Structured Query Language (SQL)

SQL is a standard Database language which is used to create, maintain and retrieve the relational database.

What Can SQL do?

- SQL can execute queries against a database
- SQL can retrieve data from a database
- SQL can insert records in a database
- SQL can update records in a database
- SQL can delete records from a database
- SQL can create new databases
- SQL can create new tables in a database, etc.

Characteristics of SQL:

- SQL is an ANSI and ISO standard computer language for creating and manipulating databases.
- SQL allows the user to create, update, delete, and retrieve data from a database.
- SQL is very simple and easy to learn.
- SQL works with database programs like DB2, Oracle, MS Access, Sybase, MS SQL Sever etc.

Advantages of SQL:

High Speed: SQL Queries can be used to retrieve large amounts of records from a database quickly and efficiently.

Well Defined Standards Exist: SQL databases use long-established standard, which is being adopted by ANSI & ISO. There are no standards adhered by the non-SQL databases.

No Coding Required: It is easier to manage database systems without having to write substantial amount of code.

Portability: SQL can be used in the program in PCs, servers, laptops, and even some of the mobile phones.

Multiple data views: With the help of SQL language, the users can make different views of database structure and databases for the different users.

Disadvantages of SQL:

Difficult Interface: It has a complex interface that makes it difficult for some users to access it.

Partial Control: The programmers who use SQL doesn’t have a full control over the database because of the hidden business rules.

Implementation: Some of the databases go to the proprietary extensions to standard SQL for ensuring the vendor lock-in.

Cost: The operating cost of some SQL versions makes it difficult for some programmers to access it.

SQL Constraints: Constraints are the rules that we can apply on the type of data in a table, to maintain the accuracy and integrity of the data inside table.

Constraints can be divided into following two types:

Column level constraints: limits only column data

Table level constraints: limits whole table data

The following constraints are commonly used in SQL:

NOT NULL - Ensures that a column cannot have a NULL value

UNIQUE - Ensures that all values in a column are different

PRIMARY KEY - A combination of a NOT NULL and UNIQUE. Uniquely identifies each row in a table

FOREIGN KEY - Uniquely identifies a row/record in another table

CHECK - Ensures that all values in a column satisfies a specific condition

DEFAULT - Sets a default value for a column when no value is specified

INDEX - Used to create and retrieve data from the database very quickly

SQL Data Types

A data type defines what kind of value a column can hold.

These are the different data types into the following categories:

- **Numeric:** This type of data stores numerical values. Data types that fall in this category include Integer, Float, Real, Numeric, or Decimal.
- **Character String:** This type of data stores character values. The two common types are CHAR(*n*) and VARCHAR(*n*).
- **Date/Datetime:** This type of data allows us to store data or datetime in a database table.
- **Binary:** This type of data allows us to store binary objects in a database table.

Basic Types

char(*n*): A fixed-length character string with user-specified length.

varchar(*n*): A variable-length character string with user-specified maximum length *n*.

int: An integer(a finite subset of the integers of the integers that is machine).

smallint: A small integer(a machine-dependent subset of the integer type).

numderic(p,d): A fixed-point number with user-specified precision.

real, double precision: Floating-point and double-position floating-point numbers with machine-dependent precision.

float(*n*): A floating-point number, with precision of at least *n* digits.

Date and Time Types in SQL

date: A calendar date containing a (four-digit) year, month, and day of the month.

time: The time of day, in hours, minutes, and seconds.

timestamp: A combination of date and time.

Literals

In SQL, a literal is the same as a constant. A literal is an explicit numeric, character, string, or Boolean value not represented by an identifier.

Some basic literals are given below:

Integer Literals: Integer literals can be either positive numbers or negative numbers, but do not contain decimals. Example: 536, +536, -536, etc.

Decimal Literals: Decimal literals can be either positive numbers or negative numbers and contain decimals. Example: 24.7, +24.7, -24.7, etc.

Character Literals: Character literals are always surrounded by single quotes (') and contain a single character. Example: ‘A’, ‘%’, ‘9’, ‘)’, etc.

String Literals: String literals are also surrounded by single quotes but it contain group of characters. Example: 'Hello world!', 'XYZ', '123', etc.

BOOLEAN Literals: Boolean literals hold values ‘TRUE’, ‘FALSE’, or ‘NULL’.

Date and Time Literals: Datetime literals are character representations of datetime values that are enclosed in single quotes. Example: '2015/04/30', '2015/04/30 08:34:25'

SQL Commands

SQL commands are instructions, coded into SQL statements, which are used to communicate with the database to perform specific tasks, work, functions and queries with data.

SQL commands are grouped into four major categories depending on their functionality:

- **Data Definition Language (DDL) -** These SQL commands are used for creating, modifying, and dropping the structure of database objects. The commands are CREATE, ALTER, DROP, RENAME, and TRUNCATE.
- **Data Manipulation Language (DML) -** These SQL commands are used for storing, retrieving, modifying, and deleting data. These Data Manipulation Language commands are: SELECT, INSERT, UPDATE, and DELETE.
- **Transaction Control Language (TCL) -** These SQL commands are used for managing changes affecting the data. These commands are COMMIT, ROLLBACK, and SAVEPOINT.
- **Data Control Language (DCL) -** These SQL commands are used for providing security to database objects. These commands are GRANT and REVOKE.

