**SQL Interview Questions**

**What are user-defined functions?**

We can use User-defined functions in PL/SQL or Java to provide functionality that is not available in SQL or SQL built-in functions. SQL functions and User-defined functions can appear anywhere, that is, wherever an expression occurs.

For example, it can be used in:

* Select a list of SELECT statements.
* Condition of the WHERE clause.
* CONNECT BY, ORDER BY, START WITH, and GROUP BY
* The VALUES clause of the INSERT statement.
* The SET clause of the UPDATE statement.

Types:

**Scalar User-Defined Function** : A Scalar user-defined function returns one of the scalar data types. Text, image, and timestamp data types are not supported. These are the type of user-defined functions that most developers are used to in other programming languages. You pass in 0 to many parameters and you get a return value.

**Inline Table-Value User-Defined Function :** An Inline Table-Value user-defined function returns a table data type and is an exceptional alternative to a view as the user-defined function can pass parameters into a T-SQL select command and, in essence, provide us with a parameterized, non-updateable view of the underlying tables.

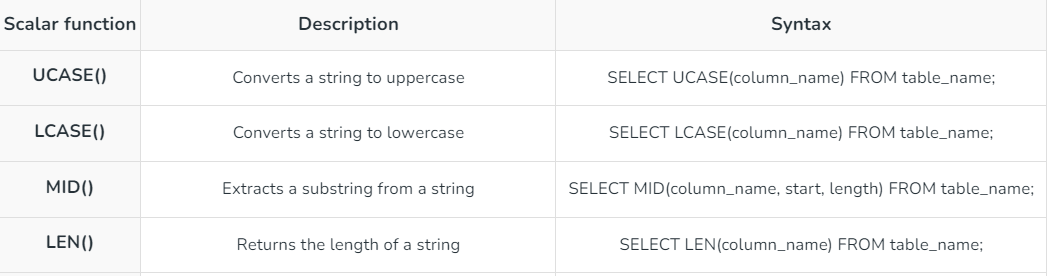
**Multi-statement Table-Value User-Defined Function**:  Multi-Statement Table-Value user-defined function returns a table and is also an exceptional alternative to a view, as the function can support multiple T-SQL statements to build the final result where the view is limited to a single SELECT statement.

**What are aggregate and scalar functions?**

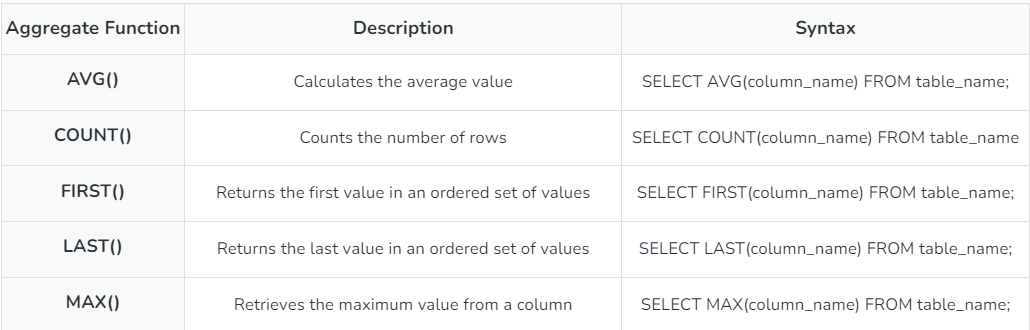
For doing operations on data SQL has many built-in functions, they are categorized into two categories and further sub-categorized into seven different functions under each category. The categories are:

**SQL Scalar Functions**are built-in functions that operate on a single value and return a single value.

Scalar functions in SQL helps in efficient data manipulation and simplification of complex calculations in SQL queries.



