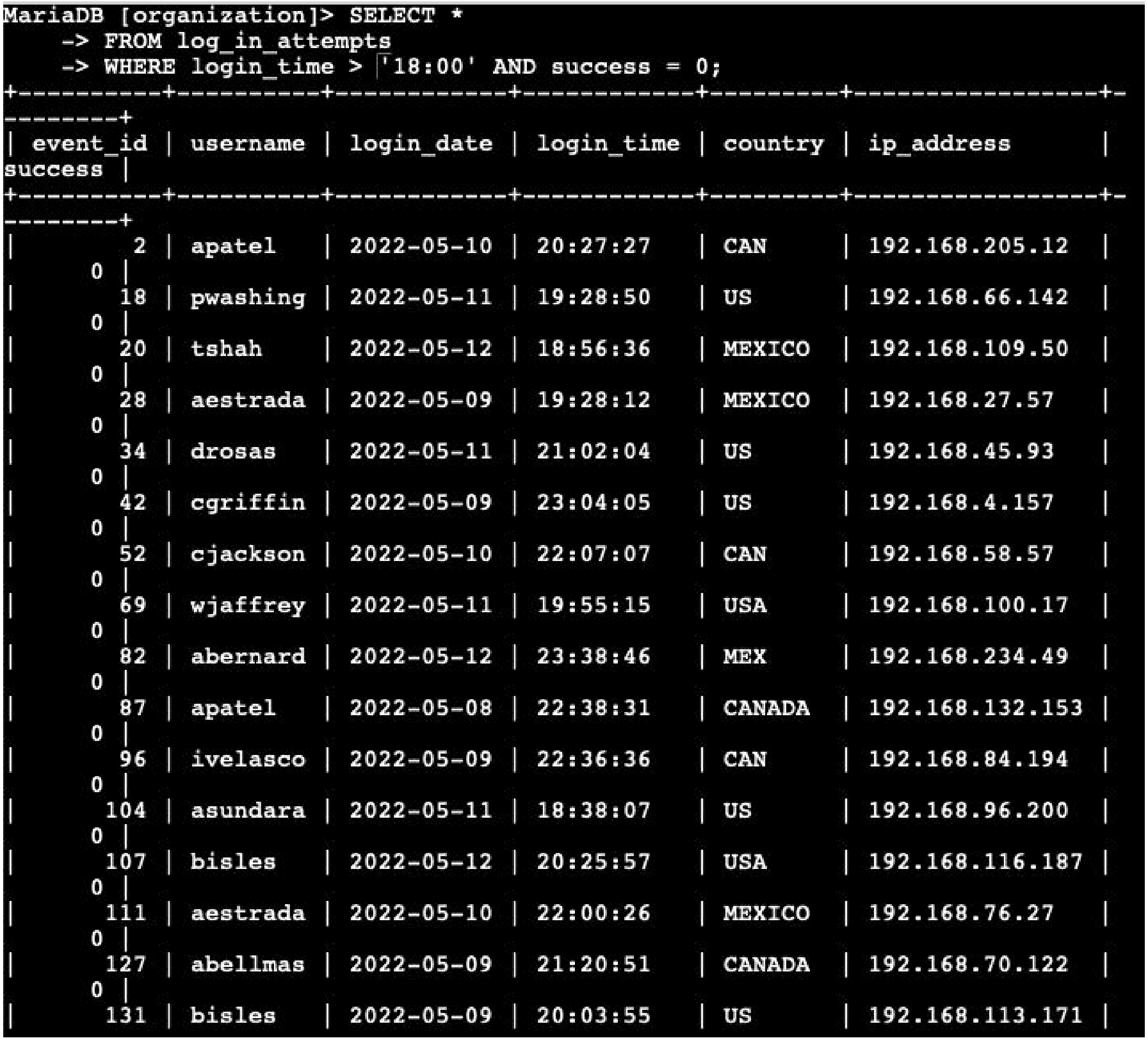
Apply filters to SQL queries

Project description

[I will be using SQL to search for information on employees, and their login attempts. I will be using the various SQL Clauses, like the AND, OR, and LIKE, to name a few. I will be adding these clauses to narrow down my searches.]