Operators

An operator is a symbol specifying an action that is performed on one or more expressions. Operations are performed on operands or arguments.

Commonly used SQL operators are given below:

1. **Arithmetic Operator:** Arithmetic operator used to perform mathematical operations.

Sr. No.	**Operator**	**Symbol**	**Description**
1.	Add	+	Use to perform add operation
2.	Subtract	-	Use to perform subtract operation
3.	Multiply	*	Use to perform multiply
4.	Divide	/	Use to perform divide
5.	Modulus	%	Use to get remainder

2. **Comparison Operator:** These operators are used to compare data in column to the data specified in SQL statement.

Sr. No.	**Operator**	**Symbol**	**Description**
1.	Equal to	=	When value of two variables or fields is equal, then condition becomes true.
2.	Not equal to	!=	When value of two variables or fields is not equal, then condition becomes true.
3.	Greater than	>	When the value of left operand is greater than the value of right operand, then condition becomes true.
4.	Less than	<	When the value of left operand is less than the value of right operand, then condition becomes true.
5.	Greater than or equal to	>=	When the value of left operand is greater than or equal to the value of right operand, then condition becomes true.
6.	Less than or equal to	<=	When the value of left operand is less than or equal to the value of right operand, then condition becomes true.

3. **Logical Operator:** Logical Operators are used to perform logical operations. AND, OR, NOT are the three types of logical operators.

Sr. No.	**Operator**	**Description**
1.	**AND**	The AND operator displays a record if all the conditions separated by AND is TRUE.
2.	**OR**	The OR operator displays a record if any of the conditions separated by OR is TRUE.
3.	**NOT**	The NOT operator displays a record if the condition(s) is NOT TRUE.

Tables

Tables are database objects that contain all the data in a database. In tables, data is logically organized in a row-and-column format similar to a spreadsheet.

Views

In SQL, a view is a virtual table based on the result-set of an SQL statement. A view contains rows and columns, just like a real table.

Index

The indexes are special objects which built on top of tables or views. Indexes are used to retrieve data from the database very fast.

Aggregate Functions

Aggregate functions perform a calculation on a set of values and return a single value.
The ISO standard defines five (5) aggregate functions namely:
1) COUNT: It returns the total number of values in a given column.
2) SUM: It returns the sum of the numeric values in a given column.
3) AVG: It returns the average value of a given column.
4) MIN: It returns the smallest value in a given column.
5) MAX: It returns the largest value in a given column.

Basic Commands

1) **INSERT command:** Insert command is used to insert data into a table.
   Syntax: INSERT into table-name values (data1, data2,..)

2) **UPDATE command:** Update command is used to update a row of a table.
   Syntax: UPDATE table-name set column-name = value where condition;

3) **Delete command:** Delete command is used to delete data from a table. Delete command can also be used with condition to delete a particular row.
   Syntax: DELETE from table-name;

SQL Joins

A JOIN clause is used to combine rows from two or more tables, based on a related column between them.

General Syntax:

```
SELECT column-names
FROM table-name1 JOIN table-name2
ON column-name1 = column-name2
WHERE condition
```

Types of Join

Here are the different types of the JOINs in SQL:

(INNER) JOIN: It returns records that have matching values in both tables.

Syntax:

```
SELECT column-name-list
from table-name1
INNER JOIN
table-name2
WHERE table-name1.column-name = table-name2.column-name;
```

LEFT (OUTER) JOIN: It returns all records from the left table, and the matched records from the right table.

Syntax:

```
SELECT column-name-list
from table-name1
LEFT OUTER JOIN
table-name2
on table-name1.column-name = table-name2.column-name;
```

RIGHT (OUTER) JOIN: It returns all records from the right table, and the matched records from the left table.

Syntax:

```
select column-name-list
from table-name1
RIGHT OUTER JOIN
table-name2
on table-name1.column-name = table-name2.column-name;
```

FULL (OUTER) JOIN: It returns all records when there is a match in either left or right table.

Syntax:

```
select column-name-list
from table-name1
FULL OUTER JOIN
table-name2
on table-name1.column-name = table-name2.column-name;
```

Union

The SQL UNION clause/operator is used to combine the results of two or more SELECT statements without returning any duplicate rows.

Syntax:

```
SELECT column_name(s) FROM table1
UNION
SELECT column_name(s) FROM table2;
```

Union ALL

The UNION ALL operator is used to combine the results of two SELECT statements including duplicate rows.

Syntax:

```
SELECT column_name(s) FROM table1
UNION ALL
SELECT column_name(s) FROM table2;
```

What is the difference between UNION and UNION ALL?

- UNION removes duplicate rows.
- UNION ALL does not remove duplicate rows.

Intersect

Intersect operation is used to combine two SELECT statements, but it only retuns the records which are common from both SELECT statements.

Syntax:

```
SELECT column_name(s) FROM table1
INTERSECT
SELECT column_name(s) FROM table2;
```

Minus

It takes all the results from the first SQL statement, and then subtract out the ones that are present in the second SQL statement to get the final result set.

Syntax:

```
[SQL Statement 1]

MINUS

[SQL Statement 2];
```