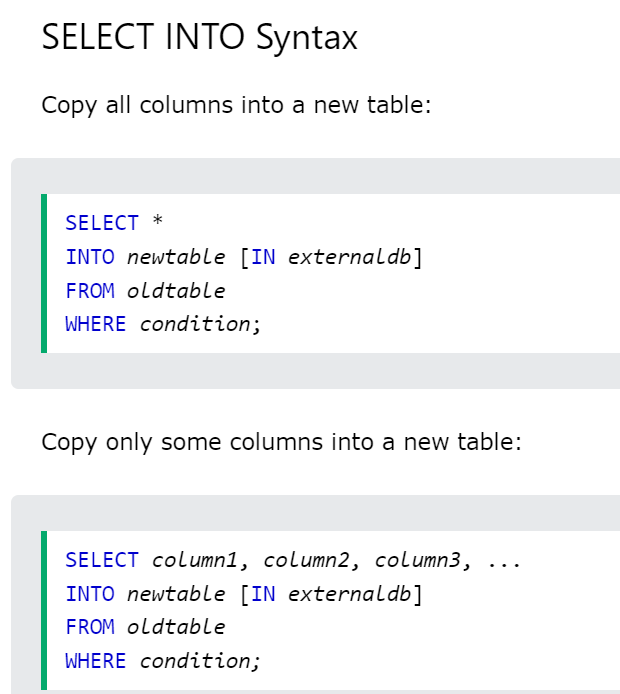
[**SQL Aggregate Functions**](https://www.geeksforgeeks.org/aggregate-functions-in-sql/) operate on a data group and return a singular output. They are mostly used with the [**GROUP BY**](https://www.geeksforgeeks.org/sql-group-by/)clause to summarize data.



### How to copy tables in SQL?

If you want to copy the data of one SQL table into another SQL table in the same SQL server, then it is possible by using the SELECT INTO statement in SQL.

The SELECT INTO statement in Structured Query Language copies the content from one existing table into the new table. SQL creates the new table by using the structure of the existing table.



**Global & local temp tables**

**Local**

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer error

AI-generated content may be incorrect.

**Global temp table**

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

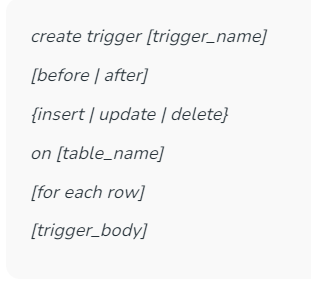
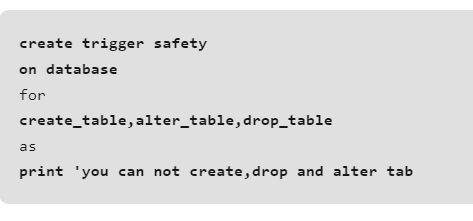
**Triggers**

A trigger is a set of SQL statements that reside in system memory with unique names. It is a specialized category of stored procedure that is called automatically when a database server event occurs. Each trigger is always associated with a table.

A **trigger is called a special procedure** because it cannot be called directly like a stored procedure. The key distinction between the trigger and procedure is that a trigger is called automatically when a data modification event occurs against a table. A stored procedure, on the other hand, must be invoked directly.

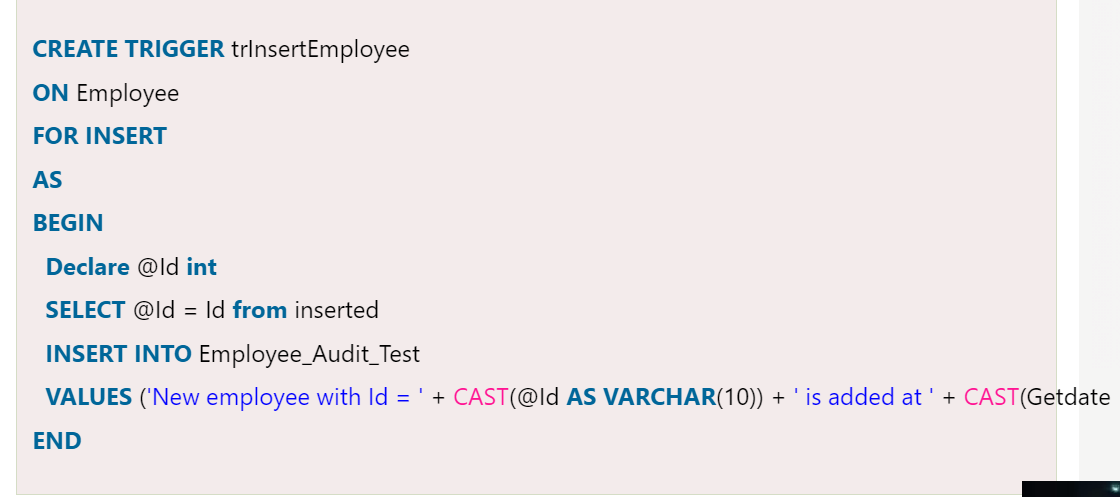
* We cannot manually execute/invoked triggers.
* Triggers have no chance of receiving parameters.
* A transaction cannot be committed or rolled back inside a trigger.

