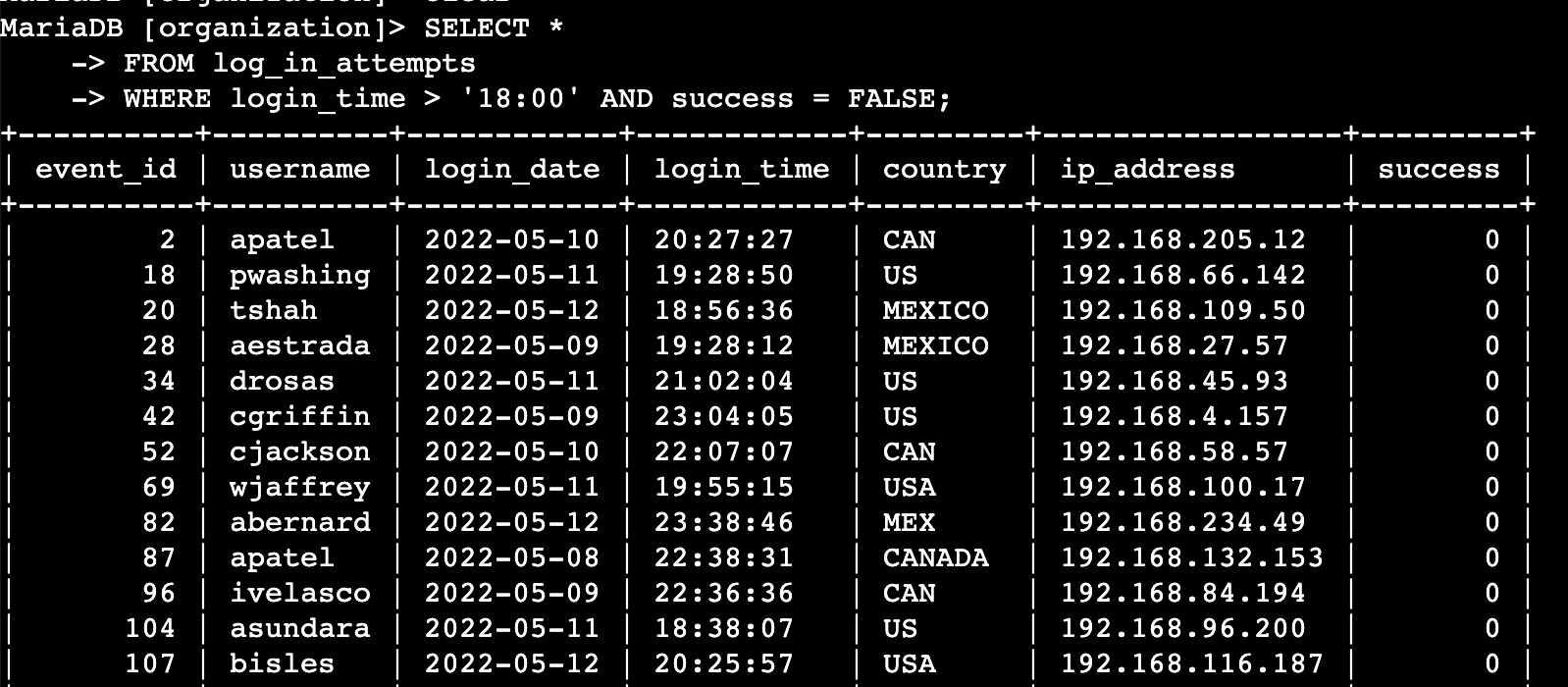
# Apply filters to SQL queries

## Project description

## To boost our system's security, my task is to check for any security gaps, and update computers used by our staff accordingly. I perform these tasks using SQL commands, which are set up with certain filters to help me pinpoint and address security issues effectively.

