

UNIT 1:

The SQL Language

SQL

- SQL stands for Structured Query Language. It is used for storing and managing data in relational database management system (RDMS).
- It is a standard language for Relational Database System. It enables a user to create, read, update and delete relational databases and tables.
- All the RDBMS like MySQL, Informix, Oracle, MS Access and SQL Server use SQL as their standard database language.
- SQL allows users to query the database in a number of ways, using English-like statements.

Rules:

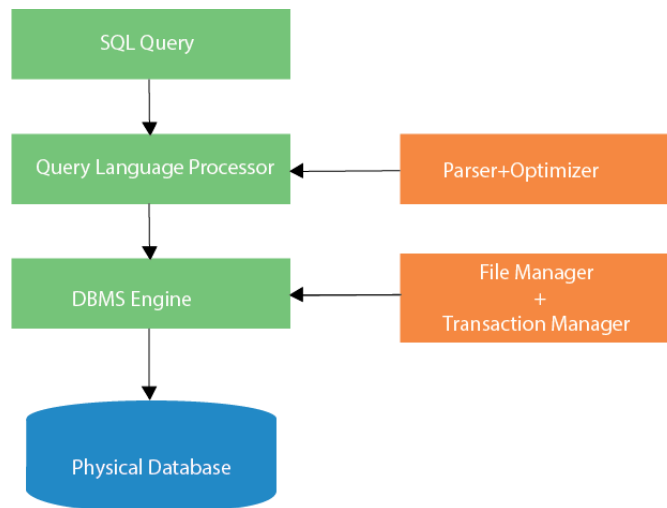
SQL follows the following rules:

- Structure query language is not case sensitive. Generally, keywords of SQL are written in uppercase.
- Statements of SQL are dependent on text lines. We can use a single SQL statement on one or multiple text line.
- Using the SQL statements, you can perform most of the actions in a database.
- SQL depends on tuple relational calculus and relational algebra.

SQL process:

- When an SQL command is executing for any RDBMS, then the system figure out the best way to carry out the request and the SQL engine determines that how to interpret the task.

- In the process, various components are included. These components can be optimization Engine, Query engine, Query dispatcher, classic, etc.
- All the non-SQL queries are handled by the classic query engine, but SQL query engine won't handle logical files.



Oracle Architecture

Oracle Database is a relational database management system (RDBMS) that Oracle Corporation created and marketed. It is one of the most popular RDBMSs on the market and is used to store and retrieve data for a wide range of applications. Oracle Database is well-known for its dependability, scalability, and performance, and it is compatible with a wide range of programming languages and development frameworks. It includes data warehousing, online transaction processing, and advanced analytics, as well as high availability, disaster recovery, and security.

Oracle Database employs a client-server architecture, with the database server and client connections running in separate processes. The server process manages data and resources, and client connections communicate with the server to access and manipulate data. SQL, PL/SQL (Oracle's proprietary procedural language), and Java are among the programming interfaces supported by Oracle Database. It also includes tools for database management and administration, such as Oracle Enterprise Manager and SQL Plus.

The oracle database architecture consists of:

- Memory structure(Instances)
- Database system
- Processes

Oracle Instances

The instance is a collection of two things:

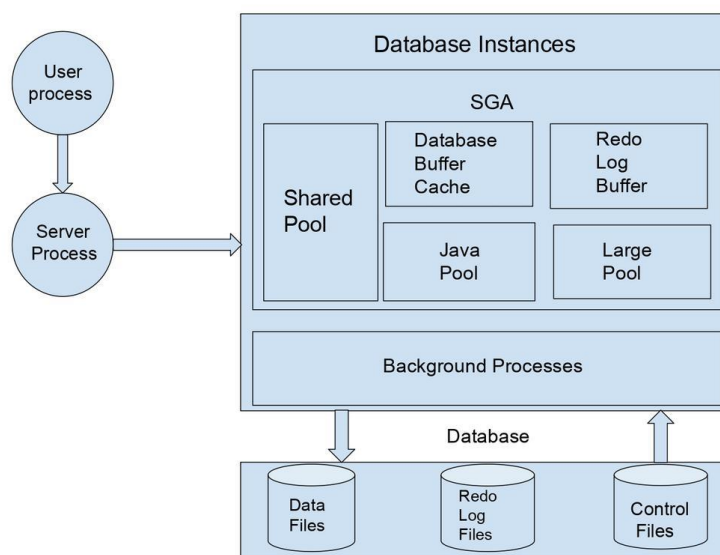
SGA: It stands for System Global Area. It is a shared memory area. Whenever a database instance starts, some memory gets allocated and that memory is termed SGA. Along with memory allocation, one or more background processes will. SGA is used to store data as well as control information about one database instance through its various subcomponents, Where each component is dedicated to a specific purpose. Various Components are:

- **Database buffer cache:** The Buffer Cache is a portion of the SGA that stores copies of data blocks read from datafiles. It is used to cache frequently accessed data blocks, reducing the number of disks I/Os required to access the data. This can improve performance by reducing disk I/O time and increasing the speed of data retrieval.
- **Redo log Buffer:** The most crucial structure for recovery operations is the redo log, which consists of two or more preallocated files that store all changes made to the database as they occur. Every instance of an Oracle Database has an associated redo log to protect the database in case of an instance failure.
- **Java pool:** The Java Pool is an optional portion of the SGA that is used by Java Virtual Machine (JVM) and related components. It is used to store Java-related data structures, such as Java classes and objects. This pool is used when the Oracle Database is configured to run Java applications or when using Oracle JVM.
- **Large Pool:** The Large Pool is an optional portion of the SGA that can be used for large memory allocation, such as backup and restore operations and I/O server processes. It is typically used to improve the performance of these operations by reducing the amount of disk I/O required.
- **Shared pool:** The Shared Pool is a portion of the SGA that contains shared memory structures, such as shared SQL and PL/SQL areas. It is used to store the parsed representation of SQL statements, execution plans, and PL/SQL program units. This allows for the efficient reuse of frequently executed statements, reducing the need for reparsing and improving performance.

Background processes: Oracle has a collection of processes that are called background processes. These processes are responsible for managing memory, performing I/O operations, and other maintenance activities. Following are some important background processes that are required:

- **System Monitor Process (SMON):** These processes are responsible for performing system-level recovery and maintenance activities.

- **Process Monitor Process (PMON):** The task of these processes is to monitor other background processes.
- **Database Writer Process (DBWR):** This process performs the task of writing data blocks from the Database Buffer Cache (present in SGA) to physical data files (Present in the Database system).
- **Log Writer Process (LGWR):** This process writes the Redo blocks from Redo Log Buffer (present in SGA) to Redo Log Files (present in the Database system).
- **CheckPoint (CKPT):** This process maintains data files and control files with the most recent checkpoint information.



Database System

The database system is suited to the storage system of a computer. The Database system is simply the storage of files. There are three categories of files that are situated in the database system and those are:-

- **Data files:** These files hold the actual data in the database.
- **Redo log files:** These files are used to hold the changes made in the database. Redo log files can be utilized during the database recovery process to retrieve the original information.
- **Control files:** It is a binary file that holds database status-related information like Database Name, Data File, and Redo Log file Names, File Locations, and Log Sequence Number.

There are other categories of files that contribute to database management.

- **Parameter file:** This file contains the parameters which define the way the database is expected to start up.
- **Password file:** This file holds the user passwords and thus maintains the security of databases.

Processes

There are two types of processes:

- **User process:** It is also known as the client process. The user actually connects to the instance with the help of user processes. The user process is stabilized when the user sends a connection request to the Oracle server.
- **Server process:** The server process connects the user to the database and performs the activities on the client's behalf as executing SQL statements or retrieving data from the database.

