Scenario

You are a security professional at a large organization. Part of your job is to investigate security issues to help keep the system secure. You recently discovered some potential security issues that involve login attempts and employee machines.
Your task is to examine the organization's data in their employees and log_in_attempts tables. You'll need to use SQL filters to retrieve records from different datasets and investigate the potential security issues.
Note: This scenario involves the same queries as the ones the Filter with AND, OR, and NOT lab. You can revisit the lab to get screenshots to include in your portfolio document. If you choose, it's also possible to complete this activity without revisiting the lab by typing your queries in the template

1. **Apply Filters to SQL Queries Portfolio**
    a. As a security professional at a large organization, part of my role is to investigate security issues and ensure system integrity. In this project, I use SQL queries to analyze login attempts and employee records, identifying potential security threats and filtering relevant data using AND, OR, NOT, and LIKE operators. These queries help in detecting suspicious activities, such as failed login attempts outside business hours, unauthorized access from specific locations, and department-based machine updates.
2. **Retrieve After Hours Failed Login Attempts**
    a. To investigate security incidents outside business hours, I used the following query to identify failed login attempts after 18:00:
        i. **SELECT * FROM log_in_attempts**
        ii. **WHERE login_time > '18:00' AND success = 0;**
    b. This query filters records where the login attempt time is greater than 18:00 and the success column indicates a failed attempt (value of 0). This helps identify unauthorized access attempts after working hours.
3. **Retrieve Login Attempts on Specific Dates**
    a. To analyze suspicious events on 2022-05-09 and the previous day, I used:
        i. **SELECT * FROM log_in_attempts**
        ii. **WHERE login_date = '2022-05-09' OR login_date = '2022-05-08';**
    b. This query retrieves all login attempts that occurred on May 8th and 9th, allowing the security team to review login patterns around the suspicious event.
4. **Retrieve Login Attempts Outside of Mexico**
    a. To filter out login attempts originating from Mexico, I used:
        i. **SELECT * FROM log_in_attempts**
        ii. **WHERE country NOT LIKE 'MEX%' AND country NOT LIKE 'MEXICO';**
    b. The NOT LIKE operator ensures that login attempts from both "MEX" and "MEXICO" are excluded, focusing on activities outside of Mexico.
5. **Retrieve Employees in Marketing**
    a. To retrieve employees from the Marketing department located in the East building, I used:
        i. **SELECT * FROM employees**
        ii. **WHERE department = 'Marketing' AND office LIKE 'East-%';**
    b. This query filters employees who belong to the Marketing department and have an office located in the East building.
6. **Retrieve Employees in Finance or Sales**
    a. To find employees in either the Finance or Sales department, I executed:
        i. **SELECT * FROM employees**
        ii. **WHERE department = 'Finance' OR department = 'Sales';**
    b. The OR operator ensures that employees from both departments are included in the results.
7. **Retrieve All Employees Not in IT**
    a. To identify employees outside the IT department, I used:
        i. **SELECT * FROM employees**
        ii. **WHERE department <> 'Information Technology';**
    b. The NOT operator (<>) excludes all employees from the IT department, listing only those who require security updates.

This project demonstrates the use of SQL queries to filter data effectively, aiding in security investigations. By leveraging operators such as AND, OR, NOT, and LIKE, I successfully retrieved relevant data on failed login attempts, suspicious activities, and employee department details. These queries are essential for identifying potential threats and ensuring system security in a large organization.