SQL (Structured Query Language) is used to perform operations on the records stored in the database, such as updating records, inserting records, deleting records, creating and modifying database tables, views, etc.

SQL is not a database system, but it is a query language. "E.F. Codd" in 1970.

**Why we use it ?**

* The basic use of SQL for data professionals and SQL users is to insert, update, and delete the data from the relational database.
* SQL allows the data professionals and users to retrieve the data from the relational database management systems.
* It also helps them to describe the structured data.
* It allows SQL users to create, drop, and manipulate the database and its tables.
* It also helps in creating the view, stored procedure, and functions in the relational database.
* It allows you to define the data and modify that stored data in the relational database.
* It also allows SQL users to set the permissions or constraints on table columns, views, and stored procedures.
* Process of SQL
* When we are executing the command of SQL on any Relational database management system, then the system automatically finds the best routine to carry out our request, and the SQL engine determines how to interpret that particular command.
* Advantages of SQL
* **1. No programming needed**
* SQL does not require a large number of coding lines for managing the database systems. We can easily access and maintain the database by using simple SQL syntactical rules. These simple rules make the SQL user-friendly.
* **2. High-Speed Query Processing**
* A large amount of data is accessed quickly and efficiently from the database by using SQL queries. Insertion, deletion, and updation operations on data are also performed in less time.

**3. Standardized Language**

* SQL follows the long-established standards of ISO and ANSI, which offer a uniform platform across the globe to all its users.
* **4. Portability**
* The structured query language can be easily used in desktop computers, laptops, tablets, and even smartphones. It can also be used with other applications according to the user's requirements.
* **5. Interactive language**
* We can easily learn and understand the SQL language. This language is also used for receiving the answers to complex queries in a few seconds.
* **6. More than one Data View**
* The SQL language also helps in making the multiple views of the database structure for the different database users.
* Disadvantages of SQL
* **1. Cost**
* The operation cost of some SQL versions is high. That's why some programmers cannot use the Structured Query Language.
* **2. Interface is Complex**
* Another big disadvantage is that the interface of Structured query language is difficult, which makes it difficult for SQL users to use and manage it.
* **3. Partial Database control**
* The business rules are hidden. So, the data professionals and users who are using this query language cannot have full database control.
* What is SQL Operator?
* The SQL reserved words and characters are called operators, which are used with a WHERE clause in a SQL query. In SQL, an operator can either be a unary or binary operator. The unary operator uses only one operand for performing the unary operation, whereas the binary operator uses two operands for performing the binary operation.

|  |  |
| --- | --- |
| **SQL Operator Symbols** | **Operators** |
| \*\* | Exponentiation operator |
| +, - | Identity operator, Negation operator |
| \*, / | Multiplication operator, Division operator |
| +, -, || | Addition (plus) operator, subtraction (minus) operator, String Concatenation operator |
| =, !=, <, >, <=, >=, IS NULL, LIKE, BETWEEN, IN | Comparison Operators |
| NOT | Logical negation operator |
| && or AND | Conjunction operator |
| OR | Inclusion operator |

## Types of Operator

1. SQL Arithmetic Operators
2. SQL Comparison Operators
3. SQL Logical Operators
4. SQL Set Operators
5. SQL Bit-wise Operators
6. SQL Unary Operators

## SQL Arithmetic Operators

The **Arithmetic Operators** perform the mathematical operation on the numerical data of the SQL tables. These operators perform addition, subtraction, multiplication, and division operations on the numerical operands.

SQL Comparison Operators

The **Comparison Operators** in SQL compare two different data of SQL table and check whether they are the same, greater, and lesser. The SQL comparison operators are used with the WHERE clause in the SQL queries

1. SQL Equal Operator (=)
2. SQL Not Equal Operator (!=)
3. SQL Greater Than Operator (>)
4. SQL Greater Than Equals to Operator (>=)
5. SQL Less Than Operator (<)\
6. SQL Less Than Equals to Operator (<=)

## SQL Logical Operators

The **Logical Operators** in SQL perform the Boolean operations, which give two results **True and False.** These operators provide **True** value if both operands match the logical condition.

## SQL Logical Operators

The **Logical Operators** in SQL perform the Boolean operations, which give two results **True and False.** These operators provide **True** value if both operands match the logical condition.

