Apply Filters to SQL Queries Portfolio Activity

Project Description

As a security professional at a large organization, I am responsible for investigating security issues related to login attempts and employee machines. This project focuses on using SQL queries to filter and retrieve specific records from the `log_in_attempts` and `employees` tables. Through these queries, I identify security-related patterns, such as failed login attempts after business hours, suspicious activities on specific dates, and unauthorized access from outside Mexico. Additionally, I retrieve employee data based on department and office locations to facilitate security updates.

Retrieve After-Hours Failed Login Attempts

Query:

```
SELECT *
FROM log_in_attempts
WHERE login_time > '18:00:00' AND success = 0;
```

Explanation:

This query retrieves all failed login attempts (`success = 0`) that occurred after 18:00 (6 PM). The `login_time` column is filtered using the `>` operator to detect suspicious activity outside regular working hours.

Screenshot Placeholder: Add Screenshot Here

Retrieve Login Attempts on Specific Dates

Query:

```
SELECT *
FROM log_in_attempts
WHERE login_date = '2022-05-09' OR login_date = '2022-05-08';
```

Explanation:

This query filters login attempts that occurred on either May 8, 2022, or May 9, 2022. The `OR` operator is used to include both dates, as they are relevant to the investigation.

Screenshot Placeholder: Add Screenshot Here

Retrieve Login Attempts Outside of Mexico

Query:

```
SELECT *
FROM log_in_attempts
WHERE NOT country LIKE 'MEX%';
```

Explanation:

This query filters out login attempts from Mexico by excluding values that start with "MEX" or "MEXICO." The `LIKE` operator with `%` is used to match different variations of the country name.

Screenshot Placeholder: Add Screenshot Here

Retrieve Employees in Marketing (East Building)

Query:

```
SELECT *
FROM employees
WHERE department = 'Marketing' AND office LIKE 'East-%';
```

Explanation:

This query retrieves all employees working in the Marketing department who are in an office located in the East building. The `LIKE` operator with `East-%` ensures we capture all room numbers within the East wing.

Screenshot Placeholder: Add Screenshot Here

Retrieve Employees in Finance or Sales

Query:

```
SELECT *
FROM employees
WHERE department = 'Finance' OR department = 'Sales';
```

Explanation:

This query retrieves all employees who work in either the Finance or Sales departments. The `OR` operator is used to include employees from both departments in the results.

Screenshot Placeholder: Add Screenshot Here

Retrieve All Employees Not in IT

Query:

```
SELECT *
FROM employees
WHERE NOT department = 'Information Technology';
```

Explanation:

This query retrieves all employees who are **not** in the Information Technology department. The `NOT` operator is used to exclude IT employees, as they have already received the necessary security update.

Screenshot Placeholder: Add Screenshot Here

Summary

Through this project, I used SQL queries with `AND`, `OR`, and `NOT` operators to filter data efficiently for security analysis. The queries helped identify failed login attempts outside business hours, suspicious login activity on specific dates, and unauthorized access from outside Mexico. Additionally, I retrieved employee information for security updates based on department and office location. These SQL filtering techniques enable precise data extraction, supporting effective security investigations and system protection.