



## Inside the Order of Execution and Writing of SQL Queries: What You Need to Know?



Durgesh Kumar

Founder @LetsBeAnalyst | Driving growth through data-backed solutions for individuals, entrepreneurs, and small...

February 17, 2023

In SQL, understanding the order of execution is critical to writing efficient and effective queries. The order of execution determines the sequence in which the different parts of the query are evaluated, and can have a significant impact on performance.

The order of execution in SQL follows a specific set of rules. First, the FROM clause is evaluated, followed by the WHERE clause, then the SELECT clause. The GROUP BY clause is evaluated next, followed by the HAVING clause, and finally, the ORDER BY clause.

It's important to understand how each of these clauses works and interacts with the others in order to optimize the performance of your SQL queries. By following best practices and understanding the order of execution, you can improve the efficiency of your queries and get the results you need more quickly.

here are the details of each step in the order of execution of SQL queries:

### FROM clause:

The FROM clause is the first step in the order of execution of SQL queries. In this step, the tables to be used in the query are specified.

### WHERE clause:

The WHERE clause filters the data based on the specified conditions. It is used to filter the data in the tables to the required subset for processing.

### GROUP BY clause:

The GROUP BY clause groups the data based on the specified columns. This step is used to create groups of data based on the values of one or more columns.

#### HAVING clause:

The HAVING clause filters the data based on the grouped data. It is used to filter the grouped data to the required subset for processing.

#### SELECT clause:

The SELECT clause specifies the columns to be displayed in the output. In this step, the columns are selected, and the required calculations are performed on them.

#### DISTINCT clause:

The DISTINCT clause removes the duplicate rows from the output.

#### ORDER BY clause:

The ORDER BY clause sorts the rows in the output based on the specified columns and sort order.

#### LIMIT clause:

The LIMIT clause is used to limit the number of rows in the output.

It's important to note that not all queries will use all of these steps, and some may use additional steps, such as subqueries. The order of execution may also be affected by the specific syntax of the query being used. Understanding the order of execution is important for optimizing query performance and avoiding errors.

### SQL Query Writing: order of writing

The order of writing SQL queries refers to the sequence in which you write the various parts of a query to retrieve data from a database. Here is a brief overview of the order in which different parts of an SQL query are typically written:

**SELECT:** This is usually the first keyword in a SQL query and is used to specify the columns or expressions that you want to retrieve from the database.

**FROM:** This is the next keyword in the query and is used to specify the table or tables from which you want to retrieve the data.

**JOIN:** If you want to combine data from multiple tables, you use the JOIN keyword to specify the type of join you want to perform and the tables you want to join.

**WHERE:** This keyword is used to filter the data based on specific criteria. You can specify one or more conditions to filter the data.

**GROUP BY:** This keyword is used to group the data based on one or more columns in the SELECT statement. You can also use aggregate functions like SUM or COUNT to perform calculations on the grouped data.

**HAVING:** This keyword is used to filter the grouped data based on specific criteria. It works similarly to the WHERE keyword but filters the data after it has been grouped.

**ORDER BY:** This keyword is used to sort the data based on one or more columns. You can sort the data in ascending or descending order.

**LIMIT/OFFSET:** These keywords are used to limit the number of rows returned by the query or to skip a certain number of rows.

It's important to note that the order of writing SQL queries is not always rigid and can vary depending on the specific query and the database being used. However, following a standard order can make your queries more organized and easier to understand.

Enjoyed this article?

Follow to never miss an update.

Comments

👍 29 • 4 comments



Durgesh Kumar



Founder @LetsBeanalyst | Driving growth through data-backed solutions for individuals, entrepreneurs, and small enterprises. | Delivering custom analytics training, interactive dashboards, and robust data architectures.

Like

Comment

Add a comment...

+ Follow

Most recent

- About

Professional Community Policies

Privacy & Terms

Sales Solutions

Safety Center
- Accessibility

Careers

Ad Choices

Mobile
- Talent Solutions

Marketing Solutions

Advertising

Small Business
- Questions?

Visit our Help Center.

Manage your account and privacy

Go to your Settings.

Recommendation transparency

Learn more about Recommended Content.
- Select Language

English (English)