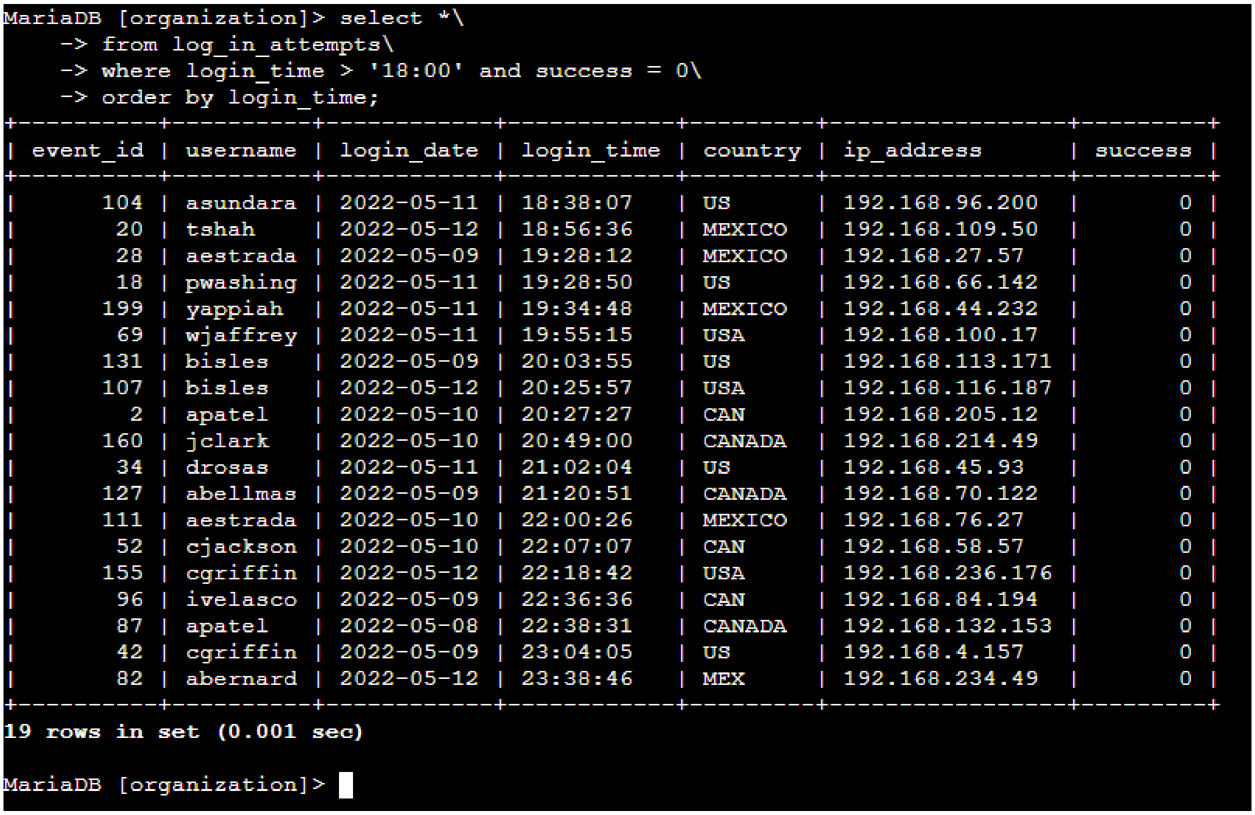
Apply filters to SQL queries

Project description

Part of my job is to investigate security issues to help keep the system secure. I recently discovered some potential security issues that involve login attempts and employee machines. My task is to examine the organization’s data in their employees and log_in_attempts tables. I’ll need to retrieve records from different datasets and investigate the potential security issues.

Retrieve after hours failed login attempts

I recently discovered a potential security incident that occurred after business hours. To investigate this, I need to query the log_in_attempts table and review after hours login activity. I will use filters in SQL to create a query that identifies all failed login attempts that occurred after 18:00.



