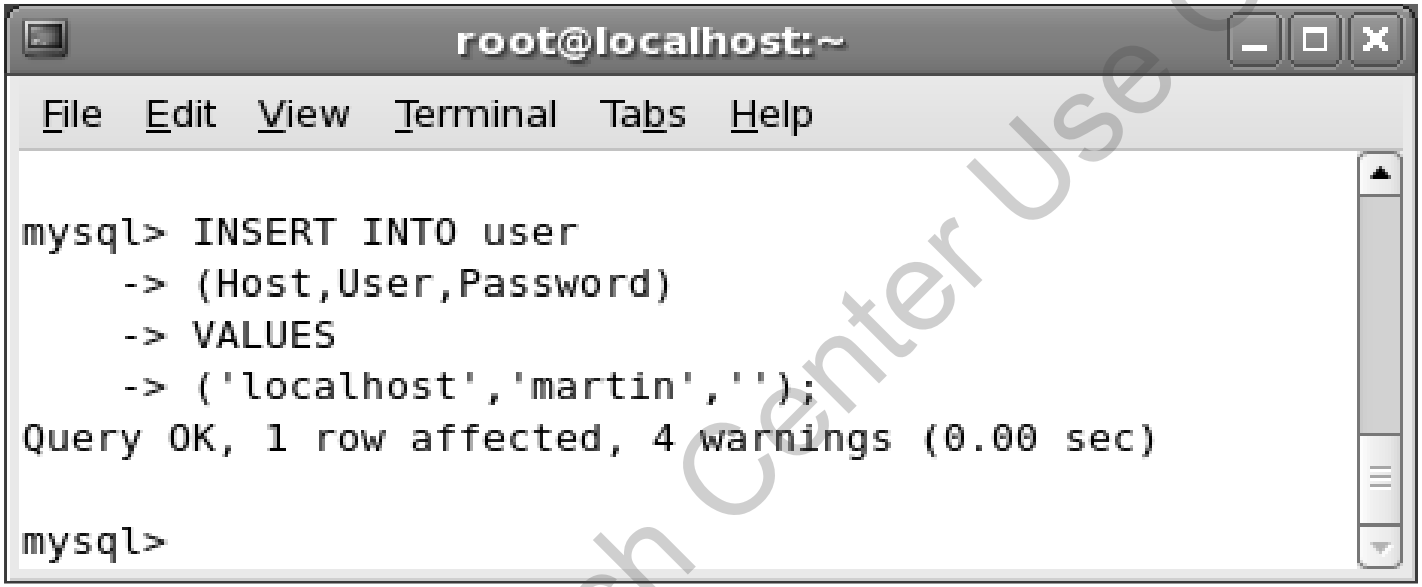


- To create an account without any access and password privileges, enter the following command at the command prompt:

```
INSERT INTO user (Host, User, Password) VALUES
('localhost','martin', ' ');
```

- The command adds a new entry into the user table and specifies the hostname, user name, and password

Figure displays the output of the command



- **Ensuring Valid Connections**
  - MySQL uses security for all connections and queries
  - The server controls access by ensuring that the following checks are performed:
    - Checks whether the user has privileges to connect to the server
    - Verifies the privileges assigned to the user after the connection is established

Rename User Account

- MySQL enables you to change the name of an existing user account
- You can edit the name of the account using the `RENAME USER` command
- MySQL enables you to change the name of the account without modifying the privileges
- For example, to rename a user account, enter the following command at the command prompt:

```
RENAME USER 'temp_user'@'localhost' TO 'temp-user1'@'localhost';
```

- The command changes the name of the object by specifying the type of object to rename

Altering Privileges

- Managing privileges of users involves assigning and revoking user rights and privileges
- You can use database-independent SQL commands, such as `GRANT` and `REVOKE` to manage user privileges
- To alter privileges for the user, use `GRANT` command with selective privileges

Altering Privileges

- For example, the command to grant permission to a user to read data from a specified table is as follows:

```
GRANT SELECT ON table_name TO user_name;
```

where,

`GRANT` – assigns privileges or rights

`SELECT` – specifies the type of privilege or right to assign

`table_name` – specifies the name of the table

`user_name` – specifies the name of the user or account

- The command assigns the `SELECT` permission to the user specified in the `user_name` clause of the statement

