Apply filters to SQL queries

Project description

My organization wants to make sure that the system is safe. I have been tasked to investigate some security concerns, ensure that the system is safe and to update some devices when necessary. The following is what I do to ensure that the system is safe using some SQL commands.

Retrieve after hours failed login attempts

The organization closes at 1800 hours. A security incident that happened after hours needs to be investigated. This code demonstrates how I use SQL commands to retrieve all failed login attempts that took place after hours.
I select all columns (* selects all) from the log_in_attempts table in our database.
I place my condition which is initiated by WHERE and mention the column of interest, login_time.
I then use the operator >(greater than) followed by closing time to return all login attempts that happened after 1800 hours.
Finally, I use AND to indicate another condition that must be true, which is that the attempt to log in failed. 0 in success = 0 represents failed attempts and the boolean value of FALSE in mysql is 0.

```
MariaDB [organization]> clear
MariaDB [organization]> SELECT *
    -> FROM log_in_attempts
    -> WHERE login_time > '18:00' AND success = 0;
+----------+----------+------------+------------+---------+--------------
--+---------+
| event_id | username | login_date | login_time | country | ip_address
  | success |
+----------+----------+------------+------------+---------+--------------
```

Retrieve login attempts on specific dates

After a suspicious event occurred on 2022-05-09, there was a need to investigate the login attempts that happened on the day as well as the previous day 2022-05-08.
In this case I used OR to filter so that the system could return login attempts on both dates. There are two conditions there and at least one of them has to be met and at most both of

them are met. The first part is that the login date is 2022-05-09 and the second part is that the login date is 2022-05-08.

```
MariaDB [organization]> SELECT *
    -> FROM log_in_attempts
    -> WHERE login_date = '2022-05-09' OR login_date = '2022-05-08';
+----------+----------+------------+------------+---------+--------------
--+----------+
| event_id | username | login_date | login_time | country | ip_address
  | success |
+----------+----------+------------+------------+---------+--------------
```

Retrieve login attempts outside of Mexico

I then discovered that there was another threat to the login attempts and this happened outside of Mexico. My investigation had to exclude Mexico.
I wrote a SQL command that leaves out Mexico when it returns all the log_in_attempts table.

In this case I wrote 'Mex%' as there are instances where a login attempt in Mexico is listed as Mex instead of Mexico. % substitutes for any number of characters that come after Mex, therefore even if the country is listed as Mex or Mexi or Mexic or Mexico, it would be returned. When using %, you also use LIKE in order to detect patterns.

```
MariaDB [organization]> SELECT *
    -> FROM log_in_attempts
    -> WHERE NOT country LIKE 'Mex%';
+----------+----------+------------+------------+---------+--------------
--+----------+
| event_id | username | login_date | login_time | country | ip_address
  | success |
+----------+----------+------------+------------+---------+--------------
```

Retrieve employees in Marketing

My team needs to update computers in the Marketing department, specifically those in the East Offices. Now there are different East offices, and they are listed with numeric value that follows East-.
I had to get details on these devices from the employees table.
I used the AND condition because two conditions have to be met when the data is returned. The first condition is that the department is Marketing and the second condition is that the office follows a pattern where it starts with East and is followed by any other characters.

```
MariaDB [organization]> SELECT *
    -> FROM employees
    -> WHERE department = 'Marketing' AND office LIKE 'East%';
+-------------+-------------+----------+------------+----------+
| employee_id | device_id   | username | department | office   |
+-------------+-------------+----------+------------+----------+
```

Retrieve employees in Finance or Sales

Devices from the Finance and Sales department now also need to be updated and I have to get information on all the employees in both departments.
The following code selects data where either the department is Finance or Sales or both and that is returned.
OR means that either condition can be met.

```
MariaDB [organization]> SELECT *
    -> FROM employees
    -> WHERE department = 'Finance' OR department = 'Sales';
+-------------+-------------+----------+------------+-------------+
| employee_id | device_id   | username | department | office      |
+-------------+-------------+----------+------------+-------------+
```

Retrieve all employees not in IT

All machines have to be updated except for those in the IT department as they have been updated already.
In this case I will have to get all information from the employees table except for the IT department.
In order to exclude the IT department, I used the NOT keyword. This negates the condition that the department is Information Technology.

```
MariaDB [organization]> SELECT *
    -> FROM employees
    -> WHERE NOT department = 'Information Technology';
```

Summary

I used two different tables from a database, log_in_attempts and employees to filter and return only the information that was relevant to the scenario. I used AND, NOT and OR as my filters. I also used LIKE with % to investigate patterns instead of known values. Then I also used some Operators like = and > for date and time values.