## Retrieve after hours failed login attempts

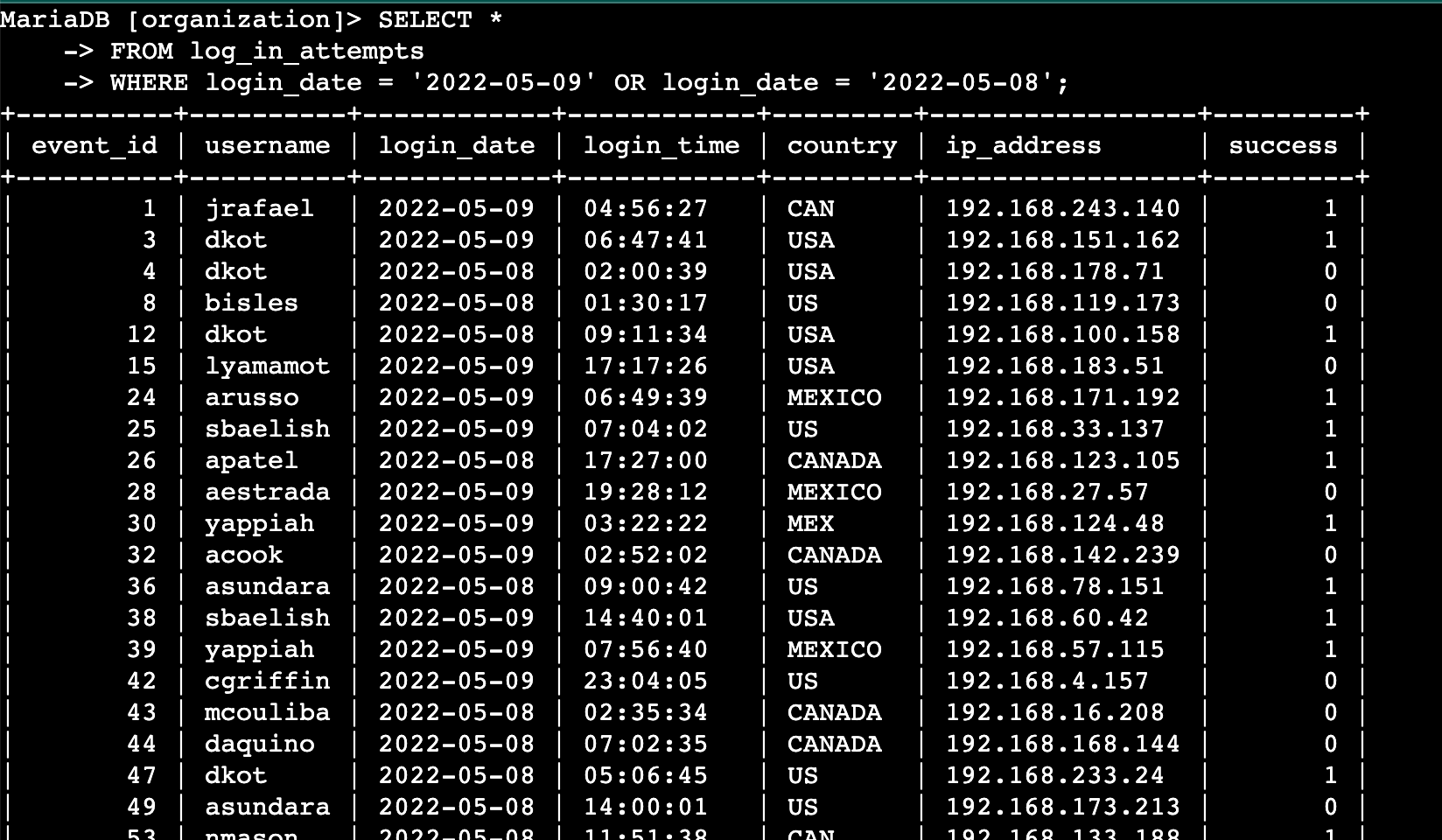
To address a possible security event that happened post-work hours, I crafted a SQL query to isolate failed login attempts made after 6 PM(18;00) for investigation.



## In my SQL query, I looked at the log\_in\_attempts table to find out who tried to log in after 6 PM and didn't succeed. I chose records that were after 18:00 and where the success was marked as FALSE.

