**Introduction**

SQL or Structured Query Language is the language that we use to work with these relational database management systems.

SQL(Originally called sequel) is a language that communicates with databases. Most relational databases, including MySQL, ORACLE, SQL Server, etc., support (SQL) query language.

**Note: SQL is not case-sensitive.**

 SQL was firstly developed by IBM for querying and altering relational databases, using declarative statements.SQL became a standard of the American National Standards Institute (ANSI) in 1986, and of the International Organization for Standardization (ISO) in 1987. Although it is often called a “query language”,  SQL is much more than that. SQL can define data structure, modify data in a database, specify security constraints, and can perform many more tasks such as

* 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
* SQL can create stored procedures in a database
* SQL can create views in a database
* SQL can set permissions on tables, procedures, and views
* SQL can execute queries against a database

Before we delve into the different features offered by SQL, let us understand what are functions.

**What are Functions?**

Functions are methods used to perform data operations. SQL has many in-built functions used to perform string concatenations, mathematical calculations, etc.

SQL functions are categorized into the following categories:

1. Aggregate Functions
2. Scalar Functions

Let us look into each one of them, one by one.

**1. Aggregate SQL Functions**

An aggregate function in SQL calculates a group of values and returns a single result.  
Following are a few of the most commonly used Aggregate Functions:

|  |  |
| --- | --- |
| Function | Description |
| SUM() | SUM() is used to return the sum of a group of values. |
| COUNT() | Returns the number of rows either based on a condition or without a condition. |
| AVG() | AVG() is used to calculate the average value of a numeric column. |
| MIN() | MIN() function returns the minimum value of a column. |
| MAX() | MAX() function returns a maximum value of a column. |
| FIRST() | FIRST() is used to return the first value of the column. |
| LAST() | LAST() function returns the last value of the column. |

**2. Scalar SQL Functions**

SQL Scalar Functions return a single value based on the given input value.  
Following are a few of the most commonly used Aggregate Functions

|  |  |
| --- | --- |
| LCASE() | LCASE() is used to convert string column values to lowercase |
| UCASE() | This function is used to convert a string column values to Uppercase. |
| LEN() | LEN() returns the length of the text values in the column. |
| MID() | MID() extracts substrings in SQL from column values having String data type. |
| ROUND() | Rounds off a numeric value to the nearest integer. |
| NOW() | NOW() function is used to return the current system date and time. |
| FORMAT() | FORMAT() is used to format how a field must be displayed. |

Now that we have a fair idea about Structured Query Language(SQL), you must be thinking “What are the features of SQL?”.   
So let’s discuss all the features of SQL one by one.

**Features of SQL**

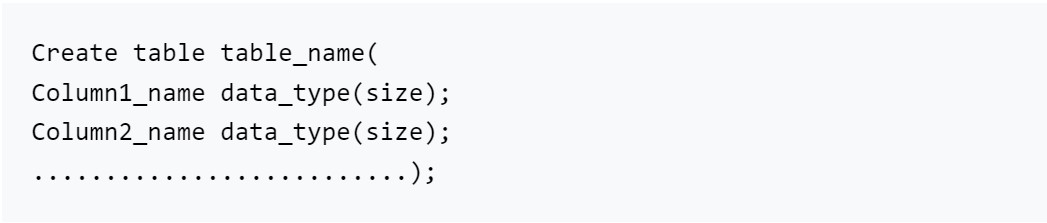
**1. Data Definition Language(DDL):**

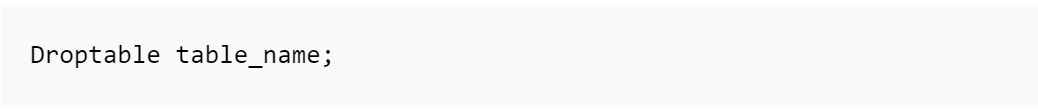
SQL provides many commands, one of which is Data Definition Language(DDL).

**Formal definition**: DDL stands for Data Definition Language. It provides commands for defining relation schemas, deleting relations, and modifying relational schemas. All the structural operations performed on a database are controlled by this language. Sometimes it is also referred to as data description language.

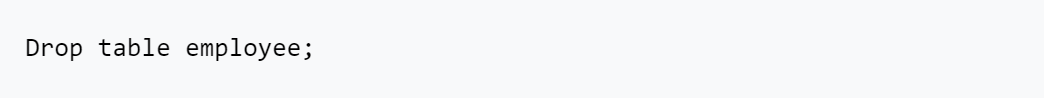
There are 5 commands in DDL:

**i) Create command**:   
If we want to make a new table, the create command is used.  
Syntax: for creating a table

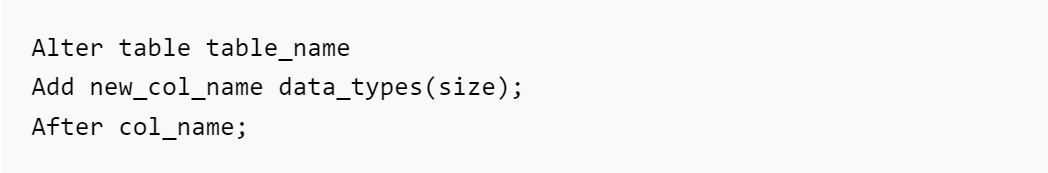


**ii) Drop command**:   
If we want to delete a table including all attributes then the drop command is used.  
Syntax: for drop command

