Task 1. Retrieve after hours failed login attempts

Your team is investigating failed login attempts that were made after business hours. You want to retrieve this information from the login activity. You'll identify all unsuccessful attempts after 18:00.

The `login_time` column in the `log_in_attempts` table contains information on when login attempts were made. Office hours end at `'18:00'`.

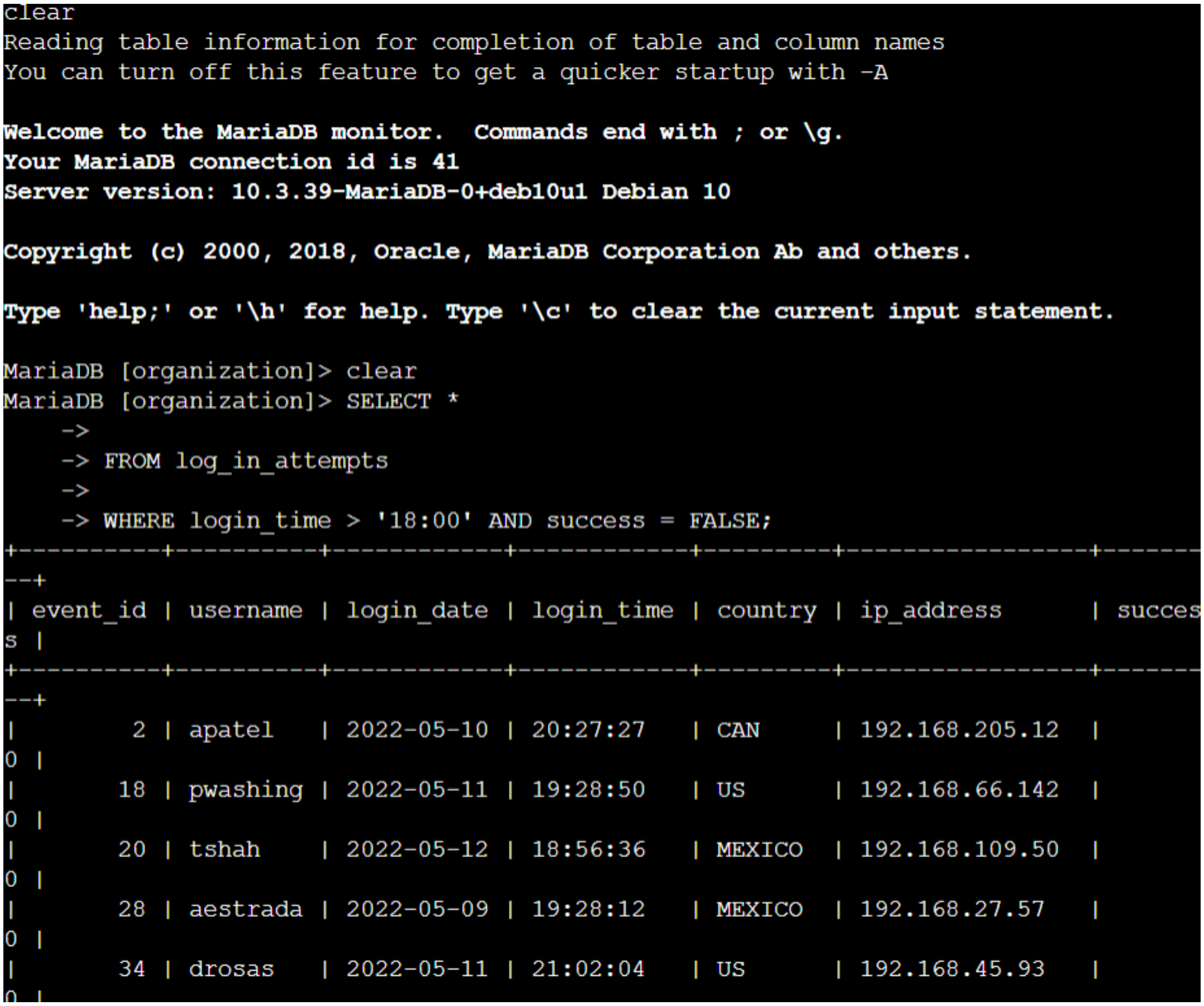
The `success` column in the `log_in_attempts` table contains values of `TRUE` or `FALSE` to indicate whether the login was successful. MySQL stores Boolean values as `1` for `TRUE`, and `0` for `FALSE`. This means that `TRUE` is represented as `1`, and `FALSE` represented as `0` in the `success` column.

