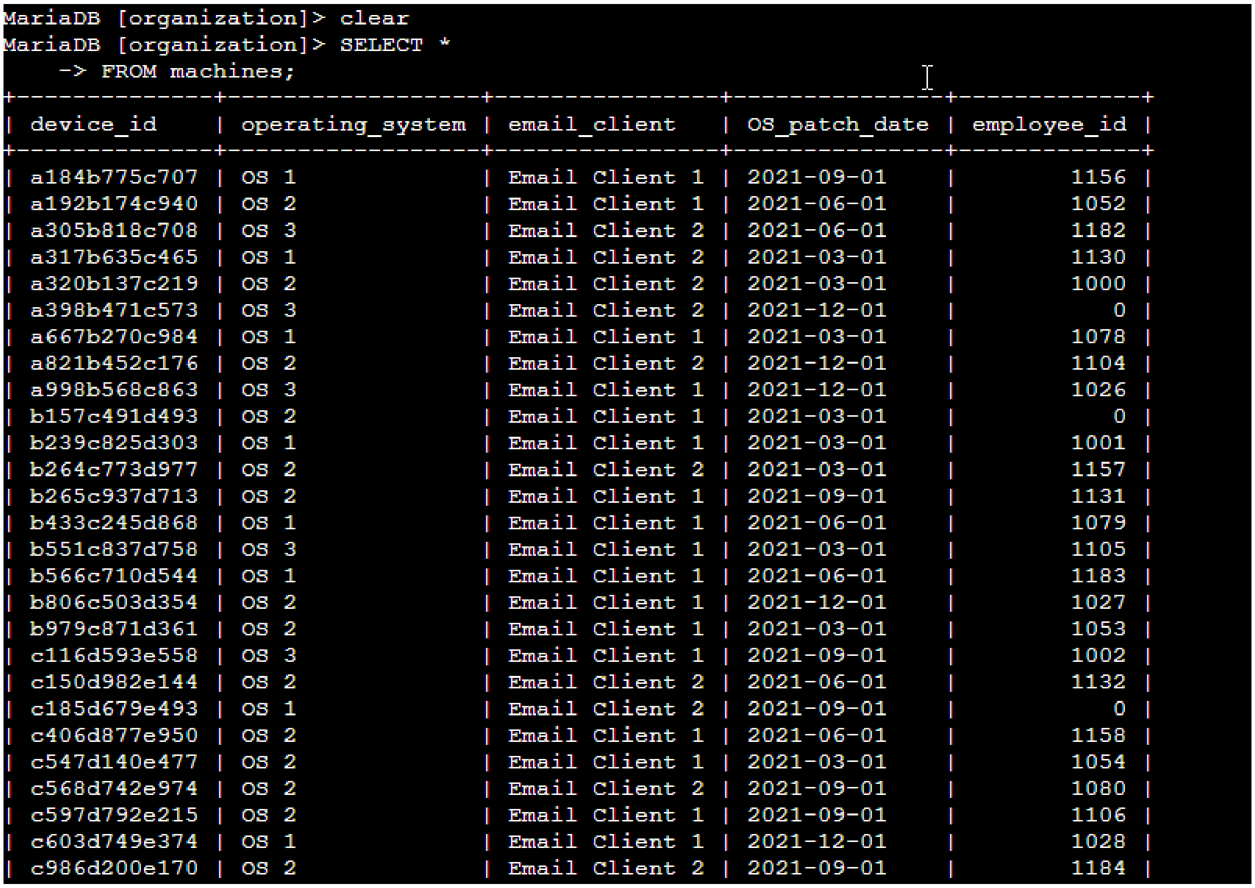
Joining data with SQL: Combine information effectively using joins.

Project description:

SQL joins allow you to combine tables that have a common column. This is useful when you need to connect information that appears in different tables.