Syntax:

When to use triggers:

Triggers will be helpful when we need to execute some events automatically on certain desirable scenarios. **For example**, we have a constantly changing table and need to know the occurrences of changes and when these changes happen. If the primary table made any changes in such scenarios, we could create a trigger to insert the desired data into a separate table.

Eg: 

**What is the difference between primary key and unique constraints?**

The primary key cannot have NULL values, the unique constraints can have NULL values. There is only one primary key in a table, but there can be multiple unique constraints. The primary key creates the clustered index automatically but the unique key does not.

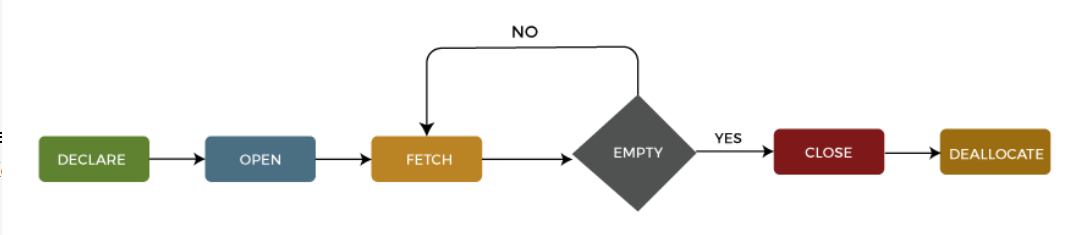
### What is the On Delete cascade constraint?

An ‘ON DELETE CASCADE’ constraint is used in MySQL to delete the rows from the child table automatically when the rows from the parent table are deleted. For more details, please read [MySQL – On Delete Cascade constraint](https://www.geeksforgeeks.org/mysql-on-delete-cascade-constraint)article.

### What is a Cursor?

A cursor in SQL Server is a database object that allows us to retrieve each row at a time and manipulate its data. A cursor is nothing more than a pointer to a row. It's always used in conjunction with a SELECT statement. It is usually a collection of [SQL](https://www.javatpoint.com/sql-tutorial) logic that loops through a predetermined number of rows one by one

The [SQL Server](https://www.javatpoint.com/sql-server-tutorial) **cursor's purpose is to update the data row by row, change it, or perform calculations that are not possible when we retrieve all records at once**. It's also useful for performing administrative tasks like SQL Server database backups in sequential order. Cursors are mainly used in the development, DBA, and ETL processes.



Demerits

* Cursor consumes network resources by requiring a network roundtrip each time it fetches a record.
* A cursor is a memory resident set of pointers, which means it takes some memory that other processes could use on our machine.
* It imposes locks on a portion of the table or the entire table when processing data.
* The cursor's performance and speed are slower because they update table records one row at a time.
* Cursors are quicker than while loops, but they do have more overhead.