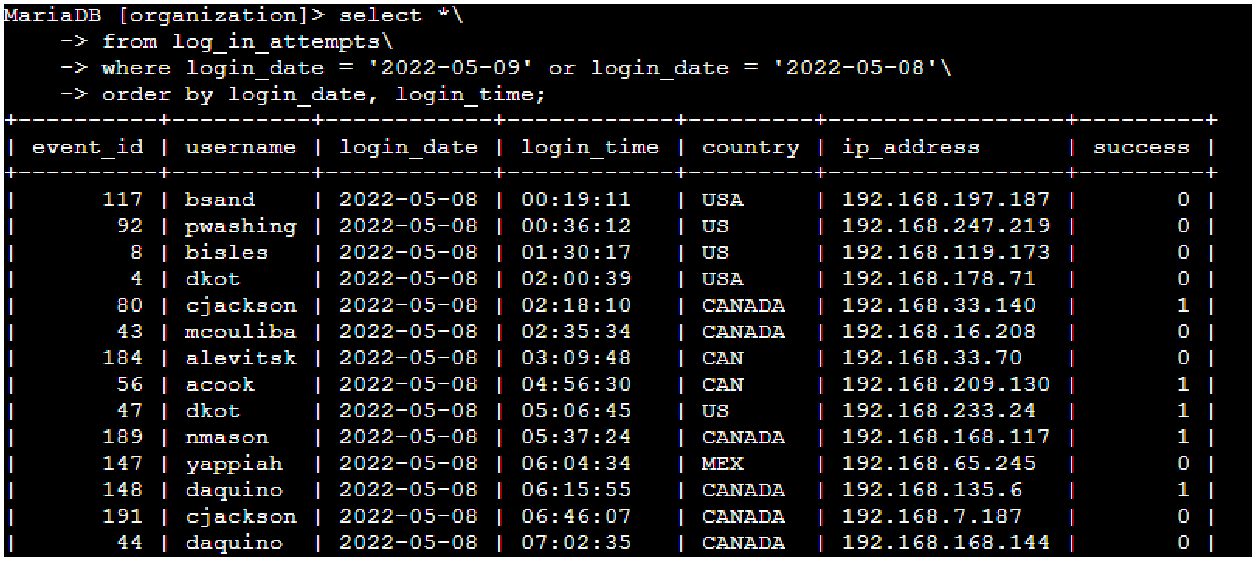
```
MariaDB [organization]> select *\
    -> from log_in_attempts\
    -> where login_time > '18:00' and success = 0\
    -> order by login_time;
+----------+----------+------------+------------+---------+-----------------+---------+
| event_id | username | login_date | login_time | country | ip_address      | success |
+----------+----------+------------+------------+---------+-----------------+---------+
|      104 | asundara | 2022-05-11 | 18:38:07   | US      | 192.168.96.200  |       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 |
|       18 | pwashing | 2022-05-11 | 19:28:50   | US      | 192.168.66.142  |       0 |
|      199 | yappiah  | 2022-05-11 | 19:34:48   | MEXICO  | 192.168.44.232  |       0 |
|       69 | wjaffrey | 2022-05-11 | 19:55:15   | USA     | 192.168.100.17  |       0 |
|      131 | bisles   | 2022-05-09 | 20:03:55   | US      | 192.168.113.171 |       0 |
|      107 | bisles   | 2022-05-12 | 20:25:57   | USA     | 192.168.116.187 |       0 |
|        2 | apatel   | 2022-05-10 | 20:27:27   | CAN     | 192.168.205.12  |       0 |
|      160 | jclark   | 2022-05-10 | 20:49:00   | CANADA  | 192.168.214.49  |       0 |
|       34 | drosas   | 2022-05-11 | 21:02:04   | US      | 192.168.45.93   |       0 |
|      127 | abellmas | 2022-05-09 | 21:20:51   | CANADA  | 192.168.70.122  |       0 |
|      111 | aestrada | 2022-05-10 | 22:00:26   | MEXICO  | 192.168.76.27   |       0 |
|       52 | cjackson | 2022-05-10 | 22:07:07   | CAN     | 192.168.58.57   |       0 |
|      155 | cgriffin | 2022-05-12 | 22:18:42   | USA     | 192.168.236.176 |       0 |
|       96 | ivelasco | 2022-05-09 | 22:36:36   | CAN     | 192.168.84.194  |       0 |
|       87 | apatel   | 2022-05-08 | 22:38:31   | CANADA  | 192.168.132.153 |       0 |
|       42 | cgriffin | 2022-05-09 | 23:04:05   | US      | 192.168.4.157   |       0 |
|       82 | abernard | 2022-05-12 | 23:38:46   | MEX     | 192.168.234.49  |       0 |
+----------+----------+------------+------------+---------+-----------------+---------+
19 rows in set (0.001 sec)

MariaDB [organization]>
```