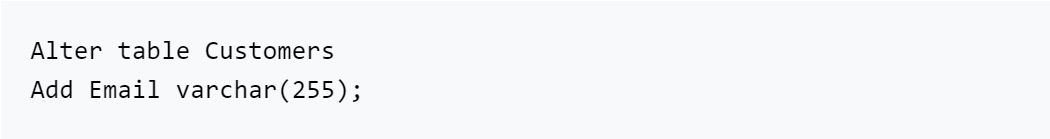
Example:  
To drop the table employee, the query would be like



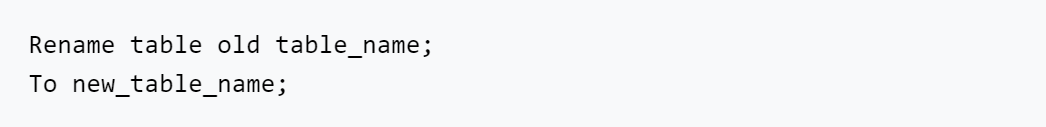
**iii) Alter:**  
The alter command is used to modify the structure of a table.  
Syntax: to add a new column to a table

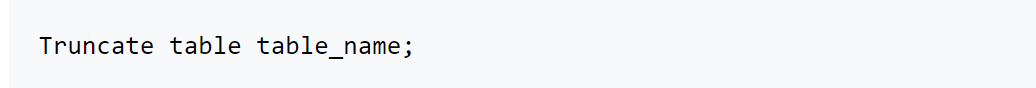


Example:  
The following SQL adds an “Email” column to the “Customers” table:



**iv) Rename**:   
The rename command is used to change the name of a table.  
Syntax: for renaming a table



**v) Truncate:**  
The truncate command is used to delete all the data in a table without affecting the table structure.  
Syntax: for using the truncate command

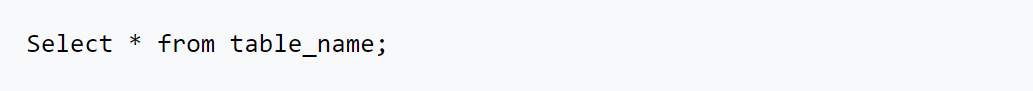
### 2. Data Manipulation Language(DML)

**Formal definition**: DML which stands for Data Manipulation Language is a computer programming language used for managing or manipulating data in a table.

Data manipulation language(DML) provides 4 commands:

**i) Select command:**  
It is a very important command in SQL that is used to select a set of data or information from a table or simply we can say that it is used to query information from a table.  
The syntax depends upon the requirement because the select command is used with different conditions and clauses.

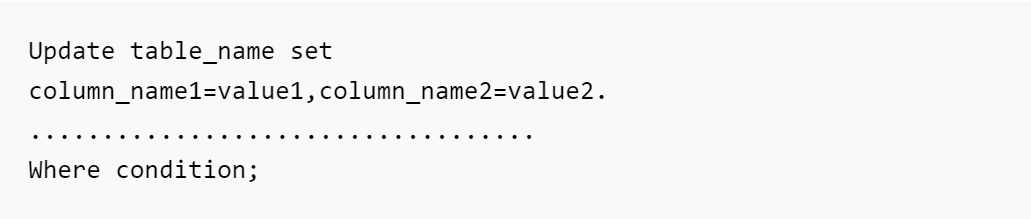
So let’s see a basic syntax for displaying all the information in a table:



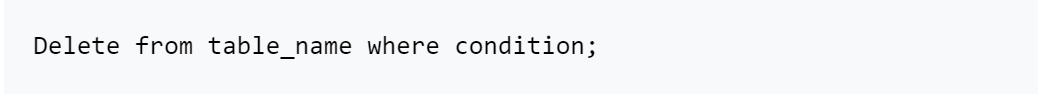
**Note: While using the select command, the table should not be empty.**

**ii) Insert command:**  
The insert command is used to add new records or rows to a table  
Syntax: to use the insert command

**iii) Update command:**  
The update command is used when we want to change or modify data in a table:  
Syntax: to use the update command



**iv) Delete command:**  
The delete command is used when we want to delete a record or data in a table.  
Syntax: to use the delete command



**3. Relational Foundation**

SQL is mainly used for relational databases. A relational database’s tabular structure provides an intuitive user interface, making SQL easy to learn and use. Additionally, relational models have a strong theoretical foundation that has guided the development and implementation of relational databases. In the wake of the success of the relational model, SQL has become the database language for relational databases.

**4. High-performance**

A large amount of data is retrieved quickly and efficiently. In addition, simple operations like inserting, deleting, and manipulating data can also be accomplished in a short amount of time. The reason SQL is so fast is that for a database product to be successful, it must return sets of data quickly when queried. In most database organizations, many of the company’s brightest people work constantly on the query engine so it creates ‘optimal’ query plans that work quickly.

**5. Scalability**

SQL database is vertically scalable, which means that you can increase the load on a single server by adding more RAM, SSDs, or CPUs. Due to the way data is stored (related tables vs unrelated collections), SQL databases generally support vertical scaling only – horizontal scaling is only possible for NoSQL databases.

**6. Security and authentication**

SQL Server includes several security-enabling features, including encrypted communication over SSL/TLS, Windows Data Protection API (DPAPI) to encrypt data at rest, authentication, and authorization.