## How can we avoid cursors?

The main job of cursors is to traverse the table row by row. The easiest way to avoid cursors are given below:

**Using the SQL while loop**

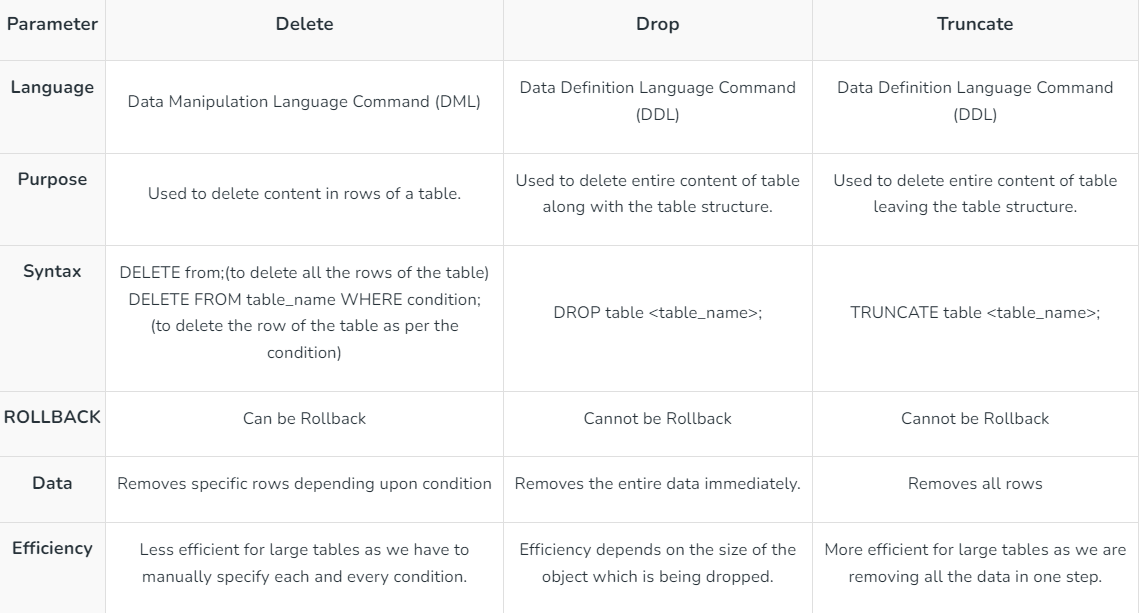
The easiest way to avoid the use of a cursor is by using a while loop that allows the inserting of a result set into the temporary table.

**User-defined functions**

Sometimes cursors are used to calculate the resultant row set. We can accomplish this by using a user-defined function that meets the requirements.