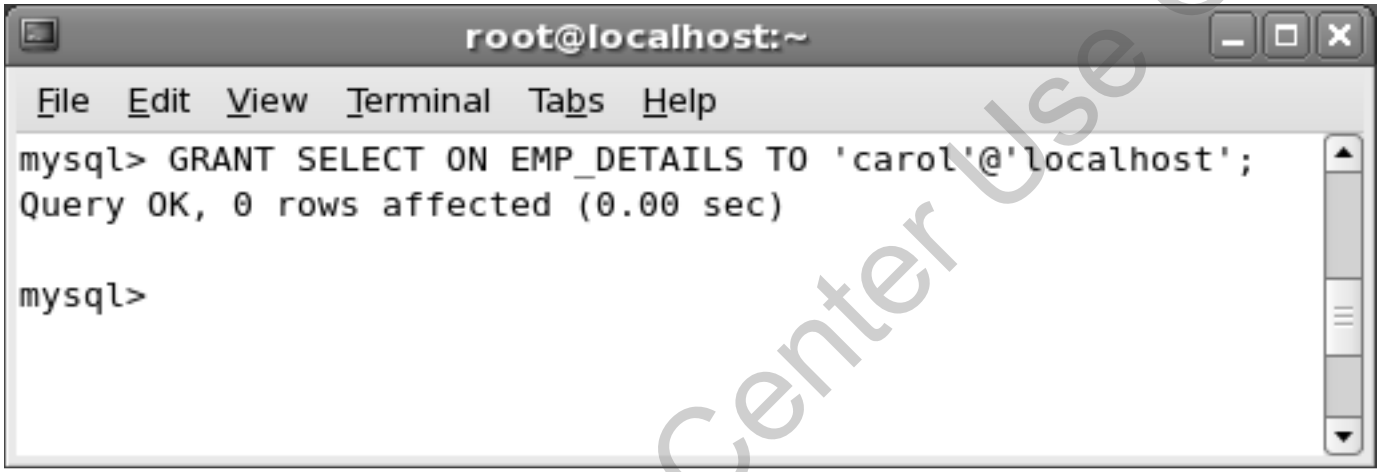
Altering Privileges

- To create a user who has the permission to read data from the `EMP_DETAILS` table, enter the following command at the command prompt:

```
GRANT SELECT ON EMP_DETAILS TO 'carol'@'localhost';
```

- The command will assign the `SELECT` privilege to the user, `carol`

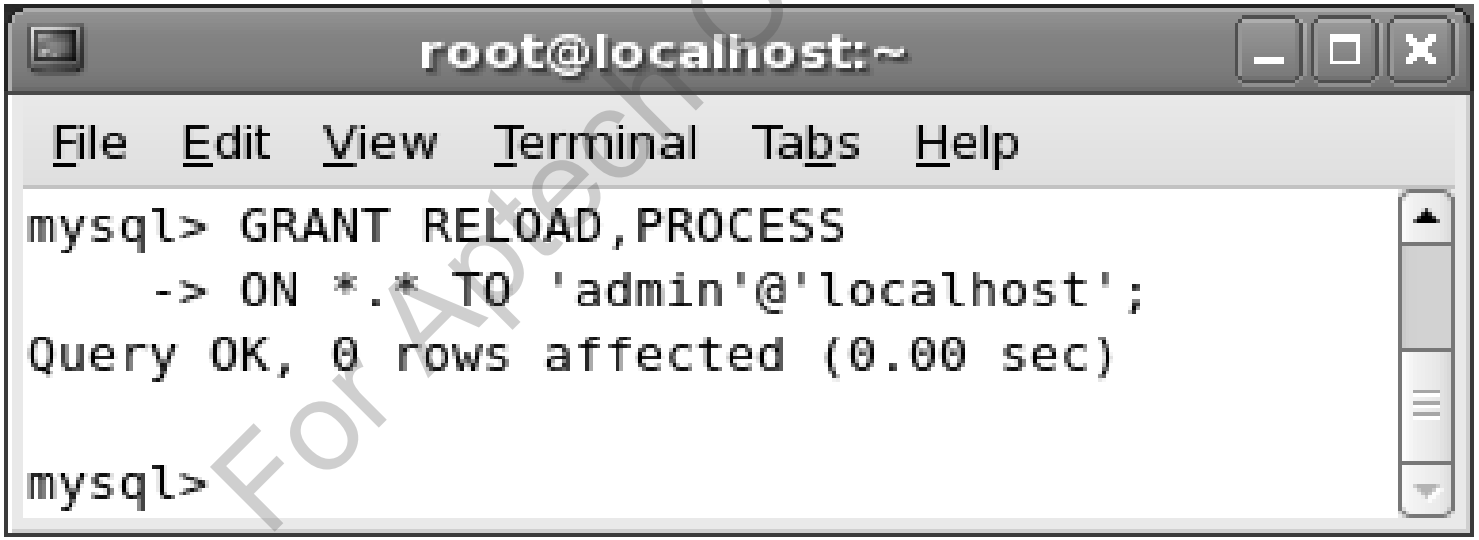
Figure displays the output of the command