In this query, I'm selecting all columns from the log_in_attempts table where the login_status is 'failed' and the login_time is greater than '18:00'. This will give me a list of all failed login attempts that occurred after 6:00 PM.

Retrieve login attempts on specific dates

To investigate the suspicious event that occurred on 2022-05-09, I want to review all login attempts that occurred on this day and the day before. I will use filters in SQL to create a query that identifies all login attempts that occurred on 2022-05-09 or 2022-05-08.

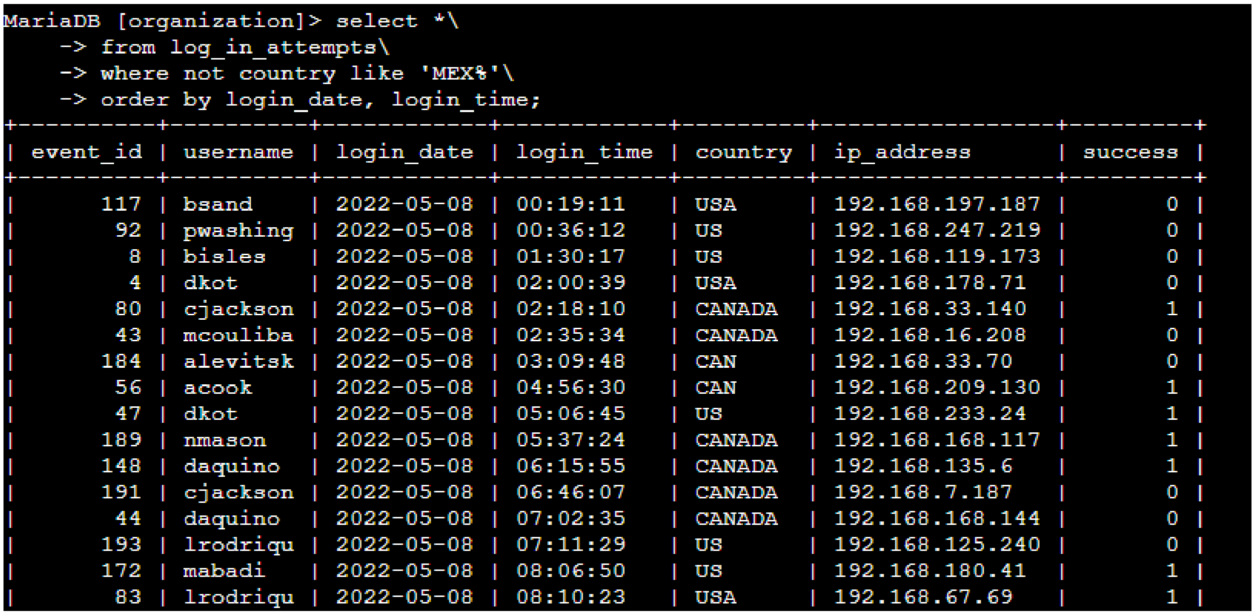


```
MariaDB [organization]> select *\
    -> from log_in_attempts\
    -> where login_date = '2022-05-09' or login_date = '2022-05-08'\
    -> order by login_date, login_time;
+----------+----------+------------+------------+---------+-----------------+---------+
| event_id | username | login_date | login_time | country | ip_address      | success |
+----------+----------+------------+------------+---------+-----------------+---------+
|      117 | bsand    | 2022-05-08 | 00:19:11   | USA     | 192.168.197.187 |       0 |
|       92 | pwashing | 2022-05-08 | 00:36:12   | US      | 192.168.247.219 |       0 |
|        8 | bisles   | 2022-05-08 | 01:30:17   | US      | 192.168.119.173 |       0 |
|        4 | dkot     | 2022-05-08 | 02:00:39   | USA     | 192.168.178.71  |       0 |
|       80 | cjackson | 2022-05-08 | 02:18:10   | CANADA  | 192.168.33.140  |       1 |
|       43 | mcouliba | 2022-05-08 | 02:35:34   | CANADA  | 192.168.16.208  |       0 |
|      184 | alevitsk | 2022-05-08 | 03:09:48   | CAN     | 192.168.33.70   |       0 |
|       56 | acook    | 2022-05-08 | 04:56:30   | CAN     | 192.168.209.130 |       1 |
|       47 | dkot     | 2022-05-08 | 05:06:45   | US      | 192.168.233.24  |       1 |
|      189 | nmason   | 2022-05-08 | 05:37:24   | CANADA  | 192.168.168.117 |       1 |
|      147 | yappiah  | 2022-05-08 | 06:04:34   | MEX     | 192.168.65.245  |       0 |
|      148 | daquino  | 2022-05-08 | 06:15:55   | CANADA  | 192.168.135.6   |       1 |
|      191 | cjackson | 2022-05-08 | 06:46:07   | CANADA  | 192.168.7.187   |       0 |
|       44 | daquino  | 2022-05-08 | 07:02:35   | CANADA  | 192.168.168.144 |       0 |
```

