Apply filters to SQL queries

Project description

In this project, my fictitious organization has focused on making their systems more secure. My job is to ensure a safe perimeter by investigating any potential security issues, and updating any employee computers that are out of date. I will explain my process on how I used SQL filters and queries to perform security-related tasks.in a very detailed manner.

Retrieve after hours failed login attempts

The organization informed me of a potential incident that occurred after business hours. We know that after hours starts at 6pm, which can be represented in SQL as `'18:00'`. Every log in attempt after this time should be investigated. To do this we employ the following SQL query to filter for failed login attempts outside of business hours.

```
MariaDB [organization]> SELECT *
    -> FROM log_in_attempts
    -> WHERE login_time > '18:00' AND success = FALSE;
+----------+----------+------------+------------+---------+-----------------+---------+
| event_id | username | login_date | login_time | country | ip_address      | success |
+----------+----------+------------+------------+---------+-----------------+---------+
|        2 | apatel   | 2022-05-10 | 20:27:27   | CAN     | 192.168.205.12  |       0 |
|       18 | pwashing | 2022-05-11 | 19:28:50   | US      | 192.168.66.142  |       0 |
|       20 | tshah    | 2022-05-12 | 18:56:36   | MEXICO  | 192.168.109.50  |       0 |
|       28 | aestrada | 2022-05-09 | 19:28:12   | MEXICO  | 192.168.27.57   |       0 |
|       34 | drosas   | 2022-05-11 | 21:02:04   | US      | 192.168.45.93   |       0 |
```

The first half of the screenshot are my queries, and the chart in the second half is the database output. First, I started by selecting all the data (`*`) from the `log_in_attempts` table. Then, I used a `WHERE` clause with an `AND` operator to filter the results to output only login attempts that happened after 6pm and were ineffective. The first condition is `login_time > '18:00'`, which filters for the login attempts that occurred after 18:00. The second condition is `success = FALSE`, which filters for the failed login attempts.

Retrieve login attempts on specific dates

The organization has reported to me that a suspicious event has occurred on 2022-05-09. They want me to investigate any events on this day or the day before.
The following code demonstrates how I created and SQL query to filter for login attempts on these specific dates.

```
MariaDB [organization]> SELECT *
    -> FROM log_in_attempts
    -> WHERE login_date = '2022-05-08' OR login_date = '2022-05-09';
+----------+----------+------------+------------+---------+-----------------+---------+
| event_id | username | login_date | login_time | country | ip_address      | success |
+----------+----------+------------+------------+---------+-----------------+---------+
|        1 | jrafael  | 2022-05-09 | 04:56:27   | CAN     | 192.168.243.140 |       1 |
|        3 | dkot     | 2022-05-09 | 06:47:41   | USA     | 192.168.151.162 |       1 |
|        4 | dkot     | 2022-05-08 | 02:00:39   | USA     | 192.168.178.71  |       0 |
|        8 | bisles   | 2022-05-08 | 01:30:17   | US      | 192.168.119.173 |       0 |
```

The first half of the screenshot are my queries, and the chart in the second half is the database output. The first two lines stay the same as last query, then I used the `WHERE` clause with the `OR` operator to output only login attempts that occurred on either 2022-05-09 or 2022-05-08. The first condition is `login_date = '2022-05-09'`, which filters for logins on 2022-05-09. The second condition is `login_date = '2022-05-08'`, which filters for logins on 2022-05-08.

Retrieve login attempts outside of Mexico

Upon further investigating, I came up with the strong hypothesis that the issue has to come from a device outside of Mexico, so these login attempts should be investigated.
To achieve this, I input the following code into the SQL database:

```
MariaDB [organization]> SELECT *
    -> FROM log_in_attempts
    -> WHERE NOT country LIKE 'MEX%';
+----------+----------+------------+------------+---------+-----------------+---------+
| event_id | username | login_date | login_time | country | ip_address      | success |
+----------+----------+------------+------------+---------+-----------------+---------+
|        1 | jrafael  | 2022-05-09 | 04:56:27   | CAN     | 192.168.243.140 |       1 |
|        2 | apatel   | 2022-05-10 | 20:27:27   | CAN     | 192.168.205.12  |       0 |
|        3 | dkot     | 2022-05-09 | 06:47:41   | USA     | 192.168.151.162 |       1 |
|        4 | dkot     | 2022-05-08 | 02:00:39   | USA     | 192.168.178.71  |       0 |
```

The first half of the screenshot are my queries, and the chart in the second half is the database output. The first two lines stay the same as the last query, then, I used a `WHERE` clause with `NOT` to filter for countries other than Mexico. I used `LIKE` with `MEX%` as the pattern to match because the dataset represents Mexico as `MEX` and `MEXICO`. The percentage sign (`%`) represents any number of unspecified characters when used with `LIKE`.