1. SQL ALL operator
2. SQL AND operator
3. SQL OR operator
4. SQL BETWEEN operator
5. SQL IN operator
6. SQL NOT operator
7. SQL ANY operator
8. SQL LIKE operator

### **SQL ALL Operator**

This operator is always used with the following statement:

1. SELECT,
2. HAVING, and
3. WHERE.

### **SQL ANY Operator**

The **ANY operator** in SQL shows the records when any of the values returned by the sub-query meet the condition.

The ANY logical operator must match at least one record in the inner query and must be preceded by any SQL comparison operator.

### **SQL LIKE Operator**

The **LIKE operator** in SQL shows those records from the table which match with the given pattern specified in the sub-query.

The percentage (%) sign is a wildcard which is used in conjunction with this logical operator.

This operator is used in the WHERE clause with the following three statements:

1. SELECT statement
2. UPDATE statement
3. DELETE statement

## SQL Set Operators

The **Set Operators** in SQL combine a similar type of data from two or more SQL database tables. It mixes the result, which is extracted from two or more SQL queries, into a single result.

Set operators combine more than one select

statement in a single query and return a specific result set.

**Following are the various set operators which are performed on the similar data stored in the two SQL database tables:**

1. SQL Union Operator
2. SQL Union ALL Operator
3. SQL Intersect Operator
4. SQL Minus Operator

SQL Unary Operators

The **Unary Operators** in SQL perform the unary operations on the single data of the SQL table, i.e., these operators operate only on one operand.

These types of operators can be easily operated on the numeric data value of the SQL table.

1. SQL Unary Positive Operator
2. SQL Unary Negative Operator
3. SQL Unary Bitwise NOT Operator

SQL Bitwise Operators

The **Bitwise Operators** in SQL perform the bit operations on the Integer values. To understand the performance of Bitwise operators, you just knew the basics of Boolean algebra.

**Following are the two important logical operators which are performed on the data stored in the SQL database tables:**

1. Bitwise AND (&)
2. Bitwise OR(|)

# **SQL Create Database**

In SQL, the 'Create Database' statement is a first step for storing the structured data in the database.

* The database we want to create should be a simple and unique name, which can be easily identified.
* Database name should be no more than 128 characters.

### **Syntax of Create Database statement in SQL**

1. **CREATE** **DATABASE** Database\_Name;

# **SQL DROP Database**

The SQL Drop Database statement deletes the existing database permanently from the database system

### **Syntax of Drop Database Statement in SQL**

1. **DROP** **DATABASE** Database\_Name;

# **SQL RENAME Database**

In some situations, database users and administrators want to change the name of the database for some technical reasons. So, the **Rename Database** statement in SQL is used to change the name of the existing database.

### **Syntax of Rename Database in MySQL**

1. RENAME **DATABASE** old\_database\_name **TO** new\_database\_name;

# **SQL Table**

Table is a collection of data, organized in terms of rows and columns. In DBMS term, table is known as relation and row as tuple.

Table is the simple form of data storage. A table is also considered as a convenient representation of relations.

1. **create** **table** "tablename"
2. ("column1" "data type",
3. "column2" "data type",
4. ...
5. "columnN" "data type");

Ex.

1. **CREATE** **TABLE** Employee
2. (
3. EmployeeID **int**,
4. FirstName **varchar**(255),
5. LastName **varchar**(255),
6. Email **varchar**(255),
7. AddressLine **varchar**(255),
8. City **varchar**(255)
9. );

# **SQL DROP TABLE**

A SQL DROP TABLE statement is used to delete a table definition and all data from a table.

1. **DROP** **TABLE** "table\_name";

# **SQL DELETE TABLE**

The DELETE statement is used to delete rows from a table. If you want to remove a specific row from a table you should use WHERE condition.

1. **DELETE** **FROM** table\_name [**WHERE** condition];

# **SQL RENAME TABLE**

Any database user can easily change the name by using the RENAME TABLE and ALTER TABLE statement in Structured Query Language.

The RENAME TABLE and ALTER TABLE syntax help in changing the name of the table.

## Syntax of RENAME statement in SQL

1. RENAME old\_table \_name To new\_table\_name ;

# **SQL TRUNCATE TABLE**

A truncate SQL statement is used to remove all rows (complete data) from a table. It is similar to the DELETE statement with no WHERE clause.

1. **TRUNCATE** **TABLE** table\_name;
2. **TRUNCATE** **TABLE** Employee;

# **SQL COPY TABLE**

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.

## Syntax of SELECT INTO statement in SQL

1. SELECT \* INTO New\_table\_name FROM old\_table\_name;

SELECT \* INTO Car\_Details FROM Cars;