- Use the `AND` operator to retrieve the failed login attempts that occurred after business hours. Replace the `X` and `Y` with the correct values to filter for the records you need:

***Note:** Values of `TRUE` and `FALSE` are not placed in single quotes because they are not string data. They are Boolean data, which is another data type.*

The command to complete this step:

```sql
SELECT *
FROM log_in_attempts
WHERE login_time > '18:00' AND success = FALSE;
```



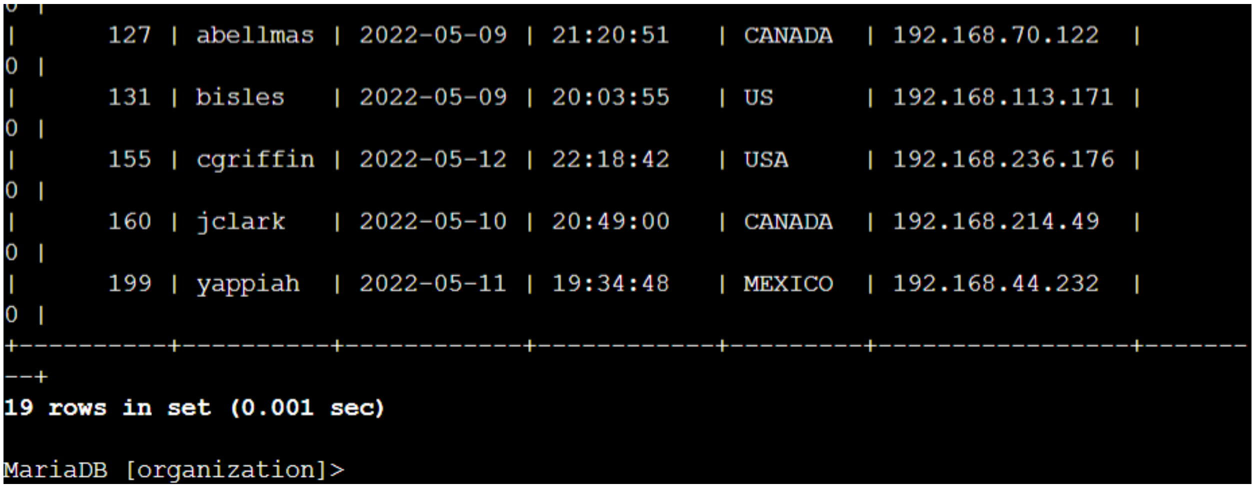
```
clear
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 41
Server version: 10.3.39-MariaDB-0+deb10u1 Debian 10

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [organization]> clear
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      | succes
s |
+----------+----------+------------+------------+---------+-----------------+-------
--+
|        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 |
```



```
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 (0.001 sec)

MariaDB [organization]>
```