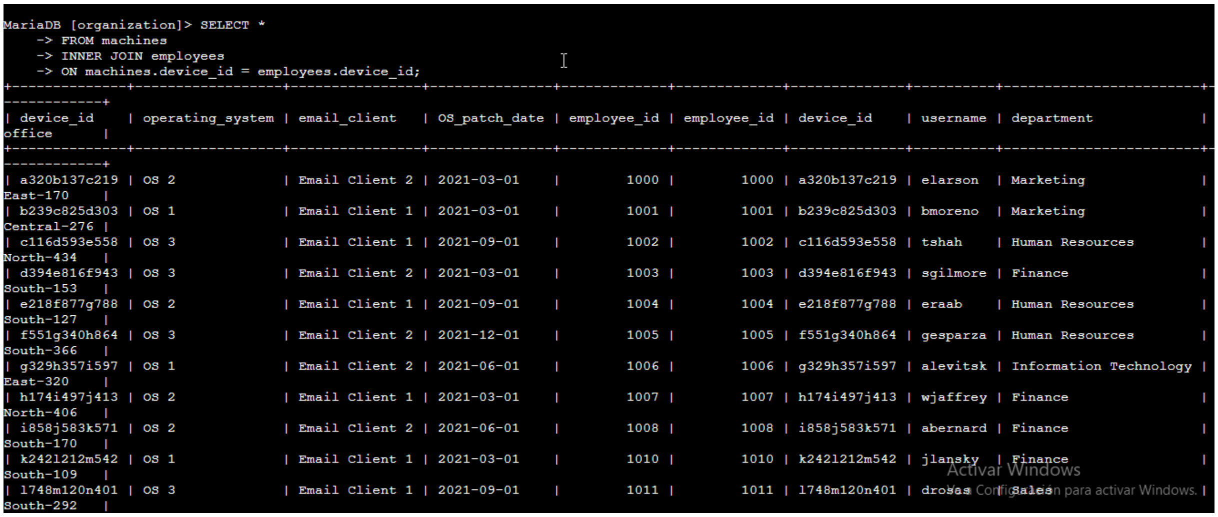
Match employees with their machines:

First, I need to identify which employee uses each machine. The data is located in the machines and employees tables. I will use an SQL inner join to retrieve the records I need based on a related column. In this case, both tables include the device_id column. This will be used to perform the join.



```
MariaDB [organization]> clear
MariaDB [organization]> SELECT *
    -> FROM machines;
+--------------+------------------+----------------+---------------+-------------+
| device_id    | operating_system | email_client   | OS_patch_date | employee_id |
+--------------+------------------+----------------+---------------+-------------+
| a184b775c707 | OS 1             | Email Client 1 | 2021-09-01    |        1156 |
| a192b174c940 | OS 2             | Email Client 1 | 2021-06-01    |        1052 |
| a305b818c708 | OS 3             | Email Client 2 | 2021-06-01    |        1182 |
| a317b635c465 | OS 1             | Email Client 2 | 2021-03-01    |        1130 |
| a320b137c219 | OS 2             | Email Client 2 | 2021-03-01    |        1000 |
| a398b471c573 | OS 3             | Email Client 2 | 2021-12-01    |           0 |
| a667b270c984 | OS 1             | Email Client 1 | 2021-03-01    |        1078 |
| a821b452c176 | OS 2             | Email Client 2 | 2021-12-01    |        1104 |
| a998b568c863 | OS 3             | Email Client 1 | 2021-12-01    |        1026 |
| b157c491d493 | OS 2             | Email Client 1 | 2021-03-01    |           0 |
| b239c825d303 | OS 1             | Email Client 1 | 2021-03-01    |        1001 |
| b264c773d977 | OS 2             | Email Client 2 | 2021-03-01    |        1157 |
| b265c937d713 | OS 2             | Email Client 1 | 2021-09-01    |        1131 |
| b433c245d868 | OS 1             | Email Client 1 | 2021-06-01    |        1079 |
| b551c837d758 | OS 3             | Email Client 1 | 2021-03-01    |        1105 |
| b566c710d544 | OS 1             | Email Client 1 | 2021-06-01    |        1183 |
| b806c503d354 | OS 2             | Email Client 1 | 2021-12-01    |        1027 |
| b979c871d361 | OS 2             | Email Client 1 | 2021-03-01    |        1053 |
| c116d593e558 | OS 3             | Email Client 1 | 2021-09-01    |        1002 |
| c150d982e144 | OS 2             | Email Client 2 | 2021-06-01    |        1132 |
| c185d679e493 | OS 1             | Email Client 2 | 2021-09-01    |           0 |
| c406d877e950 | OS 2             | Email Client 1 | 2021-06-01    |        1158 |
| c547d140e477 | OS 2             | Email Client 1 | 2021-03-01    |        1054 |
| c568d742e974 | OS 2             | Email Client 2 | 2021-09-01    |        1080 |
| c597d792e215 | OS 2             | Email Client 1 | 2021-09-01    |        1106 |
| c603d749e374 | OS 1             | Email Client 1 | 2021-12-01    |        1028 |
| c986d200e170 | OS 2             | Email Client 2 | 2021-09-01    |        1184 |
```