Authentication means a process of identifying a user or a person based on their username and password**.**SQL Server authenticates its users by their credentials.

SQL Server supports two authentication modes, Windows authentication mode, and mixed-mode.

* **Windows authentication** is the default authentication method and is also known as integrated security because it is tightly integrated with Windows. Specific Windows user and group accounts are trusted to log in to SQL Server. Users who have already been authenticated do not need to provide additional credentials.
* **Mixed-mode** supports authentication both by Windows and by SQL Server. Usernames and passwords are maintained in SQL Server.

**7. Vendor Independence**

With SQL support available from all the major DBMS vendors, no new DBMS product has been highly successful over the last decade. It’s possible to migrate SQL-based databases and programs from one DBMS to another vendor’s DBMS with minimal conversion effort. Thus vendor independence is one of the most important features of SQL and an important reason for its early popularity.

**8. Portability across different computer systems**

SQL-based database products support a variety of platforms such as mainframes, PCs, workstations, specialized servers, and even handhelds.

Due to this feature of SQL:

* Applications that begin on single-user or departmental servers can be migrated to larger server systems as they grow.
* It is also possible to extract and download data from corporate SQL databases into departmental or personal databases.
* A prototype of a SQL-based database application can be developed on an affordable personal computer before it is passed to a more costly multiuser system.

**9. IBM Endorsement and Commitment (DB2)**

Researchers at IBM created SQL, which has since become a strategic product built on IBM’s flagship DB2 database. From personal computers to midrange systems to IBM mainframes, SQL is supported on all major IBM products. With IBM’s initial work, other database and system vendors followed IBM’s lead early on in the development of SQL and relational databases, and as a result of IBM’s wide support and commitment, SQL was accepted more quickly by the market. In addition to IBM’s products, the SQL-based products it has developed run on hardware from emerging competitors such as HP and Sun.

**10. English-Like Structure**

SQL is simple and easy to learn as it contains English-like sentences such as create, select, delete, update, etc. In SQL databases, columns and tables can have long, descriptive names. Therefore, most SQL statements have a clear meaning, allowing them to be interpreted as natural sentences.

**11. Programmatic Database Access**

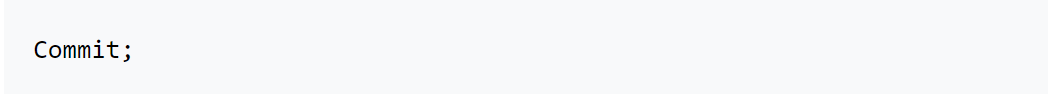
Programmers use SQL to access databases when they write applications. In contrast to traditional databases in which one set of tools are for programmatic access and another for unscheduled requests, Interactive and programmatic access to the database is achieved by using the same SQL statements which allow database access parts of a program to be tested first with interactive SQL and then embedded into a program.

**12. Transaction Control Language:**

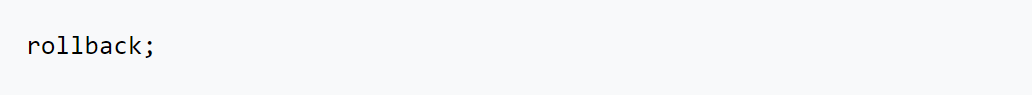
The propagation of a change in the database is called a transaction.

In DBMS, transactions are an important element, and to handle them, TCL(Transaction Control Language) is used, which includes commands such as commit, rollback, and savepoint.

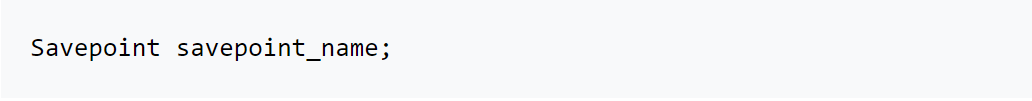
**Commit command:**  
The commit command is used to save the changes.  
The syntax for commit command:



**Rollback command:**  
The rollback command is used to roll back the changes to the previous transaction.  
The syntax for rollback command:



**Savepoint command:**  
The savepoint command is used when we want to go back to the previous transaction without going back to the entire transaction.  
The syntax for savepoint command:



### 13. Various perspectives on data

With SQL, the creator of a database can give different users different views of the database’s structure and content. For eg:-An organization’s database can be constructed in such a way that each user can access only his or her department’s data. Additionally, data from different tables in the database can be combined and displayed as a simple row/column table for the user.

### 14. Dynamic

One of the major advantages of SQL over other static databases is that, even while users are accessing database content, it is possible to change and expand a database’s structure dynamically. As a result, SQL provides maximum flexibility, allowing online applications to continue uninterrupted while a database can adapt to changing requirements.

