# Apply filters to SQL queries

## Project description

I am a security professional at a large organization. Part of my job is to investigate security issues to help keep the system secure. I recently discovered some potential security issues that involve login attempts and employee machines.

My task is to examine the organization’s data in their employees and log\_in\_attempts tables. I willl use SQL filters to retrieve records from different datasets and investigate the potential security issues.

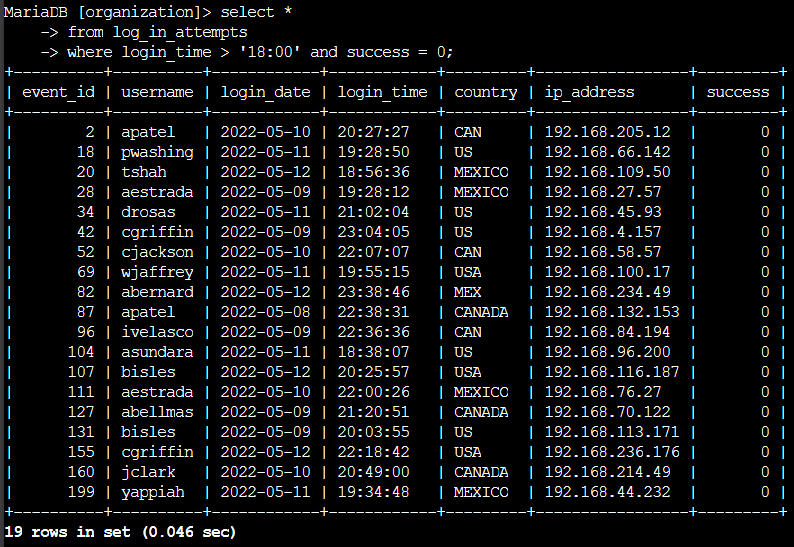
## Retrieve after hours failed login attempts

My team is investigating failed login attempts that were made after business hours. You want to retrieve this information from the login activity. I’ll identified all unsuccessful attempts after 18:00. The login\_time column in the log\_in\_attempts table contains information on when login attempts were made. Office hours end at '18:00'.

The first part of the screenshot is my query, and the second part is a portion of the output. This query filters for failed login attempts that occurred after 18:00. First, I started by selecting all data from the log\_in\_attempts table. Then, I used a WHERE clause with an AND operator to filter my results to output only login attempts that occurred after 18:00 and were unsuccessful. The first condition is login\_time > '18:00', which filters for the login attempts that occurred after 18:00. The second condition is success = 0, which filters for the failed login attempts.

I use the following commands to retrieve the information

SELECT \* FROM log\_in\_attempts WHERE login\_time > '18:00' AND success = 0;



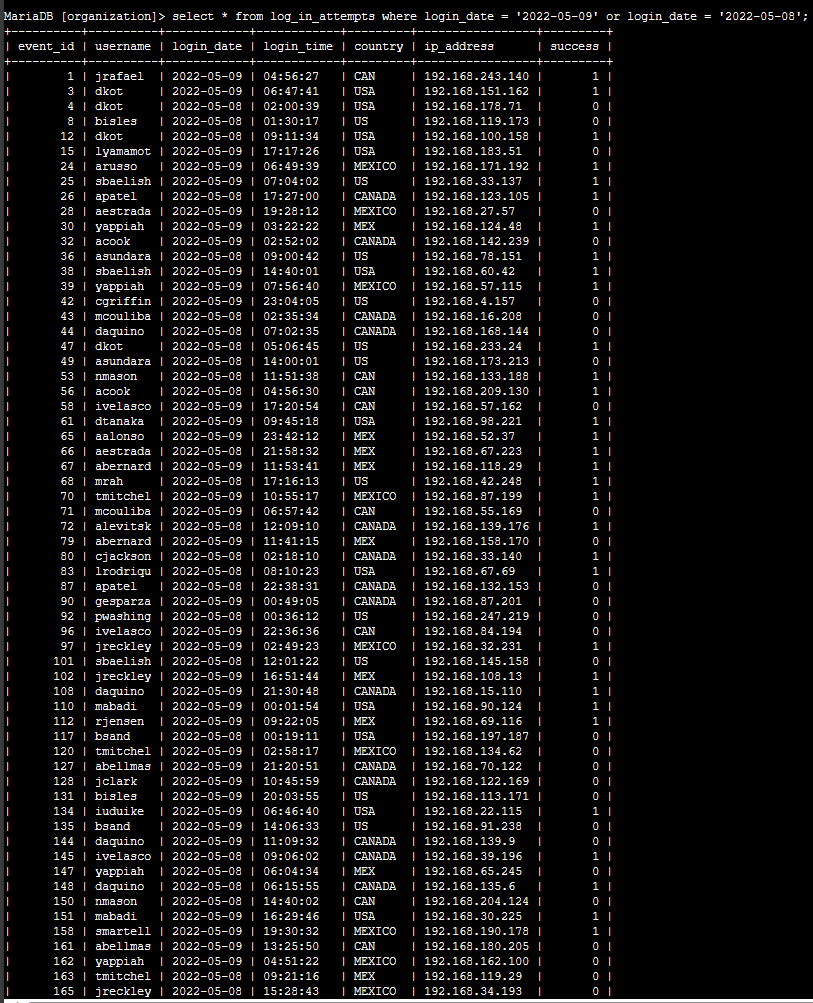
## Retrieve login attempts on specific dates

