# Apply filters to SQL queries

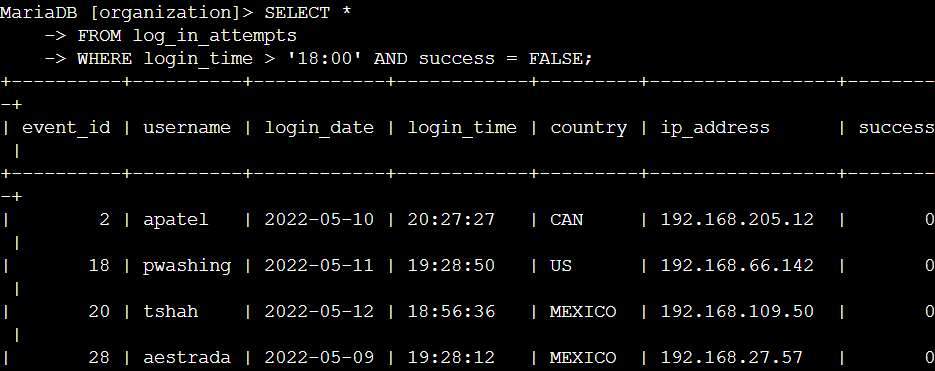
## Project description

As a security professional at a large organization, I am responsible for investigating security issues to maintain system integrity. Recently, I have uncovered potential security concerns involving login attempts and employee machines. The following steps provide examples of how I used SQL with filters to perform security-related tasks.

## Retrieve after hours failed login attempts

A potential security incident happened after business hours (post 18:00). We need to investigate all failed login attempts that occurred during these hours.

The following code demonstrates how I created a SQL query to filter for failed login attempts that occurred after business hours.



In the first part of the screenshot, you'll see the query I used, and the second part shows the results. This query targets failed login attempts after 18:00. I began by selecting all entries from the log\_in\_attempts table, then applied a WHERE clause with an AND operator to narrow the results to those occurring after 18:00 that were unsuccessful. The condition login\_time > '18:00' filters for attempts made post 18:00, and success = FALSE filters for those that failed.

## Retrieve login attempts on specific dates

A suspicious event took place on 2022-05-09. We need to examine any login activity from that date as well as the day before, 2022-05-08.

The following code demonstrates how I created a SQL query to filter for login attempts that occurred on specific dates.

A computer screen with white text

Description automatically generated

In the first part of the screenshot, you'll see my query, and the second part shows a segment of the output. This query fetches all login attempts made on either 2022-05-09 or 2022-05-08. I began by selecting all data from the log\_in\_attempts table. Then, I applied a WHERE clause with an OR operator to filter the results, focusing only on login attempts from those two specific dates. The condition login\_date = '2022-05-09' captures logins on 2022-05-09, while login\_date = '2022-05-08' captures logins on 2022-05-08.

## Retrieve login attempts outside of Mexico

After reviewing the organization's login attempt data, I suspect there's an issue with attempts originating outside of Mexico. These attempts need further investigation.

The following code demonstrates how I created a SQL query to filter for login attempts that occurred outside of Mexico.

A screen shot of a computer

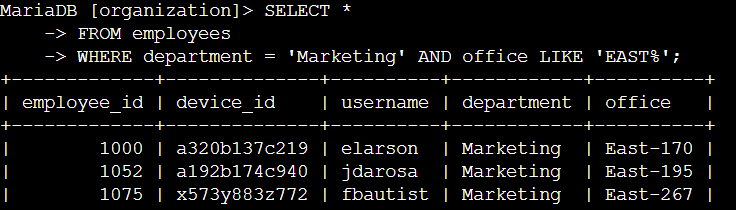
Description automatically generated

The first part of the screenshot shows my query, while the second part displays a snippet of the results. This query retrieves all login attempts from countries other than Mexico. I started by selecting all records from the log\_in\_attempts table, then applied a WHERE clause using NOT to filter out Mexico. The pattern MEX% with LIKE was used to catch entries labeled either 'MEX' or 'MEXICO', with % acting as a wildcard for any number of characters.

## Retrieve employees in Marketing

My team needs to upgrade the computers for some employees in the Marketing department. To proceed, I need to identify which employee machines require updates.

The following code demonstrates how I created a SQL query to filter for employee machines from employees in the Marketing department in the East building.



In the screenshot, the first part shows my query, and the second part displays part of the output. This query fetches all employees in the Marketing department located in the East building. I started by selecting all data from the employees table. Then, I used a WHERE clause with AND to filter for employees working in Marketing and in the East building. I used LIKE with East% to match the office column, as it represents the East building followed by specific office numbers. The first condition department = 'Marketing' filters for Marketing employees, and the second condition office LIKE 'East%' filters for those in the East building.

## Retrieve employees in Finance or Sales

The machines for employees in the Finance and Sales departments also require updates. Given that this update differs from others, I need to gather information specifically about employees from these two departments.

The following code demonstrates how I created a SQL query to filter for employee machines from employees in the Finance or Sales departments.

A screenshot of a computer

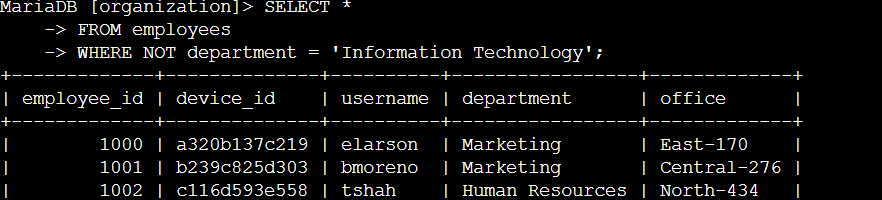
Description automatically generated

The screenshot's first part shows my query, and the second part presents some of the results. This query fetches all employees from the Finance and Sales departments. I began by selecting all data from the employees table. Then, I applied a WHERE clause with the OR operator to filter for employees in either the Finance or Sales departments. The first condition, department = 'Finance', filters for Finance employees, while the second condition, department = 'Sales', filters for Sales employees.

## Retrieve all employees not in IT

My team has one final security update to apply to employees outside the Information Technology department. To proceed, I need to gather information on these employees.

The following demonstrates how I created a SQL query to filter for employee machines from employees not in the Information Technology department.



The screenshot's first section displays my query, while the subsequent section shows some results. This query fetches all employees outside the Information Technology department. I began by selecting all data from the employees table, then applied a WHERE clause with NOT to filter out employees in this department.

## Summary

I used filters in SQL queries to extract specific details on login attempts and employee machines. This involved two tables: log\_in\_attempts and employees. To hone in on the needed information, I utilized the AND, OR, and NOT operators, along with the LIKE operator and the % wildcard for pattern matching.