### 15. Client/Server Architecture

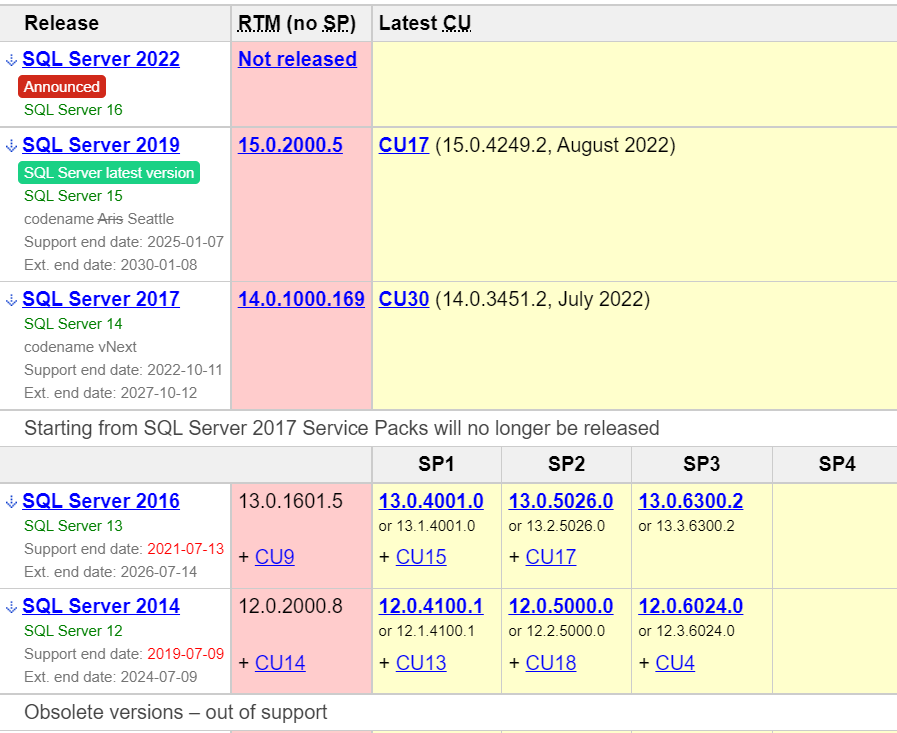
A client-server relationship is defined as a relationship in which a client (many) is connected to a server (one).

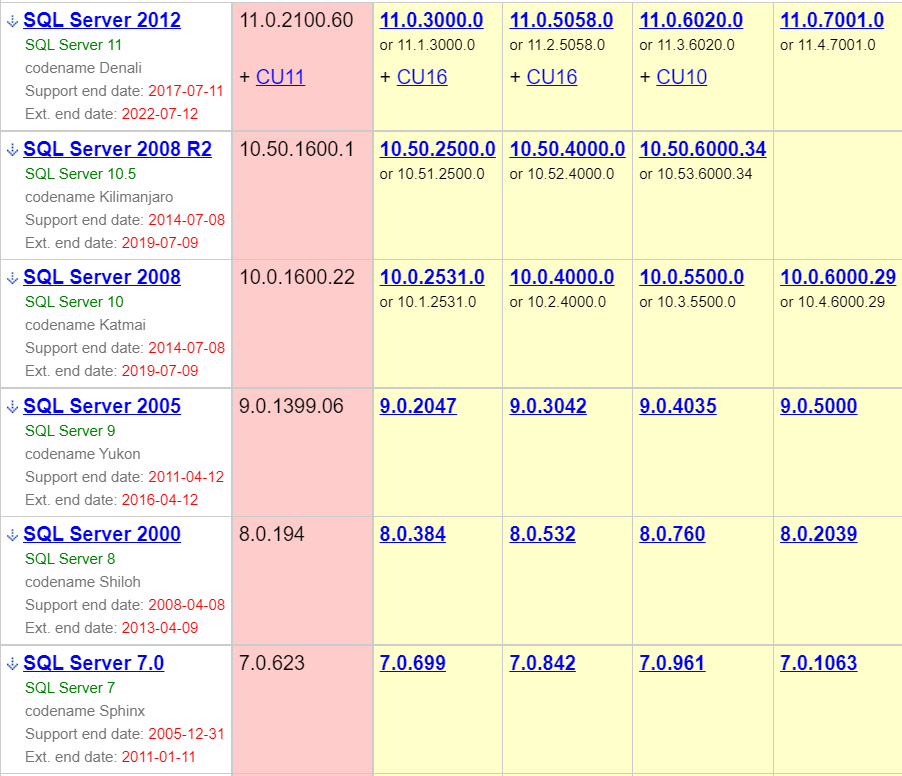
Applications based on distributed, client/server architectures are natural candidates for SQL implementation. A SQL database serves as the glue between “front-end” computer systems geared toward user interaction and “back-end” systems that focus on database management, giving each system the ability to do what it does best. In addition, SQL allows personal computers to serve as a front end to network servers or mainframe databases, allowing access to corporate data from a personal computer application.

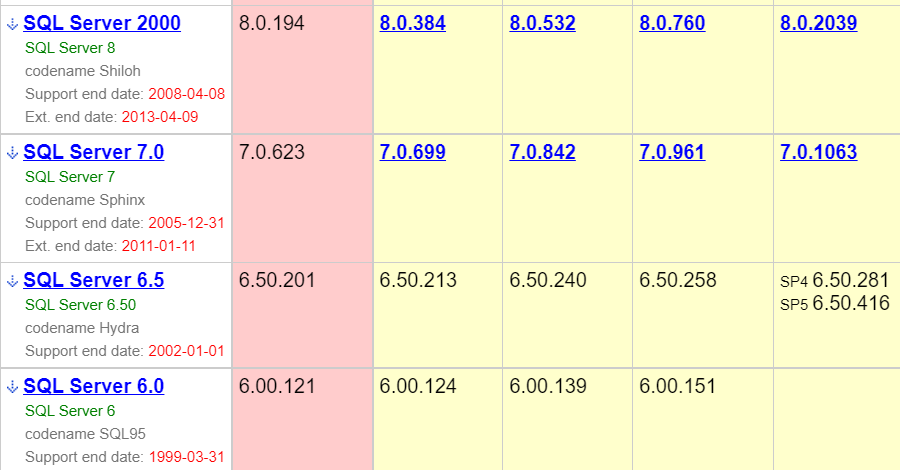
### 16. Integration of SQL with Java

In recent years, the integration of SQL with Java has been a major area of SQL development. Sun Microsystems (the developer of JAVA) introduced Java Database Connectivity JDBC(a standard API that allows Java programs to use SQL for database access), to link the Java language to existing relational databases. It ensured that SQL was still relevant in the new era of Java-based programming.