- To create a user who is granted only with the `RELOAD` and the `PROCESS` rights for the entire database, with no password and has access only from the host named, localhost, enter the following command at the command prompt:

```
GRANT RELOAD,PROCESS ON *.* TO 'admin'@'localhost';
```

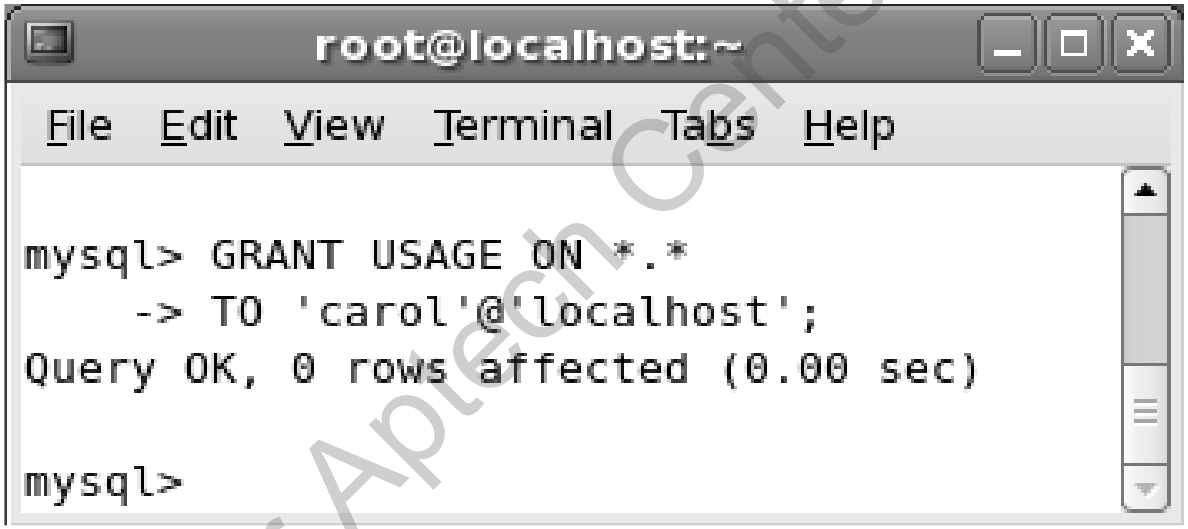
Figure displays the output of the command



- To grant the USAGE privilege to the user, enter the following command at the command prompt:

```
GRANT USAGE ON *.* TO 'carol'@'localhost';
```