**Using Joins**

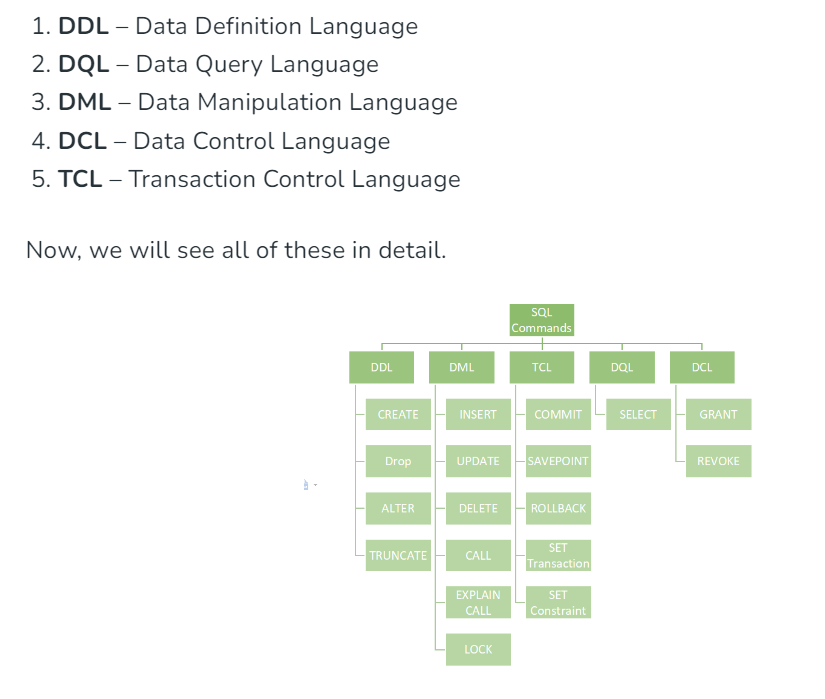
**Delete vs Drop v Truncate**



### What is a Live Lock?

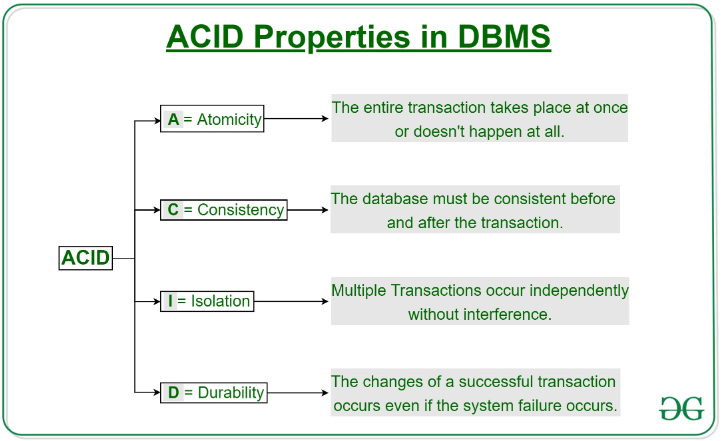
**Livelock** occurs when two or more processes continually repeat the same interaction in response to changes in the other processes without doing any useful work. These processes are not in the waiting state, and they are running concurrently. This is different from a deadlock because in a deadlock all processes are in the waiting state.

**DML VS DDL VS DCL**



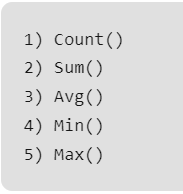
### What are ACID properties?

A [**transaction**](https://www.geeksforgeeks.org/sql-transactions) is a single logical unit of work that accesses and possibly modifies the contents of a database. Transactions access data using read-and-write operations. In order to maintain consistency in a database, before and after the transaction, certain properties are followed. These are called **ACID** properties. **ACID** (Atomicity, Consistency, Isolation, Durability) is a set of properties that guarantee that database transactions are processed reliably.



### What is the need for group functions in SQL?

In database management, group functions, also known as aggregate functions,  is a function where the values of multiple rows are grouped together as input on certain criteria to form a single value of more significant meaning.

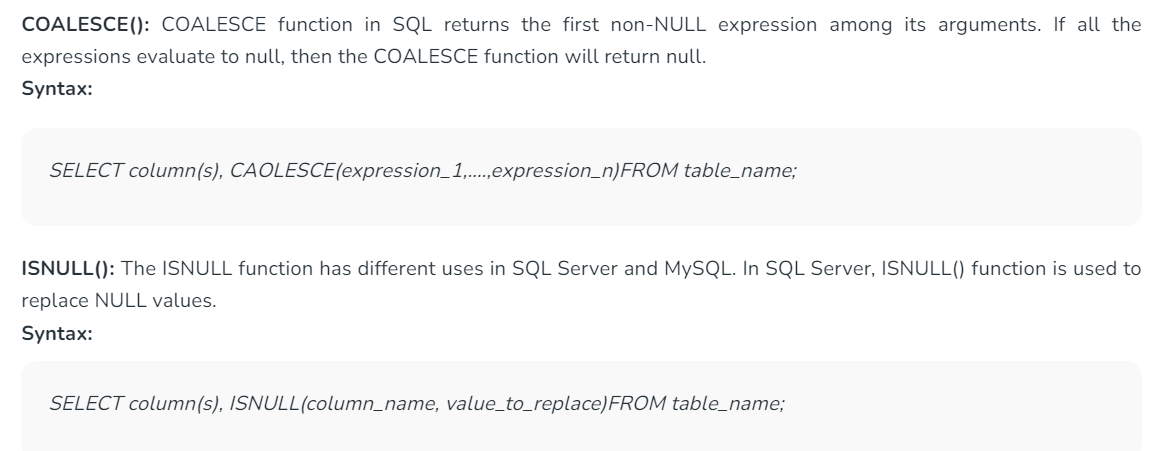


### How do we avoid getting duplicate entries in a query without using the distinct keyword?

DISTINCT is useful in certain circumstances, but it has drawbacks that it can increase the load on the query engine to perform the sort (since it needs to compare the result set to itself to remove duplicates). We can remove duplicate entries using the following options:

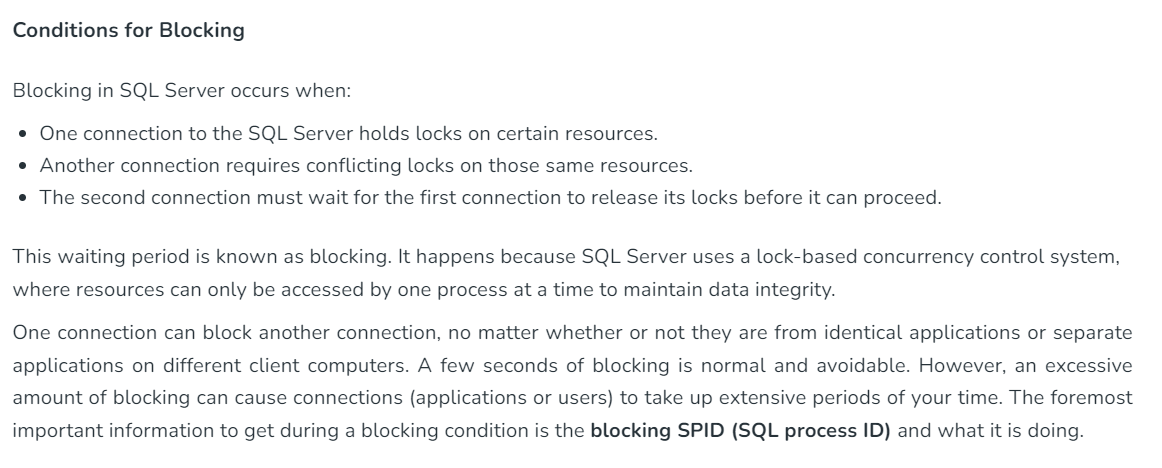
* Remove duplicates using row numbers.
* Remove duplicates using self-Join.
* Remove duplicates using group by.

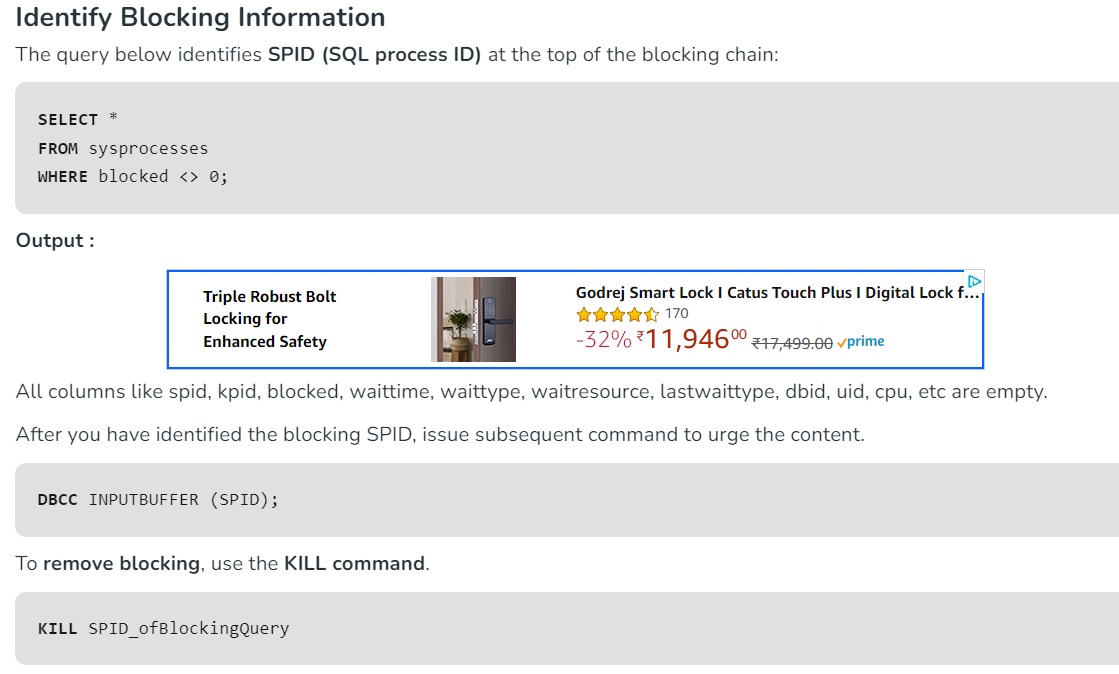
### What is the difference between COALESCE() & ISNULL()?



