Apply filters to SQL queries

Project description

In this project, I used SQL to investigate a potential security incident within an organization. By applying SQL filters such as `AND`, `OR`, `NOT`, `LIKE`, and time/date comparisons, I retrieved specific information from the `log_in_attempts` and `employees` tables. These queries simulate the work of a cybersecurity analyst investigating login activity and preparing updates to employee machines.

Retrieve after hours failed login attempts

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

This query filters for all failed login attempts (success = 0) that occurred after 6:00 PM (18:00:00). These failed attempts could indicate suspicious activity happening outside regular business hours and are important to investigate from a security perspective.

Retrieve login attempts on specific dates

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

This query retrieves all login attempts that occurred on May 8 and May 9, 2022. It uses the OR operator to combine two date conditions. These dates are being examined due to a suspicious event reported on May 9, making it critical to analyze login behavior around that time.

Retrieve login attempts outside of Mexico

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

This query retrieves login attempts where the country is not Mexico, excluding both 'MEX' and 'MEXICO' by using the NOT LIKE 'MEX%' condition. It's used to narrow the search to logins that originated outside Mexico, which helps isolate suspicious international activity.

Retrieve employees in Marketing

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

This query retrieves all employees who are in the Marketing department and work in any office located in the East building. It uses LIKE 'East-%' to match all office values that start with "East-", ensuring the results include multiple East-located offices such as East-170, East-320, etc.

Retrieve employees in Finance or Sales

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

This query finds all employees who belong to either the Finance or Sales departments. The OR operator is used to include employees from both departments so that their machines can be updated as part of a security routine.

Retrieve all employees not in IT

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

This query filters out employees who are in the Information Technology department. Using NOT excludes that group and returns everyone else, helping to identify devices that still need updates since IT machines are already covered.

Summary

In this project, I used SQL queries to investigate failed login attempts, filter login records by date and location, and retrieve specific groups of employees based on department and office location. Each query targeted data needed for **security analysis** or **system updates**. By applying filtering techniques such as `LIKE`, `NOT`, `AND`, and `OR`, I demonstrated the ability to retrieve precise information — a key skill for cybersecurity professionals managing large datasets.