Figure displays the output of the command

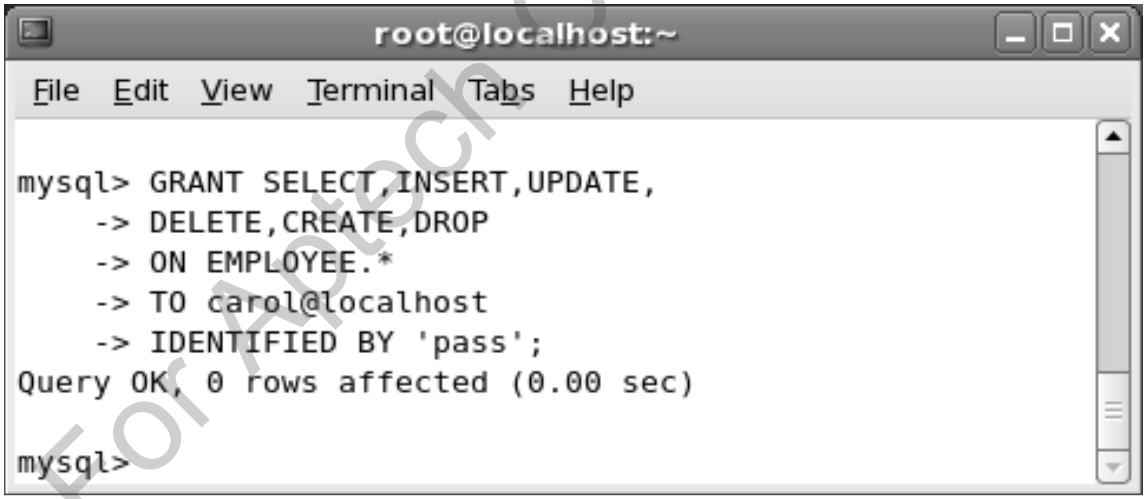


Altering Privileges

- To create a user who has access to the `EMPLOYEE` database from `localhost` only and with selective rights, enter the following command at the command prompt:

```
GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP ON
EMPLOYEE.* TO carol@localhost IDENTIFIED BY 'pass';
```

Figure displays the output of the command



- The syntax to grant access to a specific database is:

```
GRANT privilege ON {db_name.*} TO user_name [IDENTIFIED BY
[PASSWORD] 'password' [WITH GRANT OPTION]
```

where,

`GRANT` – specifies to assign privileges

`privilege` – specifies the type of permission to assign

`db_name` – specifies the name of the

`databaseuser_name` – specifies the name of the account

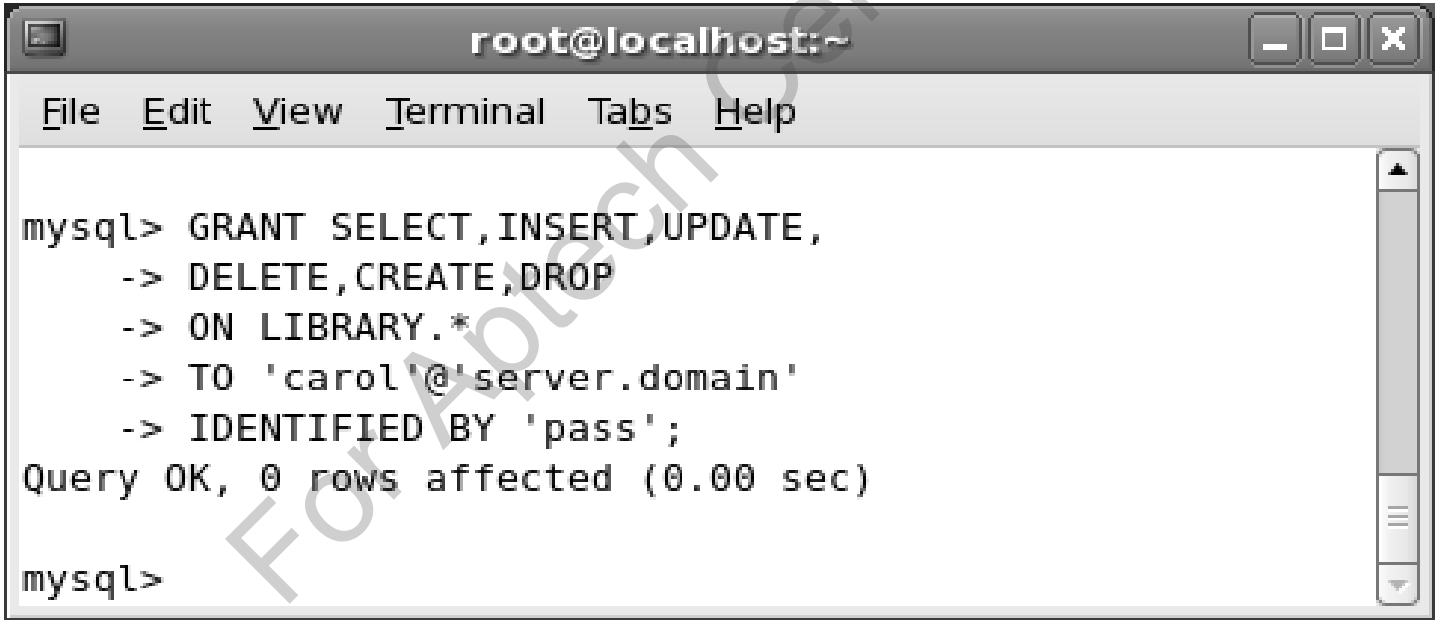
`IDENTIFIED BY [PASSWORD]` – sets a password for the user account