**SQL Versions**

****

****

****

**MySQL**

## What is Database?

It is very important to understand the database before learning MySQL. A database is an application that stores the organized collection of records. It can be accessed and manage by the user very easily. It allows us to organize data into tables, rows, columns, and indexes to find the relevant information very quickly. Each database contains distinct [API](https://www.javatpoint.com/api-full-form) for performing database operations such as creating, managing, accessing, and searching the data it stores. Today, many databases available like MySQL, Sybase, [Oracle](https://www.javatpoint.com/what-is-oracle), [MongoDB](https://www.javatpoint.com/mongodb-tutorial), [PostgreSQL](https://www.javatpoint.com/postgresql-tutorial), [SQL Server](https://www.javatpoint.com/sql-server-tutorial), etc. In this section, we are going to focus on MySQL mainly.

What is MySQL?

MySQL is currently the most popular database management system software used for managing the relational database. It is open-source database software, which is supported by Oracle Company. It is fast, scalable, and easy to use database management system in comparison with Microsoft SQL Server and Oracle Database. It is commonly used in conjunction with [PHP](https://www.javatpoint.com/php-tutorial) scripts for creating powerful and dynamic server-side or web-based enterprise applications.

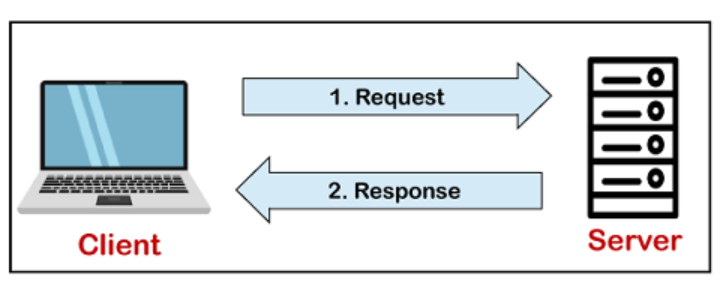
It is developed, marketed, and supported by **MySQL AB, a Swedish company**, and written in [C programming language](https://www.javatpoint.com/c-programming-language-tutorial) and [C++ programming language](https://www.javatpoint.com/cpp-tutorial). The official pronunciation of MySQL is not the My Sequel; it is ***My Ess Que Ell****. However, you can pronounce it in your way.* Many small and big companies use MySQL. MySQL supports many Operating Systems like [Windows](https://www.javatpoint.com/windows), [Linux](https://www.javatpoint.com/linux-tutorial), MacOS, etc. with C, C++, and [Java languages](https://www.javatpoint.com/java-tutorial).

MySQL is a [Relational Database Management System](https://www.javatpoint.com/what-is-rdbms) (RDBMS) software that provides many things, which are as follows:

* It allows us to implement database operations on tables, rows, columns, and indexes.
* It defines the database relationship in the form of tables (collection of rows and columns), also known as relations.
* It provides the Referential Integrity between rows or columns of various tables.
* It allows us to updates the table indexes automatically.
* It uses many SQL queries and combines useful information from multiple tables for the end-users.

How MySQL Works?

MySQL follows the working of Client-Server Architecture. This model is designed for the end-users called clients to access the resources from a central computer known as a server using network services. Here, the clients make requests through a graphical user interface (GUI), and the server will give the desired output as soon as the instructions are matched. The process of MySQL environment is the same as the client-server model.

****

The core of the MySQL database is the MySQL Server. This server is available as a separate program and responsible for handling all the database instructions, statements, or commands. The working of MySQL database with MySQL Server are as follows:

1. MySQL creates a database that allows you to build many tables to store and manipulate data and defining the relationship between each table.
2. Clients make requests through the GUI screen or command prompt by using specific SQL expressions on MySQL.
3. Finally, the server application will respond with the requested expressions and produce the desired result on the client-side.

A client can use any MySQL [GUI](https://www.javatpoint.com/gui-full-form). But, it is making sure that your GUI should be lighter and user-friendly to make your data management activities faster and easier. Some of the most widely used MySQL GUIs are MySQL Workbench, SequelPro, DBVisualizer, and the Navicat DB Admin Tool. Some GUIs are commercial, while some are free with limited functionality, and some are only compatible with MacOS. Thus, you can choose the GUI according to your needs.

