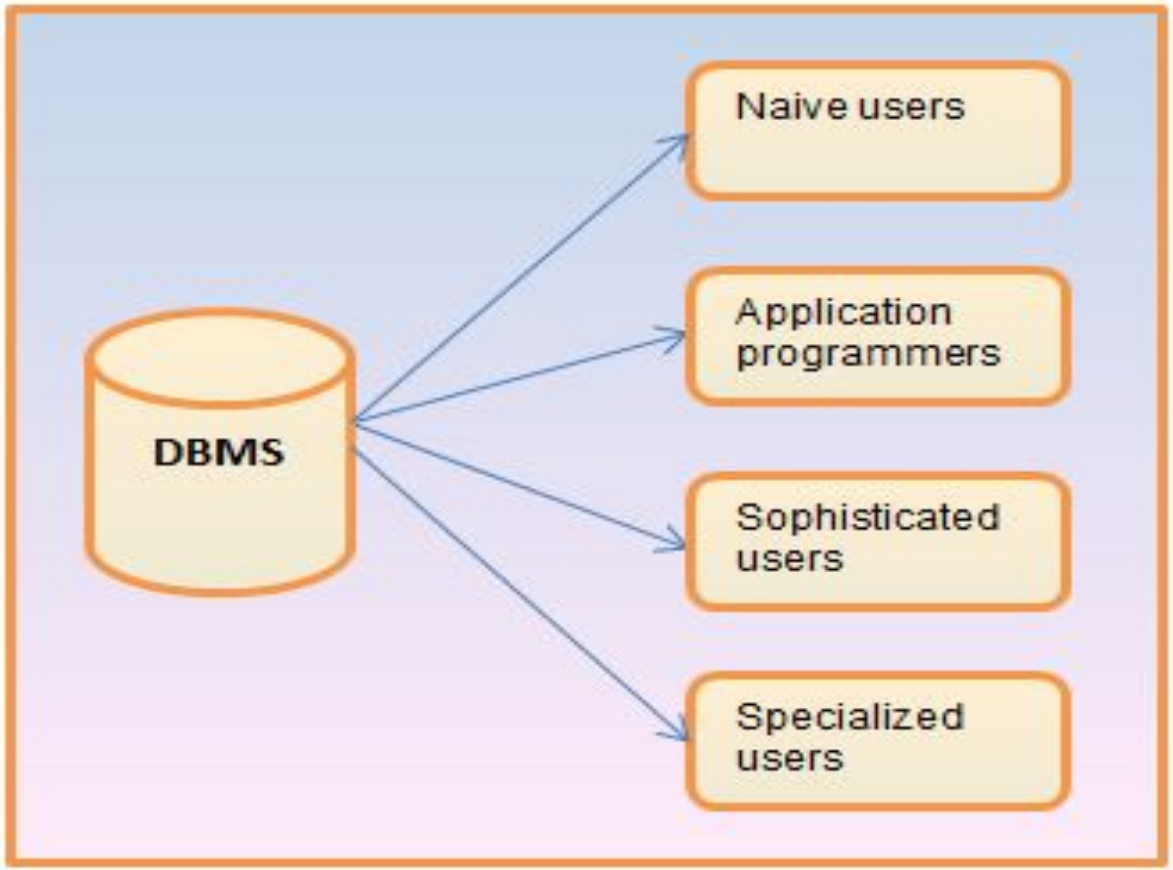
DATABASE MANAGEMENT SYSTEMS

UNIT – 1

S-12, S-13

DATABASE USERS



NAÏVE USERS

- Naïve users are unsophisticated users who interact with the system by invoking one of the application programs that have been written previously.

APPLICATION PROGRAMMERS

- Application programmers are computer professionals who write application programs.
- Rapid application development (RAD) tools are tools that enable an application programmer to construct forms and reports without writing a program(develop user interface).

SOPHISTICATED USERS

- Sophisticated users interact with the system without writing programs. Instead, they form their requests in a database query language.
- They submit each such query to a query processor, whose function is to break down DML statements into instructions that the storage manager understands.
- Analysts who submit queries to explore data in the database fall in this category.

SPECIALIZED USERS

- Specialized users are sophisticated users who write specialized database applications that do not fit into the traditional data-processing framework.

DATABASE ADMINISTRATOR (DBA)

- One of the main reasons for using DBMSs is to have central control of both the data and the programs that access those data.
- A person who has such central control over the system is called a database administrator (DBA).
- **Schema definition:** The DBA creates the original database schema by executing a set of data definition statements in the DDL.
- **Storage structure and access-method definition.**
- **Schema and physical-organization modification:** The DBA carries out changes to the schema and physical organization to reflect the changing needs of the organization, or to alter the physical organization to improve performance.
- **Granting of authorization for data access:** By granting different types of authorization, the database administrator can regulate which parts of the database various users can access.
- **Routine maintenance:** Periodically backing up the database, Ensuring that enough free disk space, Monitoring jobs running on the database.