## Retrieve login attempts on specific dates

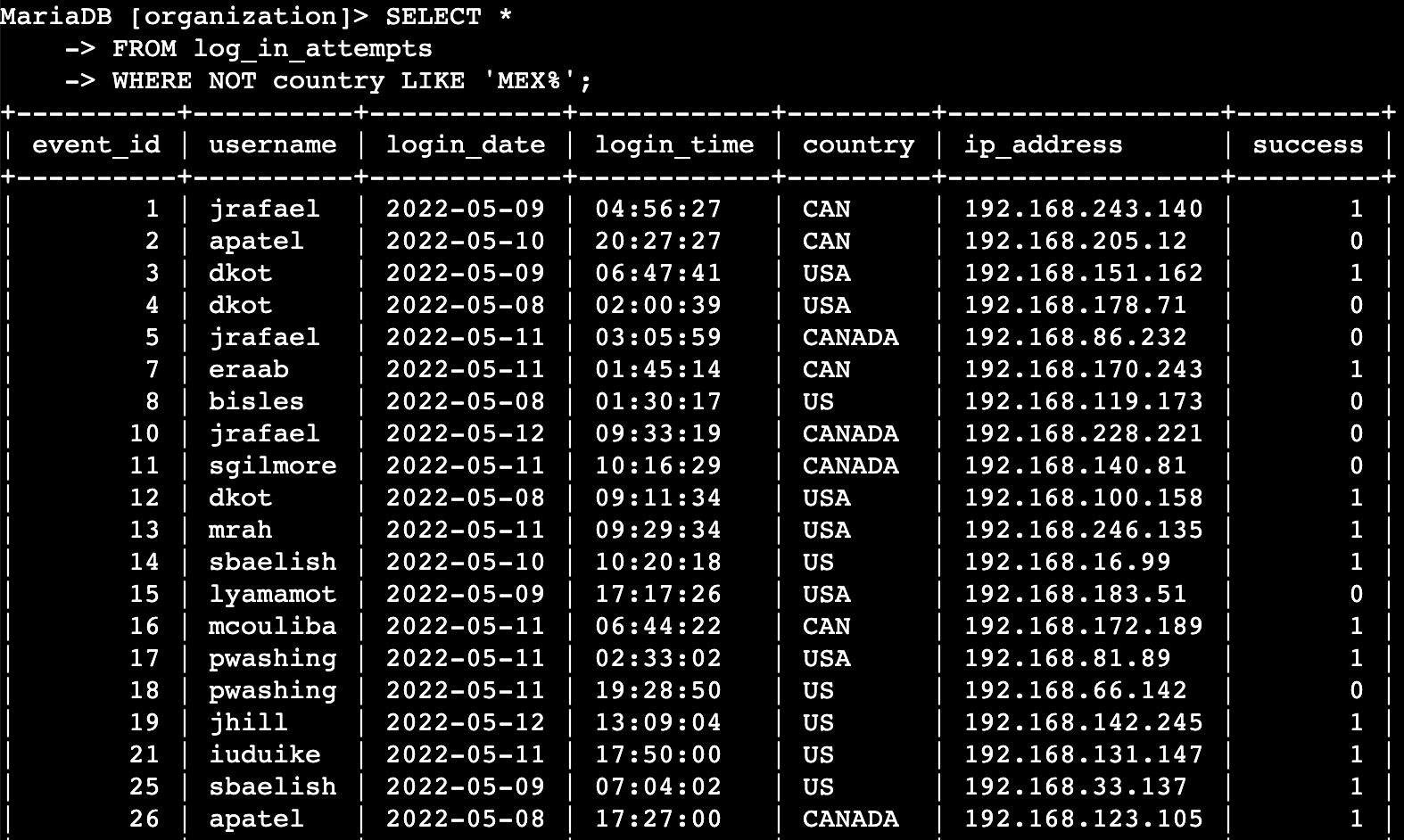
To look into a suspicious incident from May 9, 2022, I used a SQL query to focus on login attempts from that day and the day before it.



