

# Query Gate Write-up

---

## Database

Database is a system for storing, managing and accessing data in an organized and efficient way.

Databases provide fast and secure access to data by storing data in an organized way. Data is kept in the form of tables, columns and rows. Thanks to this method, we can access data quickly and securely.

## Database Functions

Data Storage: Stores data in a secure and organized manner.

Data Querying: Provides fast and efficient access to data.

Data Updating: Provides the capability to update data.

Security and Access Control: Prevents unauthorized access and ensures data security.

## SQL

SQL (Structured Query Language) is a language used to manage and query data in databases. These operations include inserting, updating, deleting and retrieving data from the database.

SQL is a way of interacting with the database. Learning the SQL language and being able to write SQL queries is important for understanding and using databases.

## Database Types

Relational Databases (SQL): It stores data in tables. Each table consists of columns and rows. Some common SQL databases: MySQL, PostgreSQL.

Document Based Databases (NoSQL): It stores data in document form using structures such as JSON, XML. Some common NoSQL databases: MongoDB, CouchDB.

Key-Value Databases: Stores data using simple key-value pairs. Provides fast access to data. Some common key-value databases: Redis, DynamoDB.

## MySQL

MySQL is a popular relational database management system (RDBMS) widely used worldwide. It is known for being open source, having a large community of users and developers, and being available on various operating systems.

MySQL uses Structured Query Language (SQL) for data manipulation and querying. SQL, also used in MySQL, is a standard language for interacting with databases.

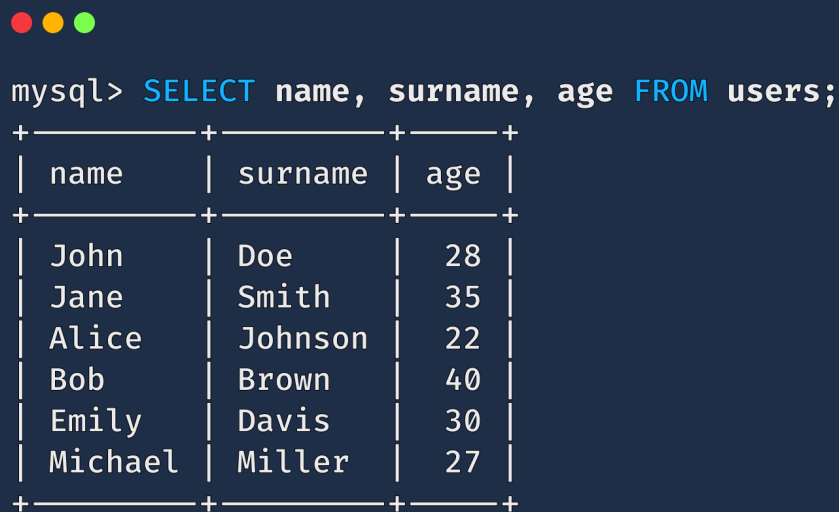
Below are examples of SQL commands used in MySQL for various database and table operations.

### SELECT

Used to select and display specific data from the database.

```
SELECT name, surname, age FROM users;
```

This query selects the "name", "surname", "age" columns from the "users" table and retrieves the data in them.

A terminal window with a dark blue background and three colored window control buttons (red, yellow, green) in the top left corner. The prompt 'mysql>' is followed by the command 'SELECT name, surname, age FROM users;'. Below the command, the output is displayed as a table with three columns: 'name', 'surname', and 'age'. The data rows are: John Doe 28, Jane Smith 35, Alice Johnson 22, Bob Brown 40, Emily Davis 30, and Michael Miller 27.

```
mysql> SELECT name, surname, age FROM users;
```

name	surname	age
John	Doe	28
Jane	Smith	35
Alice	Johnson	22
Bob	Brown	40
Emily	Davis	30
Michael	Miller	27

## INSERT

Used to insert new data into the table.

```
INSERT INTO users (name, surname, age) VALUES ('Lynn', 'Spence', 37);
```

This query creates a new row in the "users" table and adds "Lynn", "Spence" and 37 to the "name", "surname" and "age" columns respectively.

```
mysql> INSERT INTO users (name, surname, age) VALUES ('Lynn', 'Spence', 37);  
Query OK, 1 row affected (0.01 sec)
```

## UPDATE

It is used to update data in the table.

```
UPDATE users SET age = 34 WHERE name = 'Lynn';
```

This query updates the "age" information to 34 for the records in the "users" table where the "name" column contains "Lynn".

```
mysql> UPDATE users SET age = 34 WHERE name = 'Lynn';  
Query OK, 1 row affected (0.01 sec)  
Rows matched: 1 Changed: 1 Warnings: 0
```