Task 2: Retrieve login attempts on specific dates

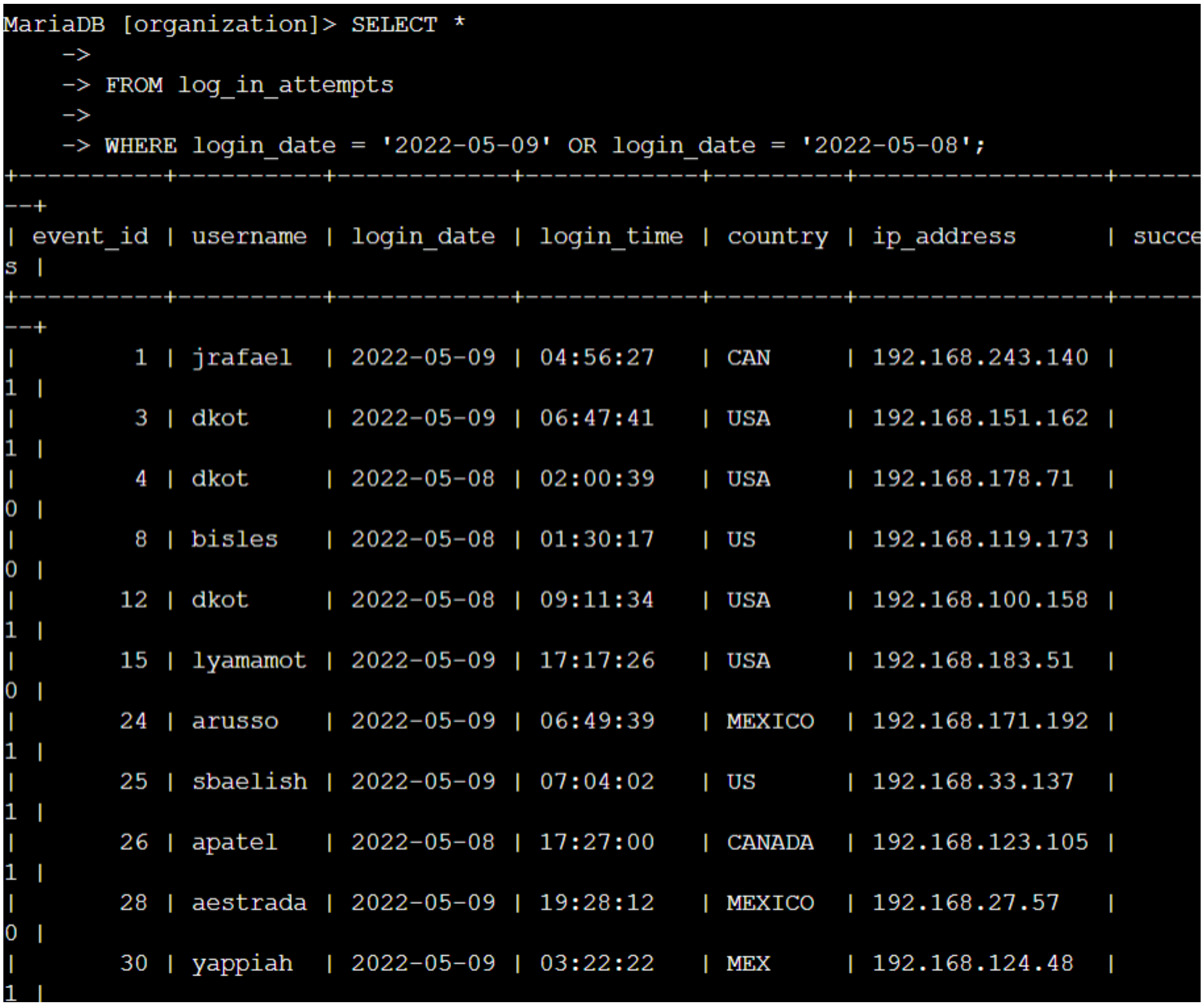
Your team is investigating a suspicious event that occurred on `'2022-05-09'`. You want to retrieve all login attempts that occurred on this day and the day before (`'2022-05-08'`).

The `login_date` column in the `log_in_attempts` table contains information on the dates when login attempts were made.

- Use the `OR` operator to retrieve the failed login attempts on the specified days. Replace the `X` and `Y` with the correct values to filter for the records you need:

The correct query to solve this step:

```
SELECT *
FROM log_in_attempts
WHERE login_date = '2022-05-09' OR login_date = '2022-05-08';
```