Retrieve after hours failed login attempts

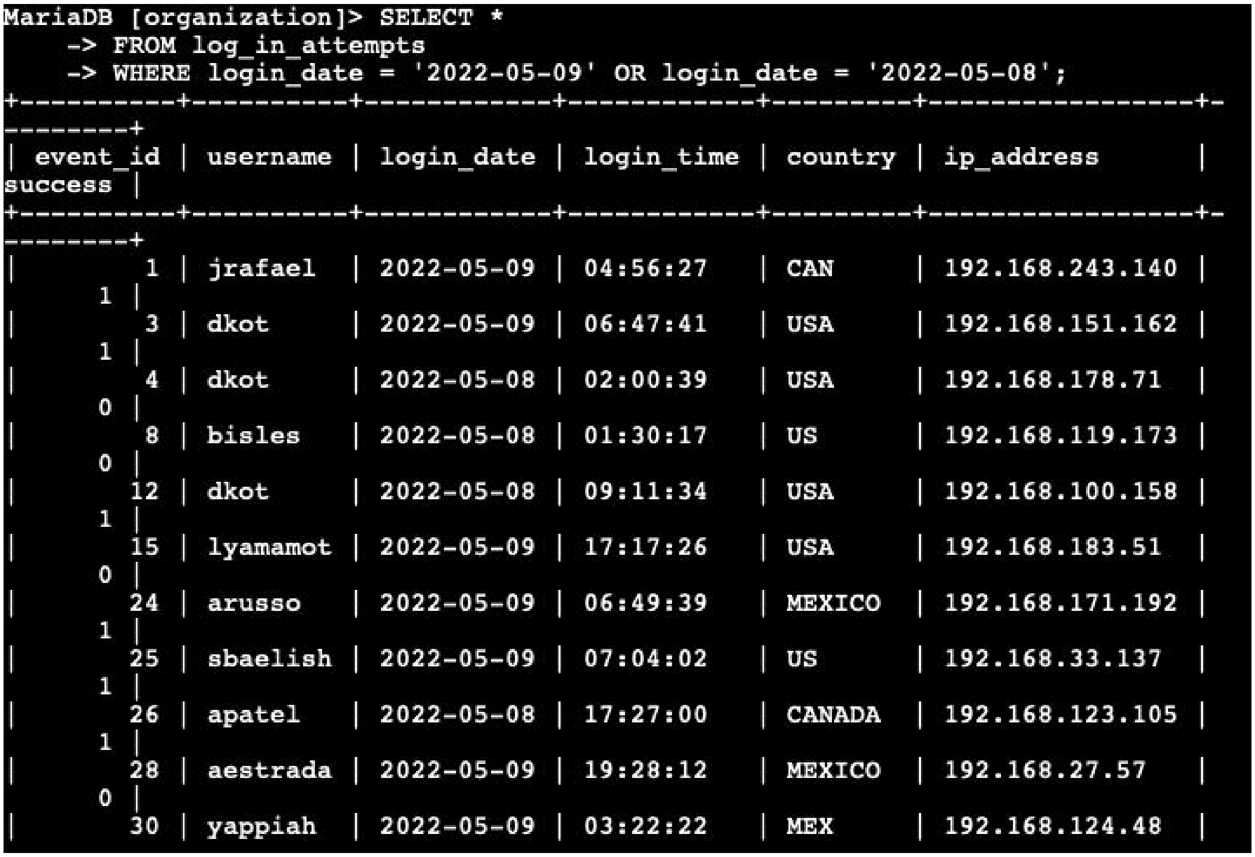
[



```
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 |
```

] I used the SELECT * (all) asterisk from the log_in_attempts table. Then, I narrowed down the search by using the WHERE Clause. I searched for login_times that took place (>) after 18:00 hours. Lastly, I also added a search criteria to check for all failed login attempts by adding a search clause using the AND operator to combine a search through the success table. I used 0 to represent all failed login attempts.

