Chapter 5

Data Control Language (DCL)

“Data is a precious thing and will last longer than the systems themselves.” - Tim Berners-Lee, inventor of the World Wide Web.

5.1. Data Control Language (DCL)

DCL command is a statement used to perform the work related to the rights, permissions, and other control of the database system.

The Need For DCL commands:

1. Unauthorized access to the data should be prevented to achieve security in our database
2. DCL commands maintain the database more effectively than anyone else other than the database administrator is not allowed to access the data without permission.
3. These commands allow the data administrator to set and remove database permissions in a granular fashion.

The two most important DCL commands are:

1. GRANT

   This command is used to grant permission to the user to perform a particular operation on a particular object. If you are a database administrator and want to restrict user access, such as one who only views the data or may only update the data. You can give the privilege permission to the users according to your wish.

   Syntax:

   ```
   GRANT privilege_list
   ON Object_name
   TO user_name;
   ```

2. REVOKE

   This command is used to take permission/access back from the user. If you want to return permission from the database you have granted to the users at that time, you need to run the REVOKE command.

   Syntax:

   ```
   REVOKE privilege_list
   ON object_name
   FROM user_name;
   ```

Privileges list:

PRIVILEGE	DESCRIPTION
SELECT	Select statement on the tables
INSERT	Insert statement on the tables
DELETE	Delete statements on the tables
INDEX	Create an index on the existing table
CREATE	Create table statement
ALTER	Ability to perform ALTER TABLE to change the table definition
DROP	Drop table statement
ALL	Grant all permissions
UPDATE	Update the statement on the table
GRANT	Allows to grant the privilege

The difference between GRANT and REVOKE are:

GRANT	REVOKE
This DCL command grants permissions to the user on the database objects.	This DCL command removes permissions if any, granted to the users on database objects.
It assigns access rights to users.	It revokes the user access rights of users.
For each user, you need to specify the permissions.	If access for one user is removed, all the particular permissions provided by that user to others will be removed.
When the access is decentralized granting permissions will be easy.	If decentralized access removes the granted permissions is difficult.

5.2. TYPES OF PRIVILEGE BASED ON LEVEL COVERAGE

MySQL provides various levels of privilege. Each user can be restricted from being able to access either a certain database, certain tables, or even only certain columns. Based on this grouping, we can divide MySQL access rights into 4 levels, namely:

1. Global privilege (***.***)
   This permission means the user can have access for all databases in MySQL.
   Syntax:
   ```
   GRANT SELECT ON *.* TO 'user'@'localhost';
   ```

2. Database level privilege (**database_name.***)
   This permission means that the user has full access to a database.
   Syntax:
   ```
   GRANT SELECT ON nama_database.* TO 'user'@'localhost';
   ```

3. Table Level Privilege (database_name.table_name)
   This privilege means that the user has access to a table that is in a database. The user's access is only limited to the level of a table.
   Syntax:

   GRANT SELECT ON nama_database.nama_tabel TO 'user'@'localhost';

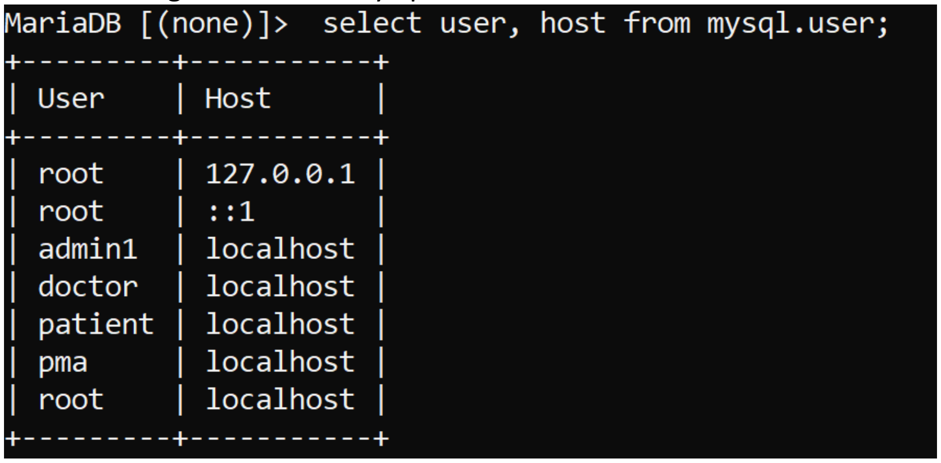
4. Column Level Privilege (column_name)
   This permission is the smallest permission that can be granted to a user. With column-level permissions, the user only has access rights to certain columns in a table.
   Syntax:

   GRANT SELECT (column1, column2) ON database_name.table_name TO 'user'@'localhost';

<u>Exercise:</u>
CREATE USER 'nama_user@'localhost;

a. admin1@localhost
b. doctor@localhost
c. patient@localhost

Show existing users in the mysql database



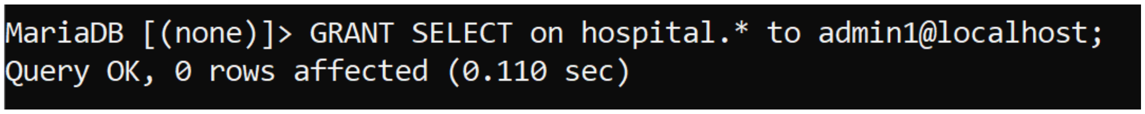
```
MariaDB [(none)]>  select user, host from mysql.user;
+---------+-----------+
| User    | Host      |
+---------+-----------+
| root    | 127.0.0.1 |
| root    | ::1       |
| admin1  | localhost |
| doctor  | localhost |
| patient | localhost |
| pma     | localhost |
| root    | localhost |
+---------+-----------+
```

5.3. Give privilege to users with the GRANT command

Grant is used to grant privileges to defined tables to other users. Privileges for users in the grant command are defined using the privilege names. Privilege names make it easier for administrators to grant privileges without knowing what field and table names must be filled in.
Syntax:

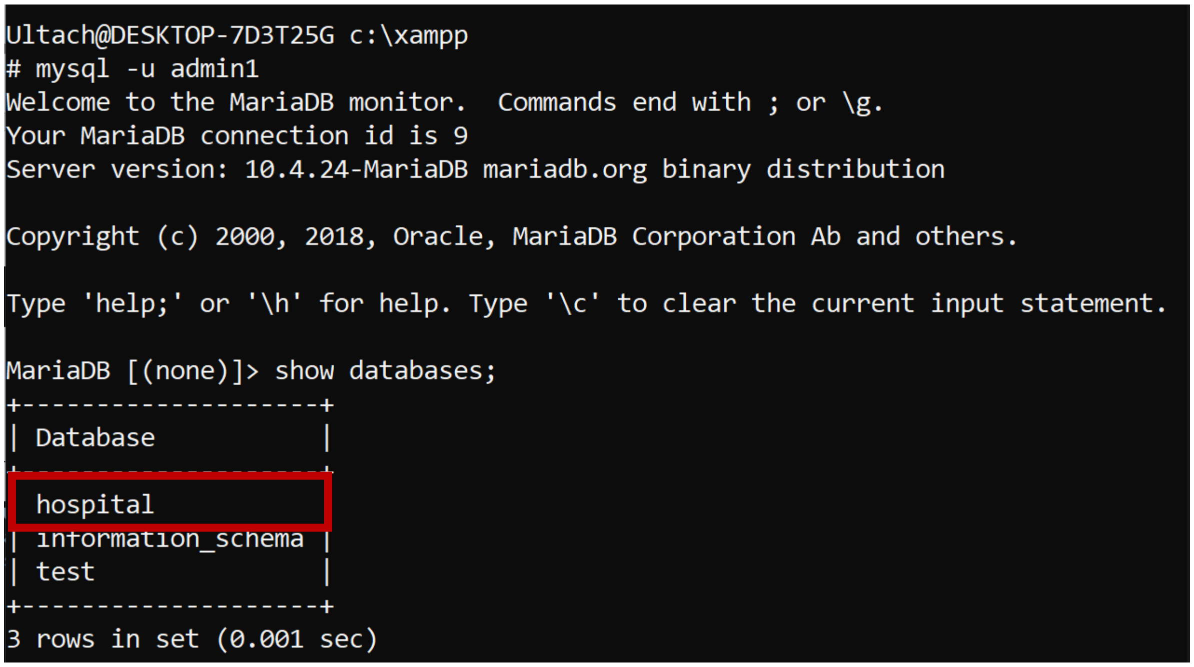
GRANT privilege_names ON database_name.table_name TO 'user_name'@'user_location';

Example:



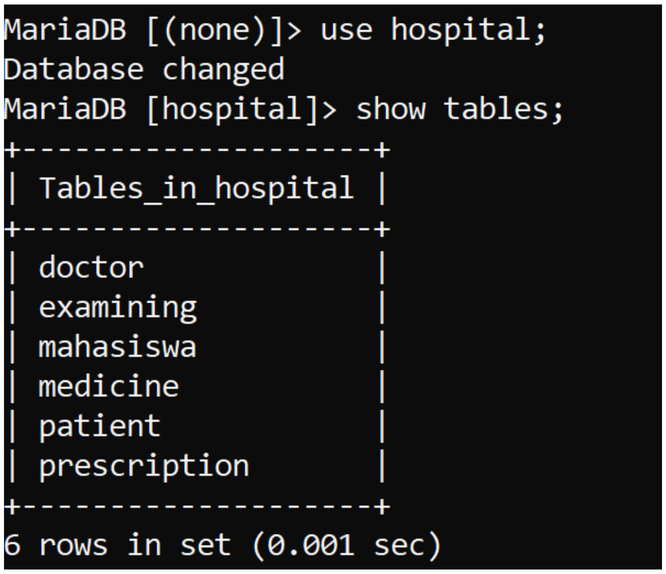
```
MariaDB [(none)]> GRANT SELECT on hospital.* to admin1@localhost;
Query OK, 0 rows affected (0.110 sec)
```

Then login as admin1 and display the database.



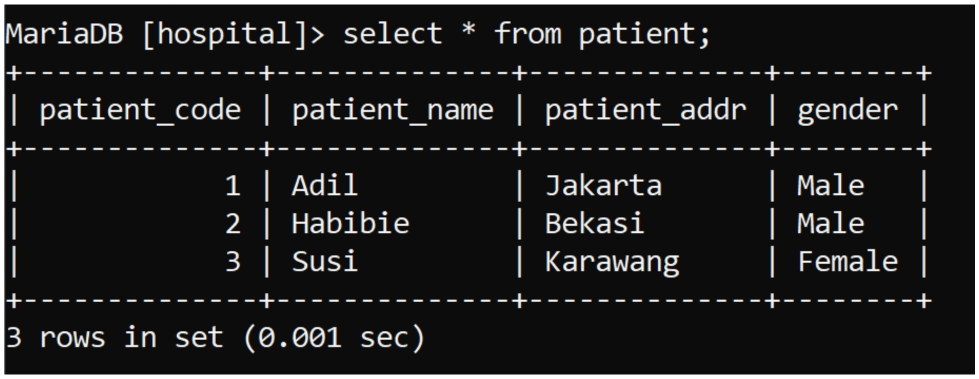
```
Ultach@DESKTOP-7D3T25G c:\xampp
# mysql -u admin1
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 9
Server version: 10.4.24-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| hospital           |
| information_schema |
| test               |
+--------------------+
3 rows in set (0.001 sec)
```

Use the hospital database and then display the tables in the database.



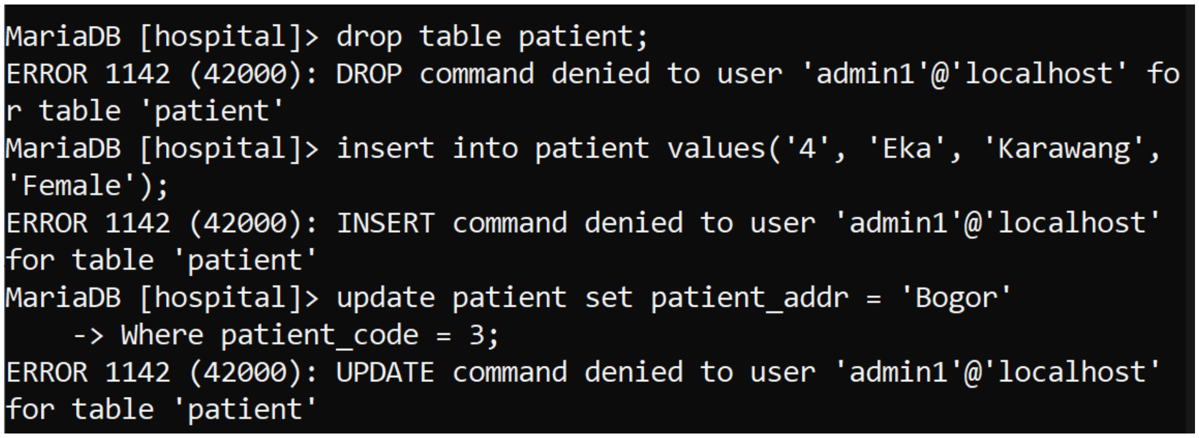
```
MariaDB [(none)]> use hospital;
Database changed
MariaDB [hospital]> show tables;
+--------------------+
| Tables_in_hospital |
+--------------------+
| doctor             |
| examining          |
| mahasiswa          |
| medicine           |
| patient            |
| prescription       |
+--------------------+
6 rows in set (0.001 sec)
```