Client-Server Model

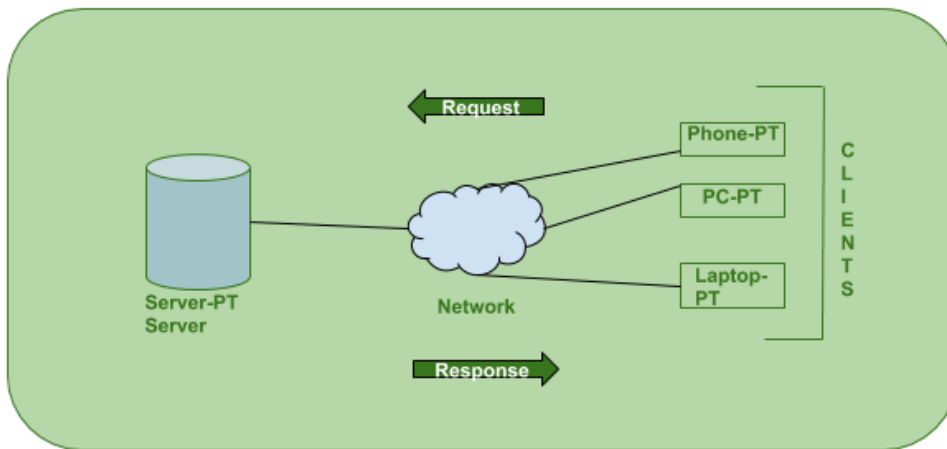
The Client-server model is a distributed application structure that partitions task or workload between the providers of a resource or service, called servers, and service requesters called clients. In the client-server architecture, when the client computer sends a request for data to the server through the internet, the server accepts the requested process and delivers the data packets requested back to the client. Clients do not share any of their resources. Examples of Client-Server Model are Email, World Wide Web, etc.

How the Client-Server Model works ?

In this article we are going to take a dive into the **Client-Server** model and have a look at how the **Internet** works via web browsers. This article will help us in having a solid foundation of the WEB and help in working with WEB technologies with ease.

- **Client:** When we talk the word **Client**, it means to talk of a person or an organization using a particular service. Similarly in the digital world a **Client** is a computer (**Host**) i.e. capable of receiving information or using a particular service from the service providers (**Servers**).
- **Servers:** Similarly, when we talk the word **Servers**, it means a person or medium that serves something. Similarly in this digital world a **Server** is a remote computer which provides information (data) or access to particular services.

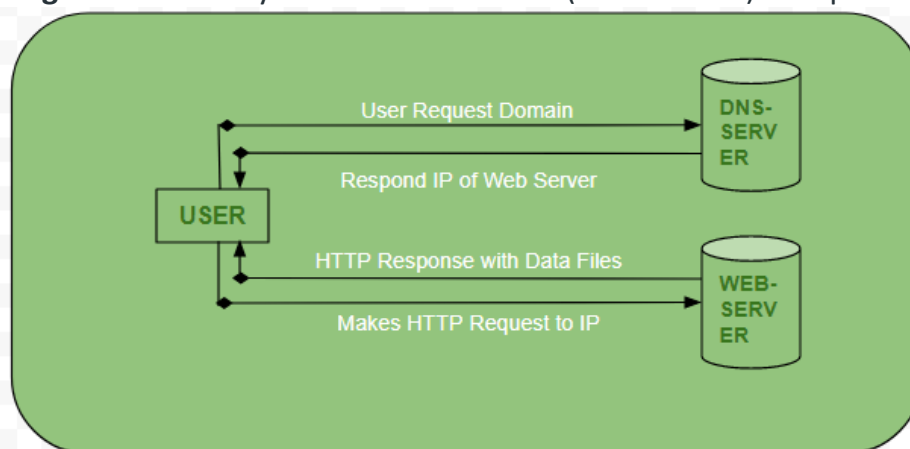
So, it's basically the **Client** requesting something and the **Server** serving it as long as it's present in the database.



How the browser interacts with the servers ?

There are few steps to follow to interact with the servers a client.