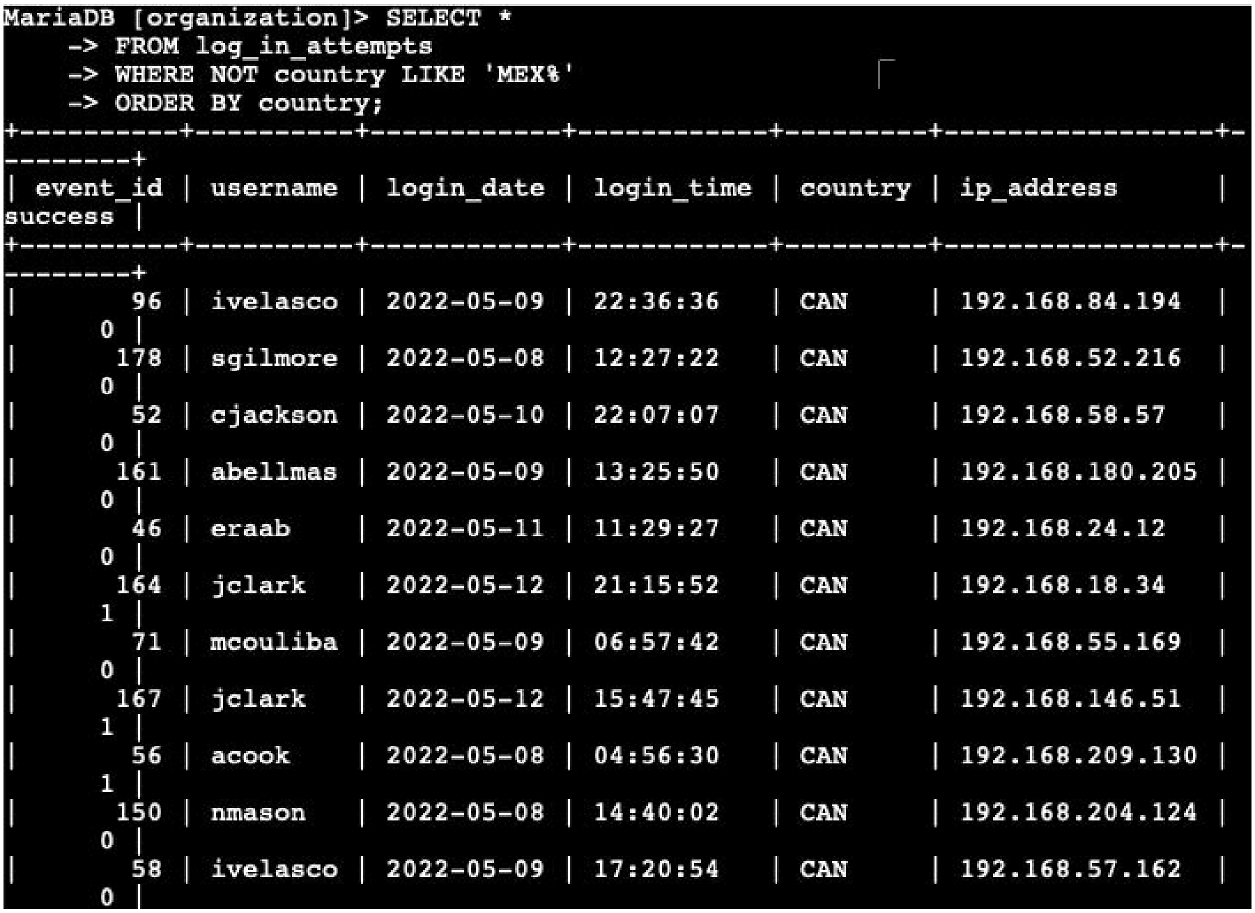
Retrieve login attempts on specific dates



```
MariaDB [organization]> SELECT *
    -> FROM log_in_attempts
    -> WHERE login_date = '2022-05-09' OR login_date = '2022-05-08';
+----------+----------+------------+------------+---------+-----------------+-
--------+
| 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 |
|       12 | dkot     | 2022-05-08 | 09:11:34   | USA     | 192.168.100.158 |
      1 |
|       15 | lyamamot | 2022-05-09 | 17:17:26   | USA     | 192.168.183.51  |
      0 |
|       24 | arusso   | 2022-05-09 | 06:49:39   | MEXICO  | 192.168.171.192 |
      1 |
|       25 | sbaelish | 2022-05-09 | 07:04:02   | US      | 192.168.33.137  |
      1 |
|       26 | apatel   | 2022-05-08 | 17:27:00   | CANADA  | 192.168.123.105 |
      1 |
|       28 | aestrada | 2022-05-09 | 19:28:12   | MEXICO  | 192.168.27.57   |
      0 |
|       30 | yappiah  | 2022-05-09 | 03:22:22   | MEX     | 192.168.124.48  |
```

