# Apply filters to SQL queries

## Project description

My organization is enhancing its system security, and my role is to ensure its safety by investigating potential security issues and updating employee computers as needed. The following steps illustrate how I utilized SQL with filters to carry out security-related tasks.

## Retrieve after hours failed login attempts

A potential security incident took place after business hours (post 18:00). All failed login attempts during this time must be investigated.

The following code demonstrates how I crafted an SQL query to filter failed login attempts occurring outside business hours.



The screenshot consists of two parts: my query and a portion of the output. This query filters failed login attempts occurring after 18:00.

I began by selecting all data from the log\_in\_attempts table. Then, I applied a WHERE clause with an AND operator to refine the results, displaying only unsuccessful login attempts made after 18:00. The first condition, login\_time > '18:00', filters for login attempts occurring past 18:00, while the second condition, success = FALSE, ensures only failed attempts are included.

## Retrieve login attempts on specific dates

A suspicious event took place on May 9, 2022. To investigate, all login activity from that day and the previous day must be reviewed.

The following code demonstrates how I wrote an SQL query to filter login attempts for specific dates.



This query retrieves all login attempts made on May 8 or May 9, 2022. 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, displaying only login attempts from either of the specified dates. The first condition, login\_date = '2022-05-09', filters for logins on May 9, while the second condition, login\_date = '2022-05-08', captures logins from May 8.

## Retrieve login attempts outside of Mexico

After analyzing the organization's login attempt data, I identified a potential issue with logins originating outside of Mexico. These attempts require further investigation.

The following code demonstrates how I wrote an SQL query to filter login attempts made outside of Mexico.



This query retrieves all login attempts from countries other than Mexico.

I began by selecting all data from the log\_in\_attempts table. Then, I applied a WHERE clause with NOT to exclude entries from Mexico. To account for variations in how Mexico is represented in the dataset (e.g., "MEX" and "MEXICO"), I used LIKE 'MEX%', where the % represents any number of unspecified characters when used with LIKE.

## Retrieve employees in Marketing

My team plans to update computers for specific employees in the Marketing department. To accomplish this, I need to identify which employee machines require updates.

The following code demonstrates how I wrote an SQL query to filter for employee machines belonging to Marketing department staff located in the East building.



This query retrieves all employees in the Marketing department who work in the East building.

I began by selecting all data from the employees table. Then, I applied a WHERE clause with AND to filter for employees who belong to the Marketing department and are located in the East building. Since the office column includes specific office numbers within the East building, I used LIKE 'East%' to match all relevant entries. The first condition, department = 'Marketing', filters for Marketing employees, while the second condition, office LIKE 'East%', ensures only those in the East building are included.

## Retrieve employees in Finance or Sales

The computers for employees in the Finance and Sales departments also require updates. Since these updates differ from the previous ones, I need to gather information specifically for employees in these two departments.

The following code demonstrates how I wrote an SQL query to filter employee machines belonging to Finance or Sales department staff.



This query retrieves all employees in 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 department. I used OR because I want to include employees from either department. The first condition, department = 'Finance', filters for employees in the Finance department, while the second condition, department = 'Sales', filters for employees in the Sales department.

## Retrieve all employees not in IT

My team needs to apply one final security update for employees who are not in the Information Technology department. To proceed, I first need to gather information about these employees. I created a SQL query to filter for employee machines from employees not in the Information Technology department.



This query retrieves all employees who are not in the Information Technology department.

I began by selecting all data from the employees table. Then, I applied a WHERE clause with NOT to filter out employees in the Information Technology department

## Summary

I applied filters to SQL queries to retrieve specific information about login attempts and employee machines, using two tables: log\_in\_attempts and employees. I utilized the AND, OR, and NOT operators to refine the results for each task. Additionally, I used LIKE along with the percentage sign (%) wildcard to filter for specific patterns.