## My SQL query checks for login attempts on May 9 and 8, 2022. I selected records from log\_in\_attempts, using a WHERE clause and OR to get results for these two dates.

## Retrieve login attempts outside of Mexico

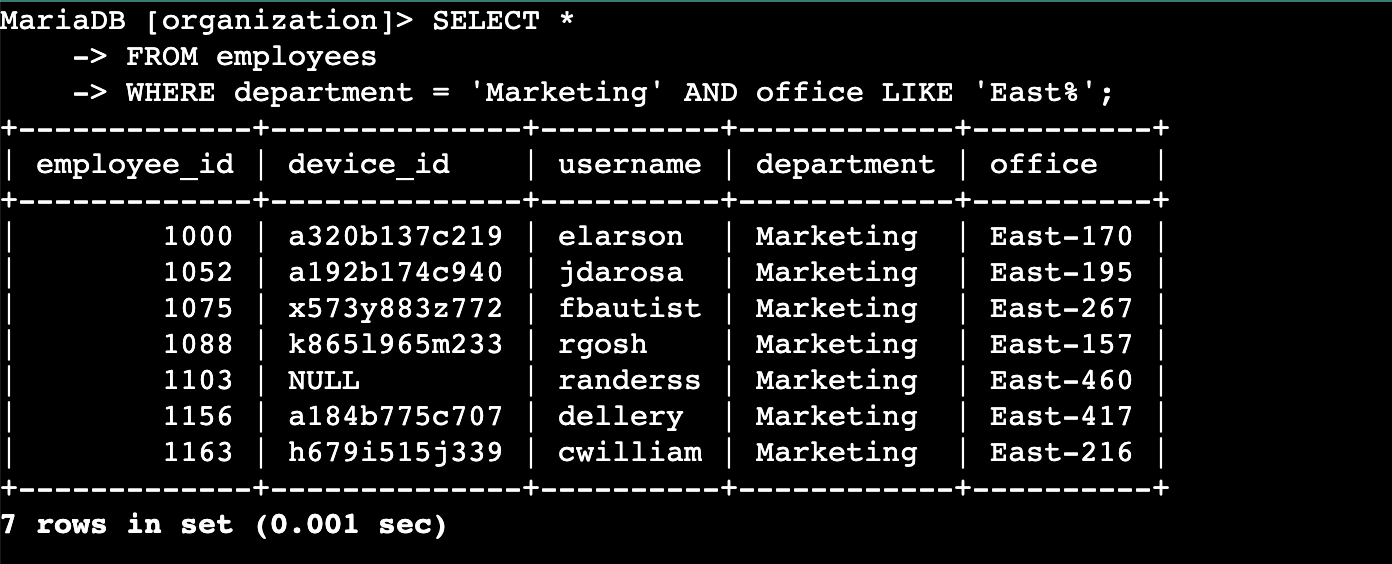
To probe login attempts from outside Mexico, I constructed a SQL query to filter through our records for any such instances.



## The query I crafted filters out login attempts made from locations excluding Mexico. By selecting from the log\_in\_attempts table and using a WHERE clause with NOT and LIKE, I targeted all entries that do not start with 'MEX', accounting for variations like 'MEX' or 'MEXICO'. The '%' wildcard ensures all such records are captured.

