Rhiannon Goebel Portfolio | SQL Login Attempts

Project description

The company is working on making their systems more secure and recently discovered some potential security issues that involve login attempts and employee machines. My task was to examine the organization's data to investigate any potential security issues and find computers that needed updating. The following steps show how I used filters in SQL queries to perform security tasks.

Retrieve after hours failed login attempts

There was a potential security incident that occurred after business hours. Below is the code I used to find all after hours login attempts that failed and would need to be investigated.

```
MariaDB [organization]> SELECT * FROM log_in_attempts
    -> WHERE login_time > '18:00' AND success = 0;
```

Retrieve login attempts on specific dates

A suspicious event occurred on 2022-05-09, so login activity that happened on 2022-05-09 or on the day before needed to be investigated. Below is the code I used to find all activity that happened on 2022-05-09 or on the day before and would need to be investigated.

```
MariaDB [organization]> SELECT * FROM log_in_attempts
    -> WHERE login_date = '2022-05-08' OR login_date = '2022-05-09';
```

Retrieve login attempts outside of Mexico

It seemed like there was an issue with the login attempts from outside of Mexico. Below is the code I used to find the login activity from outside this region and would need to be investigated. The data was inconsistently using 'MEXICO' and 'MEX' to refer to the region, so I utilized LIKE to ensure all instances referring to Mexico were removed from the data set.

```
MariaDB [organization]> SELECT * FROM log_in_attempts
    -> WHERE country NOT LIKE 'MEX%';
```

Retrieve employees in Marketing

The team needed to update the devices for employees in the Marketing department that worked in an offices within the East building. Below is the code I used to find information on which employee machines were being used by those in the Marketing department to identify the machines that would need to be updated. I also utilized the WHERE clause and LIKE to limit the results to show all devices in any of the East offices (within the Marketing department).

```
MariaDB [organization]> SELECT * FROM employees
    -> WHERE department = 'Marketing' AND office LIKE 'East%';
+------------+--------------+----------+-----------+----------+
```

Retrieve employees in Finance or Sales

The machines for employees in the Finance and Sales departments also need to be updated. Since a different security update is needed, I have to get information on employees only from these two departments. Below is the code that I used to find employees across both of these departments by utilizing the WHERE clause.

```
MariaDB [organization]> SELECT * FROM employees
    -> WHERE department = 'Sales' OR department = 'Finance';
```

Retrieve all employees not in IT

The team realized that a final security update was needed for devices of employees that were not in the Information Technology department. Below is the code that I used to pull information on these employees and the devices that needed to be updated. I used <> in this instance, but could also have used NOT to filter the information.

```
MariaDB [organization]> SELECT * FROM employees
    -> WHERE department <> 'Information Technology';
```

Summary

I applied clauses and filters to SQL queries to pull specific information on login attempts and employee machines. To do this, I used two different tables, log_in_attempts and employees. I used AND, OR, and NOT to filter the data needed, and LIKE and the percentage sign (%) to filter for patterns.