When we use select command for the patient table:



```
MariaDB [hospital]> select * from patient;
+--------------+--------------+--------------+--------+
| patient_code | patient_name | patient_addr | gender |
+--------------+--------------+--------------+--------+
|            1 | Adil         | Jakarta      | Male   |
|            2 | Habibie      | Bekasi       | Male   |
|            3 | Susi         | Karawang     | Female |
+--------------+--------------+--------------+--------+
3 rows in set (0.001 sec)
```

But when we use these commands, some errors happen. Why did it happen? Discuss it!

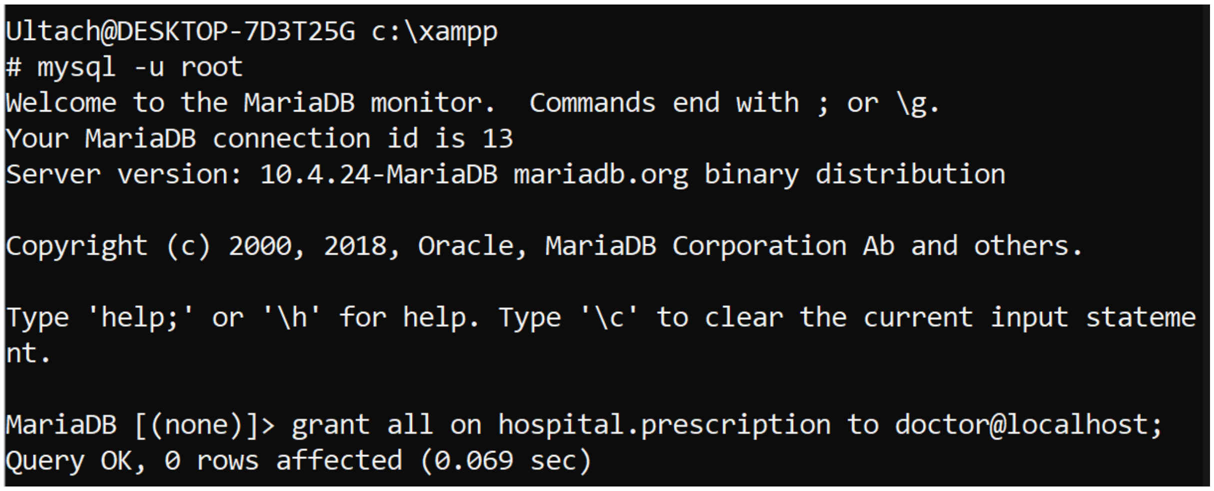


```
MariaDB [hospital]> drop table patient;
ERROR 1142 (42000): DROP command denied to user 'admin1'@'localhost' fo
r table 'patient'
MariaDB [hospital]> insert into patient values('4', 'Eka', 'Karawang',
'Female');
ERROR 1142 (42000): INSERT command denied to user 'admin1'@'localhost'
for table 'patient'
MariaDB [hospital]> update patient set patient_addr = 'Bogor'
    -> Where patient_code = 3;
ERROR 1142 (42000): UPDATE command denied to user 'admin1'@'localhost'
for table 'patient'
```

Granting All privilege (GRANT ALL)

GRANT ALL is a shorthand way of giving almost all privileges to a particular user. These permissions cover all basic queries.