In this query, I'm selecting all columns from the log_in_attempts table where the login_time is greater than or equal to '2022-05-08' and less than '2022-05-10'. This will give me a list of all login attempts that occurred on 2022-05-08 and 2022-05-09.

Retrieve login attempts outside of Mexico

To investigate the suspicious activity with login attempts outside of Mexico, I will use filters in SQL to create a query that identifies all login attempts that have a location that is not in Mexico.

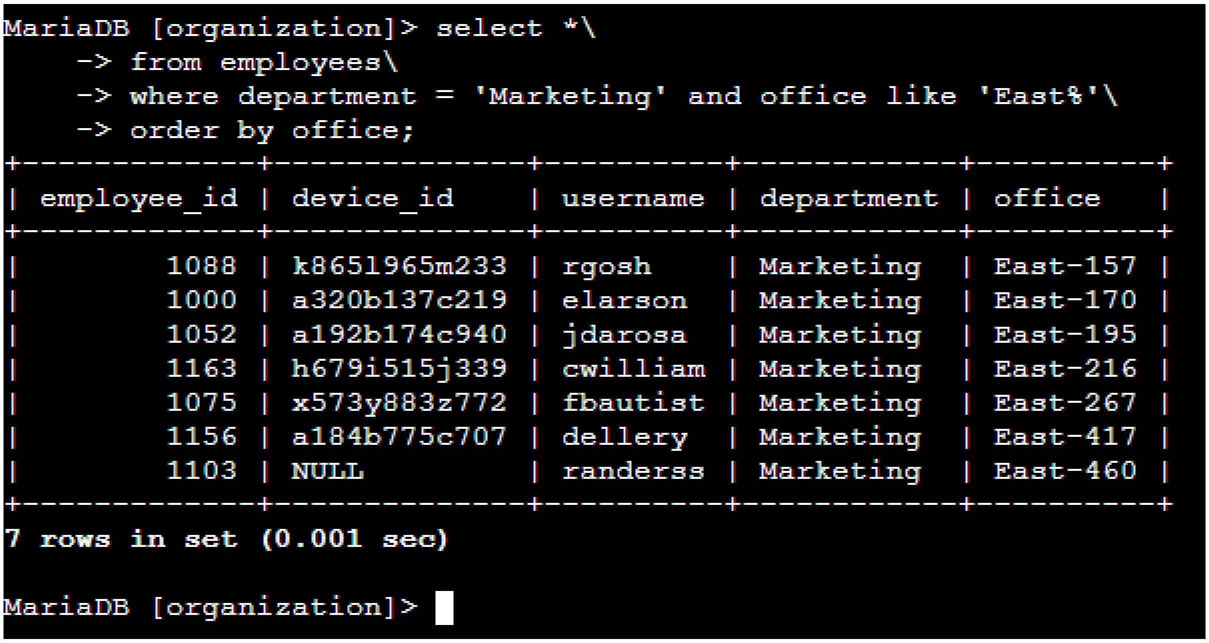


```
MariaDB [organization]> select *\
    -> from log_in_attempts\
    -> where not country like 'MEX%'\
    -> order by login_date, login_time;
+----------+----------+------------+------------+---------+-----------------+---------+
| event_id | username | login_date | login_time | country | ip_address      | success |
+----------+----------+------------+------------+---------+-----------------+---------+
|      117 | bsand    | 2022-05-08 | 00:19:11   | USA     | 192.168.197.187 |       0 |
|       92 | pwashing | 2022-05-08 | 00:36:12   | US      | 192.168.247.219 |       0 |
|        8 | bisles   | 2022-05-08 | 01:30:17   | US      | 192.168.119.173 |       0 |
|        4 | dkot     | 2022-05-08 | 02:00:39   | USA     | 192.168.178.71  |       0 |
|       80 | cjackson | 2022-05-08 | 02:18:10   | CANADA  | 192.168.33.140  |       1 |
|       43 | mcouliba | 2022-05-08 | 02:35:34   | CANADA  | 192.168.16.208  |       0 |
