Apply Filters to SQL Queries — Security Investigation Portfolio Project

Project Description

As a security professional at a large organization, I investigated suspicious authentication activity and inventoried employee machines by querying two datasets: log_in_attempts and employees. I used SQL filters with **AND**, **OR**, **NOT**, pattern matching via **LIKE**, and date/time filters to retrieve targeted records that support incident triage and access control reviews.

Task 3 — Retrieve After■Hours Failed Login Attempts Goal: Find failed attempts (success = 0/FALSE) occurring after 18:00.

```
SELECT *
FROM log_in_attempts
WHERE (success = 0 OR success = FALSE)
AND login_time > '18:00:00';
```

Why it works: Combines failure criterion with a time window using AND. Using either 0 or FALSE captures failed attempts depending on schema conventions.

Task 4 — Retrieve Login Attempts on Specific Dates Goal: Return all attempts on 2022■05■09 or 2022■05■08.

```
SELECT *
FROM log_in_attempts
WHERE login_date IN ('2022-05-08','2022-05-09');
```

Why it works: The IN list expresses OR between two exact dates cleanly. Equivalent to two equality checks joined with OR.

Task 5 — Retrieve Login Attempts Outside of Mexico Goal: Exclude rows where country is recorded as MEX or MEXICO (case/format variations).

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

Why it works: 'MEX%' matches MEX and MEXICO (and any other MEX-prefixed values). NOT LIKE filters for all other countries.

Task 6 — Retrieve Employees in Marketing (East Building Only)

Goal: Identify Marketing employees whose office starts with 'East-'.

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

Why it works: LIKE with wildcards handles values such as 'Marketing' within longer strings and office codes like East-170/East-320.

Task 7 — Retrieve Employees in Finance or Sales Goal: Pull machines/users in either department for a targeted update.

```
SELECT *
FROM employees
WHERE department LIKE '%Finance%'
OR department LIKE '%Sales%';
```

Why it works: OR returns rows where department contains either Finance or Sales. LIKE supports variations (e.g., 'Sales-EMEA').

Task 8 — Retrieve All Employees Not in IT

Goal: Select everyone except Information Technology for a remaining update.

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

Why it works: NOT LIKE excludes the IT department while returning all others. If the schema used short values ('IT'), swap the pattern accordingly.

Core SQL Filtering Concepts Used

Concept	**Usage in this project**
AND / OR	Combine multiple predicates (e.g., failure AND time window; Finance OR Sales).
NOT	Exclude a set (e.g., NOT LIKE 'MEX%' or NOT LIKE '%Information Technology%').
LIKE / Wildcards	Prefix/suffix patterns (e.g., 'East-%', '%Marketing%').
IN	Compact OR for enumerated values (dates list).
Date/Time Filters	Compare DATE and TIME types directly (e.g., > '18:00:00').

Summary

I applied focused SQL filters to investigate after-hours failures, date-scoped events, geographic exclusions, and department-based targeting. By combining AND/OR/NOT logic, pattern matching with LIKE, and precise date/time predicates, I produced practical result sets for security triage and endpoint management. These queries reflect everyday analyst workflows when correlating authentication telemetry with asset/HR data to drive containment and remediation.