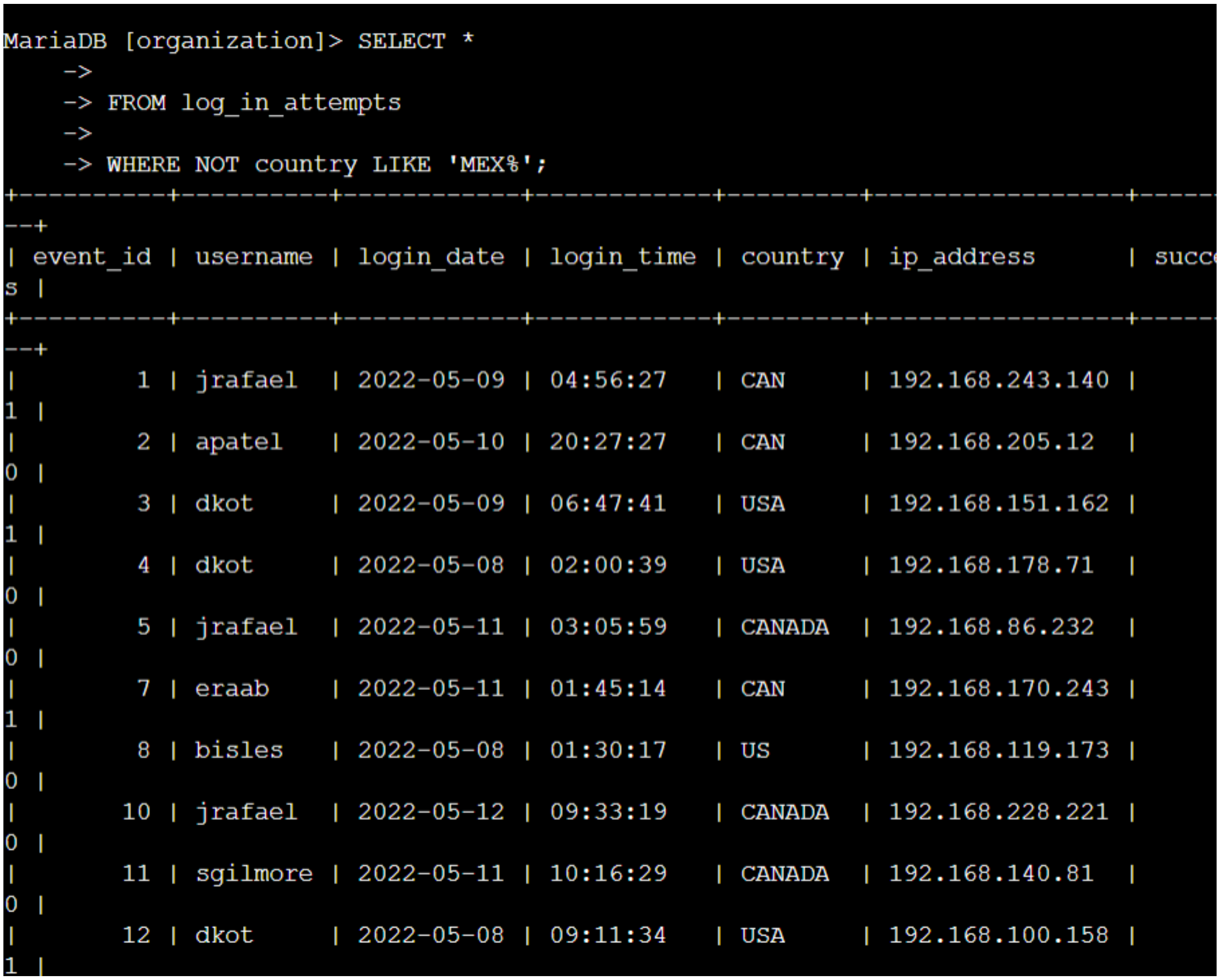
```
MariaDB [organization]> SELECT *
    ->
    -> FROM log_in_attempts
    ->
    -> WHERE login_date = '2022-05-09' OR login_date = '2022-05-08';
+----------+----------+------------+------------+---------+-----------------+------
--+
| event_id | username | login_date | login_time | country | ip_address      | succe
s |
+----------+----------+------------+------------+---------+-----------------+------
--+
|        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  |
1 |
```

Task 3. Retrieve login attempts outside of Mexico

Now, your team is investigating logins that did not originate in Mexico, and you need to find this information. Note that the country field includes entries with `'MEX'` and `'MEXICO'`. You should use the `NOT` and `LIKE` operators and the matching pattern `'MEX%'`.

- Run the following SQL query to retrieve login attempts that did not originate in Mexico. Replace `X` with the correct operator and `Y` with the correct pattern to filter for the information you need:

```
SELECT *
FROM log_in_attempts
WHERE NOT country LIKE 'MEX%';
```