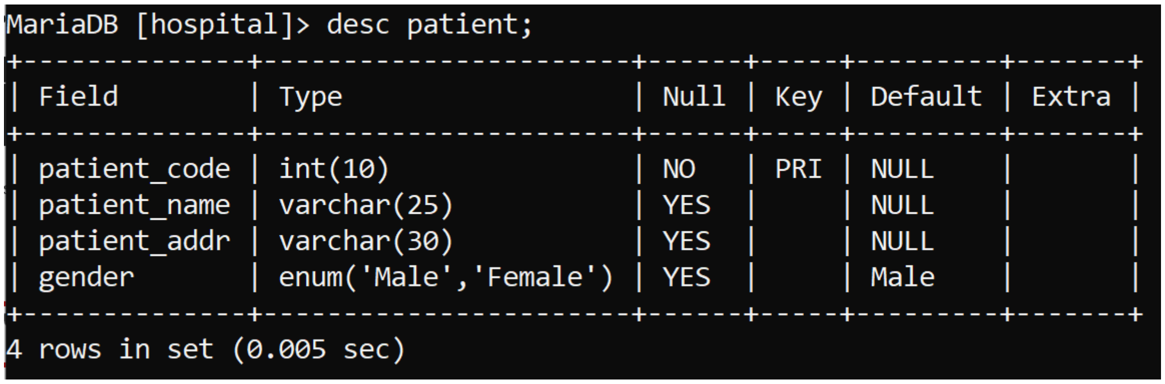
Doctors are given full access rights and can manipulate data in the prescription table.



```
Ultach@DESKTOP-7D3T25G c:\xampp
# mysql -u root
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 13
Server version: 10.4.24-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input stateme
nt.

MariaDB [(none)]> grant all on hospital.prescription to doctor@localhost;
Query OK, 0 rows affected (0.069 sec)
```

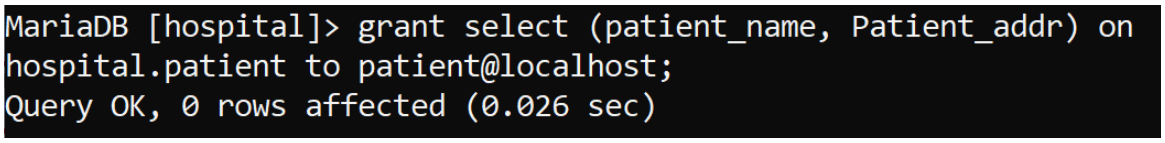
By granting GRANT ALL privileges, the users' doctor can use all basic queries on the prescription table, such as SELECT, UPDATE, and even DELETE. As an exercise, please try to log in as a doctor and perform commands such as UPDATE, DELETE, and DROP.

Give MySQL privilege at Column Level



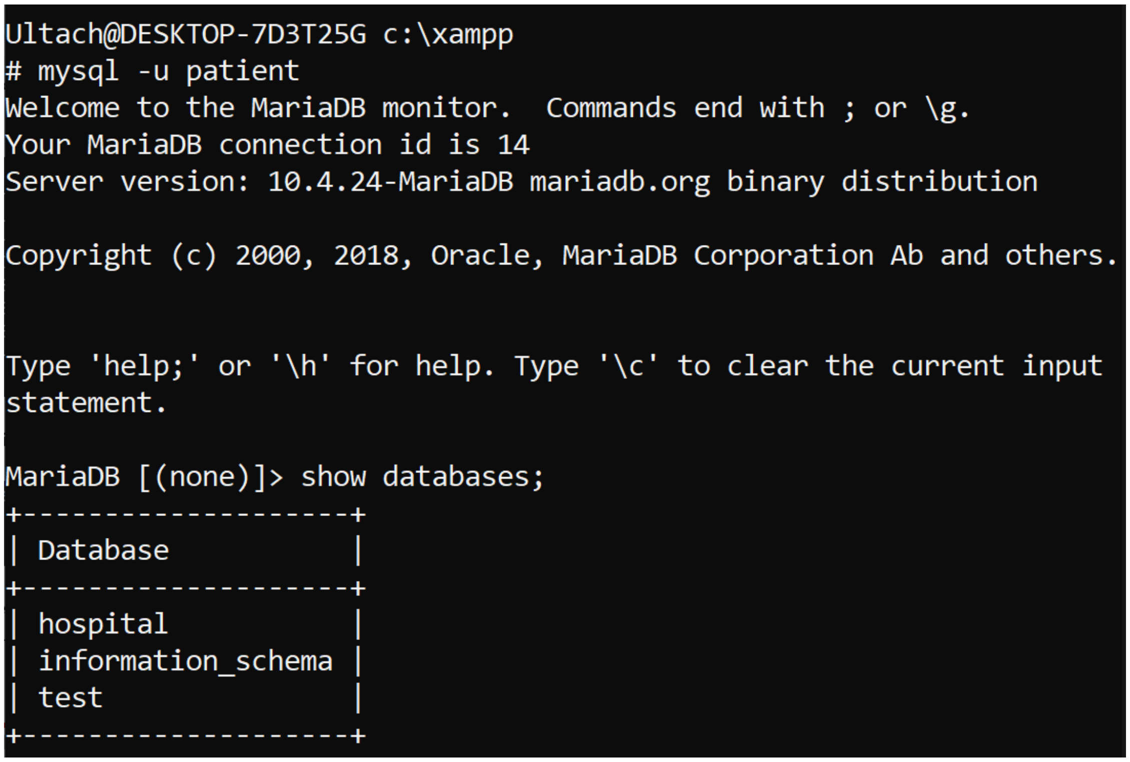
```
MariaDB [hospital]> desc patient;
+--------------+----------------------+------+-----+---------+-------+
| Field        | Type                 | Null | Key | Default | Extra |
+--------------+----------------------+------+-----+---------+-------+
| patient_code | int(10)              | NO   | PRI | NULL    |       |
| patient_name | varchar(25)          | YES  |     | NULL    |       |
| patient_addr | varchar(30)          | YES  |     | NULL    |       |
| gender       | enum('Male','Female') | YES  |     | Male    |       |
+--------------+----------------------+------+-----+---------+-------+
4 rows in set (0.005 sec)
```

