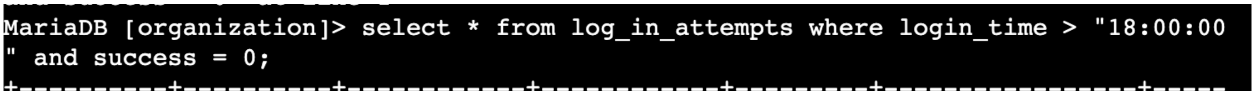
Apply filters to SQL queries

Project description

The project is intended to demonstrate the capabilities in using SQL queries to find required information in the provided table data. The information that is gathered is related to the effort to find correct information that can support security investigation.

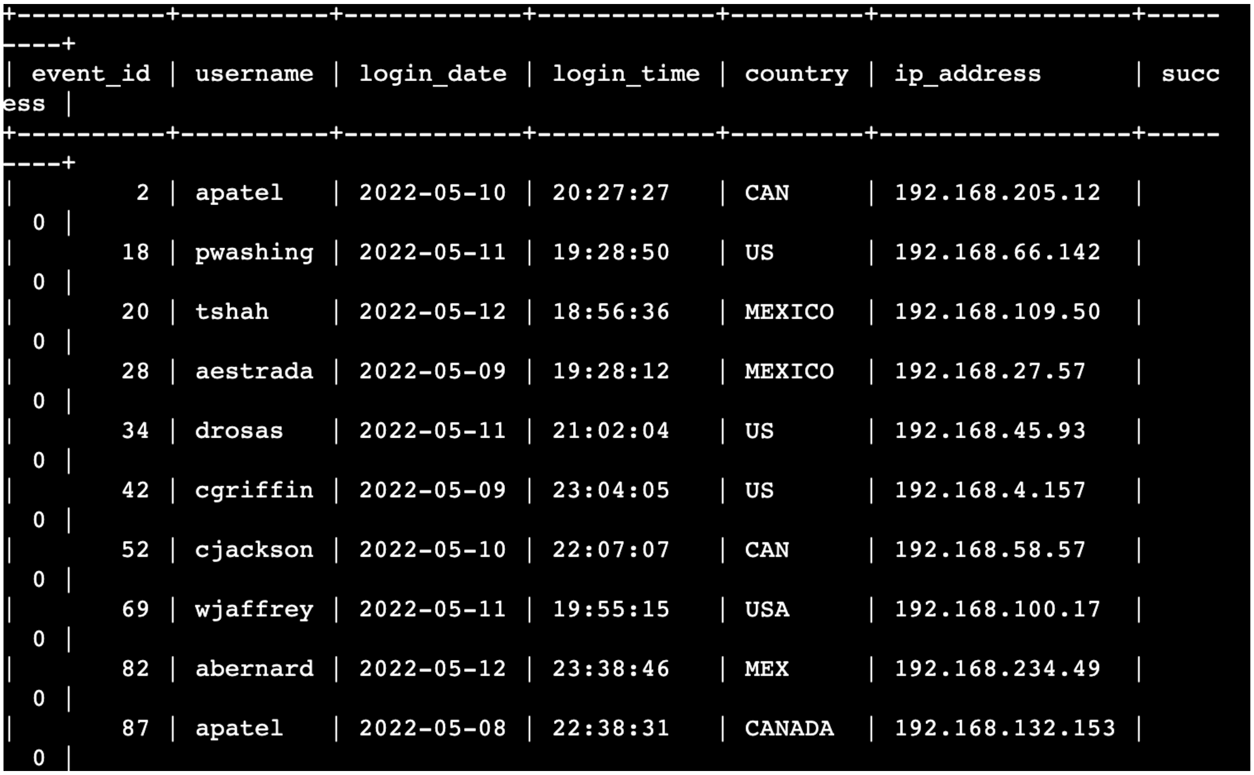
Retrieve after hours failed login attempts

In this scenario, we will find information on people who try to login after 18.00 and fail by investigating the **log_in_attempts** table. The syntax will be as follows



```
MariaDB [organization]> select * from log_in_attempts where login_time > "18:00:00
" and success = 0;
```