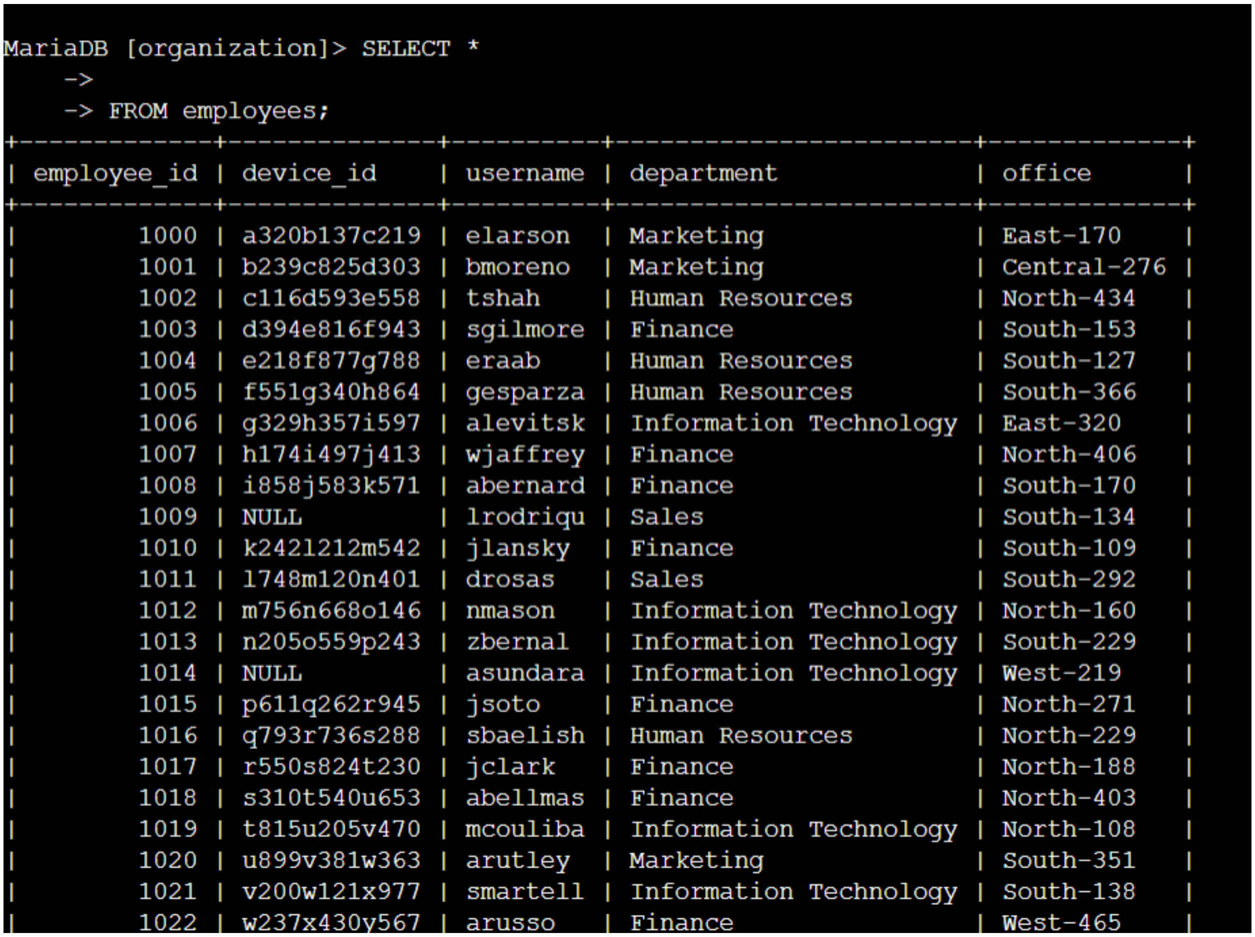
```
MariaDB [organization]> SELECT *
    ->
    -> FROM log_in_attempts
    ->
    -> WHERE NOT country LIKE 'MEX%';
+----------+----------+------------+------------+---------+-----------------+------
--+
| event_id | username | login_date | login_time | country | ip_address      | succe
s |
+----------+----------+------------+------------+---------+-----------------+------
--+
|        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 |
|        5 | jrafael  | 2022-05-11 | 03:05:59   | CANADA  | 192.168.86.232  |
0 |
|        7 | eraab    | 2022-05-11 | 01:45:14   | CAN     | 192.168.170.243 |
1 |
|        8 | bisles   | 2022-05-08 | 01:30:17   | US      | 192.168.119.173 |
0 |
|       10 | jrafael  | 2022-05-12 | 09:33:19   | CANADA  | 192.168.228.221 |
0 |
|       11 | sgilmore | 2022-05-11 | 10:16:29   | CANADA  | 192.168.140.81  |
0 |
|       12 | dkot     | 2022-05-08 | 09:11:34   | USA     | 192.168.100.158 |
1 |
```

Task 4. Retrieve employees in Marketing

For tasks 4, 5 and 6 you need to retrieve the information from the `department` and `office` columns in the `employees` table.

You can run the following SQL query if you need to view the columns and values in the `employees` table:

```
SELECT *
FROM employees;
```



```
MariaDB [organization]> SELECT *
    ->
    -> FROM employees;
+-------------+--------------+----------+------------------------+-------------+
| 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   |
|        1004 | e218f877g788 | eraab    | Human Resources        | South-127   |
|        1005 | f551g340h864 | gesparza | Human Resources        | South-366   |
|        1006 | g329h357i597 | alevitsk | Information Technology | East-320    |
|        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   |
|        1012 | m756n668o146 | nmason   | Information Technology | North-160   |
|        1013 | n205o559p243 | zbernal  | Information Technology | South-229   |
|        1014 | NULL         | asundara | Information Technology | West-219    |
|        1015 | p611q262r945 | jsoto    | Finance                | North-271   |
|        1016 | q793r736s288 | sbaelish | Human Resources        | North-229   |
|        1017 | r550s824t230 | jclark   | Finance                | North-188   |
|        1018 | s310t540u653 | abellmas | Finance                | North-403   |
|        1019 | t815u205v470 | mcouliba | Information Technology | North-108   |
|        1020 | u899v381w363 | arutley  | Marketing              | South-351   |
|        1021 | v200w121x977 | smartell | Information Technology | South-138   |
|        1022 | w237x430y567 | arusso   | Finance                | West-465    |
```