|      184 | alevitsk | 2022-05-08 | 03:09:48   | CAN     | 192.168.33.70   |       0 |
|       56 | acook    | 2022-05-08 | 04:56:30   | CAN     | 192.168.209.130 |       1 |
|       47 | dkot     | 2022-05-08 | 05:06:45   | US      | 192.168.233.24  |       1 |
|      189 | nmason   | 2022-05-08 | 05:37:24   | CANADA  | 192.168.168.117 |       1 |
|      148 | daquino  | 2022-05-08 | 06:15:55   | CANADA  | 192.168.135.6   |       1 |
|      191 | cjackson | 2022-05-08 | 06:46:07   | CANADA  | 192.168.7.187   |       0 |
|       44 | daquino  | 2022-05-08 | 07:02:35   | CANADA  | 192.168.168.144 |       0 |
|      193 | lrodriqu | 2022-05-08 | 07:11:29   | US      | 192.168.125.240 |       0 |
|      172 | mabadi   | 2022-05-08 | 08:06:50   | US      | 192.168.180.41  |       1 |
|       83 | lrodriqu | 2022-05-08 | 08:10:23   | USA     | 192.168.67.69   |       1 |
```

In this query, I'm selecting all columns from the log_in_attempts table where the location is not 'Mexico' (MEX). This will give me a list of all login attempts that occurred outside of Mexico, helping me investigate the suspicious activity.

Retrieve employees in Marketing

My team wants to perform security updates on specific employee machines in the Marketing department. My responsible for getting information on these employee machines and will need to query the employees table. Use filters in SQL to create a query that identifies all employees in the Marketing department for all offices in the East building.