DATABASE LANGUAGES

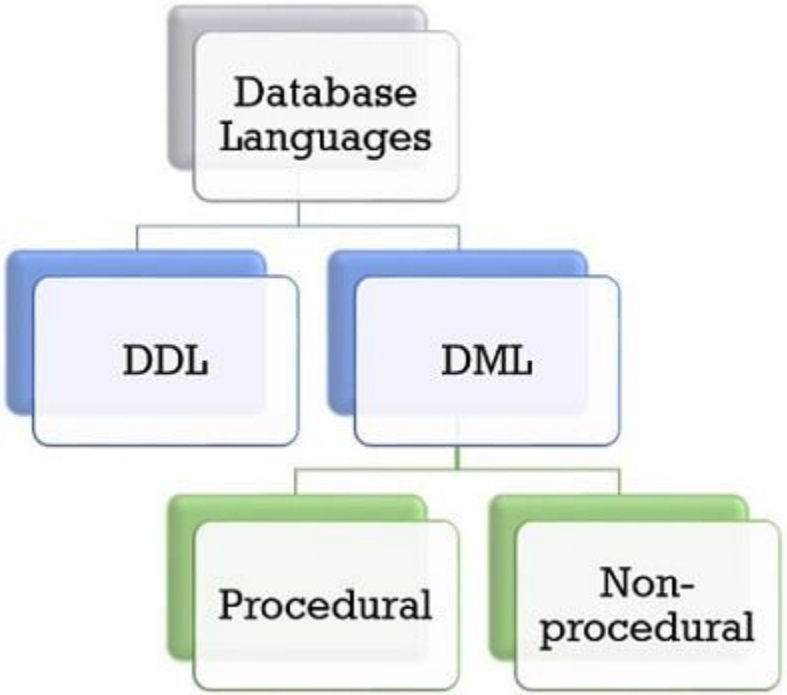
A database system provides a Data-Definition Language (DDL) to specify the database schema and a Data-Manipulation Language (DML) to express database queries and updates.

Data-Manipulation Language (DML)

A data-manipulation language (DML) is a language that enables users to access or manipulate data as organized by the appropriate data model.

The types of access are

- Retrieval of information stored in the database
- Insertion of new information into the database
- Deletion of information from the database
- Modification of information stored in the database