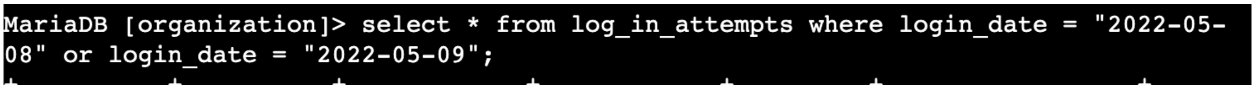
And top 10 results will be as follows:



```
+----------+----------+------------+------------+---------+-----------------+-----
----+
| event_id | username | login_date | login_time | country | ip_address      | succ
ess |
+----------+----------+------------+------------+---------+-----------------+-----
----+
|        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 |
```

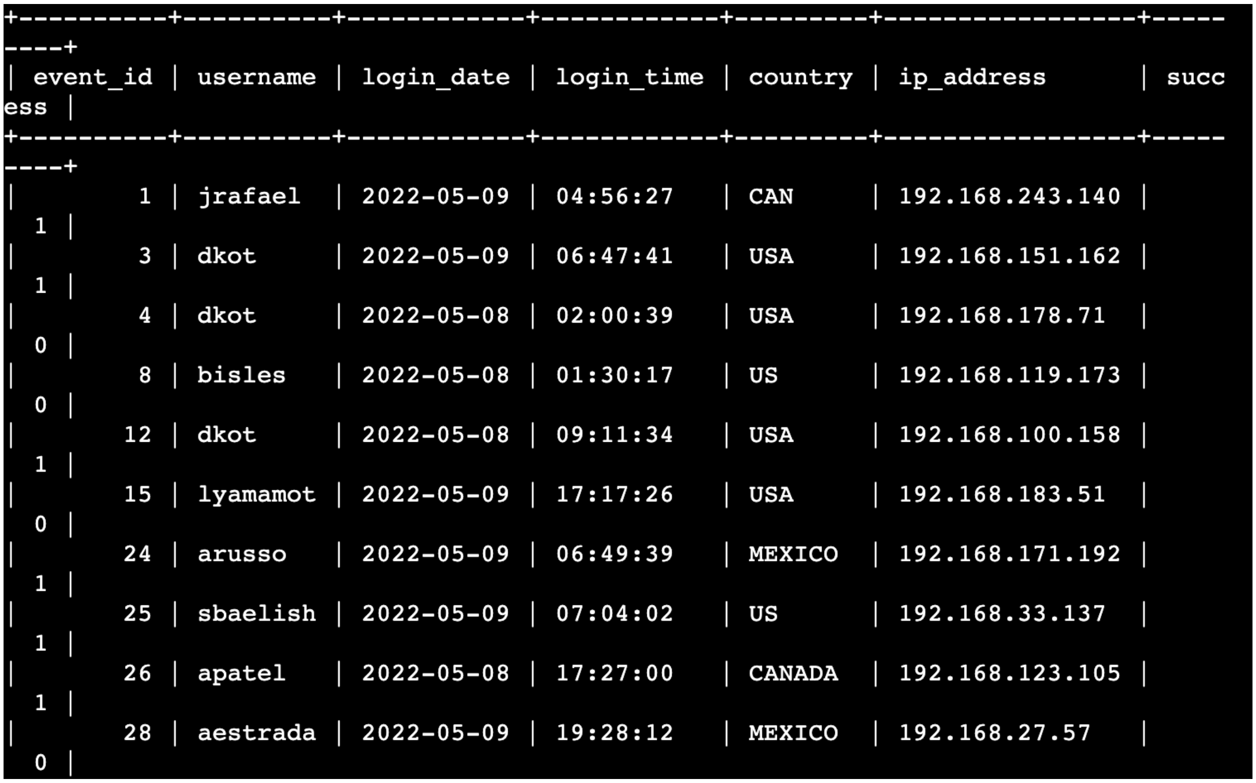
Retrieve login attempts on specific dates

In this scenario, we will investigate the login attempts for specific dates because there are suspicious activities happening on 2022-05-09. So we will investigate the **log_in_attempts** table to see what happened on that date and the previous date. The syntax will be as follows:



```
MariaDB [organization]> select * from log_in_attempts where login_date = "2022-05-
08" or login_date = "2022-05-09";
```

