Structure Query Language(SQL) is a database query language used for storing and managing data in Relational DBMS. SQL was the first commercial language introduced for E.F Codd's **Relational** model of database. Today almost all RDBMS(MySql, Oracle, Infomix, Sybase, MS Access) use **SQL** as the standard database query language. SQL is used to perform all types of data operations in RDBMS.

SQL Command

SQL defines following ways to manipulate data stored in an RDBMS.

DDL: Data Definition Language

This includes changes to the structure of the table like creation of table, altering table, deleting a table etc.

All DDL commands are auto-committed. That means it saves all the changes permanently in the database.

|  |  |
| --- | --- |
| **Command** | **Description** |
| create | to create new table or database |
| alter | for alteration |
| truncate | delete data from table |
| drop | to drop a table |
| rename | to rename a table |

DML: Data Manipulation Language

DML commands are used for manipulating the data stored in the table and not the table itself.

DML commands are not auto-committed. It means changes are not permanent to database, they can be rolled back.

|  |  |
| --- | --- |
| **Command** | **Description** |
| insert | to insert a new row |
| update | to update existing row |
| delete | to delete a row |
| merge | merging two rows or two tables |

TCL: Transaction Control Language

These commands are to keep a check on other commands and their affect on the database. These commands can annul changes made by other commands by rolling the data back to its original state. It can also make any temporary change permanent.

|  |  |
| --- | --- |
| **Command** | **Description** |
| commit | to permanently save |
| rollback | to undo change |
| savepoint | to save temporarily |

DCL: Data Control Language

Data control language are the commands to grant and take back authority from any database user.

|  |  |
| --- | --- |
| **Command** | **Description** |
| Grant | grant permission of right |
| Revoke | take back permission. |

DQL: Data Query Language

Data query language is used to fetch data from tables based on conditions that we can easily apply.

|  |  |
| --- | --- |
| **Command** | **Description** |
| select | retrieve records from one or more table |

Most commonly used datatypes for Table columns

Here we have listed some of the most commonly used datatypes used for columns in tables.

|  |  |
| --- | --- |
| **Datatype** | **Use** |
| INT | used for columns which will store integer values. |
| FLOAT | used for columns which will store float values. |
| DOUBLE | used for columns which will store float values. |
| VARCHAR | used for columns which will be used to store characters and integers, basically a string. |
| CHAR | used for columns which will store char values(single character). |
| DATE | used for columns which will store date values. |
| TEXT | used for columns which will store text which is generally long in length. For example, if you create a table for storing profile information of a social networking website, then for **about me** section you can have a column of type TEXT. |

**create** is a DDL SQL command used to create a table or a database in relational database management system.

Creating a Database

To create a database in RDBMS, **create** command is used. Following is the syntax,

CREATE DATABASE <DB\_NAME>;

Example for creating Database

CREATE DATABASE Test;

The above command will create a database named **Test**, which will be an empty schema without any table.

To create tables in this newly created database, we can again use the create command.

Creating a Table

create command can also be used to create tables. Now when we create a table, we have to specify the details of the columns of the tables too. We can specify the **names** and **datatypes** of various columns in the create command itself.

Following is the syntax,

CREATE TABLE <TABLE\_NAME>

(

column\_name1 datatype1,

column\_name2 datatype2,

column\_name3 datatype3,

column\_name4 datatype4

);

create table command will tell the database system to create a new table with the given table name and column information.

Example for creating Table

CREATE TABLE Student(

student\_id INT,

name VARCHAR(100),

age INT);

The above command will create a new table with name **Student** in the current database with 3 columns, namely student\_id, name and age. Where the column student\_id will only store integer, name will hold upto 100 characters and age will again store only integer value.

If you are currently not logged into your database in which you want to create the table then you can also add the database name along with table name, using a dot operator .

For example, if we have a database with name **Test** and we want to create a table **Student** in it, then we can do so using the following query:

CREATE TABLE Test.Student(

student\_id INT,

name VARCHAR(100),

age INT);

SQL is a programming language for Relational Databases. It is designed over relational algebra and tuple relational calculus. SQL comes as a package with all major distributions of RDBMS.

SQL comprises both data definition and data manipulation languages. Using the data definition properties of SQL, one can design and modify database schema, whereas data manipulation properties allows SQL to store and retrieve data from database.

## Data Definition Language

SQL uses the following set of commands to define database schema −

### CREATE

Creates new databases, tables and views from RDBMS.

**For example** −

Create database tutorialspoint;

Create table article;

Create view for\_students;

### DROP

Drops commands, views, tables, and databases from RDBMS.

**For example**−

Drop object\_type object\_name;

Drop database tutorialspoint;

Drop table article;

Drop view for\_students;

### ALTER

Modifies database schema.

Alter object\_type object\_name parameters;

**For example**−

Alter table article add subject varchar;

This command adds an attribute in the relation **article** with the name **subject** of string type.

## Data Manipulation Language

SQL is equipped with data manipulation language (DML). DML modifies the database instance by inserting, updating and deleting its data. DML is responsible for all forms data modification in a database. SQL contains the following set of commands in its DML section −

* SELECT/FROM/WHERE
* INSERT INTO/VALUES
* UPDATE/SET/WHERE
* DELETE FROM/WHERE

These basic constructs allow database programmers and users to enter data and information into the database and retrieve efficiently using a number of filter options.

### SELECT/FROM/WHERE

* **SELECT** − This is one of the fundamental query command of SQL. It is similar to the projection operation of relational algebra. It selects the attributes based on the condition described by WHERE clause.
* **FROM** − This clause takes a relation name as an argument from which attributes are to be selected/projected. In case more than one relation names are given, this clause corresponds to Cartesian product.
* **WHERE** − This clause defines predicate or conditions, which must match in order to qualify the attributes to be projected.

**For example** −

Select author\_name

From book\_author

Where age > 50;

This command will yield the names of authors from the relation **book\_author** whose age is greater than 50.

### INSERT INTO/VALUES

This command is used for inserting values into the rows of a table (relation).

**Syntax**−

INSERT INTO table (column1 [, column2, column3 ... ]) VALUES (value1 [, value2, value3 ... ])

Or

INSERT INTO table VALUES (value1, [value2, ... ])

**For example** −

INSERT INTO tutorialspoint (Author, Subject) VALUES ("anonymous", "computers");

### UPDATE/SET/WHERE

This command is used for updating or modifying the values of columns in a table (relation).

**Syntax** −

UPDATE table\_name SET column\_name = value [, column\_name = value ...] [WHERE condition]

**For example** −

UPDATE tutorialspoint SET Author="webmaster" WHERE Author="anonymous";

### DELETE/FROM/WHERE

This command is used for removing one or more rows from a table (relation).

**Syntax** −

DELETE FROM table\_name [WHERE condition];

**For example** −

DELETE FROM tutorialspoints

WHERE Author="unknown";

## What is a Query

A Query is a set of instruction given to the database management system, which tells RDBMS what information you would like to get from the database.