## DELETE

It is used to delete data in the table.

```
DELETE FROM users WHERE age > 60;
```

This query deletes the records in the "users" table if the data in the "age" column is greater than 60.

```
mysql> DELETE FROM users WHERE age > 60;  
Query OK, 24 rows affected (0.02 sec)
```

## List Databases

A SQL server can have many different databases. We can run the following command to list these databases.

```
SHOW DATABASES;
```

## Select Database

We can run the following command to select the database we will be working on.

```
USE <database-name>;
```

## Delete Database

To delete a database we can execute the following command.

```
DROP DATABASE <database-name>;
```

## List Tables

There can be many different tables in a database. We can run the following command to list the tables in the selected database.

```
SHOW TABLES;
```

## Delete Table

To delete a table we can execute the following command.

```
DROP TABLE <table-name>;
```

## Display Table Information

To display the columns and data types of a table we can execute the following command.

```
DESCRIBE <table-name>;
```

## Information Gathering

Let's gather information by running a port scan on our target machine.

### Task 1, Task 2

```
root@hackerbox:~# nmap 172.20.7.45
Starting Nmap 7.80 ( https://nmap.org ) at 2023-11-19 06:45 CST
Nmap scan report for 172.20.7.45
Host is up (0.00037s latency).
Not shown: 999 closed ports
PORT      STATE SERVICE
3306/tcp  open  mysql
MAC Address: 52:54:00:DB:AA:ED (QEMU virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 13.29 seconds
```

## System Access

Let's try connecting to the target MySQL server.

### Task 3, Task 4

The task asks for the most authorized user, so **root** should come directly to mind. We will try to connect to the target MySQL server with this username.

The command string we will use to connect to MySQL is as follows.

```
mysql -u root -h <target>
```

-u : Used to specify which user to connect as when connecting to the target machine.

-h : Parameter used to specify the IP address or hostname of the target machine.

-P : Parameter used to specify the port number.

*Note: If MySQL uses port 3306 by default, we do not need to specify a port number when connecting.*

```
root@hackerbox:~# mysql -u root -h 172.20.7.45
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 8.0.34 MySQL Community Server - GPL

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

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

MySQL [(none)]>
```

Yes, as we expected, we were able to access the MySQL command line by connecting to our target machine with the **root** user.

Normally, when connecting to MySQL, you connect with username and password. Unless this configuration was done intentionally, we can say that there is a misconfiguration on the target machine.

### Task 5

To see how many databases there are, let's run the **SHOW DATABASES;** command from the MySQL command line we are connected to.

```
MySQL [(none)]> SHOW DATABASES;
+-----+
| Database |
+-----+
| detective_inspector |
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0.011 sec)
```

We saw that the MySQL server we connected to has **5** databases.

## Task 6, Task 7

To view the tables in the **detective\_inspector** database that attracted our attention, we first select this database using the **USE** command. Then we will run the **SHOW TABLES;** command to list the tables.

```
MySQL [(none)]> USE detective_inspector;
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 [detective_inspector]> SHOW TABLES;
+-----+
| Tables_in_detective_inspector |
+-----+
| hacker_list                    |
+-----+
1 row in set (0.004 sec)
```

We discovered that there is a table named **hacker\_list** in the database we are targeting.

## Task 8

Let's look at the data in the table to identify the white hat hacker.

```
MySQL [(none)]> SELECT * FROM hacker_list;
+----+-----+-----+-----+-----+
| id  | firstName | lastName | nickname | type |
+----+-----+-----+-----+-----+
| 1001 | Jed       | Meadows | sp1d3r   | gray-hat |
| 1002 | Melissa  | Gamble  | c0c0net  | gray-hat |
| 1003 | Frank    | Netsi   | v3nus    | gray-hat |
| 1004 | Nancy    | Melton  | s1torml09 | black-hat |
| 1005 | Jack     | Dunn    | psyod3d  | black-hat |
| 1006 | Arron    | Eden    | r4nd0myfff | black-hat |
| 1007 | Lea      | Wells   | pumq7eggy7 | black-hat |
| 1008 | Hackviser | Hackviser | h4ckv1s3r | white-hat |
| 1009 | Xavier   | Klein   | oricy4l33 | black-hat |
+----+-----+-----+-----+-----+
9 rows in set (0.005 sec)
```

👉 We succeeded in accessing the information of the white hat hacker.

Congratulations 🍾🍾

✨ You have successfully completed all tasks in this warmup.