

Exemplar: Filter with AND, OR, and NOT

Activity overview

As a security analyst, you'll likely need to analyze data. And often finding the specific data you'll need depends on more than one factor.

To retrieve specific pieces of information from the database, you can filter for multiple conditions. You can also filter for what does not match a particular condition.

In this lab activity, you'll use the **AND**, **OR**, and **NOT** operators to create more complex filters for SQL queries.

Get ready to practice running a few complex SQL queries!

This exemplar is a walkthrough of the previous Qwiklab activity, including detailed instructions and solutions. You may use this exemplar if you were unable to complete the lab and/or you need extra guidance in competing lab tasks. You may also refer to this exemplar to prepare for the graded quiz in this module.

Scenario

In this scenario, you need to obtain specific information about employees, their machines, and the departments they belong to from the database.

Your team needs data to investigate potential security issues and to update computers.

You are responsible for filtering the required information from the database.

Here's how you'll do this task: **First**, you'll retrieve all failed login attempts after business hours. **Second**, you'll retrieve all login attempts that occurred on specific dates. **Third**, you'll retrieve logins that didn't originate in Mexico. **Fourth**, you'll retrieve information about certain employees in the Marketing department. **Fifth**, you'll retrieve information about employees in the Finance or the Sales department. **Finally**, you'll obtain information about employees who are not in the Information Technology department.

Note: *In this lab you'll be working with the organization database and the tables it contains.*

*The lab starts with the organization database in the MariaDB shell that is already open. This means you can start with the tasks as soon as you click the **Start Lab** button.*

*If you unintentionally exit the organization database in the MariaDB shell, you can reconnect by running the **sudo mysql organization** command.*

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:

```
1 SELECT *
2 FROM log_in_attempts
3 WHERE login_time > 'X' AND success = Y;
```

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:

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

How many failed login attempts occurred after 18:00?

Answer: There are 19 failed login attempts that occurred after 18:00.

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:

```
1 SELECT *
2 FROM log_in_attempts
3 WHERE login_date = 'X' OR login_date = 'Y';
```

The correct query to solve this step:

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

How many login attempts were made on these two days?

Answer: There are 75 login attempts in these two days.

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:

```
1 SELECT *
2 FROM log_in_attempts
3 WHERE X country LIKE 'Y';
```

The correct query to solve this step:

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

How many login attempts were made outside of Mexico?

Answer: There are 144 login attempts made outside of Mexico.

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:

```
1 SELECT *
2 FROM employees;
```

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:

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

What is the username of the first employee in the Marketing department in the East building?

Answer: The username of the first employee in the Marketing department in the East building is elarson.

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:

```
1 SELECT *
2 FROM employees
3 WHERE department = 'Finance' OR department = 'Sales';
```

What is the username of the first employee in the Sales department returned by the query?

Answer: The username of the first employee in the Sales department is lrodriqu.

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:

```
1  SELECT *
2  FROM employees
3  WHERE NOT department = 'Information Technology';
```

How many employees are not in the Information Technology department?

Answer: There are 161 employees who aren't in the Information Technology department.