Patients are given access to view the patient table with the columns patient_name and patient_addr.



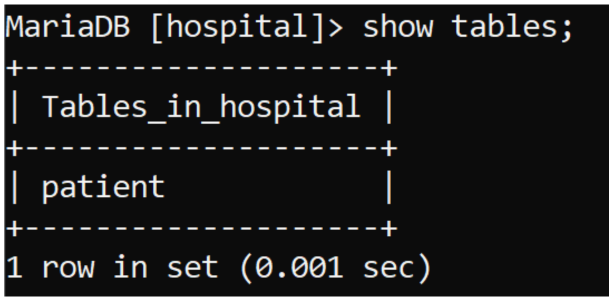
```
MariaDB [hospital]> grant select (patient_name, Patient_addr) on
hospital.patient to patient@localhost;
Query OK, 0 rows affected (0.026 sec)
```

Login as patient



```
Ultach@DESKTOP-7D3T25G c:\xampp
# mysql -u patient
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 14
Server version: 10.4.24-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.


Type 'help;' or '\h' for help. Type '\c' to clear the current input
statement.

MariaDB [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| hospital           |
| information_schema |
| test               |
+--------------------+
```

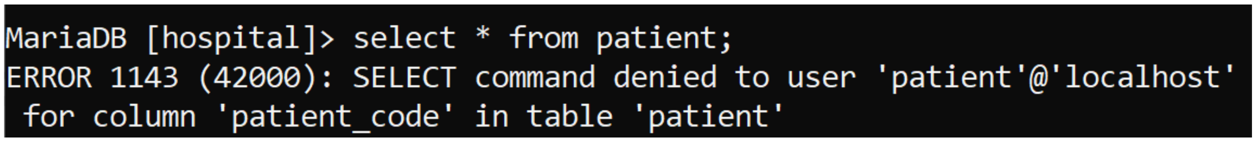
Display the table that patient can access:



```
MariaDB [hospital]> show tables;
+--------------------+
| Tables_in_hospital |
+--------------------+
| patient            |
+--------------------+
1 row in set (0.001 sec)
```