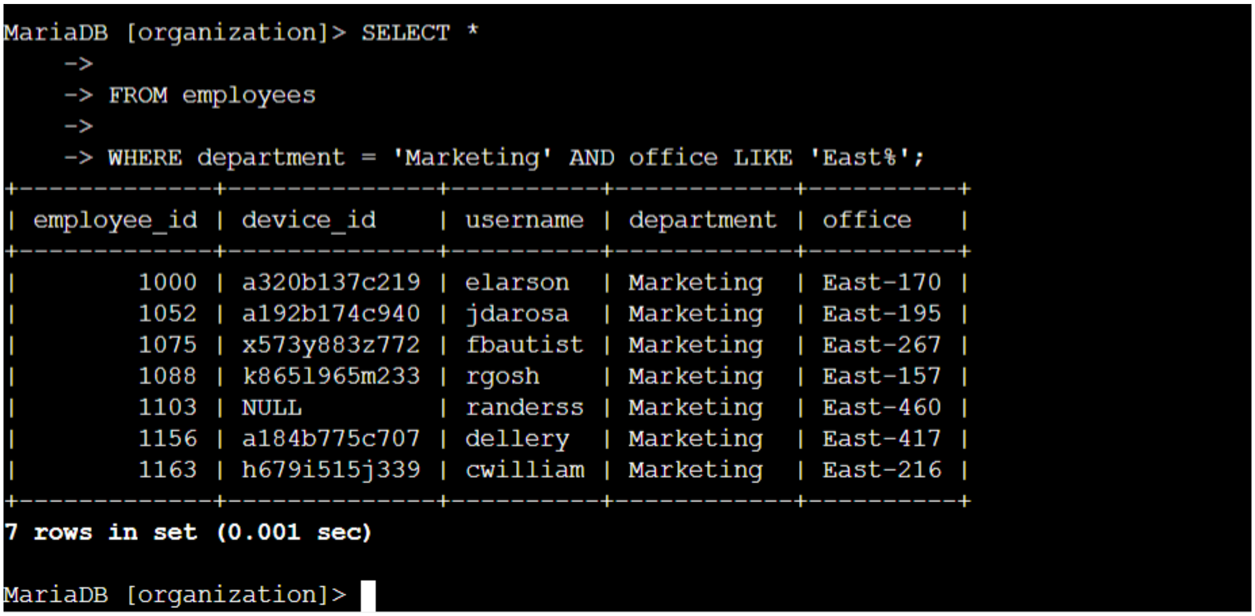
Your team is updating employee machines, and you need to obtain the information about employees in the `'Marketing'` department who are located in all offices in the East building (such as `'East-170'` or `'East-320'`).

- Write a SQL query to retrieve this information from the `employees` table. Select all columns and include filters on the `department` and `office` columns to return only the needed records.

Note: *You'll need to use the* `AND` *and* `LIKE` *operators to satisfy both of these criteria.*

The correct query to solve this step:

```
SELECT *
FROM employees
WHERE department = 'Marketing' AND office LIKE 'East%';
```



```
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 (0.001 sec)

MariaDB [organization]>
```