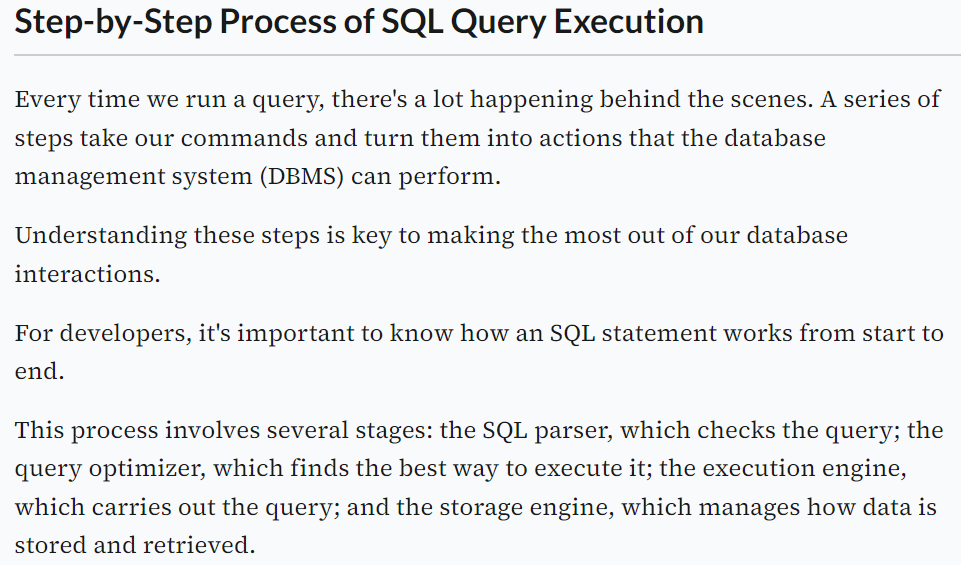
# **Blocking in SQL Server**

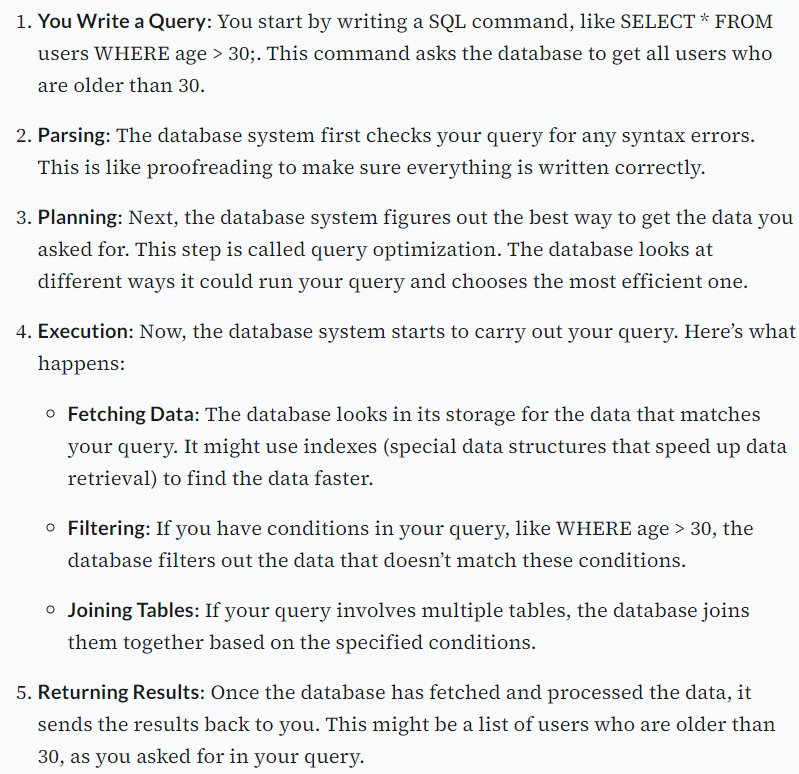
**Blocking in SQL Server** occurs when two or more processes require the same resource, but only one process can access a resource at a time. This results in a situation where if one process is using a resource, any subsequent processes will have to wait until the first process has finished using it

