**1. Object in Database**

In DB terms, an **object** is **anything we create and store in the database** that has a name and can hold data or define structure.

1. In **RDBMS labs**, common *objects* are:

* **Tables** → Store data in rows & columns.
* **Views** → Virtual tables based on queries.
* **Indexes** → Help speed up searching.
* **Sequences** → Generate numbers automatically.
* **Synonyms** → Alternative names for objects.

**Example (object = Table):**

CREATE TABLE Student (Student ID INT,Name VARCHAR(50), Age INT);

Here, **Student** is the **object**.

2. **Attribute in Database**

An **attribute** is a **characteristic of an object**.  
In tables, attributes are simply the **columns/fields**.  
Each attribute has:

* **Name** (e.g., StudentID)
* **Data type** (e.g., INT, VARCHAR)
* **Constraints** (optional, e.g., NOT NULL, PRIMARY KEY)

**Example (attributes of Student table):**

* StudentID (integer)
* Name (string)
* Age (integer)

|  |  |  |
| --- | --- | --- |
| **StudentID** | **Name** | **Age** |
| 1 | Meena | 20 |
| 2 | Rahul | 21 |
|  |  |  |

**DDL( DATA DEFINITION LANGUAGE)**

* Defines and modifies database **structure** (tables, schemas, indexes).
* Changes the schema or layout of the database.
* CREATE, ALTER, DROP, TRUNCATE, RENAME
* CREATE TABLE Students (ID INT, Name VARCHAR(50));
* When creating, modifying, or deleting database objects

**DATA MANIPULATION LANGUAGE**

* Works with the **data** stored in tables.
* Changes the actual rows/records in the database.
* INSERT, UPDATE, DELETE, SELECT
* INSERT INTO Students VALUES (1, 'Meena');
* When adding, editing, or viewing data in those objects.

|  |  |
| --- | --- |
| **DDL (Data Definition Language)** | **DML (Data Manipulation Language)** |
| Defines and modifies database **structure** (tables, schemas, indexes). | Works with the **data** stored in tables. |
| Changes the schema or layout of the database. | Changes the actual rows/records in the database. |
| CREATE, ALTER, DROP, TRUNCATE, RENAME | INSERT, UPDATE, DELETE, SELECT |
| Auto-commits (changes are permanent immediately). | Can be rolled back (undo) before commit. |
| CREATE TABLE Students (ID INT, Name VARCHAR(50)); | INSERT INTO Students VALUES (1, 'Meena'); |
| When creating, modifying, or deleting database objects. | When adding, editing, or viewing data in those objects. |

EXERCISE 2 : DDL AND DML COMMANDS

Aim:

To understand and execute operations for creating, modifying, inserting, retrieving, and

deleting data in relational databases.

Description:

1.CREATE Command

Used to create a new table in the database with specified columns and their data types.

Syntax:

CREATE TABLE table\_name (column1\_name data type, column2\_name data type, ...);

2. DROP Command

(i) Delete (Drop) a table

Used to **permanently delete** an existing table from the database, along with all of its data and structure. Once dropped, the table cannot be recovered unless recreated.

Syntax:

DROP TABLE table\_name;

2. ALTER Command

Used to modify the structure of an existing table without deleting its data.

(i) Add a column:

ALTER TABLE table\_name ADD (column\_name data type);

(ii) Delete (drop) a column:

Used to permanently remove a specific column from an existing table along with all its data.

**Syntax:**

ALTER TABLE table\_name DROP COLUMN column\_name;

e.g: ALTER TABLE employee DROP COLUMN email;

3. TRUNCATE command

Used to remove **all rows** from a table instantly, while keeping the table structure(column) intact for future use

**Syntax:**

TRUNCATE TABLE table\_name;

e.g., TRUNCATE TABLE employee;

4. RENAME command

Used to change the name of an existing database object. The table structure and data remain the same; only the name changes.

**Syntax:**

RENAME old\_name to new\_name;

e.g., rename employee to emp1;

**DML(Data manipulation language)**

1. INSERT Command

 Used to add new records (rows) into an existing table.

 Types:

1. Single Row Insert:

o Adds one row at a time.

2. Multiple Row Insert:

o Adds multiple rows in a single query.

**Syntax:**

INSERT INTO table\_name VALUES (value1, value2, ...);

2. SELECT (RETRIEVE) Command

 Used to fetch data from one or more tables.

**Syntax:**

SELECT \* FROM table\_name;

SELECT COLUMN\_1, COLUMN\_2 FROM table\_name;

SELECT DISTINCT (Column\_name) FROM table\_name;

SELECT COLUMN\_1, COLUMN\_2 FROM table\_name WHERE condition;

3.UPDATE Command

Used to change existing data in one or more columns

Can be with or without a condition

**Syntax:**

UPDATE table\_name SET column\_name=value;

UPDATE table\_name SET column\_name=value WHERE condition;

6. DELETE Command

 Used to delete records (rows) from a table.

Syntax:

DELETE FROM table\_name; - this deletes all records if no condition is specified.

To delete specific rows, use: - DELETE FROM table\_name WHERE condition;

RESULT:

Thus the program was successfully executed and the DDL and DML commands performed the intended

operations.