Task 5. Retrieve employees in Finance or Sales

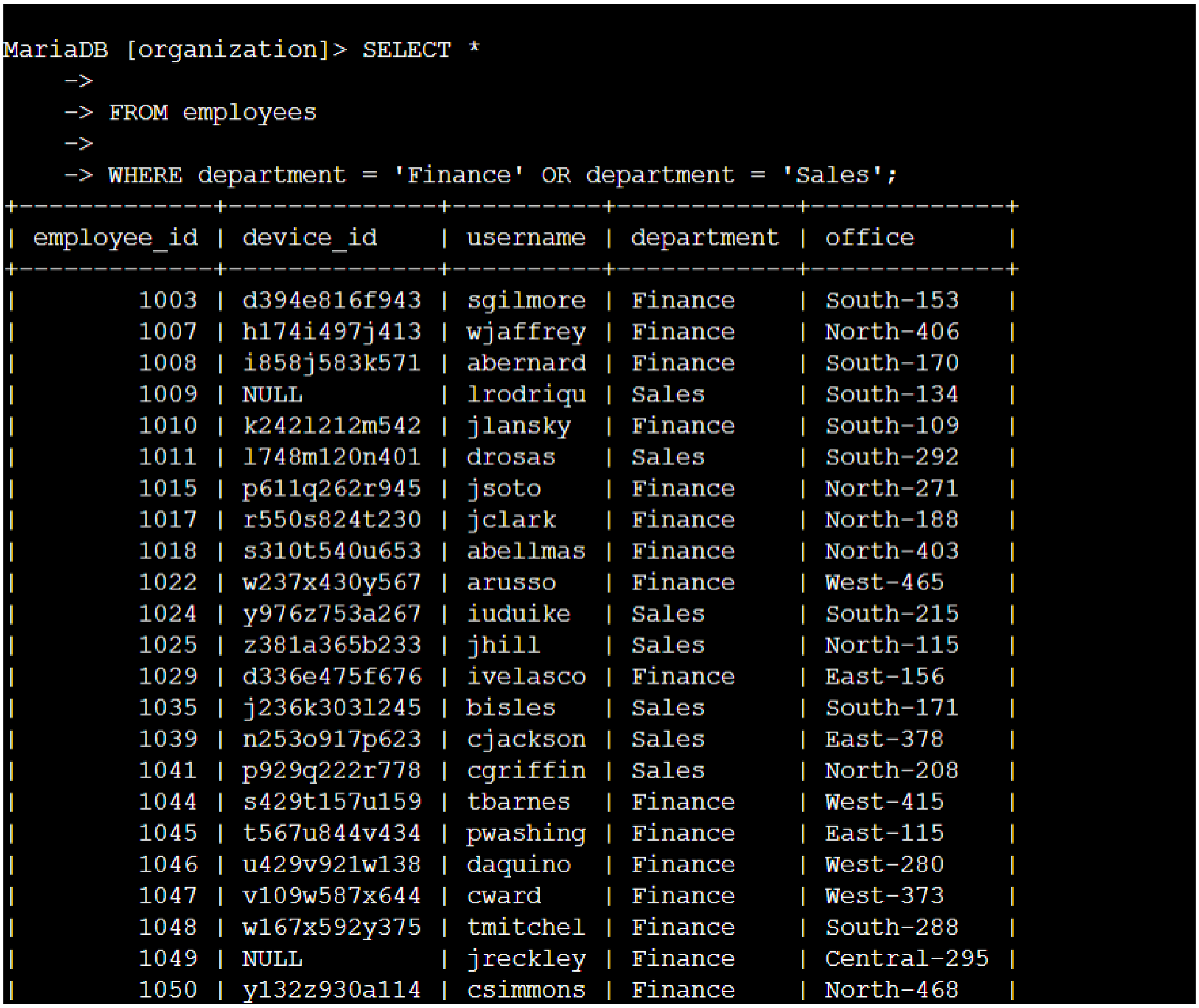
Now, your team needs to perform a different update to the computers of all employees in the Finance or the Sales department, and you need to locate information on these employees.

- Write a SQL query to retrieve records for employees in the `'Finance'` or the `'Sales'` department.

***Note:** Even though both conditions are based on the same column, you need to write out both full conditions. This means that you must specify `department` as the column in both conditions.*

The correct query to solve this step:

```
SELECT *
FROM employees
WHERE department = 'Finance' OR department = '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    |
|        1041 | p929q222r778 | cgriffin | Sales      | North-208   |
|        1044 | s429t157u159 | tbarnes  | Finance    | West-415    |
|        1045 | t567u844v434 | pwashing | Finance    | East-115    |
|        1046 | u429v921w138 | daquino  | Finance    | West-280    |
|        1047 | v109w587x644 | cward    | Finance    | West-373    |
|        1048 | w167x592y375 | tmitchel | Finance    | South-288   |
|        1049 | NULL         | jreckley | Finance    | Central-295 |
|        1050 | y132z930a114 | csimmons | Finance    | North-468   |
```

