DDL and DML Commands

DDL

Stands for "Data Definition Language." A DDL is a language used to define data structures and modify data. For example, DDL commands can be used to add, remove, or modify tables within in a database. DDLs used in database applications are considered a subset of SQL, the Structured Query Language. However, a DDL may also define other types of data, such as XML.

A Data Definition Language has a pre-defined syntax for describing data. For example, to build a new table using SQL syntax, the CREATE command is used, followed by parameters for the table name and column definitions. The DDL can also define the name of each column and the associated data type. Once a table is created, it can be modified using the ALTER command. If the table is no longer needed, the DROP command can be used to delete the table.

Since DDL is a subset of SQL, it does not include all the possible SQL commands. For example, commands such as SELECT and INSERT are considered part of the Data Manipulation Language (DML), while access commands such as CONNECT and EXECUTE are part of the Data Control Language (DCL). The DDL, DML, and DCL languages include most of the commands supported by SQL.

Examples of DDL commands:

- **CREATE** – is used to create the database or its objects (like table, index, function, views, store procedure and triggers).
- **DROP** – is used to delete objects from the database.
- **ALTER**-is used to alter the structure of the database.
- **TRUNCATE**–is used to remove all records from a table, including all spaces allocated for the records are removed.
- **COMMENT** –is used to add comments to the data dictionary.

- **RENAME**–is used to rename an object existing in the database.

DML

A data manipulation language (DML) is a computer programming language used for adding (inserting), deleting, and modifying (updating) data in a database. A DML is often a sublanguage of a broader database language such as SQL, with the DML comprising some of the operators in the language. Read-only selecting of data is sometimes distinguished as being part of a separate data query language (DQL), but it is closely related and sometimes also considered a component of a DML; some operators may perform both selecting (reading) and writing.

A popular data manipulation language is that of Structured Query Language (SQL), which is used to retrieve and manipulate data in a relational database. Other forms of DML are those used by IMS/DLI, CODASYL databases, such as IDMS and others.

Examples of DML:

- **INSERT** – is used to insert data into a table.
- **UPDATE** – is used to update existing data within a table.
- **DELETE** – is used to delete records from a database table.

Difference between DDL and DML

DDL	DML
It stands for Data Definition Language.	It stands for Data Manipulation Language.
It is used to create database schema and can be used to define some constraints as well.	It is used to add, retrieve or update the data.

It basically defines the column (Attributes) of the table.	It add or update the row of the table. These rows are called as tuple.
It doesn't have any further classification.	It is further classified into Procedural and Non-Procedural DML.
Basic command present in DDL are CREATE, DROP, RENAME, ALTER etc.	BASIC command present in DML are UPDATE, INSERT, MERGE etc.

Introduction to SQL: Aggregate Functions

SQL Aggregate Function: An aggregate function allows you to perform a calculation on a set of values to return a single scalar value. We often use aggregate functions with the GROUP BY and HAVING clauses of the SELECT statement.

The following are the most commonly used SQL aggregate functions:

AVG – calculates the average of a set of values.

COUNT – counts rows in a specified table or view.

MIN – gets the minimum value in a set of values.

MAX – gets the maximum value in a set of values.

SUM – calculates the sum of values.

Notice that all aggregate functions above ignore NULL values except for the COUNT function.

The following table shows the SQL Server aggregate functions:

Aggregate function	Description
AVG	The AVG() aggregate function calculates the average of non-NULL values in a set.
CHECKSUM_AGG	The CHECKSUM_AGG() function calculates a checksum value based on a group of rows.
COUNT	The COUNT() aggregate function returns the number of rows in a group, including rows with NULL values.
COUNT_BIG	The COUNT_BIG() aggregate function returns the number of rows (with BIGINT data type) in a group, including rows with NULL values.
MAX	The MAX() aggregate function returns the highest value (maximum) in a set of non-NULL values.
MIN	The MIN() aggregate function returns the lowest value (minimum) in a set of non-NULL values.
STDEV	The STDEV() function returns the statistical standard deviation of all values provided in the expression based on a sample of the data population.
SUM	The SUM() aggregate function returns the summation of all non-NULL values a set.
VAR	The VAR() function returns the statistical variance of values in an expression based on a sample of the specified population.
VARP	The VARP() function returns the statistical variance of values in an expression but does so based on the entire data population.