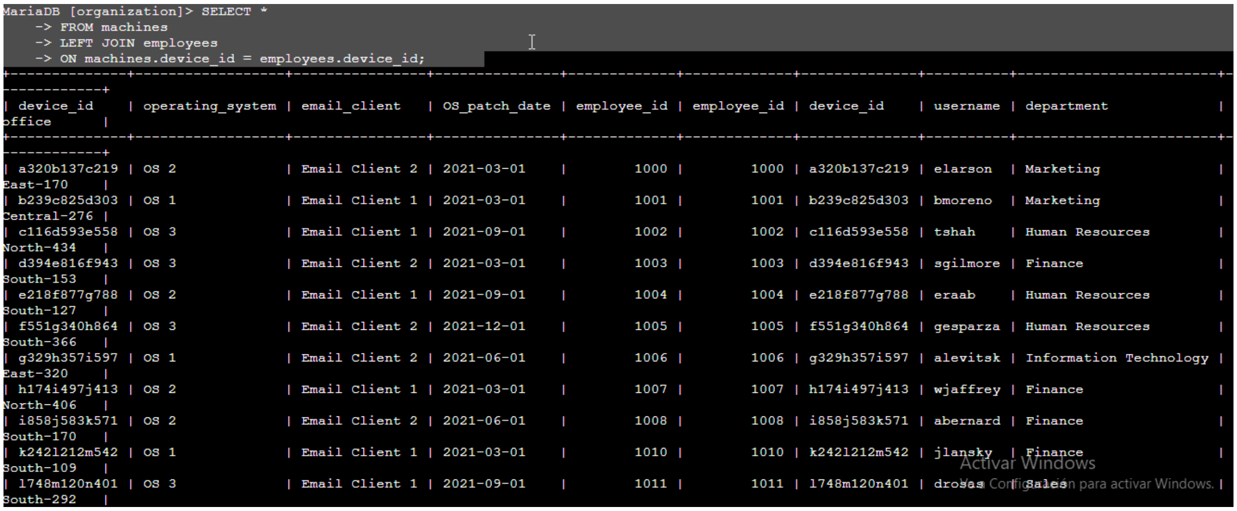
This SQL code I'm executing performs a query to combine data from the "machines" and "employees" tables using an inner join. The result of the query will include all columns from both tables. The join is done using the condition "machines.device_id = employees.device_id", which means that records from the two tables are combined based on the value of the "device_id" column. This will allow me to obtain the information of which employee is using each machine.



```
MariaDB [organization]> SELECT *
    -> FROM machines
    -> INNER JOIN employees
    -> ON machines.device_id = employees.device_id;
+--------------+------------------+----------------+---------------+-------------+-------------+--------------+----------+------------------------+-
-------------+
| device_id    | operating_system | email_client   | OS_patch_date | employee_id | employee_id | device_id    | username | department             |
office       |
+--------------+------------------+----------------+---------------+-------------+-------------+--------------+----------+------------------------+-
-------------+
| a320b137c219 | OS 2             | Email Client 2 | 2021-03-01    |        1000 |        1000 | a320b137c219 | elarson  | Marketing              |
East-170     |
| b239c825d303 | OS 1             | Email Client 1 | 2021-03-01    |        1001 |        1001 | b239c825d303 | bmoreno  | Marketing              |
Central-276 |
| c116d593e558 | OS 3             | Email Client 1 | 2021-09-01    |        1002 |        1002 | c116d593e558 | tshah    | Human Resources        |
North-434    |
| d394e816f943 | OS 3             | Email Client 2 | 2021-03-01    |        1003 |        1003 | d394e816f943 | sgilmore | Finance                |
South-153    |
| e218f877g788 | OS 2             | Email Client 1 | 2021-09-01    |        1004 |        1004 | e218f877g788 | eraab    | Human Resources        |
South-127    |
| f551g340h864 | OS 3             | Email Client 2 | 2021-12-01    |        1005 |        1005 | f551g340h864 | gesparza | Human Resources        |
South-366    |
| g329h357i597 | OS 1             | Email Client 2 | 2021-06-01    |        1006 |        1006 | g329h357i597 | alevitsk | Information Technology |
East-320     |
| h174i497j413 | OS 2             | Email Client 1 | 2021-03-01    |        1007 |        1007 | h174i497j413 | wjaffrey | Finance                |
North-406    |
| i858j583k571 | OS 2             | Email Client 2 | 2021-06-01    |        1008 |        1008 | i858j583k571 | abernard | Finance                |
South-170    |
| k2421212m542 | OS 1             | Email Client 1 | 2021-03-01    |        1010 |        1010 | k2421212m542 | jlansky  | Finance                |
South-109    |
| l748m120n401 | OS 3             | Email Client 1 | 2021-09-01    |        1011 |        1011 | l748m120n401 | drosas   | Sales                  |
South-292    |
```