I used the SELECT *(all) from the log_in_attempts table and to narrow down the search to the specific aforementioned dates. I used the WHERE clause and input the date that needed to be searched. In conjunction, I added the OR clause to search the database for another date, too.

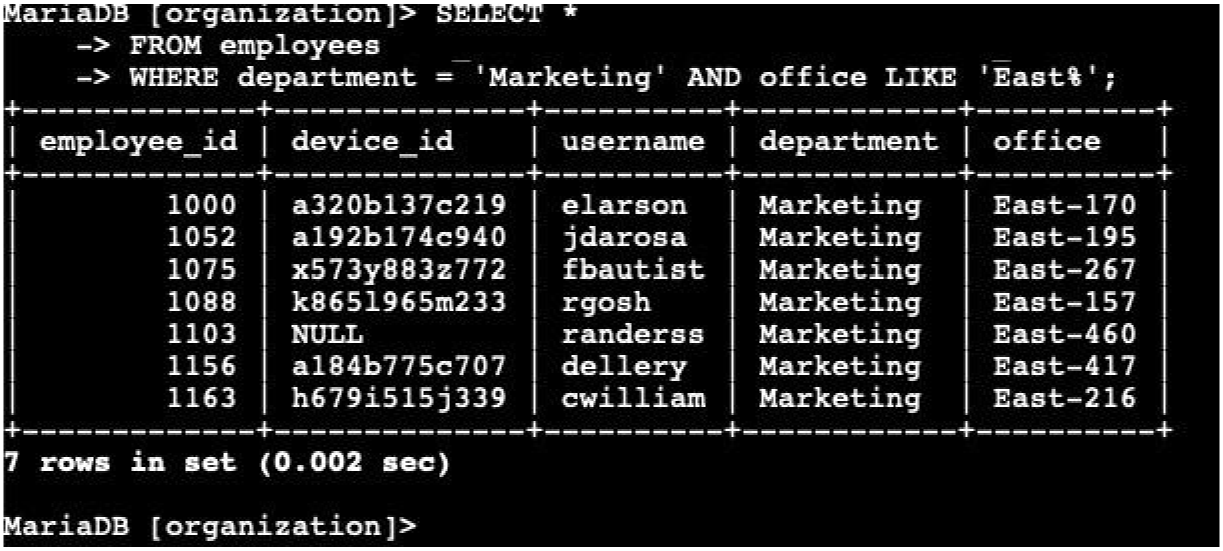
Retrieve login attempts outside of Mexico



```
MariaDB [organization]> SELECT *
    -> FROM log_in_attempts
    -> WHERE NOT country LIKE 'MEX%'
    -> ORDER BY country;
+----------+----------+------------+------------+---------+-----------------+-
--------+
| event_id | username | login_date | login_time | country | ip_address      |
success |
+----------+----------+------------+------------+---------+-----------------+-
--------+
|       96 | ivelasco | 2022-05-09 | 22:36:36   | CAN     | 192.168.84.194  |
      0 |
|      178 | sgilmore | 2022-05-08 | 12:27:22   | CAN     | 192.168.52.216  |
      0 |
|       52 | cjackson | 2022-05-10 | 22:07:07   | CAN     | 192.168.58.57   |
      0 |
|      161 | abellmas | 2022-05-09 | 13:25:50   | CAN     | 192.168.180.205 |
      0 |
|       46 | eraab    | 2022-05-11 | 11:29:27   | CAN     | 192.168.24.12   |
      0 |
|      164 | jclark   | 2022-05-12 | 21:15:52   | CAN     | 192.168.18.34   |
      1 |
|       71 | mcouliba | 2022-05-09 | 06:57:42   | CAN     | 192.168.55.169  |
      0 |
|      167 | jclark   | 2022-05-12 | 15:47:45   | CAN     | 192.168.146.51  |
      1 |
|       56 | acook    | 2022-05-08 | 04:56:30   | CAN     | 192.168.209.130 |
      1 |
|      150 | nmason   | 2022-05-08 | 14:40:02   | CAN     | 192.168.204.124 |
      0 |
|       58 | ivelasco | 2022-05-09 | 17:20:54   | CAN     | 192.168.57.162  |
      0 |
```