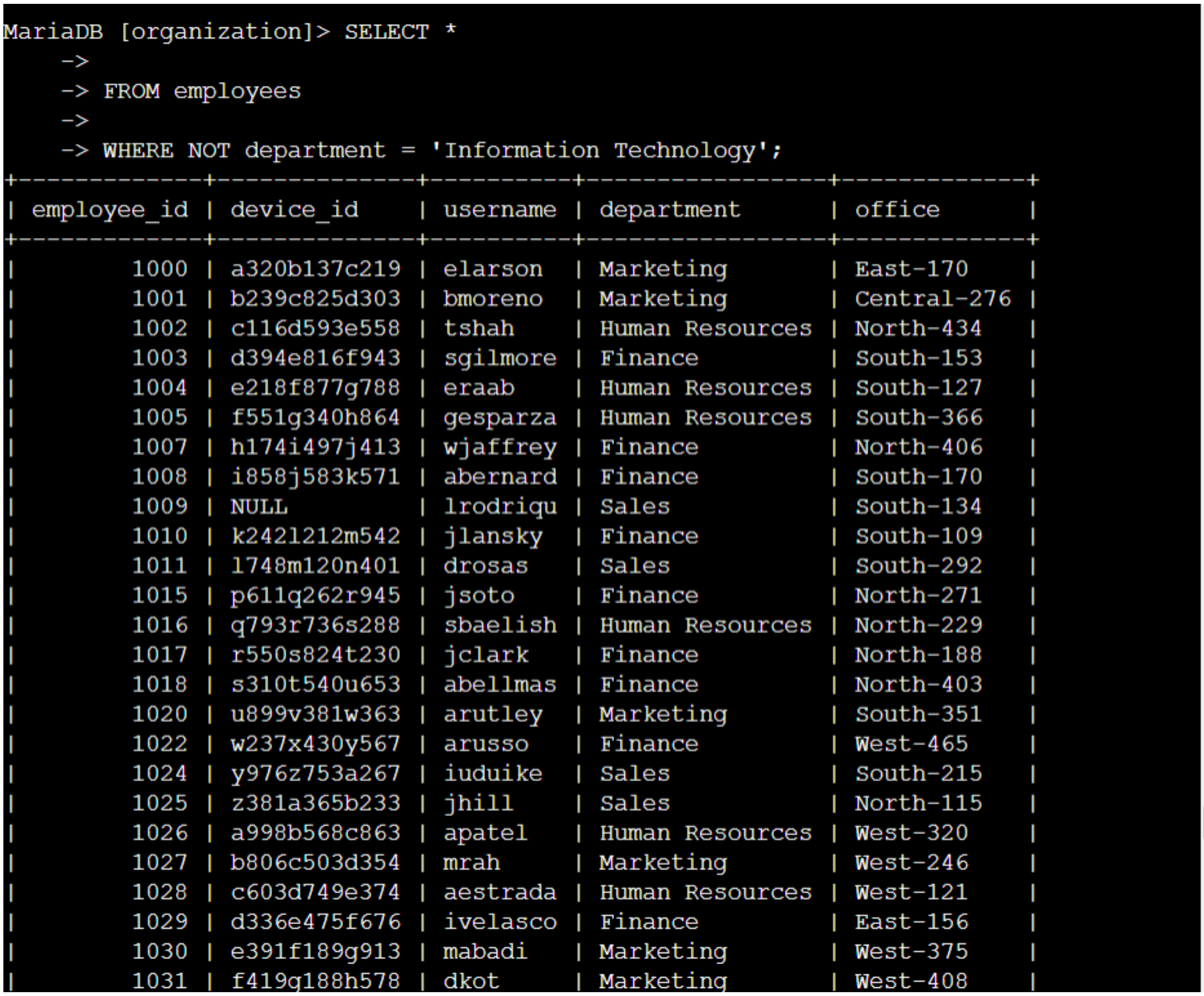
Task 6. Retrieve all employees not in IT

Your team needs to make one more update. This update was already made to employee computers in the Information Technology department. The team needs information about employees who are not in that department. You should use the `NOT` operator to identify these employees.

- Write a SQL query to retrieve records for employees who are not in the `'Information Technology'` department.

The correct query to solve this step:

```sql
SELECT *
FROM employees
WHERE NOT department = 'Information Technology';
```



```
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   |
|        1004 | e218f877g788 | eraab    | Human Resources  | South-127   |
|        1005 | f551g340h864 | gesparza | Human Resources  | South-366   |
|        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   |
|        1016 | q793r736s288 | sbaelish | Human Resources  | North-229   |
|        1017 | r550s824t230 | jclark   | Finance          | North-188   |
|        1018 | s310t540u653 | abellmas | Finance          | North-403   |
|        1020 | u899v381w363 | arutley  | Marketing        | South-351   |
|        1022 | w237x430y567 | arusso   | Finance          | West-465    |
|        1024 | y976z753a267 | iuduike  | Sales            | South-215   |
|        1025 | z381a365b233 | jhill    | Sales            | North-115   |
|        1026 | a998b568c863 | apatel   | Human Resources  | West-320    |
|        1027 | b806c503d354 | mrah     | Marketing        | West-246    |
|        1028 | c603d749e374 | aestrada | Human Resources  | West-121    |
|        1029 | d336e475f676 | ivelasco | Finance          | East-156    |
|        1030 | e391f189g913 | mabadi   | Marketing        | West-375    |
|        1031 | f419g188h578 | dkot     | Marketing        | West-408    |
```