**SQL**

**SQL:**

Structured Query Language (SQL) is a standardized programming language that is used to manage relational databases and perform various operations on the data in them.

SQL is used for the following:

* Modifying database table and index structures
* Adding, updating and deleting rows of data
* Retrieving subsets of information from within relational database management systems.

**Table:**

A table is the most basic unit of a database and consists of rows and columns of data. A single table holds records, and each record is stored in a row of the table. Tables are the most used type of database objects or structures that hold or reference data in a relational database.

* Views are logical representations of data assembled from one or more database tables.
* Indexes are lookup tables that help speed up database lookup functions.
* Reports consist of data retrieved from one or more tables, usually a subset of that data that is selected based on search criteria.

**DDL:**

**DDL –** Data Definition Language (DDL) commands are also called data definition commands because they are used to define data tables. This command is sometimes grouped with the DML commands.

**CREATE (Create Table):**

* **SQL command:** “CREATE TABLE table\_name (column1 datatype, column2 datatype, …);
* **Usage**: Defines a new table with the specified columns and their data types.

**ALTER (Alter Table):**

* **SQL Command:** “ALTER TABLE table\_name ADD COLUMN column\_name datatype;”
* **Usage:** Modifies the structure of an existing table, such as adding a new column.

**DROP (Drop Table):**

* **SQL command:** “DROP TABLE table\_name;”
* **Usage:** Removes an existing table and all its data from the database.

**TRUNCATE (Truncate Table):**

* **SQL command:** “TRUNCATE TABLE table\_name;”
* **Usage:** Removes all rows from a table but retains the table structure for future use.

**RENAME (Rename table):**

* **SQL: “**ALTER TABLE old\_table\_name RENAME TO new\_table\_name;**”**
* **Usage:** Renames an existing table.

**DML:**

**DML –** Data Manipulation Language (DML) commands are used to manipulate data in existing tables by adding, hanging or removing data. Unlike DDL commands that define how data is stored, DML commands operate in the tables defined with DDL commands.

**INSERT (Insert into Table):**

* **SQL Command:** “INSERT INTO table\_name (column1, column2,…) VALUES (value1, value2, …);”
* **Usage:** Adds new rows to a table with the specified values for each column.

**SELECT (Select from table):**

* **SQL command:** “SELECT column1, column2, FROM table\_name WHERE condition;”
* **Usage:** Retrieves data from a table based on the specified conditions.

**UPDATE (Update table):**

* **SQL command:** “UPDATE table\_name SET column1 = value1, column2 = value2, … WHERE condition;”
* **Usage:** Modifies existing records in a table based on the specified conditions.

**DELETE (Delete from table):**

* **SQL command: “**DELETE FROM table\_name WHERE condition;**”**
* **Usage:** Removes records from a table based on the specified conditions.

**DQL:**

**DQL –** Data Query Language (DQL) is a subset of SQL used for querying data from a database. The primary DQL statement is the SELECT statement, and it is used to retrieve information from tables.

**SELECT (Select from table):**

* **SQL command: “**SELECT column1, column2 FROM table\_name WHERE condition;**”**
* **Usage:** Retrieves data from a table based on the specified conditions.

DQL statements are mainly concerned with querying and retrieving data. They are used to extract information from the database tables based on specified criteria.

**DCL:**

**DCL –** Data Control Language (DCL) commands are used for managing permissions and access to the database objects, including tables. The two main DCL statements are GRANT and REVOKE.

**GRANT (Grant permissions):**

* **SQL command: ‘**GRANT permission ON table\_name TO user\_or\_role;**’**
* **Usage:** Assigns specific permission on a table to a user or role.

**REVOKE (Revoke permissions):**

* **SQL command: ‘**REVOKE permission ON table\_name FROM user\_or\_role;’
* **Usage:** Removes specific permissions on a table from a user or role.

**TCL:**

**TCL –** Transaction Control Language (TCL) commands are used to change the state of some data – for example, to COMMIT transaction changes or to ROLLBACK transaction changes.

**CRUD: Create, Read, Update, Delete.**

**Create:** Inserting new records into a database

**Read:** Retrieving or querying records from a database.

**Update:** Modifying existing records in a database

**Delete:** Removing records from a database.

**ACID – Atomicity, Consistency, Isolation, and Durability.**

**Atomicity:**

Atomicity ensures that a transaction is treated as a single, indivisible unit of work. Either all the changes made by the transaction are applied, or none of them are.

Example: If a bank transfer involves debiting one account and crediting another, atomicity ensures that both operations either succeed together or fail together.

**Consistency:**

Consistency ensures that a transaction brings the database from one valid state to another. It enforces integrity constraints and ensures that the database remains in a consistent state before and after the transaction.

Example: If a database has a constraint that prohibits negative balances, a transaction that would result in a negative balance should be rejected to maintain consistency.

**Isolation:**

Isolation ensures that the execution of one transaction is independent of the execution of other transactions. Even if multiple transactions are executed concurrently, the final result should be as if they were executed sequentially.

Example: Two users making changes to their bank accounts simultaneously should not interfere with each other. Each transaction should be isolated from the others.

**Durability:**

Durability guarantees that once a transaction is committed, its effects are permanent and they survive any subsequent system failures (such as power outages or crashes)

Example: If a user receives a confirmation that a transfer has been completed, that information should persist, and the user should not lose that confirmation, even if the system crashes afterward.

**MySQL:**

MySQL is an open-source relational database management system (RDBMS).

**Installation of MySQL:**

**Step 1:** Update the package list

sudo apt update

**Step 2:** Install wget (if not already installed)

sudo apt install wget

**Step 3:** Download and install the MySQL APT repository package

wget https://dev.mysql.com/get/mysql-apt-config\_0.8.15-1\_all.deb

sudo dpkg -i mysql-apt-config\_0.8.15-1\_all.deb

During the installation, it will prompt you to choose MySQL version and other settings.

Choose MySQL Server & Cluster, and MySQL 8.0.

**Step 4**: Update the package list again

sudo apt update

**Step 5:** Install MySQL Server

sudo apt install mysql-server

**Step 6:** Start and enable the MySQL service

sudo systemctl start mysql

sudo systemctl enable mysql