`[WITH GRANT OPTION]` – allows the account to set privileges

- To create a user who has access to the `LIBRARY` database from the `server.domain` host, enter the following command at the command prompt:

```
GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP ON
LIBRARY.* TO 'carol'@'server.domain' IDENTIFIED BY 'pass'
```

Figure displays the output of the command



Altering Privileges

- To give access to a specific user from any machine in a given domain, enter the following command at the command prompt:

```
GRANT ALL ON *.* TO 'carol'@'%.mydomain.com'
IDENTIFIED BY 'pass';
```

Figure displays the output of the command

Revoking Privileges

- MySQL provides the `REVOKE` command to remove rights or privileges from user accounts
- For example, to cancel or revoke the `INSERT` privilege for a user, enter the following command at the command prompt:

```
REVOKE INSERT ON *.* FROM 'temp_user' @ 'localhost';
```

- The command cancels the `INSERT` privilege assigned to `temp_user` on the `localhost` instance

Deleting User Account

- MySQL provides the `DROP USER` command to remove a user account
- You can also use this command to remove rows from the grant tables that contain privileges for user accounts
- To delete a user account, enter the following command at the command prompt:

```
DROP USER user_name;
```

where,

`DROP` – specifies to remove the object from the database

`USER` – specifies the type of object to be removed from the database

`user_name` – specifies the name of the account to be removed from the database

Deleting User Account

- You can also specify the domain to which the user account belongs while using the `DROP USER` command
- For example,

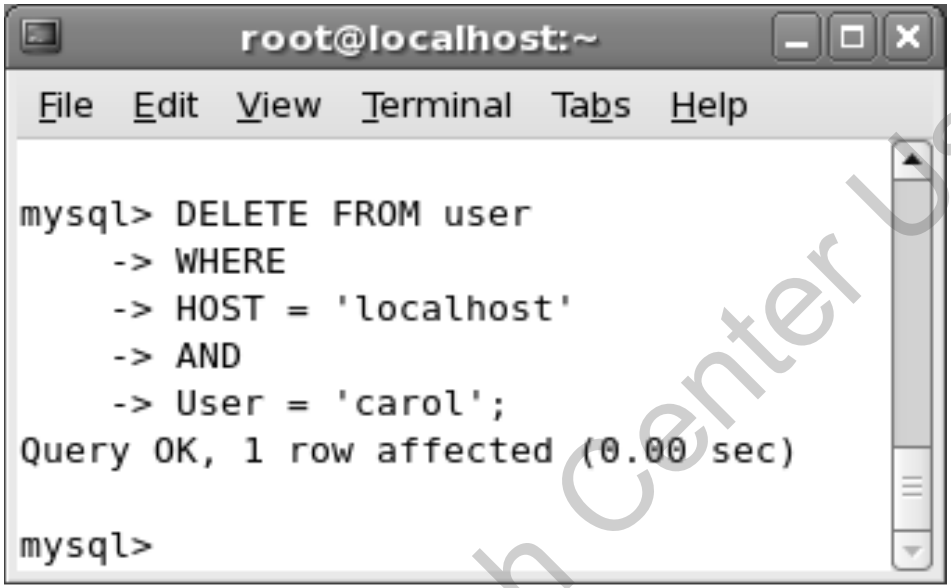
```
DROP USER 'carol'@'localhost';
```

- This command will delete the user account carol on the localhost instance of MySQL
- MySQL also enables you to remove a user account with the `DELETE` command
- To delete the user using the `DELETE` command, enter the following command at the command prompt:

```
DELETE FROM user WHERE Host='localhost' AND
User='carol';
```