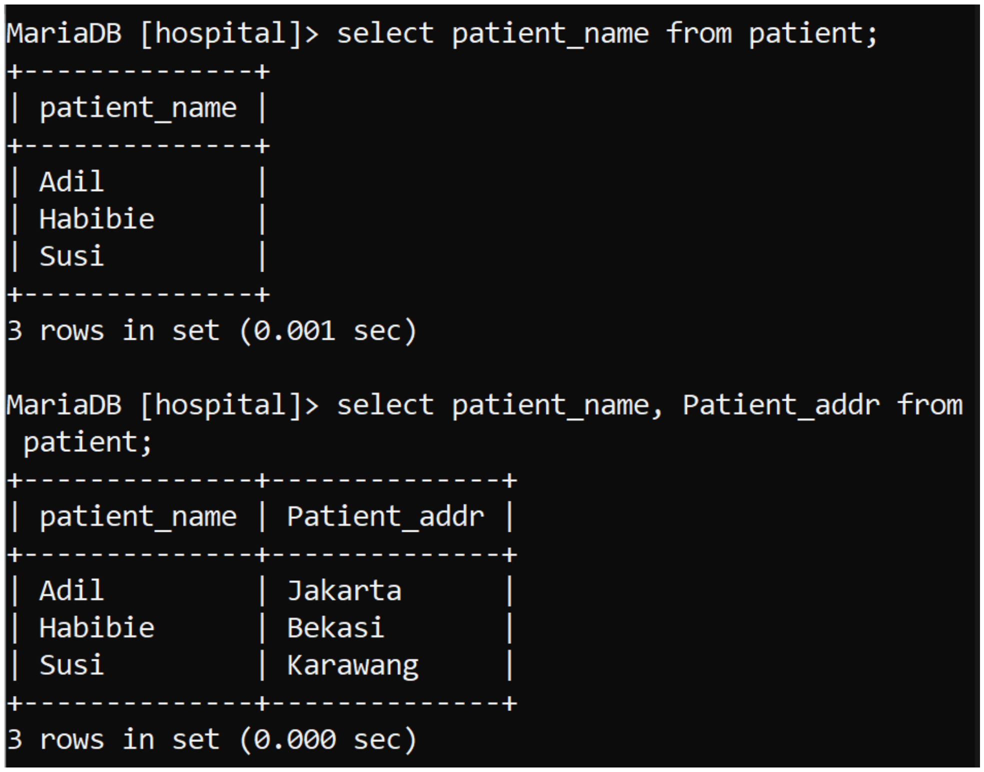
Why is it only table patient is being displayed? Discuss it!

Why is there an error when we use the select * command?



```
MariaDB [hospital]> select * from patient;
ERROR 1143 (42000): SELECT command denied to user 'patient'@'localhost'
 for column 'patient_code' in table 'patient'
```

But why is there no error when we use these commands below? Discuss it!



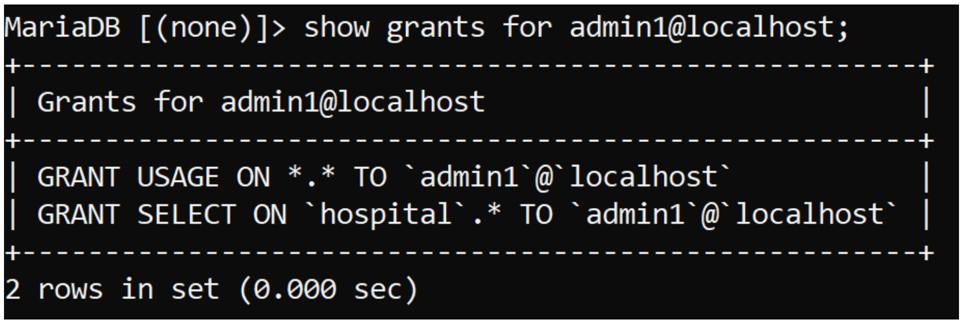
```
MariaDB [hospital]> select patient_name from patient;
+--------------+
| patient_name |
+--------------+
| Adil         |
| Habibie      |
| Susi         |
+--------------+
3 rows in set (0.001 sec)

MariaDB [hospital]> select patient_name, Patient_addr from
 patient;
+--------------+--------------+
| patient_name | Patient_addr |
+--------------+--------------+
| Adil         | Jakarta      |
| Habibie      | Bekasi       |
| Susi         | Karawang     |
+--------------+--------------+
3 rows in set (0.000 sec)
```

5.4. View MySQL User Privileges (SHOW GRANTS FOR)

The UPDATE statement in SQL is used to update the data of an existing table in the database. We can update single columns as well as multiple columns using the UPDATE statement as per our requirement.