## Retrieve employees in Marketing

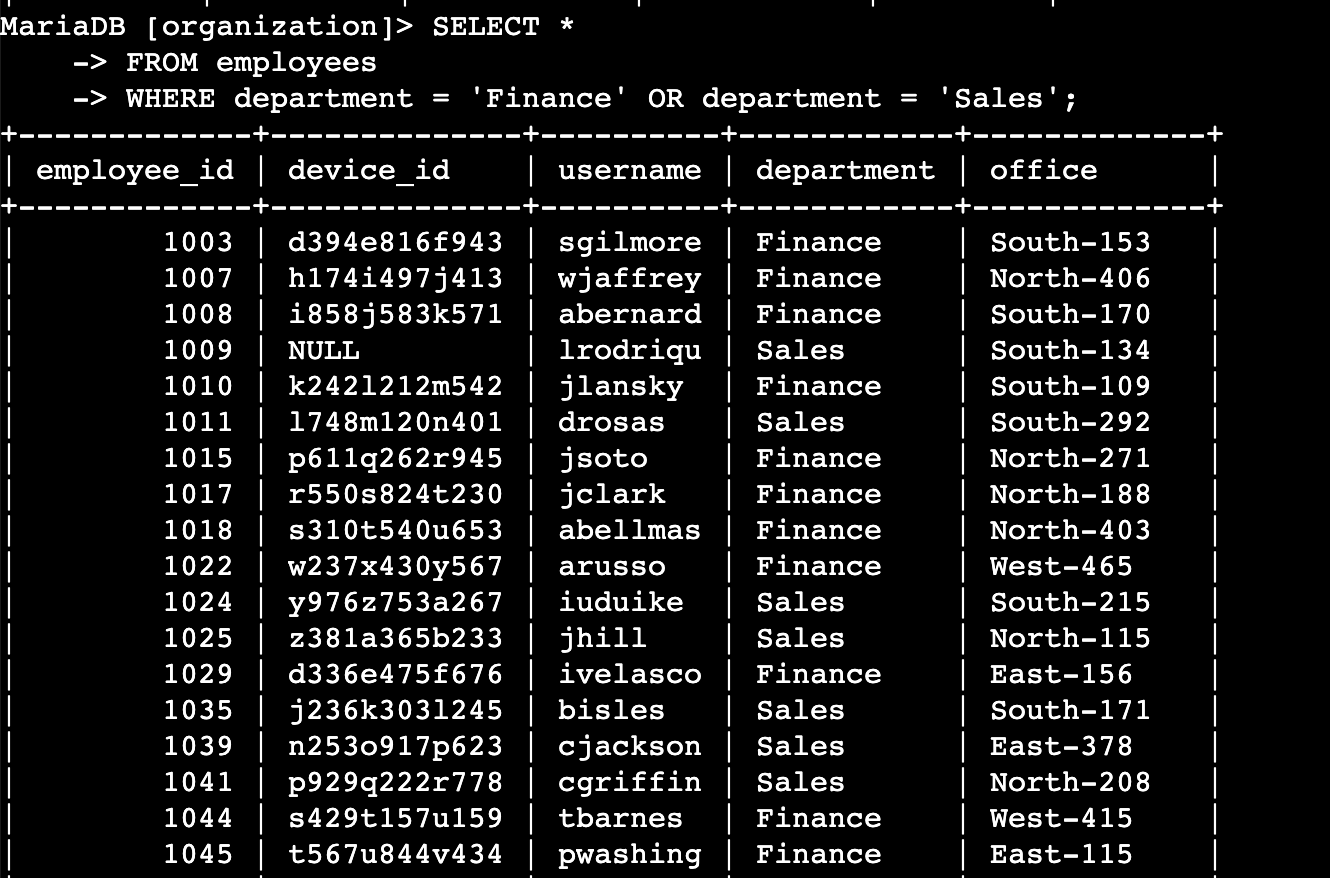
To upgrade computers for select Marketing staff in the East building, I wrote a SQL query to find the specific machines.



I formulated a SQL query that retrieves a list of employees from the Marketing department located in the East building. The search criteria were set using the WHERE clause to specify 'Marketing' for the department and LIKE 'East%' for the office location, ensuring we target only those in the specified department and building.