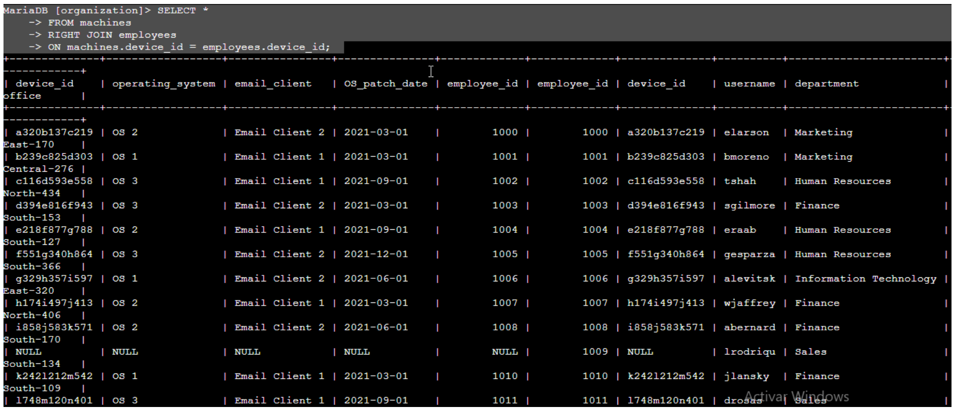
Get more data:

Now I display the information of the machines and the employees who have them assigned. To do this, I will use an inner join between the "employees" and "machines" tables, based on the device_id column. This will give me the records that match in both tables. Then, I perform a left join and a right join between the same tables to obtain the employees without assigned machines and the machines without assigned employees, respectively. I use the device_id column as the join criteria in both operations.



```
MariaDB [organization]> SELECT *
    -> FROM machines
    -> LEFT JOIN employees
    -> ON machines.device_id = employees.device_id;
+--------------+------------------+----------------+---------------+-------------+-------------+--------------+----------+------------------------+-
-------------+
| device_id    | operating_system | email_client   | OS_patch_date | employee_id | employee_id | device_id    | username | department             |
office       |
+--------------+------------------+----------------+---------------+-------------+-------------+--------------+----------+------------------------+-
-------------+
| a320b137c219 | OS 2             | Email Client 2 | 2021-03-01    |        1000 |        1000 | a320b137c219 | elarson  | Marketing              |
East-170     |
| b239c825d303 | OS 1             | Email Client 1 | 2021-03-01    |        1001 |        1001 | b239c825d303 | bmoreno  | Marketing              |
Central-276 |
| c116d593e558 | OS 3             | Email Client 1 | 2021-09-01    |        1002 |        1002 | c116d593e558 | tshah    | Human Resources        |
North-434    |
| d394e816f943 | OS 3             | Email Client 2 | 2021-03-01    |        1003 |        1003 | d394e816f943 | sgilmore | Finance                |
South-153    |
| e218f877g788 | OS 2             | Email Client 1 | 2021-09-01    |        1004 |        1004 | e218f877g788 | eraab    | Human Resources        |
South-127    |
| f551g340h864 | OS 3             | Email Client 2 | 2021-12-01    |        1005 |        1005 | f551g340h864 | gesparza | Human Resources        |
South-366    |
| g329h357i597 | OS 1             | Email Client 2 | 2021-06-01    |        1006 |        1006 | g329h357i597 | alevitsk | Information Technology |
East-320     |
| h174i497j413 | OS 2             | Email Client 1 | 2021-03-01    |        1007 |        1007 | h174i497j413 | wjaffrey | Finance                |
North-406    |
| i858j583k571 | OS 2             | Email Client 2 | 2021-06-01    |        1008 |        1008 | i858j583k571 | abernard | Finance                |
South-170    |
| k2421212m542 | OS 1             | Email Client 1 | 2021-03-01    |        1010 |        1010 | k2421212m542 | jlansky  | Finance                |
South-109    |
| l748m120n401 | OS 3             | Email Client 1 | 2021-09-01    |        1011 |        1011 | l748m120n401 | drosas   | Sales                  |
South-292    |
```