Deleting User Account

Figure displays the output of the command



Setting Restrictions

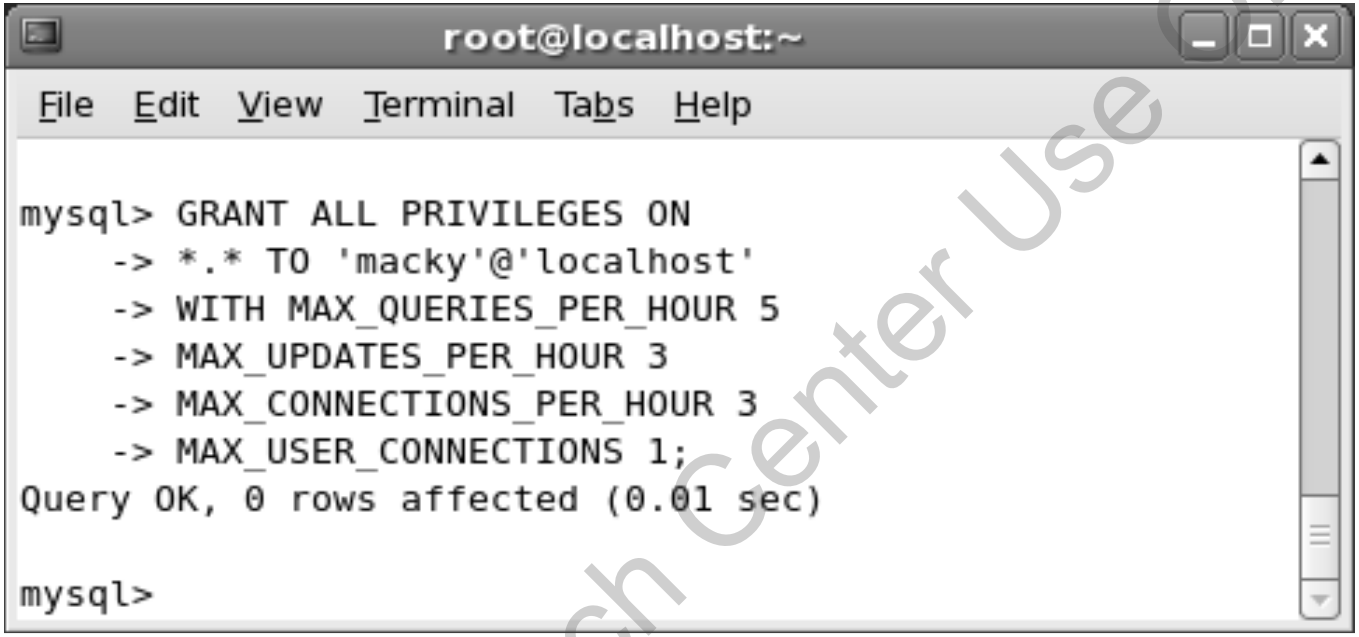
- You can ensure that single user doesnot take over the resources by setting a limit on the number of:
  - Queries per account per hour
  - Updates per account per hour
  - Connections per account per hour
  - Number of simultaneous connections per account
- You can set these restrictions using the `GRANT` command as shown:

```
GRANT … WITH MAX_QUERIES_PER_HOUR R1
            MAX_UPDATES_PER_HOUR R2
            MAX_CONNECTIONS_PER_HOUR R3
            MAX_USER_CONNECTIONS R4;
```

- For example, to grant all privileges to the user for all the databases but with resource restrictions, enter the following command at the command prompt:

```
GRANT ALL PRIVILEGES ON *.* TO 'macky'@'localhost'
WITH MAX_QUERIES_PER_HOUR 5
MAX_UPDATES_PER_HOUR 3
MAX_CONNECTIONS_PER_HOUR 3
MAX_USER_CONNECTIONS 1;
```

Figure displays the output of the command



Summary

- MySQL stores details of user accounts and privileges in the `columns_priv`, `db`, `host`, `tables_priv`, and user tables under the default database named mysql
- MySQL enables you to grant privileges to user accounts at four levels. They are global, database, table, and column level
- A new user account can be created either by using `GRANT` command or by using `INSERT` command
- The root user must have `INSERT` and `RELOAD` administrative privileges for creating user
- The `USAGE` option creates a user without any privileges. An account with just `USAGE` privilege can only be used to establish a connection

Summary

- The `GRANT OPTION` enables you to grant or revoke privileges to another user
- A super user has all the administrative rights and can access the server from anywhere
- MySQL provides the `GRANT` and `REVOKE` commands to assign and remove privileges to user accounts. To alter privileges for the user, GRANT command is used with selective privilege command
- You can execute administrative commands, such as mysql or mysqladmin to control a user's access to the database. For executing administrative commands, the user has to login as the root user
- You can ensure that no single user takes over the resources by setting a limit on the number of queries per account per hour, updates per account per hour, and so forth