My team is investigating a suspicious event that occurred on '2022-05-09'. You want to retrieve all login attempts that occurred on this day and the day before ('2022-05-08').

The login\_date column in the log\_in\_attempts table contains information on the dates when login attempts were made. I use the OR operator to retrieve the failed login attempts on the specified days.

The first part of the screenshot is my query, and the second part is a portion of the output. This query returns all login attempts that occurred on 2022-05-09 or 2022-05-08. First, I started by selecting all data from the log\_in\_attempts table. Then, I used a WHERE clause with an OR operator to filter my results to output only login attempts that occurred on either 2022-05-09 or 2022-05-08. The first condition is login\_date = '2022-05-09', which filters for logins on 2022-05-09. The second condition is login\_date = '2022-05-08', which filters for logins on 2022-05-08

I use the following commands to retrieve the information:

SELECT \* FROM log\_in\_attempts WHERE login\_date = 'X' OR login\_date = 'Y';

## Retrieve login attempts outside of Mexico

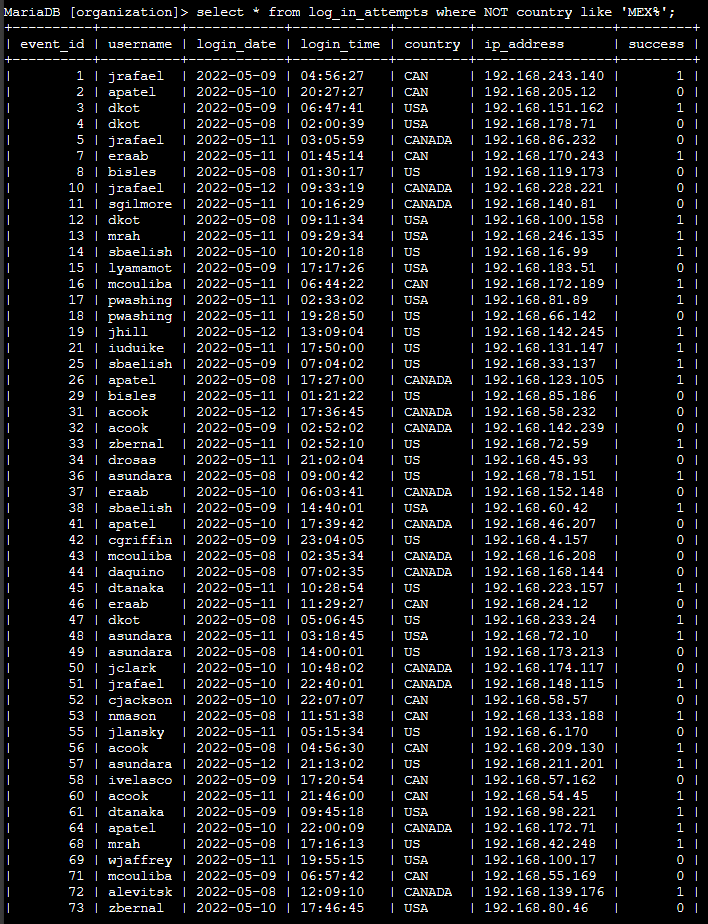
Now, my team is investigating logins that did not originate in Mexico, and you need to find this information. Note that the country field includes entries with 'MEX' and 'MEXICO'.

