

Apply filters to SQL queries

Project description

The organization wants to investigate certain security issues and update some computers that are outdated. SQL allows us to see specific data relating to security issues and employees with certain conditions relating to computer updates.

Retrieve after hours failed login attempts

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

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
42	cgriffin	2022-05-09	23:04:05	US	192.168.4.157	0
52	cjackson	2022-05-10	22:07:07	CAN	192.168.58.57	0
69	wjaffrey	2022-05-11	19:55:15	USA	192.168.100.17	0
82	abernard	2022-05-12	23:38:46	MEX	192.168.234.49	0
87	apatel	2022-05-08	22:38:31	CANADA	192.168.132.153	0
96	ivelasco	2022-05-09	22:36:36	CAN	192.168.84.194	0
104	asundara	2022-05-11	18:38:07	US	192.168.96.200	0
107	bisles	2022-05-12	20:25:57	USA	192.168.116.187	0
111	aestrada	2022-05-10	22:00:26	MEXICO	192.168.76.27	0
127	abellmas	2022-05-09	21:20:51	CANADA	192.168.70.122	0
131	bisles	2022-05-09	20:03:55	US	192.168.113.171	0
155	cgriffin	2022-05-12	22:18:42	USA	192.168.236.176	0
160	jclark	2022-05-10	20:49:00	CANADA	192.168.214.49	0
199	yappiah	2022-05-11	19:34:48	MEXICO	192.168.44.232	0

19 rows in set (0.094 sec)

This query investigates the login attempts table for attempts made outside business hours that were unsuccessful. This is done with a SELECT query and specific conditions such as login time being after 18:00 (6pm to SQL) and the success of the attempt being 0 (or FALSE). All combined the query filters the table for unsuccessful login attempts made after 6pm. In total 19 rows are returned.

Retrieve login attempts on specific dates

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

This query investigates the login attempts made on two separate specific days. This is completed by using an OR operator to ensure the query returns login attempts that meet either condition, the first day or the second specified day. Put together the query returns all login attempts on two separate specific days and no other days.

Retrieve login attempts outside of Mexico

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

This query relies on the idea that no suspicious activity originates from Mexico. This means we must exclude all login attempts from Mexico to narrow our search. The query uses a NOT operator to show all login attempts that occurred that did not originate from Mexico. In this table Mexico is listed as MEX and MEXICO meaning we have to use the LIKE operator (similar to = but less restrictive and allows the use of wildcards, like % which includes unknown following characters in the string) to include both options.

Retrieve employees in Marketing

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

This query is more specific than the others, it requires two specific conditions, one of which needs the LIKE operator to include multiple similar strings, like offices. To start the query is the same, requesting all columns from employees table but at the WHERE keyword there are two conditions separated by an AND operator. The AND operator specifies that both conditions have to be met for the row to be valid in this query. The first condition is that the employee's department is Marketing, and the second condition is that the employee's office is in the east building, or the office column is LIKE EAST% (% means unknown extra characters come after).

Retrieve employees in Finance or Sales

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

This query is quite simple, it just requests all rows of the employees table where the department is Finance or Sales. This is done with the OR operator described before.

Retrieve all employees not in IT

```
MariaDB [organization]> SELECT * FROM employees WHERE NOT department = 'Information Technology';
```

This query includes all employees except those in the Information Technology department. This is because those employees have already received this update. So to do this the query selects all employees NOT in the information technology department.

Summary

The database was queried using MariaDB and retrieved information on login attempts during certain time and date windows, as well as attempts made in certain geographical regions. We also obtained data about certain employees to see who needed the updates based on their department and which building they were in.