**Advantages**

MySQL is becoming so popular because of these following reasons:

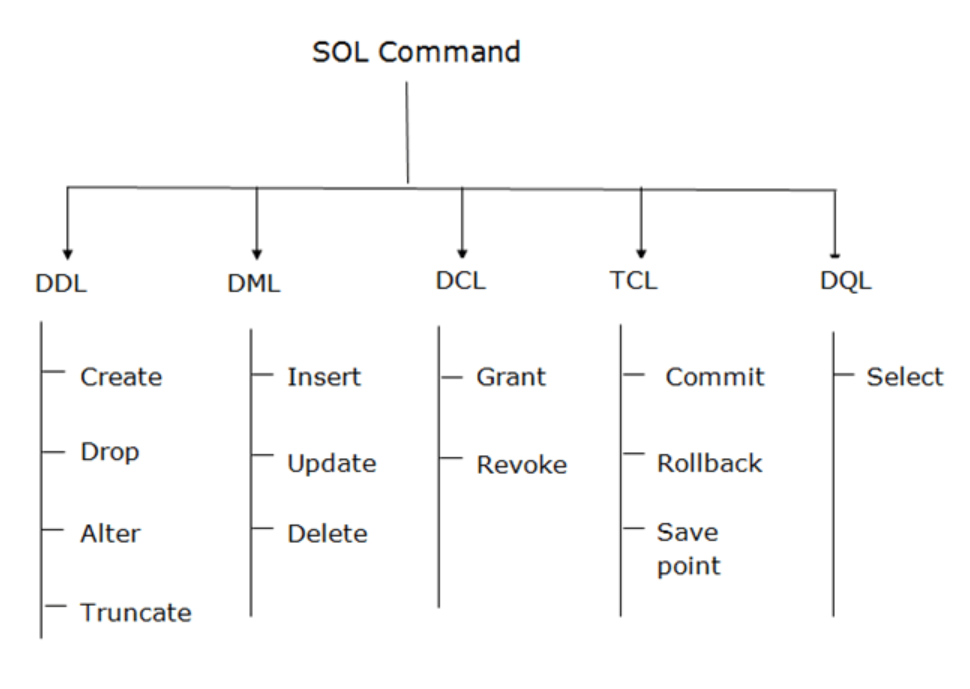
* MySQL is an open-source database, so you don't have to pay a single penny to use it.
* MySQL is a very powerful program that can handle a large set of functionality of the most expensive and powerful database packages.
* MySQL is customizable because it is an open-source database, and the open-source GPL license facilitates programmers to modify the SQL software according to their own specific environment.
* MySQL is quicker than other databases, so it can work well even with the large data set.
* MySQL supports many operating systems with many languages like PHP, PERL, C, C++, JAVA, etc.
* MySQL uses a standard form of the well-known SQL data language.
* MySQL is very friendly with PHP, the most popular language for web development.
* MySQL supports large databases, up to 50 million rows or more in a table. The default file size limit for a table is 4GB, but you can increase this (if your operating system can handle it) to a theoretical limit of 8 million terabytes (TB).

# **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.

****

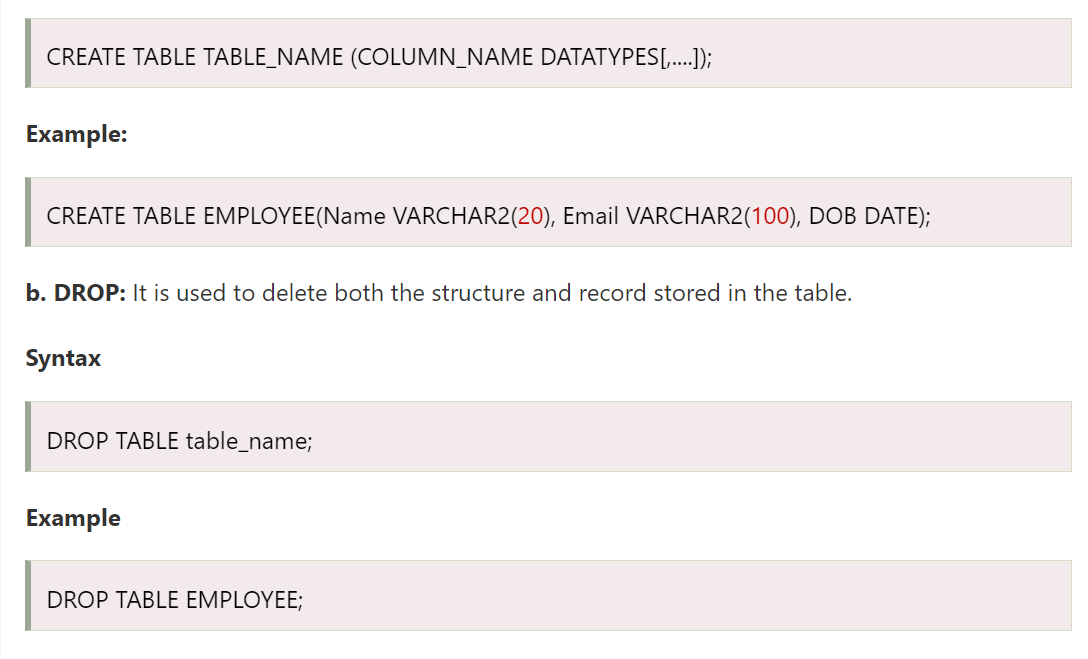
1. Data Definition Language (DDL)

* 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.