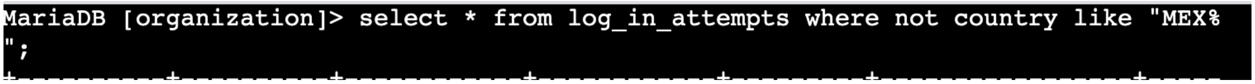
And top 10 results will be as follows:



```
+----------+----------+------------+------------+---------+-----------------+-----
----+
| event_id | username | login_date | login_time | country | ip_address      | succ
ess |
+----------+----------+------------+------------+---------+-----------------+-----
----+
|        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 |
```

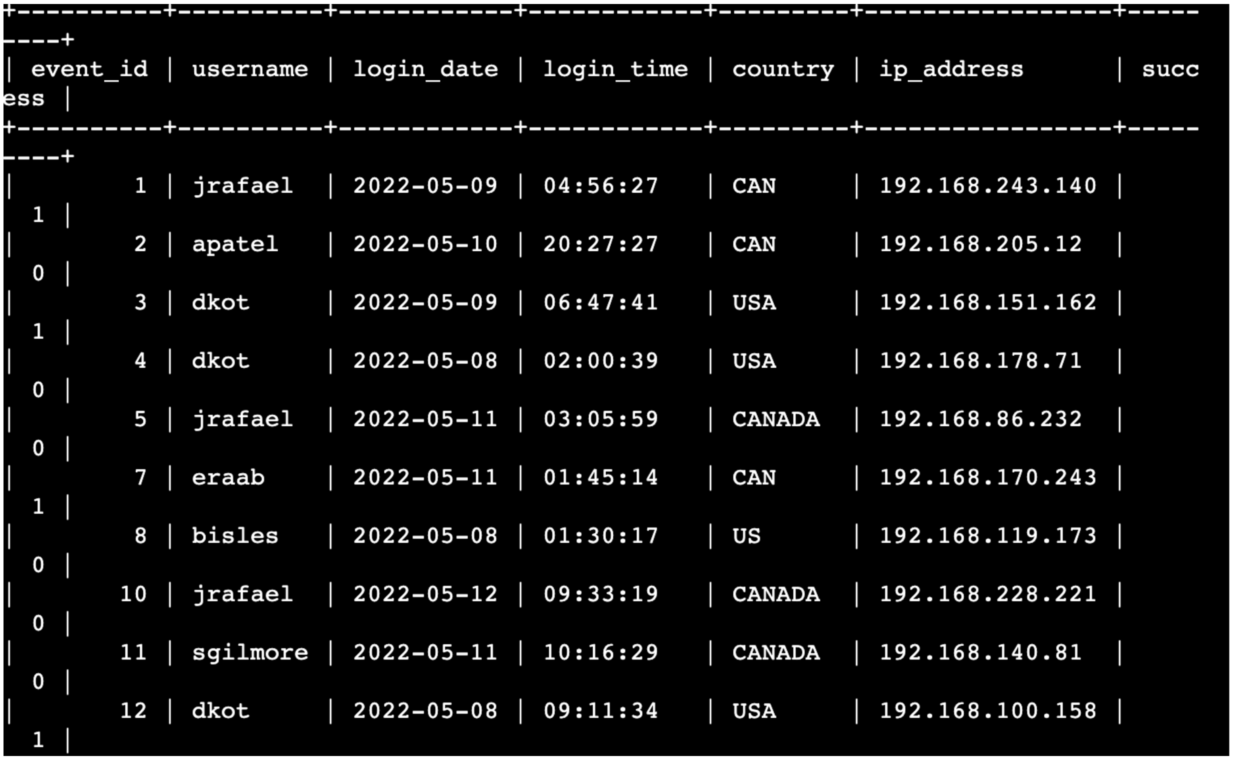
Retrieve login attempts outside of Mexico

In this scenario, we will continue the investigation and later on find out that the login is coming from a country other than Mexico. So we will investigate **log_in_attempts** table once again from login outside Mexico with following syntax:



```
MariaDB [organization]> select * from log_in_attempts where not country like "MEX%
";
```