I used the SELECT *(all) option from the log_in_attempts table. I narrowed down the search using the WHERE clause. Additionally, to only search for countries that did not include Mexico, I used the NOT clause in conjunction with the WHERE. Because Mexico had been added into the database as either fully spelled out(MEXICO) or as three letter(MEX), I used the LIKE clause with a (%) wildcard to handle searching for all countries that did not include Mexico. Lastly, I used the ORDER BY clause to set the countries in ascending order.

Retrieve employees in Marketing

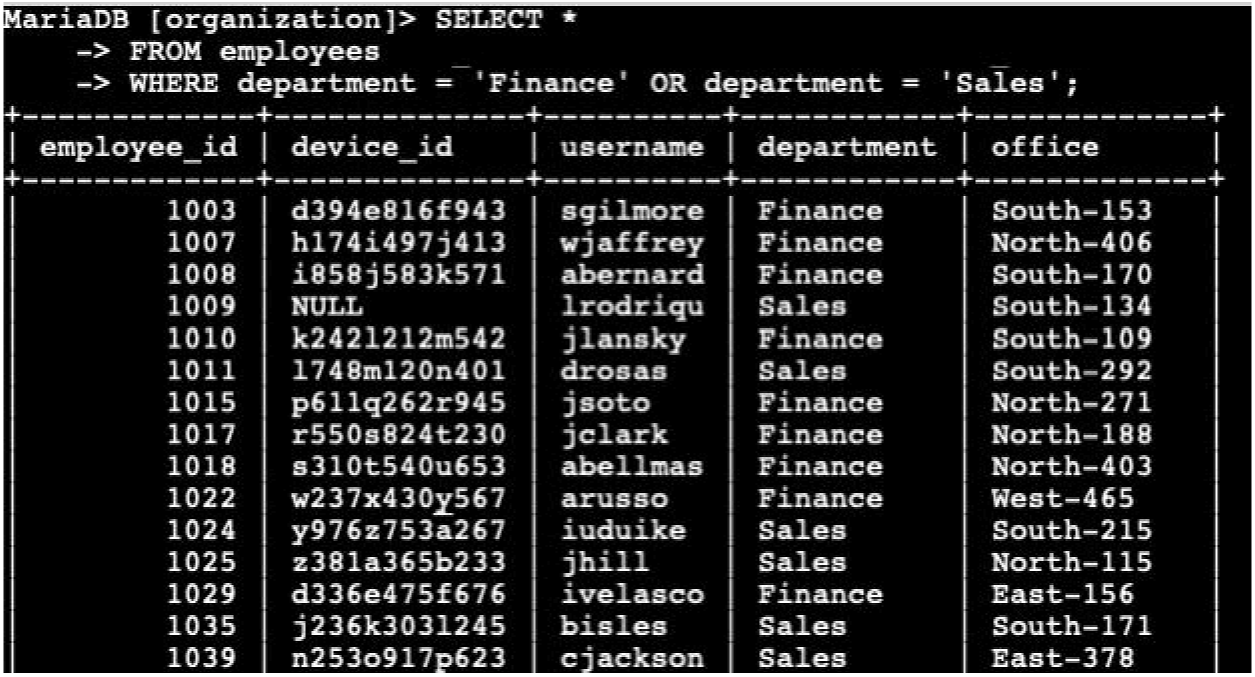


```
MariaDB [organization]> SELECT *
    -> FROM employees
    -> WHERE department = 'Marketing' AND office LIKE 'East%';
+-------------+--------------+-----------+------------+----------+
| employee_id | device_id    | username  | department | office   |
+-------------+--------------+-----------+------------+----------+
|        1000 | a320b137c219 | elarson   | Marketing  | East-170 |
|        1052 | a192b174c940 | jdarosa   | Marketing  | East-195 |
|        1075 | x573y883z772 | fbautist  | Marketing  | East-267 |
|        1088 | k8651965m233 | rgosh     | Marketing  | East-157 |
|        1103 | NULL         | randerss  | Marketing  | East-460 |
|        1156 | a184b775c707 | dellery   | Marketing  | East-417 |
|        1163 | h679i515j339 | cwilliam  | Marketing  | East-216 |
+-------------+--------------+-----------+------------+----------+
7 rows in set (0.002 sec)

MariaDB [organization]>
```