## Retrieve employees in Finance or Sales

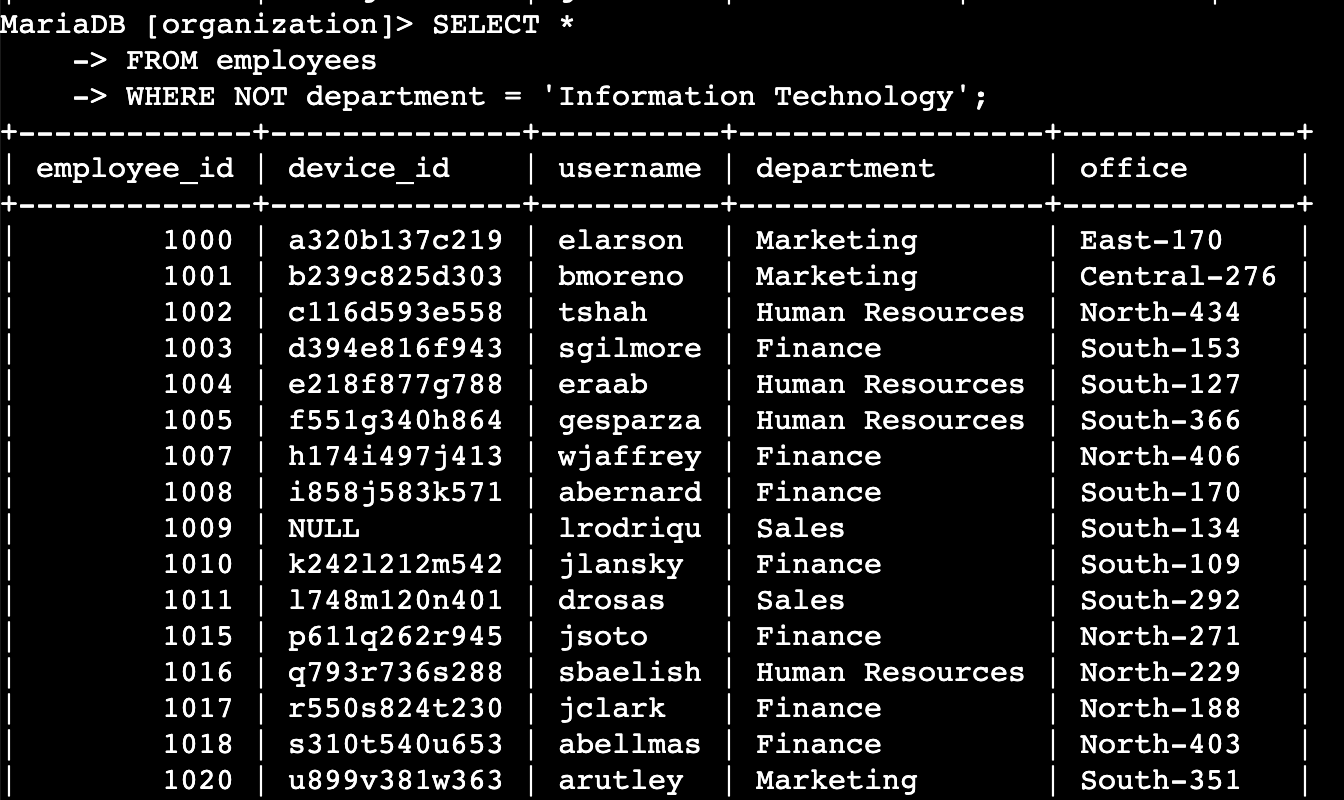
To prepare for system upgrades, I used SQL to select machines from the Finance and Sales departments. This query focused on finding the employees who need a specific security update.



## My SQL query fetches details for employees from either the Finance or Sales departments. I used the WHERE clause with OR to get employees belonging to either department from the employees table.

## Retrieve all employees not in IT

For the final security update, I crafted a SQL query to select machines of employees outside the IT department. This helps identify who needs the update.



## My SQL query is designed to retrieve details from the employees table for all staff not part of the Information Technology department, using a WHERE clause with NOT for the exclusion.

## Summary

To gather precise data, I used SQL queries with specific filters across two tables: log\_in\_attempts for login details, and employees for staff information. By using AND, OR, and NOT conditions, along with LIKE for pattern searching, I narrowed down the data to meet the distinct requirements of our security checks and updates. The percentage symbol (%) served as a wildcard, enabling more flexible pattern matching for complex queries.