W05 Paper: Case Study Working with Sets in SQL queries

You have had a profitable fourth week at your new company. You built on your skill at joining tables and learned that set theory and practices represent the concepts of joins when you combine like result sets with set operators.

Your boss gave you a new task. You should qualify what you learned about set theory and practice and explain how the process of joining sets differs between using join operators and set operators. You also should qualify how to exclude the intersection with the combination of set operators.

You should return and report with a 3–5 paragraph report that clearly explains what you learned while mastering command line semantics. This paper should qualify what you learned by experimenting with the technology.

Report

During this week, I have learned the concepts of joining sets using set operators, such as UNION, INTERSECT, and EXCEPT. The UNION operator combines the results of two or more SELECT statements into a single result set, eliminating duplicates; this is known as a compound query, while UNION ALL retains the duplicates. Unlike JOIN operations, which require a relational connection between tables, the UNION operator allows you to combine completely unrelated tables, as long as the number and types of columns in the result sets are the same.

The INTERSECT operator returns only rows that exist in both result sets and the EXCEPT operator is used to retrieve records from the first result set that do not exist in the second. However, INTERSECT and EXCEPT are not implemented in MySQL 8. There is also INTERSECT ALL and EXCEPT ALL, but they are not implemented in MySQL 8 either.

In addition, you can combine multiple set operators to achieve more complex results. For instance, you could use the EXCEPT operator to remove overlapping records between two result sets and then use UNION to merge the remaining results with data from a third table. This approach allows you to filter out intersections and bring together disparate data sets in a single result.

Additionally, sorting compound query results can be done using the ORDER BY clause after the last query, but only columns from the first query can be referenced. To avoid errors, it's best to use consistent aliases for columns across the queries. Furthermore, INTERSECT has higher precedence over other set operators, and queries are evaluated from top to bottom unless explicitly grouped with parentheses (though MySQL doesn’t support this).