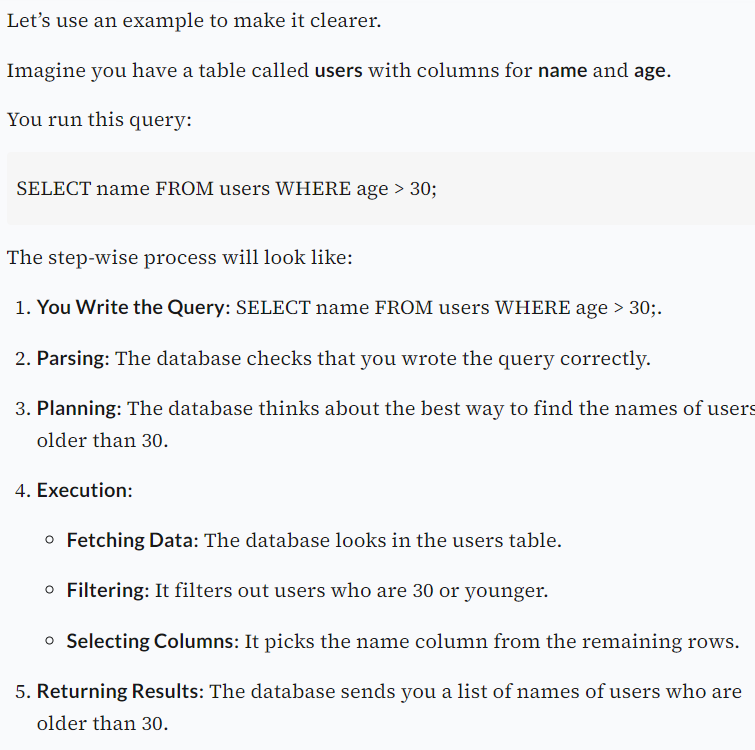


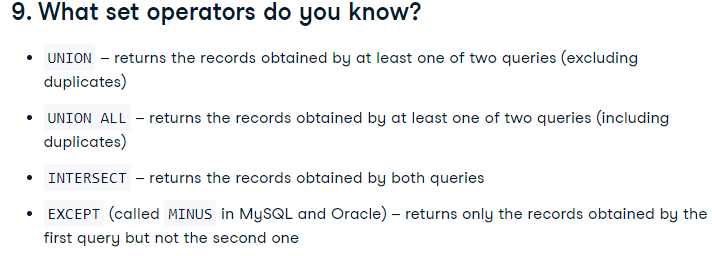


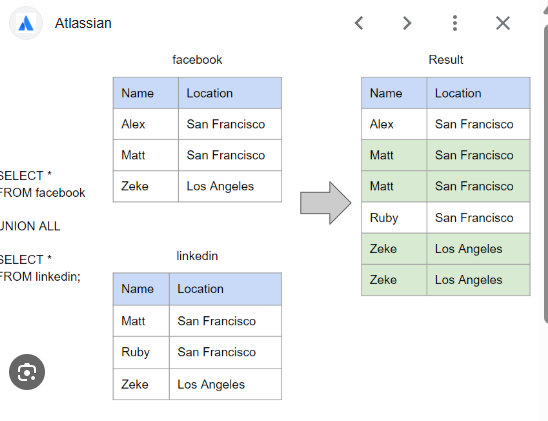
How SQL query works internally?









 A screenshot of a computer

AI-generated content may be incorrect.

