



SQL Standard

SQL Subtract

7 months ago • by John Otieno

One of the most common and popular operations when working with numerical or date dataset is determining the difference or minus between two numerical values.

In SQL, a subtraction allows us to retrieve the data that is present in one set but not in another. This comes into play as some sort of negation on the subquery. Some common usage include locating the difference between two datasets, determining the missing records, data comparisons, or locating the changes between versions of the dataset.

In this tutorial, we will learn how to use the SQL Subtract feature and learn how it can help us when working with relational databases.

SQL Subtract

Subtraction operations in SQL databases may vary depending on the database engine. For example, we can achieve this using the MINUS operator in Oracle. We can use the EXCEPT operator in SQL Server. While we use the NOT IN operator in MySQL and PostgreSQL.

For this tutorial, we will focus on the NOT IN operator as supported by both MySQL and PostgreSQL.

The following shows the basic syntax of the NOT IN operator in MySQL:

```
SELECT column1, column2, ...  
FROM table1  
WHERE column1 NOT IN (SELECT column1 FROM table2);
```

In the given example syntax, we start with a select statement and define the columns that we wish to select from the table.

Next, we use the FROM keyword to set the source table from which we wish to select the data. We also introduce the WHERE clause which allows us to filter the results based on specific conditions.

This is where we introduce the NOT IN operator which tells the database that we wish to subtract or remove the records that match a specific condition from the result.

Example: Find the Missing Records

One of the common tasks of the SQL subtract is locating the missing records. Suppose we have two tables: one that stores all employees while the other stores the former employees.

Suppose we wish to locate the employees that are still active in the company. We can use the SQL subtract operation as shown in the following query:

```
CREATE TABLE employees (  
    id INT PRIMARY KEY AUTO_INCREMENT,  
    name VARCHAR(50)  
);  
  
drop table if exists former_employees;  
  
CREATE TABLE former_employees (  
    id INT PRIMARY KEY AUTO_INCREMENT,  
    name VARCHAR(50)  
);  
  
INSERT INTO employees (name)  
VALUES  
    ('Arthur S'),  
    ('Alice K'),  
    ('Johnson A');  
  
INSERT INTO former_employees (name)  
VALUES  
    ('Brown Q'),  
    ('Alice K'),  
    ('Michael R');  
  
SELECT id, name  
FROM employees  
WHERE id NOT IN (SELECT id FROM former_employees);
```

In this query, we use the NOT IN operator to subtract the ID value that exists in the former “employee” table from the “employee” table.

Conclusion

In this tutorial, we covered the basics of SQL subtraction using the NOT IN operator as supported by both MySQL and PostgreSQL.

ABOUT THE AUTHOR



John Otieno

My name is John and am a fellow geek like you. I am passionate about all things computers from Hardware, Operating systems to Programming. My dream is to share my knowledge with the world and help out fellow geeks. Follow my content by subscribing to LinuxHint mailing list