I used the SELECT *(all) option from the employees table, and used the WHERE clause to narrow down my search for employees in the marketing department. I also used the AND operator to include a search in the office table. Because I had to find employees that worked in different offices, I used the LIKE clause and a (%) wildcard to search for all east buildings.

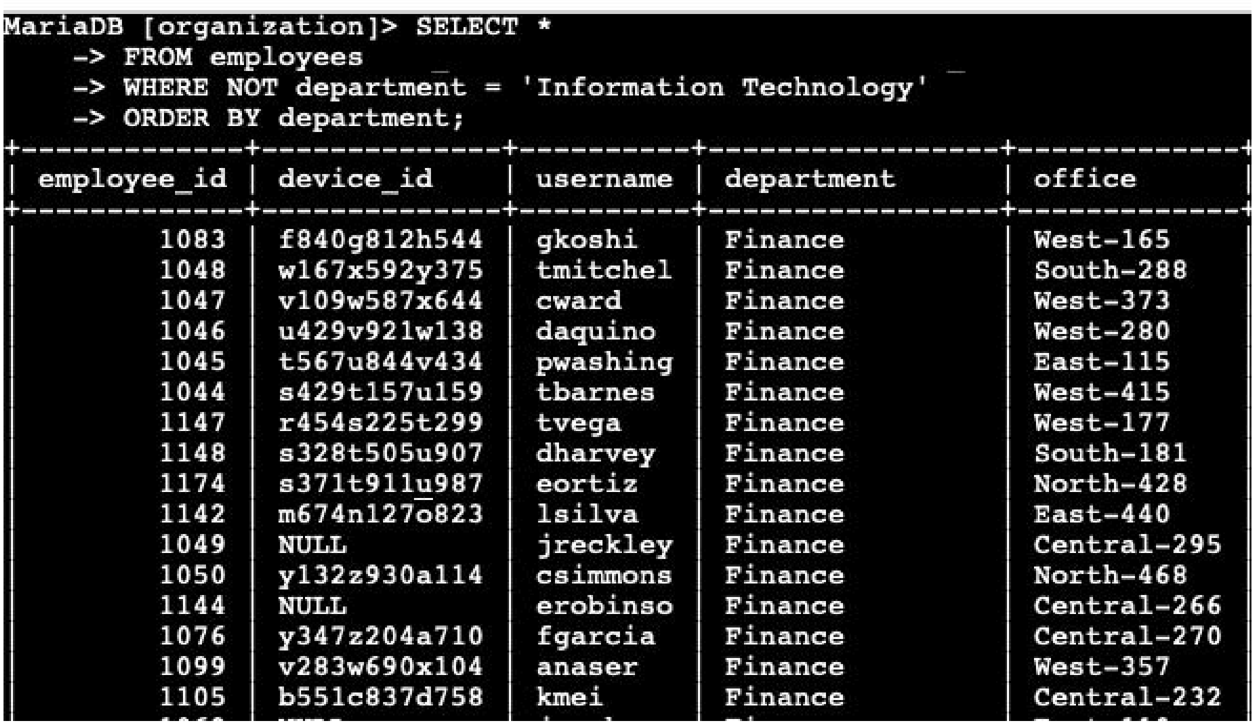
Retrieve employees in Finance or Sales



```
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   |
|        1010 | k2421212m542 | jlansky   | Finance    | South-109   |
|        1011 | 1748m120n401 | drosas    | Sales      | South-292   |
|        1015 | p611q262r945 | jsoto     | Finance    | North-271   |
|        1017 | r550s824t230 | jclark    | Finance    | North-188   |
|        1018 | s310t540u653 | abellmas  | Finance    | North-403   |
|        1022 | w237x430y567 | arusso    | Finance    | West-465    |
|        1024 | y976z753a267 | iuduike   | Sales      | South-215   |
|        1025 | z381a365b233 | jhill     | Sales      | North-115   |
|        1029 | d336e475f676 | ivelasco  | Finance    | East-156    |
|        1035 | j236k3031245 | bisles    | Sales      | South-171   |
|        1039 | n253o917p623 | cjackson  | Sales      | East-378    |
```