Syntax:

SHOW GRANTS FOR user_name@user_location;



```
MariaDB [(none)]> show grants for admin1@localhost;
+-----------------------------------------------------+
| Grants for admin1@localhost                         |
+-----------------------------------------------------+
| GRANT USAGE ON *.* TO `admin1`@`localhost`          |
| GRANT SELECT ON `hospital`.* TO `admin1`@`localhost` |
+-----------------------------------------------------+
2 rows in set (0.000 sec)
```

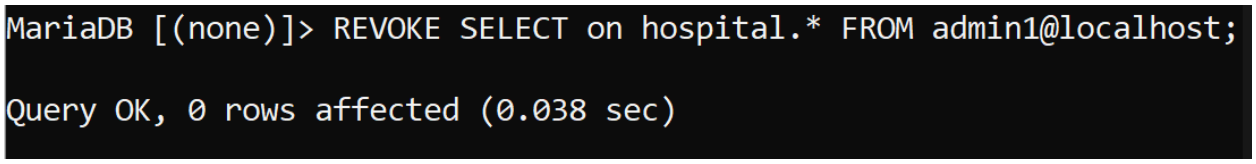
5.5. REVOKE command

The privilege granted to a user sometimes needs to be changed depending on the conditions and policies of the user. To delete a user, we can use the DROP user query, but sometimes we need to just remove the privilege without deleting the user. For this matter, MySQL provides the REVOKE command.

Syntax:

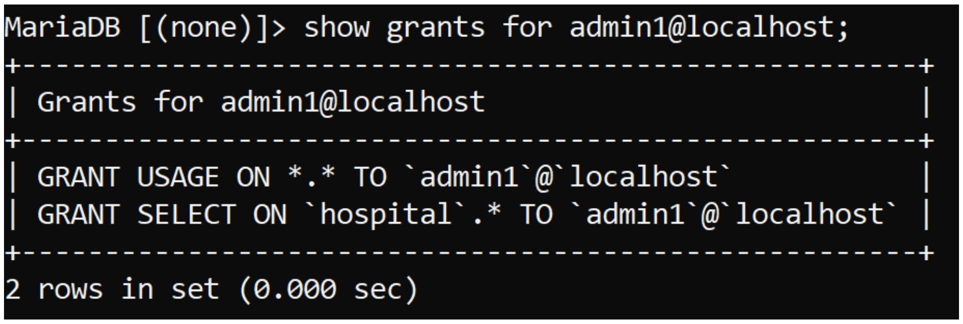
REVOKE access_type(column1, column2) ON database_name.table_name FROM user_name@user_location;

For example, we want to delete the privilege (SELECT) from admin1:



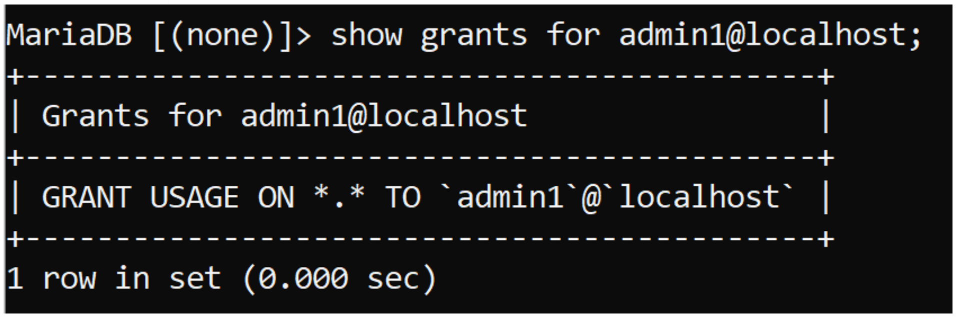
```
MariaDB [(none)]> REVOKE SELECT on hospital.* FROM admin1@localhost;

Query OK, 0 rows affected (0.038 sec)
```

Before revoke



```
MariaDB [(none)]> show grants for admin1@localhost;
+-----------------------------------------------------+
| Grants for admin1@localhost                         |
+-----------------------------------------------------+
| GRANT USAGE ON *.* TO `admin1`@`localhost`          |
| GRANT SELECT ON `hospital`.* TO `admin1`@`localhost` |
+-----------------------------------------------------+
2 rows in set (0.000 sec)
```

After revoke



```
MariaDB [(none)]> show grants for admin1@localhost;
+-----------------------------------------+
| Grants for admin1@localhost             |
+-----------------------------------------+
| GRANT USAGE ON *.* TO `admin1`@`localhost` |
+-----------------------------------------+
1 row in set (0.000 sec)
```

<u>TASK 5:</u>

1. GRANT
    - admin1 (Update and select all the tables)
    - doctor:
        i. update on the examining table
        ii. select patient name and gender on the patient table
        iii. delete on the doctor table
    - patient
        i. select on patient table
        ii. update on patient address in patient table
2. REVOKE
    - revoke doctor privilege