I use the following commands to retrieve the information:

The first part of the screenshot is my query, and the second part is a portion of the output. This query returns all login attempts that occurred in countries other than Mexico. First, I started by selecting all data from the log\_in\_attempts table. Then, I used a WHERE clause with NOT to filter for countries other than Mexico. I used LIKE with MEX% as the pattern to

match because the dataset represents Mexico as MEX and MEXICO. The percentage sign (%) represents any number of unspecified characters when used with LIKE.

Select \* from log\_in\_attempts where NOT country LIKE ‘MEX%’;



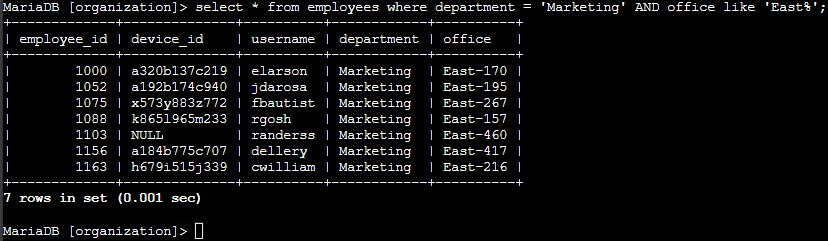
## Retrieve employees in Marketing

My team is updating employee machines, and you need to obtain the information about employees in the 'Marketing' department who are located in all offices in the East building (such as 'East-170' or 'East-320').

The first part of the screenshot is my query, and the second part is a portion of the output. This query returns all employees in the Marketing department in the East building. First, I started by selecting all data from the employees table. Then, I used a WHERE clause with AND to filter for employees who work in the Marketing department and in the East building. I used LIKE with East% as the pattern to match because the data in the office column represents the East building with the specific office number. The first condition is the department = 'Marketing' portion, which filters for employees in the Marketing department. The second condition is the office LIKE 'East%' portion, which filters for employees in the East building.

I use the following commands to retrieve the information:

Select \* from employees where department = ‘Marketing’ AND office like ‘East%’;