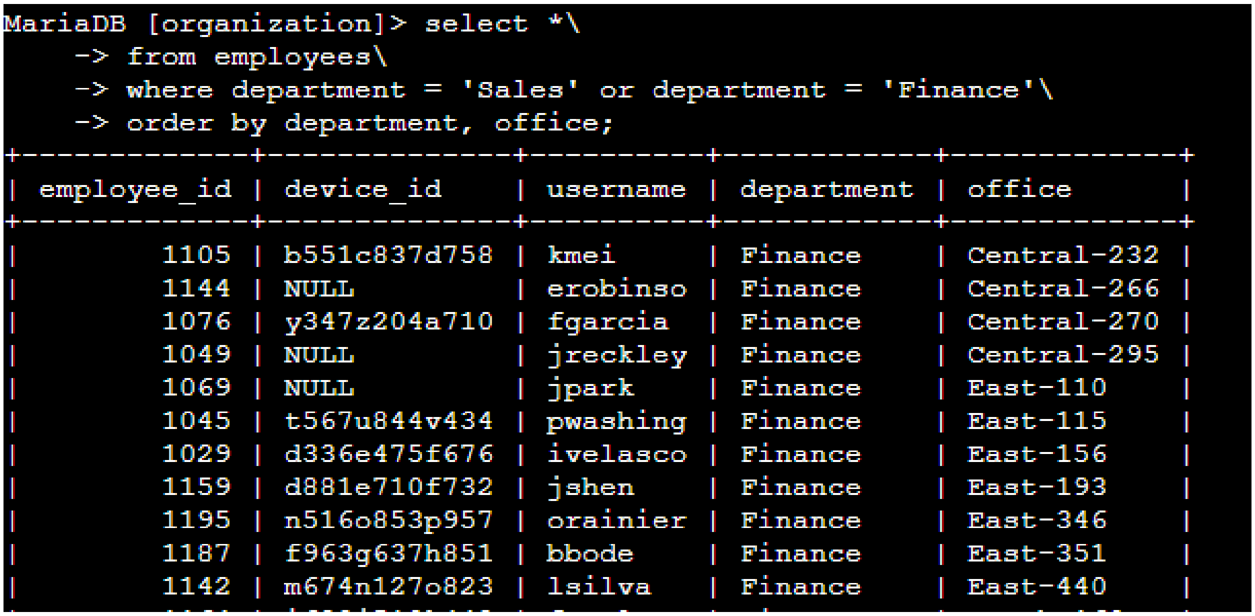
```
MariaDB [organization]> select *\
    -> from employees\
    -> where department = 'Marketing' and office like 'East%'\
    -> order by office;
+-------------+--------------+----------+------------+----------+
| employee_id | device_id    | username | department | office   |
+-------------+--------------+----------+------------+----------+
|        1088 | k8651965m233 | rgosh    | Marketing  | East-157 |
|        1000 | a320b137c219 | elarson  | Marketing  | East-170 |
|        1052 | a192b174c940 | jdarosa  | Marketing  | East-195 |
|        1163 | h679i515j339 | cwilliam | Marketing  | East-216 |
|        1075 | x573y883z772 | fbautist | Marketing  | East-267 |
|        1156 | a184b775c707 | dellery  | Marketing  | East-417 |
|        1103 | NULL         | randerss | Marketing  | East-460 |
+-------------+--------------+----------+------------+----------+
7 rows in set (0.001 sec)

MariaDB [organization]>
```

In this query, I'm selecting all columns from the employees table where the department is 'Marketing' and the office starts with 'East'. This will give me a list of all employees in the Marketing department working in offices located in the East building, allowing me to perform security updates on their machines.

Retrieve employees in Finance or Sales

Your team now needs to perform a different security update on machines for employees in the Sales and Finance departments. Use filters in SQL to create a query that identifies all employees in the Sales or Finance departments.