- User enters the **URL**(Uniform Resource Locator) of the website or file. The Browser then requests the **DNS**(DOMAIN NAME SYSTEM) Server.
- **DNS Server** lookup for the address of the **WEB Server**.
- **DNS Server** responds with the **IP address** of the **WEB Server**.
- Browser sends over an **HTTP/HTTPS** request to **WEB Server's IP** (provided by **DNS server**).
- Server sends over the necessary files of the website.
- Browser then renders the files and the website is displayed. This rendering is done with the help of **DOM** (Document Object Model) interpreter, **CSS** interpreter and **JS Engine** collectively known as the **JIT** or (Just in Time) Compilers.



Advantages of Client-Server model:

- Centralized system with all data in a single place.
- Cost efficient requires less maintenance cost and Data recovery is possible.
- The capacity of the Client and Servers can be changed separately.

Disadvantages of Client-Server model:

- Clients are prone to viruses, Trojans and worms if present in the Server or uploaded into the Server.
- Server are prone to Denial of Service (DOS) attacks.
- Data packets may be spoofed or modified during transmission.
- Phishing or capturing login credentials or other useful information of the user are common and MITM(Man in the Middle) attacks are common.

Characteristics of SQL

- SQL is easy to learn.
- SQL is used to access data from relational database management systems.
- SQL can execute queries against the database.
- SQL is used to describe the data.
- SQL is used to define the data in the database and manipulate it when needed.
- SQL is used to create and drop the database and table.
- SQL is used to create a view, stored procedure, function in a database.
- SQL allows users to set permissions on tables, procedures, and views.

What Is SQL*Plus?

SQL*Plus is essentially an interactive query tool, with some scripting capabilities. It is a non-GUI, character-based tool that has been around since the dawn of the Oracle age. Using SQL*Plus, you can enter an SQL statement, such as a SELECT query, and view the results. You can also execute Data Definition Language (DDL) commands that allow you to maintain and modify your database. You can even enter and execute PL/SQL code. In spite of SQL*Plus's age and lack of "flash," it is a workhorse tool used day in and day out by database administrators, developers, and yes, even end users. As a database administrator, it is my tool of choice for managing the databases under my care. I use it to peek under the hood —

to explore the physical implementation of my database, and to create and manage users, tables, and tablespaces. In my role as a developer, SQL*Plus is the first tool that I fire up when I need to develop a query. In spite of all the fancy, GUI-based SQL generators contained in products such as PowerBuilder, Clear Access, and Crystal Reports, I still find it quicker and easier to build up and test a complex query in SQL*Plus before transferring it to whatever development tool I am using.

Uses for SQL*Plus

Originally developed simply as a way to enter queries and see results, SQL*Plus has been enhanced with scripting and formatting capabilities, and can now be used for many different purposes. The basic functionality is very simple. With SQL*Plus, you can do the following:

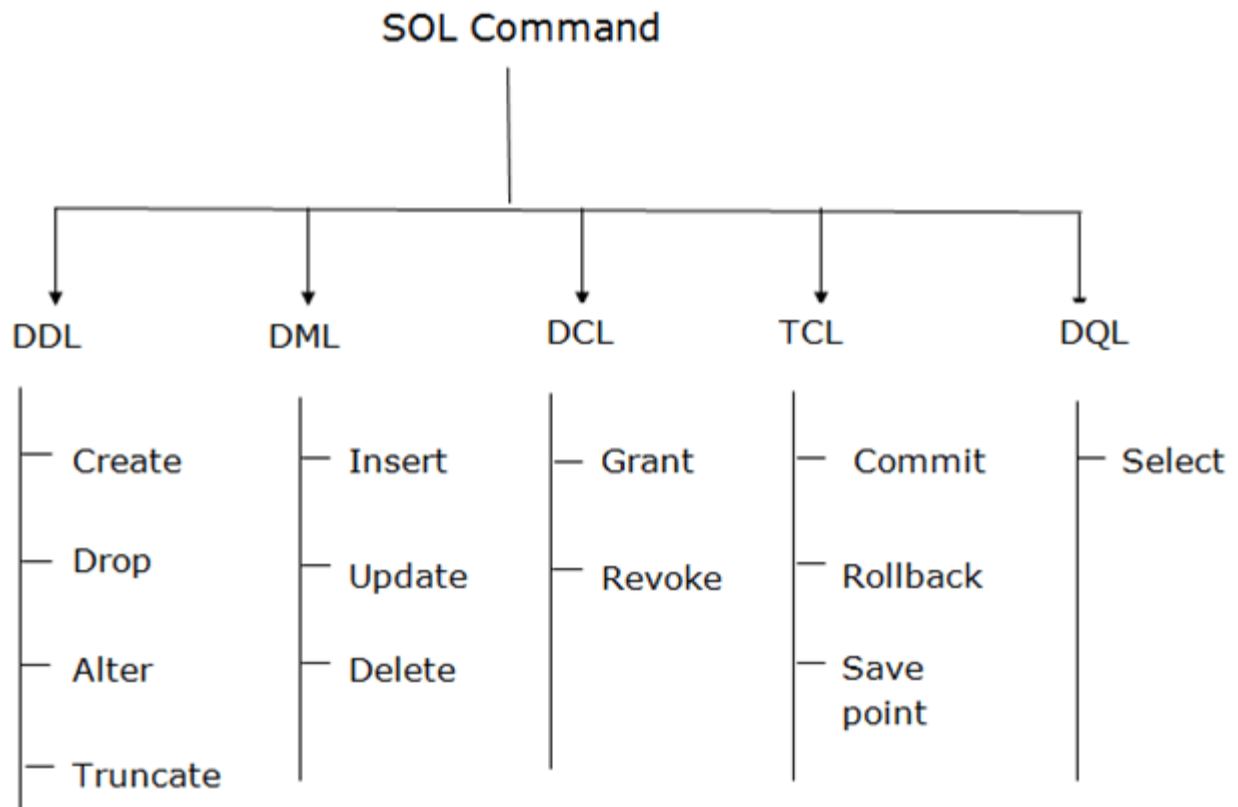
- Issue a SELECT query and view the results
- Insert, update, and delete data from database tables
- Submit PL/SQL blocks to the Oracle server for execution
- Issue DDL commands, such as those used to create, alter, or drop database objects such as tables, indexes, and users
- Execute SQL*Plus script files
- Write output to a file
- Execute procedures and functions that are stored in a database

SQL Commands

- SQL commands are instructions. It is used to communicate with the database. It is also used to perform specific tasks, functions, and queries of data.
- SQL can perform various tasks like create a table, add data to tables, drop the table, modify the table, set permission for users.

Types of SQL Commands

There are five types of SQL commands: DDL, DML, DCL, TCL, and DQL.



1. Data Definition Language (DDL)

- DDL changes the structure of the table like creating a table, deleting a table, altering a table, etc.
- All the command of DDL are auto-committed that means it permanently save all the changes in the database.

Here are some commands that come under DDL:

- CREATE
- ALTER
- DROP

- TRUNCATE

a. CREATE It is used to create a new table in the database.

Syntax:

1. CREATE TABLE TABLE_NAME (COLUMN_NAME DATATYPES[,....]);

Example:

1. CREATE TABLE EMPLOYEE(Name VARCHAR2(20), Email VARCHAR2(100), DOB DATE);

b. DROP: It is used to delete both the structure and record stored in the table.

Syntax

1. DROP TABLE table_name;

Example

1. DROP TABLE EMPLOYEE;

c. ALTER: It is used to alter the structure of the database. This change could be either to modify the characteristics of an existing attribute or probably to add a new attribute.

