## **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.

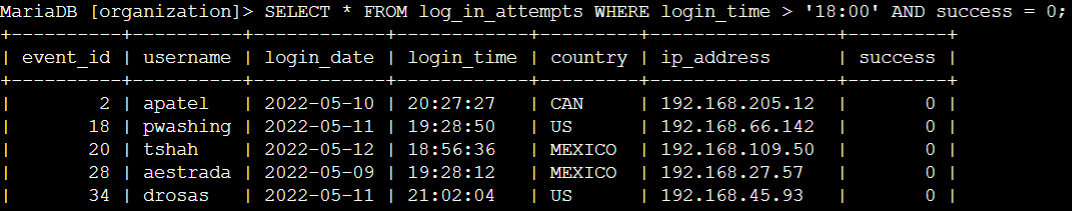
* Use the AND operator to retrieve the failed login attempts that occurred after business hours. Replace the X and Y with the correct values to filter for the records you need:

SELECT \*

FROM log\_in\_attempts

WHERE login\_time > 'X' AND success = Y;

SELECT \* FROM log\_in\_attempts WHERE login\_time > '18:00' AND success = 0;



## 

## 

## 

## **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.

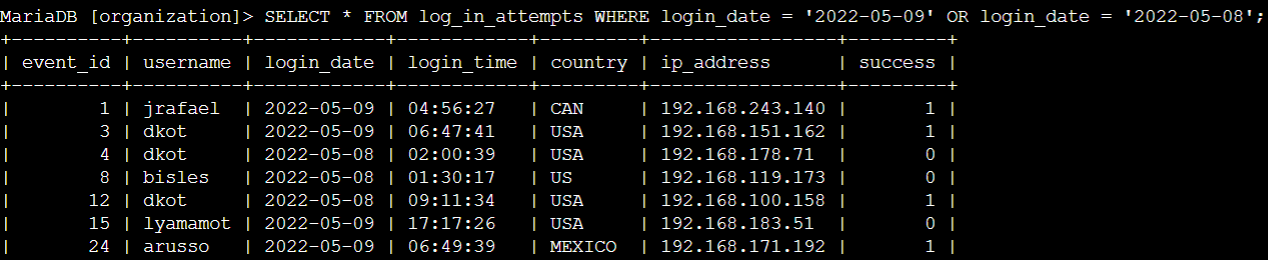
* Use the OR operator to retrieve the failed login attempts on the specified days. Replace the X and Y with the correct values to filter for the records you need:

SELECT \*

FROM log\_in\_attempts

WHERE login\_date = 'X' OR login\_date = 'Y';

SELECT \* FROM log\_in\_attempts WHERE login\_date = '2022-05-09' OR login\_date = '2022-05-08';



## **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%'.

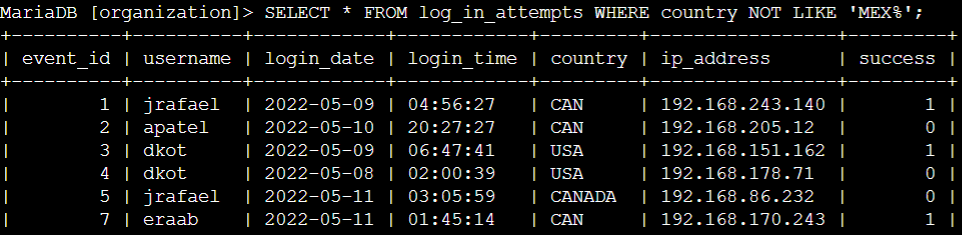
* Run the following SQL query to retrieve login attempts that did not originate in Mexico. Replace X with the correct operator and Y with the correct pattern to filter for the information you need:

SELECT \*

FROM log\_in\_attempts

WHERE X country LIKE 'Y';

SELECT \* FROM log\_in\_attempts WHERE country NOT LIKE 'MEX%';



## 

## **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;

Copied!

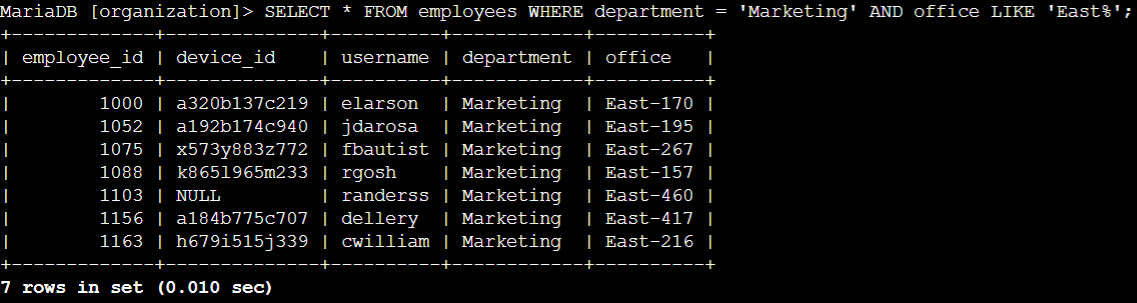
content\_copy

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.*

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



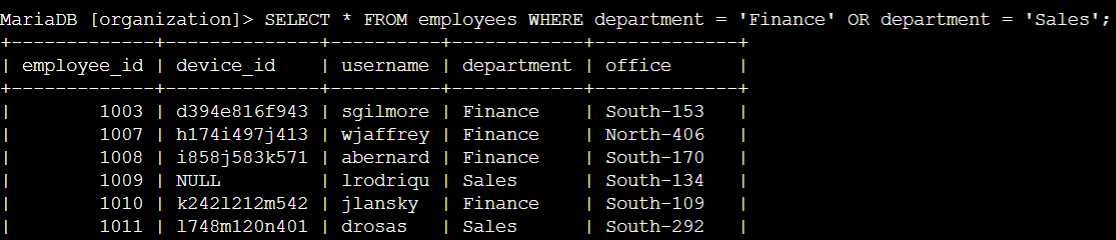
## **Task 5. Retrieve employees in Finance or Sales**

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.*

SELECT \* FROM employees WHERE department = 'Finance' OR department = 'Sales';



## 

## **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.

SELECT \* FROM employees WHERE department NOT LIKE 'Information Technology';