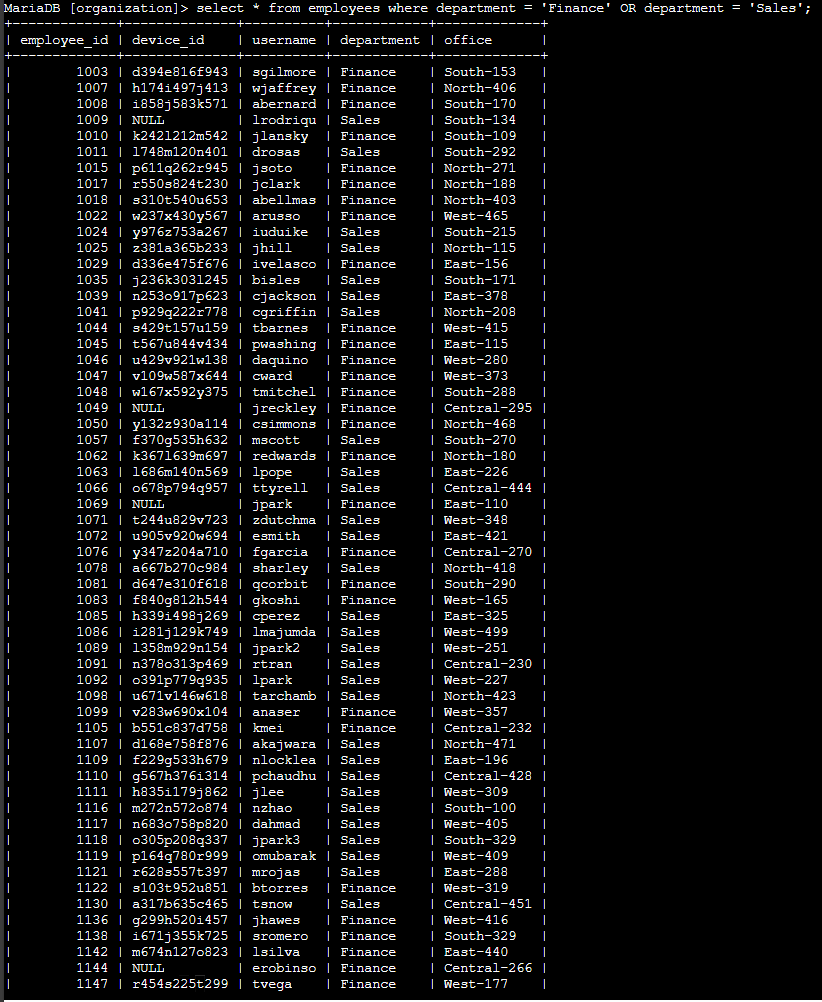


## Retrieve employees in Finance or Sales

Now, your team needs to perform a different update to the computers of all employees in the Finance or the Sales department, and you need to locate information on these employees.

I use the following commands to retrieve the information:

Select \* from employees where department = ‘Finance’ OR department = ‘Sales’;

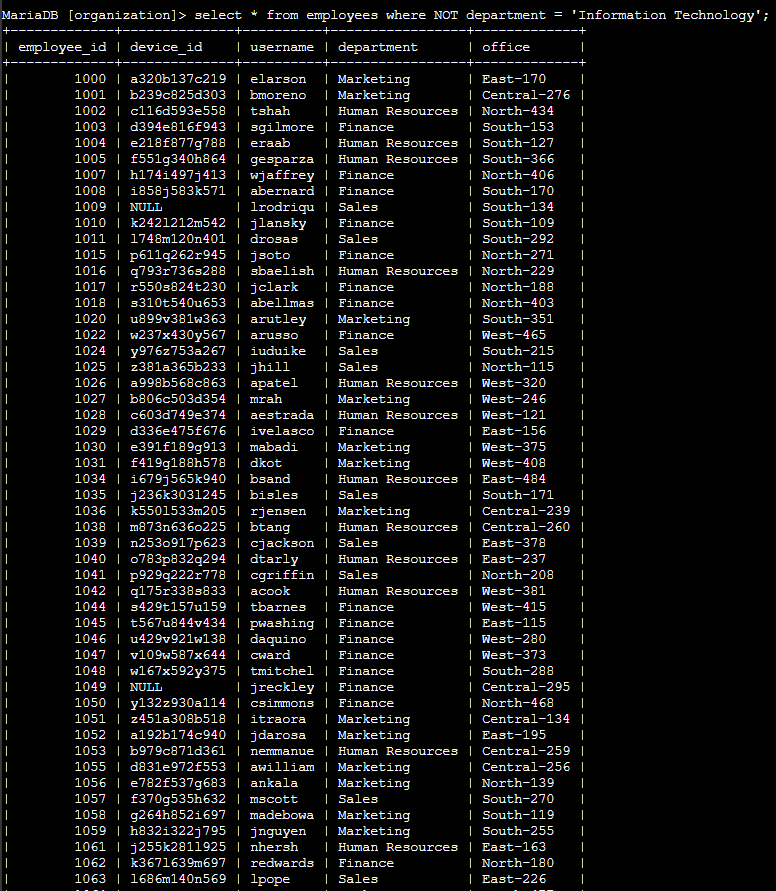


## Retrieve all employees not in IT

My team needs to make one more update. This update was already made to employee computers in the Information Technology department. The team needs information about employees who are not in that department.

I use the following commands to retrieve the information:

Select \* from employees where NOT department = ‘Information Technology’;



## Summary

Now I have demonstrated practical experience in using SQL to run SQL queries to retrieve information from a database and apply AND, OR, and NOT operators to filter SQL queries.