Syntax:

To add a new column in the table

1. ALTER TABLE table_name ADD column_name COLUMN-definition;

To modify existing column in the table:

1. ALTER TABLE table_name MODIFY(column_definitions....);

EXAMPLE

1. ALTER TABLE STU_DETAILS ADD(ADDRESS VARCHAR2(20));
2. ALTER TABLE STU_DETAILS MODIFY (NAME VARCHAR2(20));

d. TRUNCATE: It is used to delete all the rows from the table and free the space containing the table.

Syntax:

1. TRUNCATE TABLE table_name;

Example:

1. TRUNCATE TABLE EMPLOYEE;

2. Data Manipulation Language

- DML commands are used to modify the database. It is responsible for all form of changes in the database.
- The command of DML is not auto-committed that means it can't permanently save all the changes in the database. They can be rollback.

Here are some commands that come under DML:

- INSERT
- UPDATE
- DELETE

a. INSERT: The INSERT statement is a SQL query. It is used to insert data into the row of a table.

Syntax:

1. INSERT INTO TABLE_NAME
2. (col1, col2, col3,... col N)
3. VALUES (value1, value2, value3, valueN);

Or

1. INSERT INTO TABLE_NAME
2. VALUES (value1, value2, value3, valueN);

For example:

1. INSERT INTO javatpoint (Author, Subject) VALUES ("Sonoo", "DBMS");

b. UPDATE: This command is used to update or modify the value of a column in the table.

Syntax:

1. UPDATE table_name SET [column_name1= value1,...column_nameN = valueN] [WHERE CONDITION]

For example:

1. UPDATE students
2. SET User_Name = 'Sonoo'
3. WHERE Student_Id = '3'

c. DELETE: It is used to remove one or more row from a table.

Syntax:

1. DELETE FROM table_name [WHERE condition];

For example:

1. DELETE FROM javatpoint
2. WHERE Author="Sonoo";

3. Data Control Language

DCL commands are used to grant and take back authority from any database user.

Here are some commands that come under DCL:

- Grant
- Revoke

a. Grant: It is used to give user access privileges to a database.

Example

1. GRANT SELECT, UPDATE ON MY_TABLE TO SOME_USER, ANOTHER_USER;

b. Revoke: It is used to take back permissions from the user.

Example

1. REVOKE SELECT, UPDATE ON MY_TABLE FROM USER1, USER2;

4. Transaction Control Language

TCL commands can only use with DML commands like INSERT, DELETE and UPDATE only.

These operations are automatically committed in the database that's why they cannot be used while creating tables or dropping them.

Here are some commands that come under TCL:

- COMMIT
- ROLLBACK
- SAVEPOINT

a. Commit: Commit command is used to save all the transactions to the database.

Syntax:

1. COMMIT;

Example:

1. DELETE FROM CUSTOMERS
2. WHERE AGE = 25;
3. COMMIT;

b. Rollback: Rollback command is used to undo transactions that have not already been saved to the database.

Syntax:

1. ROLLBACK;

Example:

1. DELETE FROM CUSTOMERS
2. WHERE AGE = 25;
3. ROLLBACK;

c. SAVEPOINT: It is used to roll the transaction back to a certain point without rolling back the entire transaction.

Syntax:

1. SAVEPOINT SAVEPOINT_NAME;

5. Data Query Language

DQL is used to fetch the data from the database.

It uses only one command:

- o SELECT

a. SELECT: This is the same as the projection operation of relational algebra. It is used to select the attribute based on the condition described by WHERE clause.

Syntax:

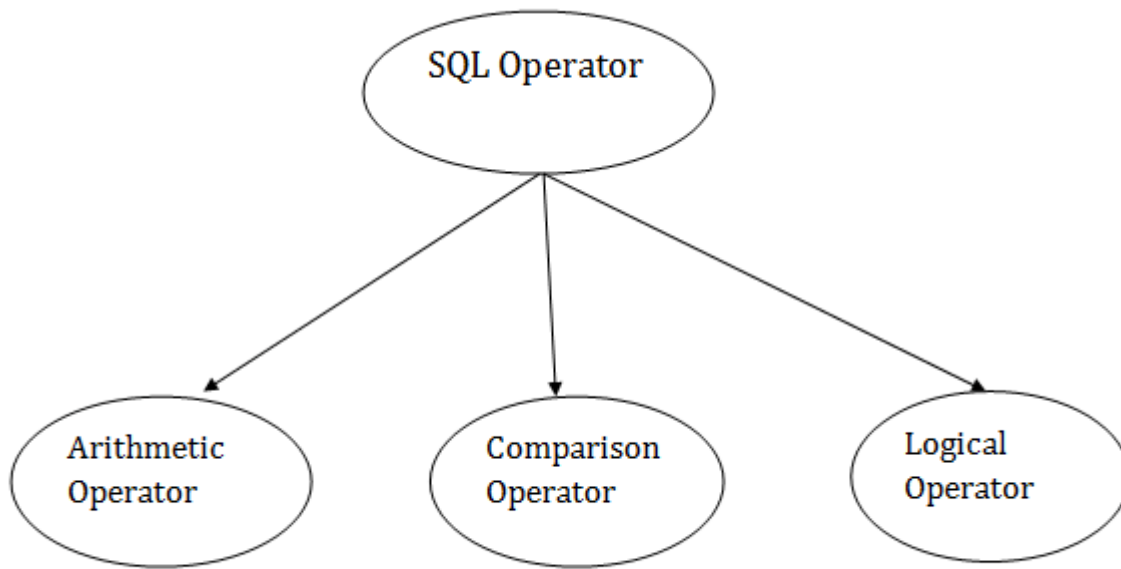
1. SELECT expressions
2. FROM TABLES
3. WHERE conditions;

For example:

1. SELECT emp_name
2. FROM employee
3. WHERE age > 20;

SQL Operator

There are various types of SQL operator:



SQL Arithmetic Operators

Let's assume 'variable a' and 'variable b'. Here, 'a' contains 20 and 'b' contains 10.

Operator	Description
+	It adds the value of both operands.
-	It is used to subtract the right-hand operand from the left-hand operand.
*	It is used to multiply the value of both operands.
/	It is used to divide the left-hand operand by the right-hand operand.
%	It is used to divide the left-hand operand by the right-hand operand and returns reminder.

SQL Comparison Operators:

Let's assume 'variable a' and 'variable b'. Here, 'a' contains 20 and 'b' contains 10.

Operator	Description	Example
=	It checks if two operands values are equal or not, if the values are equal then condition becomes true.	(a=b) is not true
!=	It checks if two operands values are equal or not, if values are not equal, then condition becomes true.	(a!=b) is true
<>	It checks if two operands values are equal or not, if values are not equal then condition becomes true.	(a<>b) is true
>	It checks if the left operand value is greater than right operand value, if yes then condition becomes true.	(a>b) is not true
<	It checks if the left operand value is less than right operand value, if yes then condition becomes true.	(a<b) is true
>=	It checks if the left operand value is greater than or equal to the right operand value, if yes then condition becomes true.	(a>=b) is not true
<=	It checks if the left operand value is less than or equal to the right operand value, if yes then condition becomes true.	(a<=b) is true
!<	It checks if the left operand value is not less than the right operand value, if yes then condition becomes true.	(a!=b) is not true
!>	It checks if the left operand value is not greater than the right operand value, if yes then condition becomes true.	(a!>b) is true

SQL Logical Operators

There is the list of logical operator used in SQL:

Operator	Description
----------	-------------

ALL	It compares a value to all values in another value set.
AND	It allows the existence of multiple conditions in an SQL statement.
ANY	It compares the values in the list according to the condition.
BETWEEN	It is used to search for values that are within a set of values.
IN	It compares a value to that specified list value.
NOT	It reverses the meaning of any logical operator.
OR	It combines multiple conditions in SQL statements.
EXISTS	It is used to search for the presence of a row in a specified table.
LIKE	It compares a value to similar values using wildcard operator.