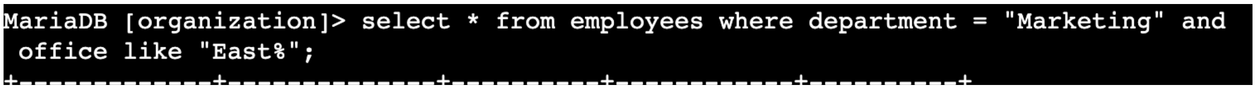
And top 10 results will be as follows:



```
+----------+----------+------------+------------+---------+-----------------+-----
----+
| event_id | username | login_date | login_time | country | ip_address      | succ
ess |
+----------+----------+------------+------------+---------+-----------------+-----
----+
|        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 |
```

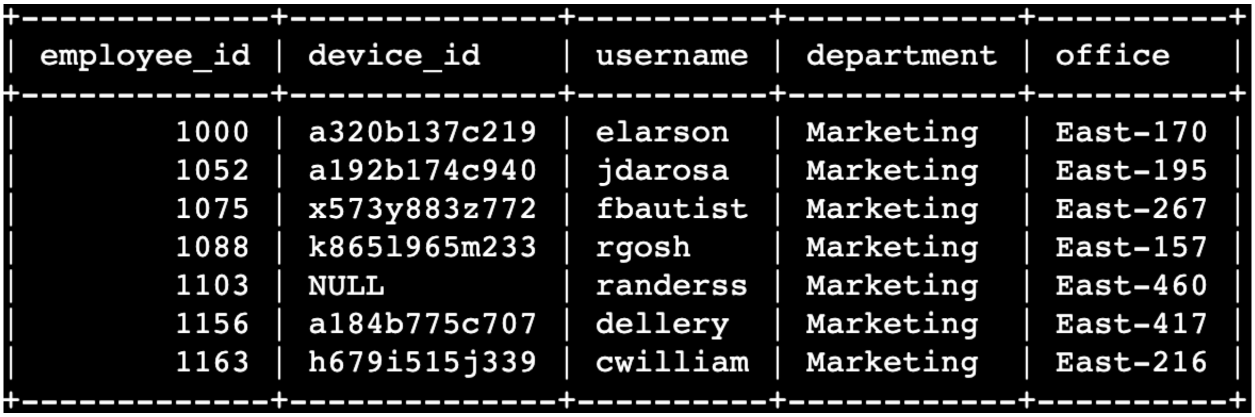
Retrieve employees in Marketing

In this scenario we will try to find employees who are in the Marketing department and work in the east building. So we will investigate **employee** table with following syntax:



```
MariaDB [organization]> select * from employees where department = "Marketing" and
 office like "East%";
+-------------+---------------+----------+------------+----------+
```

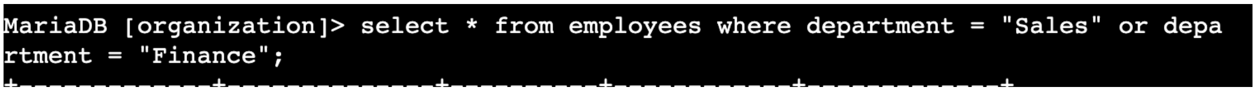
And the result will be as follows:



```
+-------------+---------------+----------+------------+----------+
| 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 |
+-------------+---------------+----------+------------+----------+
```