In this SQL query, I am selecting all columns from the "machines" table and performing a left join with the "employees" table. I use the device_id column to link the two tables. This means that I will retrieve all records from the "machines" table, even if they don't have a match in the "employees" table. If there are records in the "machines" table that don't have a match in the "employees" table, the corresponding fields in the "employees" table will be NULL in the result. This query allows me to retrieve information about the machines, including those that are not assigned to any employee.

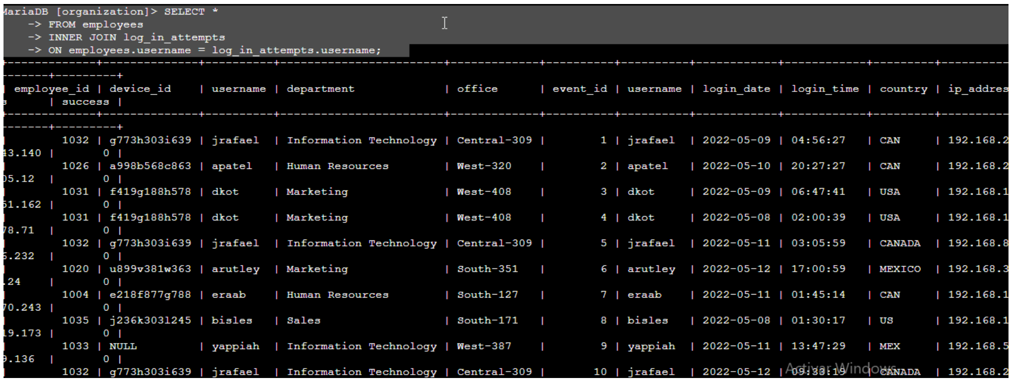


```
MariaDB [organization]> SELECT *
    -> FROM machines
    -> RIGHT JOIN employees
    -> ON machines.device_id = employees.device_id;
+--------------+------------------+----------------+---------------+-------------+-------------+--------------+----------+------------------------+
-------------+
| device_id    | operating_system | email_client   | OS_patch_date | employee_id | employee_id | device_id    | username | department             |
office       |
+--------------+------------------+----------------+---------------+-------------+-------------+--------------+----------+------------------------+
-------------+
| a320b137c219 | OS 2             | Email Client 2 | 2021-03-01    |        1000 |        1000 | a320b137c219 | elarson  | Marketing              |
East-170     |
| b239c825d303 | OS 1             | Email Client 1 | 2021-03-01    |        1001 |        1001 | b239c825d303 | bmoreno  | Marketing              |
Central-276  |
| c116d593e558 | OS 3             | Email Client 1 | 2021-09-01    |        1002 |        1002 | c116d593e558 | tshah    | Human Resources        |
North-434    |
| d394e816f943 | OS 3             | Email Client 2 | 2021-03-01    |        1003 |        1003 | d394e816f943 | sgilmore | Finance                |
South-153    |
| e218f877g788 | OS 2             | Email Client 1 | 2021-09-01    |        1004 |        1004 | e218f877g788 | eraab    | Human Resources        |
South-127    |
| f551g340h864 | OS 3             | Email Client 2 | 2021-12-01    |        1005 |        1005 | f551g340h864 | gesparza | Human Resources        |
South-366    |
| g329h357i597 | OS 1             | Email Client 2 | 2021-06-01    |        1006 |        1006 | g329h357i597 | alevitsk | Information Technology |
East-320     |
| h174i497j413 | OS 2             | Email Client 1 | 2021-03-01    |        1007 |        1007 | h174i497j413 | wjaffrey | Finance                |
North-406    |
| i858j583k571 | OS 2             | Email Client 2 | 2021-06-01    |        1008 |        1008 | i858j583k571 | abernard | Finance                |
South-170    |
| NULL         | NULL             | NULL           | NULL          |        NULL |        1009 | NULL         | lrodriqu | Sales                  |
South-134    |
| k2421212m542 | OS 1             | Email Client 1 | 2021-03-01    |        1010 |        1010 | k2421212m542 | jlansky  | Finance                |
South-109    |
| l748m120n401 | OS 3             | Email Client 1 | 2021-09-01    |        1011 |        1011 | l748m120n401 | drosas   | Sales                  |
```