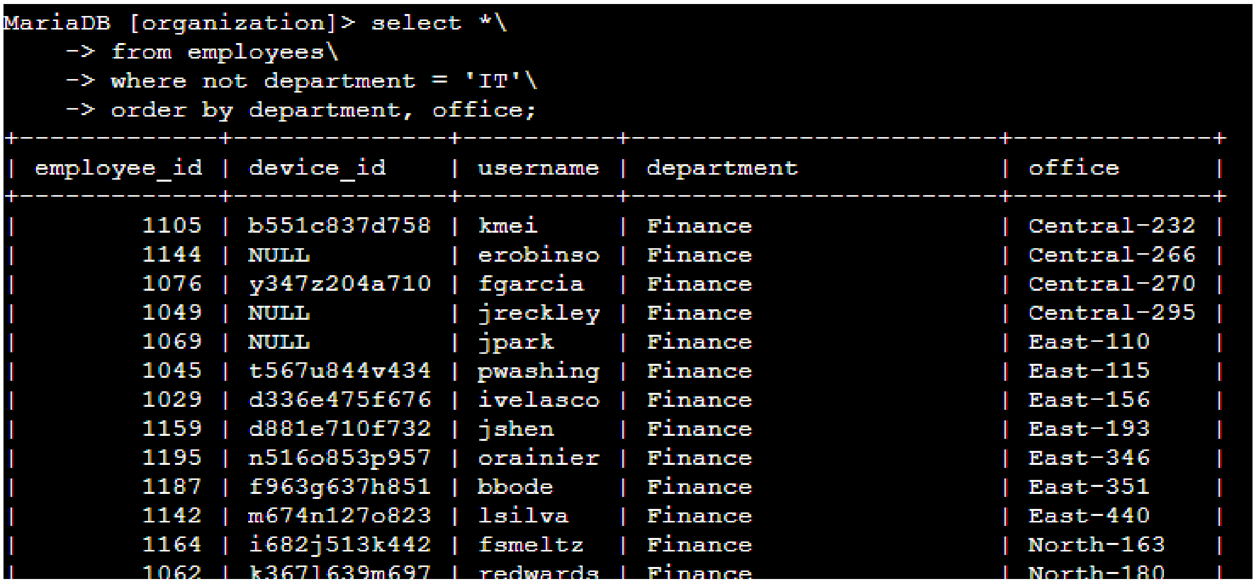


```
MariaDB [organization]> select *\
    -> from employees\
    -> where department = 'Sales' or department = 'Finance'\
    -> order by department, office;
+-------------+---------------+----------+------------+-------------+
| employee_id | device_id     | username | department | office      |
+-------------+---------------+----------+------------+-------------+
|        1105 | b551c837d758  | kmei     | Finance    | Central-232 |
|        1144 | NULL          | erobinso | Finance    | Central-266 |
|        1076 | y347z204a710  | fgarcia  | Finance    | Central-270 |
|        1049 | NULL          | jreckley | Finance    | Central-295 |
|        1069 | NULL          | jpark    | Finance    | East-110    |
|        1045 | t567u844v434  | pwashing | Finance    | East-115    |
|        1029 | d336e475f676  | ivelasco | Finance    | East-156    |
|        1159 | d881e710f732  | jshen    | Finance    | East-193    |
|        1195 | n516o853p957  | orainier | Finance    | East-346    |
|        1187 | f963g637h851  | bbode    | Finance    | East-351    |
|        1142 | m674n127o823  | lsilva   | Finance    | East-440    |
```

In this query, I'm selecting all columns from the employees table where the department is either 'Sales' or 'Finance'. This will give me a list of all employees in the Sales or Finance departments, and I can proceed with the necessary security updates on their machines.

Retrieve all employees not in IT

Your team needs to make one more update to employee machines. The employees who are in the Information Technology department already had this update, but employees in all other departments need it.



```
MariaDB [organization]> select *\
    -> from employees\
    -> where not department = 'IT'\
    -> order by department, office;
+-------------+---------------+----------+-----------------------+-------------+
| employee_id | device_id     | username | department            | office      |
+-------------+---------------+----------+-----------------------+-------------+
|        1105 | b551c837d758  | kmei     | Finance               | Central-232 |
|        1144 | NULL          | erobinso | Finance               | Central-266 |
|        1076 | y347z204a710  | fgarcia  | Finance               | Central-270 |
|        1049 | NULL          | jreckley | Finance               | Central-295 |
|        1069 | NULL          | jpark    | Finance               | East-110    |
|        1045 | t567u844v434  | pwashing | Finance               | East-115    |
|        1029 | d336e475f676  | ivelasco | Finance               | East-156    |
|        1159 | d881e710f732  | jshen    | Finance               | East-193    |
|        1195 | n516o853p957  | orainier | Finance               | East-346    |
|        1187 | f963g637h851  | bbode    | Finance               | East-351    |
|        1142 | m674n127o823  | lsilva   | Finance               | East-440    |
|        1164 | i682j513k442  | fsmeltz  | Finance               | North-163   |
|        1062 | k3671639m697  | redwards | Finance               | North-180   |
```

In this query, I'm selecting all columns from the employees table except the IT department. This will give me a list of all employees in the departments, and I can proceed with the necessary security updates on their machines.

Summary

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