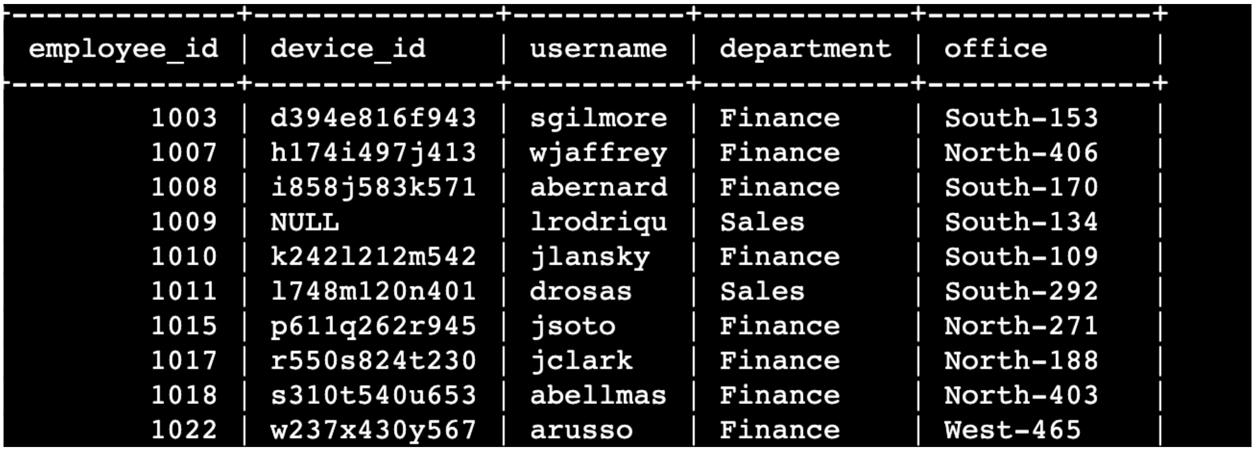
Retrieve employees in Finance or Sales

In this scenario we will try to find employees in the Finance or Sales department since they need security updates in their device. We will investigate **employees** table once again using following syntax:



```
MariaDB [organization]> select * from employees where department = "Sales" or depa
rtment = "Finance";
```

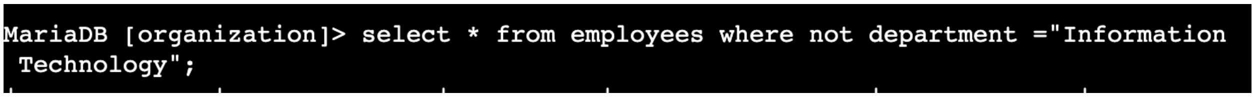
And the top 10 result will be as follows:



```
+-------------+--------------+----------+------------+-------------+
| 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    |
```

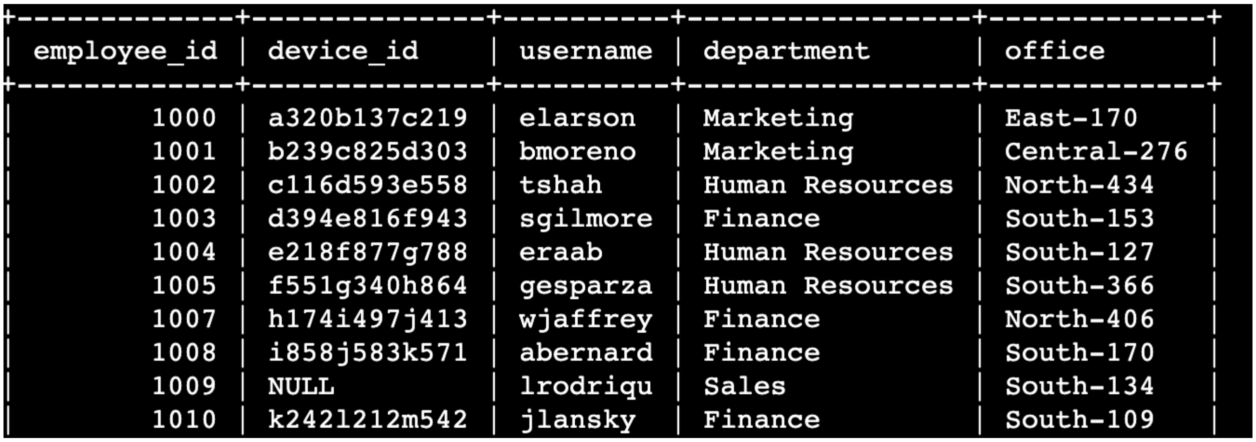
Retrieve all employees not in IT

In this scenario we will find all employees outside of the IT department since all of them have their machine updated. We will investigate **employees** table once again and exclude IT department from query with query as follows:



```
MariaDB [organization]> select * from employees where not department ="Information
 Technology";
```

And the top 10 result will be as follows:



```
+-------------+-------------+----------+-----------------+-------------+
| 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 | k242l212m542 | jlansky  | Finance         | South-109   |
```

Summary

In this exercise, we already use SQL filtering capability such as WHERE, AND, OR and NOT to find the data that we want. We also using it in several tables according to scenario that we need to solve