This SQL query performs a right join between the "machines" and "employees" tables using the device_id column as the join criterion. It retrieves all records from the "employees" table, even if they don't have a match in the "machines" table. The corresponding fields in the "machines" table will be NULL in the result.

Retrieve login attempt data:

To continue investigating the security incident, I need to retrieve the information of all employees who made login attempts. To achieve this, I perform an inner join between the "employees" and "log_in_attempts" tables, linking them using the common column "username".



```
ariaDB [organization]> SELECT *
    -> FROM employees
    -> INNER JOIN log_in_attempts
    -> ON employees.username = log_in_attempts.username;
-------------+--------------+----------+------------------------+-------------+----------+----------+------------+------------+---------+----------
------+---------+
 employee_id | device_id    | username | department             | office      | event_id | username | login_date | login_time | country | ip_addres
       | success |
-------------+--------------+----------+------------------------+-------------+----------+----------+------------+------------+---------+----------
------+---------+
        1032 | g773h303i639 | jrafael  | Information Technology | Central-309 |        1 | jrafael  | 2022-05-09 | 04:56:27   | CAN     | 192.168.2
3.140 |       0 |
        1026 | a998b568c863 | apatel   | Human Resources        | West-320    |        2 | apatel   | 2022-05-10 | 20:27:27   | CAN     | 192.168.2
5.12  |       0 |
        1031 | f419g188h578 | dkot     | Marketing              | West-408    |        3 | dkot     | 2022-05-09 | 06:47:41   | USA     | 192.168.1
1.162 |       0 |
        1031 | f419g188h578 | dkot     | Marketing              | West-408    |        4 | dkot     | 2022-05-08 | 02:00:39   | USA     | 192.168.1
8.71  |       0 |
        1032 | g773h303i639 | jrafael  | Information Technology | Central-309 |        5 | jrafael  | 2022-05-11 | 03:05:59   | CANADA  | 192.168.8
.232  |       0 |
        1020 | u899v381w363 | arutley  | Marketing              | South-351   |        6 | arutley  | 2022-05-12 | 17:00:59   | MEXICO  | 192.168.3
.24   |       0 |
        1004 | e218f877g788 | eraab    | Human Resources        | South-127   |        7 | eraab    | 2022-05-11 | 01:45:14   | CAN     | 192.168.1
0.243 |       0 |
        1035 | j236k3031245 | bisles   | Sales                  | South-171   |        8 | bisles   | 2022-05-08 | 01:30:17   | US      | 192.168.1
9.173 |       0 |
        1033 | NULL         | yappiah  | Information Technology | West-387    |        9 | yappiah  | 2022-05-11 | 13:47:29   | MEX     | 192.168.5
.136  |       0 |
        1032 | g773h303i639 | jrafael  | Information Technology | Central-309 |       10 | jrafael  | 2022-05-12 | 09:33:19   | CANADA  | 192.168.2
```

This SQL query performs an inner join between the "employees" and "log_in_attempts" tables using the "username" column as the join criteria. The result is a combination of rows from both tables where the value of the "username" column matches in both tables. This allows for retrieving the information of employees who have made login attempts recorded in the "log_in_attempts" table. The asterisk (*) in the SELECT statement indicates that all columns from both tables will be selected in the query result.

Summary:

As a junior cybersecurity analyst, performing SQL queries is a crucial part of my daily work. Through these queries, I can filter data, join tables, and obtain specific information that helps me investigate security incidents. Identifying employees in specific departments, excluding IT users, and retrieving login attempt information are just some examples of how SQL queries allow me to gather relevant insights. Additionally, using inner and outer joins provides a more comprehensive view when relating information about assigned machines and employees. In summary, mastering SQL queries is essential for conducting efficient and effective security analysis in my role as a junior cybersecurity analyst.