Retrieve employees in Marketing

My boss said that he wishes to update the computers for certain employees in the Marketing department.
The code below represents how I gathered information on which employee machines need an update.

```
MariaDB [organization]> SELECT *
    -> FROM employees
    -> WHERE department = 'Marketing' AND office LIKE 'East%';
+-------------+--------------+----------+------------+----------+
| employee_id | device_id    | username | department | office   |
+-------------+--------------+----------+------------+----------+
|        1000 | a320b137c219 | elarson  | Marketing  | East-170 |
|        1052 | a192b174c940 | jdarosa  | Marketing  | East-195 |
|        1075 | x573y883z772 | fbautist | Marketing  | East-267 |
|        1088 | k8651965m233 | rgosh    | Marketing  | East-157 |
```

The first half of the screenshot are my queries, and the chart in the second half is the database output. The first two lines stay the same as the last query (we only change to look in the `employees` table instead), then, I used a `WHERE` clause with `AND` to filter for employees who work in the Marketing department and in the East building. I used `LIKE` with `East%` as the pattern to match because the data in the `office` column represents the East building with the specific office number. The first condition is the `department = 'Marketing'` portion, which filters for employees in the Marketing department. The second condition is the `office LIKE 'East%'` portion, which filters for employees in the East building.

Retrieve employees in Finance or Sales

My team reported that the machines for employees in the Finance and Sales departments need to be updated as well.
The following code demonstrates how I filtered for employee machines that need the new security patch from employees in the Finance and Sales departments by using SQL.

```
MariaDB [organization]> SELECT *
    -> FROM employees
    -> WHERE department = 'Finance' OR department = 'Sales';
+-------------+--------------+----------+------------+-------------+
| employee_id | device_id    | username | department | office      |
+-------------+--------------+----------+------------+-------------+
|        1003 | d394e816f943 | sgilmore | Finance    | South-153   |
|        1007 | h174i497j413 | wjaffrey | Finance    | North-406   |
|        1008 | i858j583k571 | abernard | Finance    | South-170   |
|        1009 | NULL         | lrodriqu | Sales      | South-134   |
```

The first half of the screenshot are my queries, and the chart in the second half is the database output. The first two lines stay the same as the last query; then, I used a `WHERE` clause with `OR` to filter for employees who are in the Finance and Sales departments. I used the `OR` operator instead of `AND` because I want all employees who are in either department. The first condition is `department =`

`'Finance'`, which filters for employees from the Finance department. The second condition is `department = 'Sales'`, which filters for employees from the Sales department.

Retrieve all employees not in IT

Finally, my team informed me that there is still one more security update for all employees who are not in the Information Technology department.
The following code demonstrates how I created a SQL query to filter for employee machines from employees that are not in the Information Security department.

```
MariaDB [organization]> SELECT *
    -> FROM employees
    -> WHERE NOT department = 'Information Technology';
+-------------+--------------+----------+-----------------+-------------+
| employee_id | device_id    | username | department      | office      |
+-------------+--------------+----------+-----------------+-------------+
|        1000 | a320b137c219 | elarson  | Marketing       | East-170    |
|        1001 | b239c825d303 | bmoreno  | Marketing       | Central-276 |
|        1002 | c116d593e558 | tshah    | Human Resources | North-434   |
|        1003 | d394e816f943 | sgilmore | Finance         | South-153   |
```

The first half of the screenshot are my queries, and the chart in the second half is the database output. The first two lines stay the same as the last query; then, I used a `WHERE` clause with `NOT` to filter for employees not in this department.

Summary

I applied various filters to SQL queries to find specific information on login attempts as well as employee machines. I used two different tables, `log_in_attempts` and `employees`. Furthermore, I used operators such as `AND`, `OR`, and `NOT` to further filter for the specific information needed for each task. I also used `LIKE` and the percentage sign (`%`) wildcard to filter for patterns.