I used the SELECT *(all) option from the employees table. To narrow down the search for employees that worked in the sales or finance department, I used the WHERE clause. Additionally, to search for both departments, the OR clause had also been used in conjunction with the WHERE clause.

Retrieve all employees not in IT



```
MariaDB [organization]> SELECT *
    -> FROM employees
    -> WHERE NOT department = 'Information Technology'
    -> ORDER BY department;
+-------------+--------------+----------+-----------------+------------+
| employee_id | device_id    | username | department      | office     |
+-------------+--------------+----------+-----------------+------------+
|        1083 | f840g812h544 | gkoshi   | Finance         | West-165   |
|        1048 | w167x592y375 | tmitchel | Finance         | South-288  |
|        1047 | v109w587x644 | cward    | Finance         | West-373   |
|        1046 | u429v921w138 | daquino  | Finance         | West-280   |
|        1045 | t567u844v434 | pwashing | Finance         | East-115   |
|        1044 | s429t157u159 | tbarnes  | Finance         | West-415   |
|        1147 | r454s225t299 | tvega    | Finance         | West-177   |
|        1148 | s328t505u907 | dharvey  | Finance         | South-181  |
|        1174 | s371t911u987 | eortiz   | Finance         | North-428  |
|        1142 | m674n127o823 | lsilva   | Finance         | East-440   |
|        1049 | NULL         | jreckley | Finance         | Central-295|
|        1050 | y132z930a114 | csimmons | Finance         | North-468  |
|        1144 | NULL         | erobinso | Finance         | Central-266|
|        1076 | y347z204a710 | fgarcia  | Finance         | Central-270|
|        1099 | v283w690x104 | anaser   | Finance         | West-357   |
|        1105 | b551c837d758 | kmei     | Finance         | Central-232|
```

I used the SELECT*(all) option from the employees table. To narrow down the search, I used the WHERE clause. Because I was searching for all departments that were not IT, I used the NOT clause.

Summary

In summation, I used many different clauses and operators to navigate through the SQL database. I was able to search for specific login times, and login dates. I used my knowledge to narrow down my searches for countries, and employees.