****

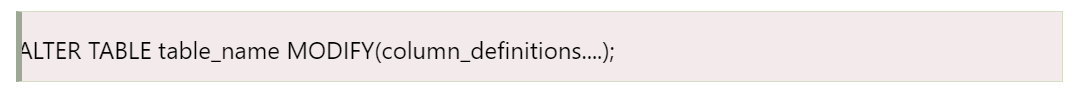
**c. ALTER:** It is used to alter the structure of the database. This change could be either to modify the characteristics of an existing attribute or probably to add a new attribute.

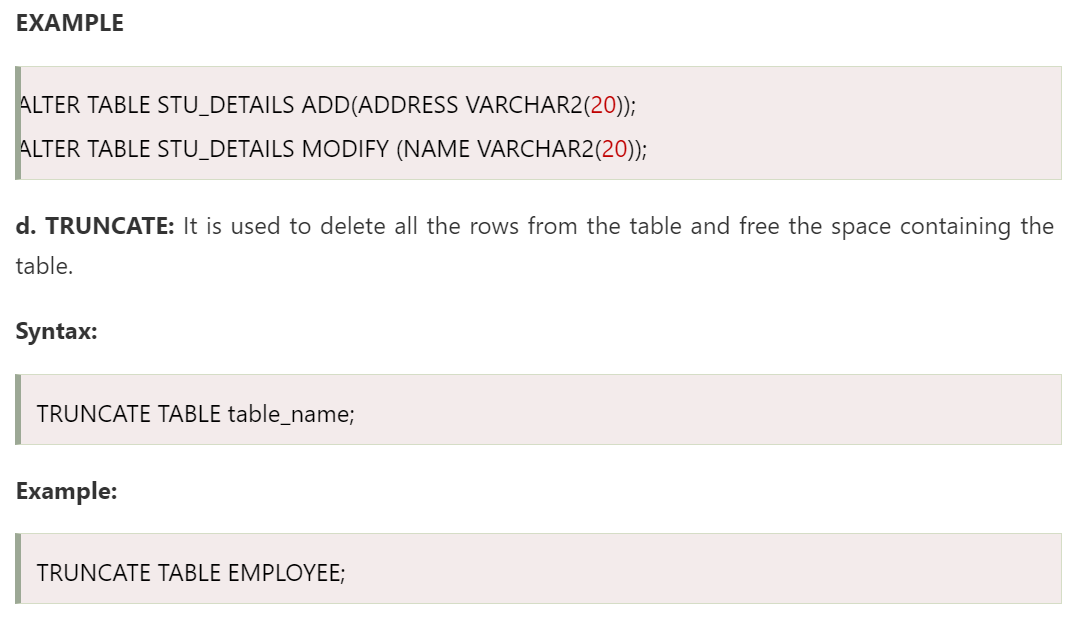
**Syntax:**

To add a new column in the table

****

To modify existing column in the table:





### 2. Data Manipulation Language

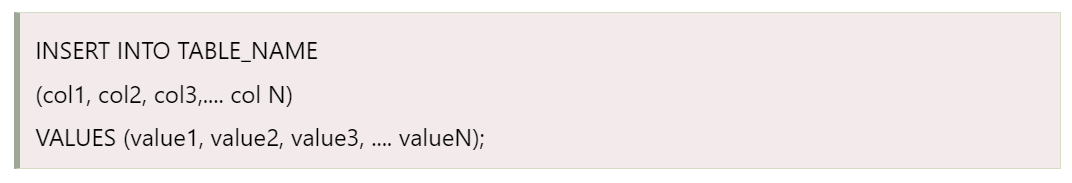
* DML commands are used to modify the database. It is responsible for all form of changes in the database.
* The command of DML is not auto-committed that means it can't permanently save all the changes in the database. They can be rollback.

Here are some commands that come under DML:

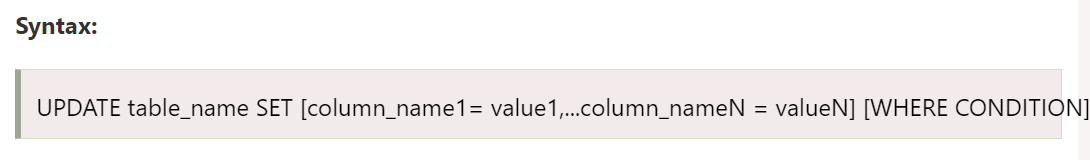
* INSERT
* UPDATE
* DELETE

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

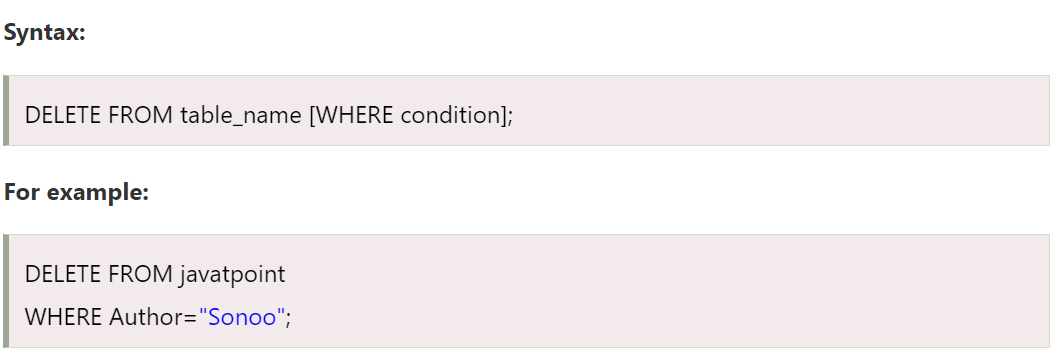
**Syntax:**



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



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



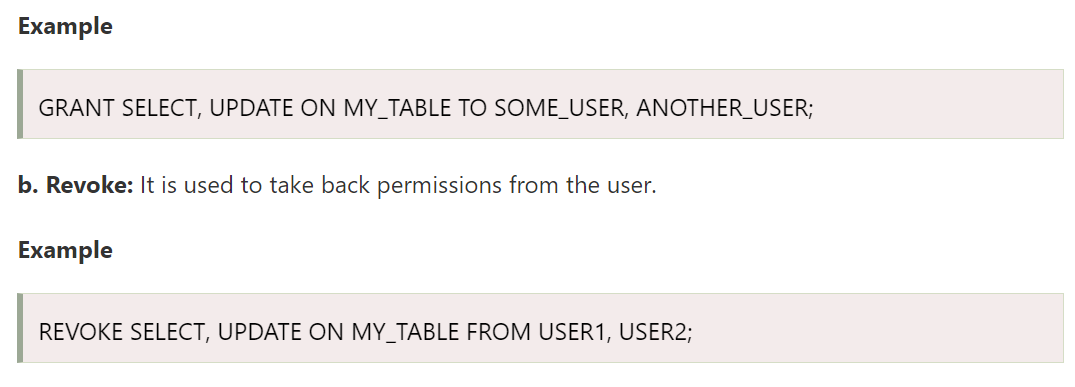
### 3. Data Control Language

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.



### 4. Transaction Control Language

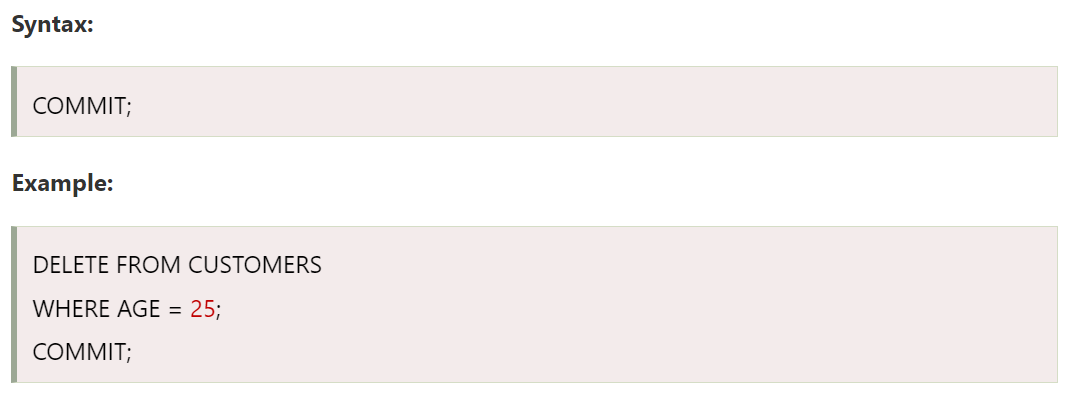
TCL commands can only use with DML commands like INSERT, DELETE and UPDATE only.

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

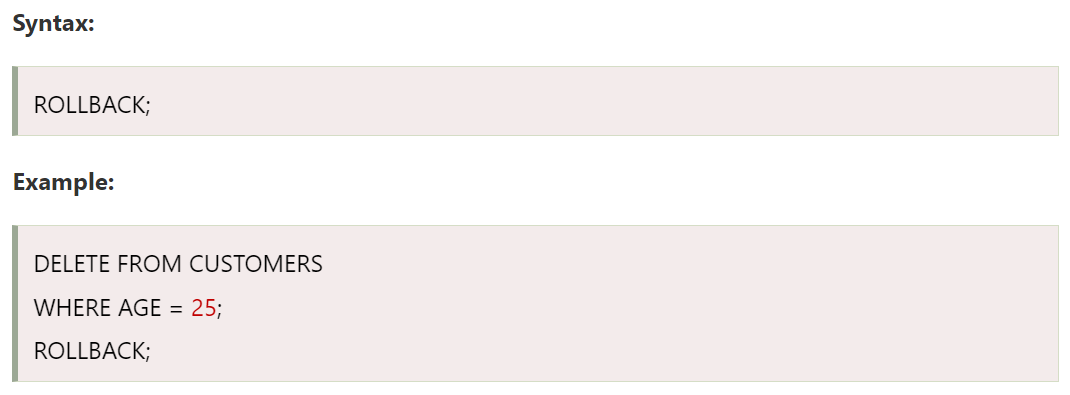
Here are some commands that come under TCL:

* COMMIT
* ROLLBACK
* SAVEPOINT

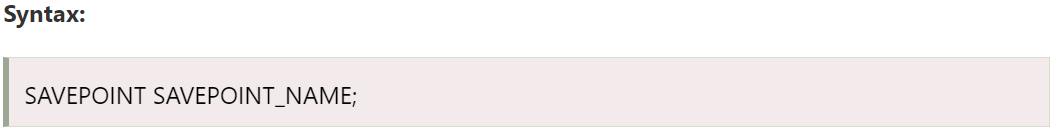
**a. Commit:** Commit command is used to save all the transactions to the database.



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



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



### 5. Data Query Language

DQL is used to fetch the data from the database.

It uses only one command:

* 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.