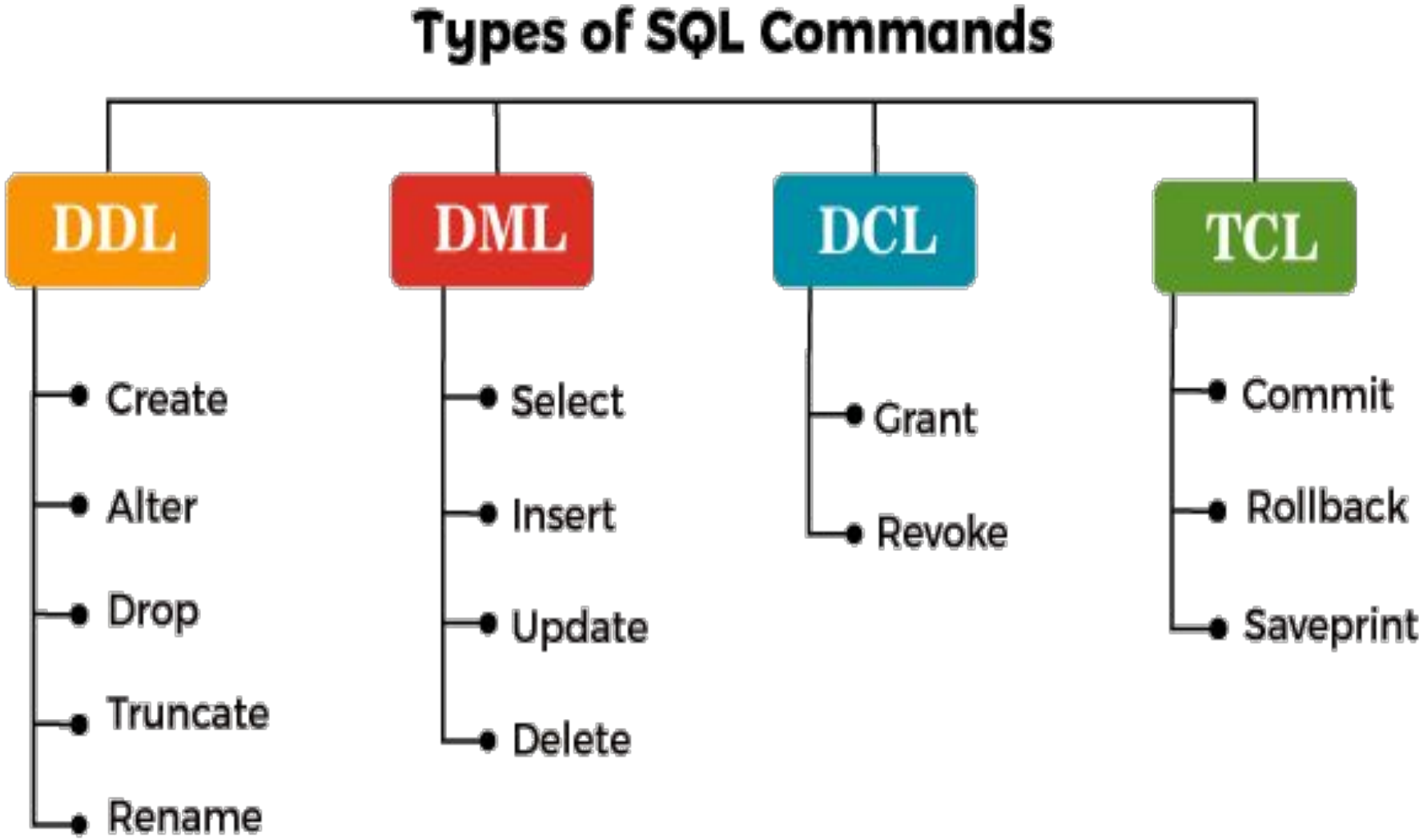


DATA-MANIPULATION LANGUGE (DML)

• **Procedural DMLs** require a user to specify what data are needed and how to get those data.

• **Declarative DMLs** (also called as nonprocedural DMLs) require a user to specify what data are needed without specifying how to get those data.

• A **query** is a statement requesting the retrieval of information. The portion of a DML that involves information retrieval is called a **query language**.


Types of SQL Commands
DDL
Create
Alter
Drop
Truncate
Rename
DML
Select
Insert
Update
Delete
DCL
Grant
Revoke
TCL
Commit
Rollback
Saveprint

DATA DEFINITION LANGUAGE

• Set of definitions expressed by a special language called a data-definition language (DDL).

• The storage structure and access methods used by the database system by a set of statements in a special type of DDL called a data storage and definition language.

• The data values stored in the database must satisfy certain consistency constraints.

• **Domain Constraints:** A domain of possible values must be associated with every attribute (for example, integer types, character types, date/time types).

• Domain constraints are the most elementary form of integrity constraint.

• **Referential Integrity:** There are cases where we wish to ensure that a value that appears in one relation for a given set of attributes also appears in a certain set of attributes in another relation (referential integrity).

• Database modifications can cause violations of referential integrity.

• **Assertions:** An assertion is any condition that the database must always satisfy. Domain constraints and referential-integrity constraints are special forms of assertions.

• **Authorization:** To differentiate among the users as far as the type of access they are permitted on various data values in the database. These differentiations are expressed in terms of authorization

DATA DEFINITION LANGUAGE (CONTD.)

- Read Authorization - which allows reading, but not modification of data.
- Insert Authorization - which allows insertion of new data but not modification of existing data.
- update authorization - which allows modification but not deletion of data.
- Delete Authorization - which allows deletion of data.

- The output of the DDL is placed in the data dictionary which contains metadata - that is, data about data.
- SQL provides a rich DDL that allows one to define tables, integrity constraints, assertions, etc.

```
create table department (dept name char (20), building char (15), budget numeric (12,2));
```

- Execution of the above DDL statement creates the department table with three columns: dept name, building, and budget, each of which has a specific data type associated with it.

DATA DEFINITION LANGUAGES

DDL is Data Definition Language statements

Some examples

- CREATE - to create objects in the database
- ALTER - alters the structure of the database
- DROP - delete objects from the database
- TRUNCATE - remove all records from a table
- COMMENT - add comments to the data dictionary

CREATING DATABASE TABLE

- **CREATE** – creates a new table in the database
- Used to create a table by defining its structure, the data type and name of the various columns, the relationships with columns of other tables etc.
- **CREATE TABLE** table_name (column_name1 data_type(size), column_name2 data_type(size),…., column_nameN data_type(size));
- E.g.:

CREATE TABLE Employee(Name varchar2(20), DOB date, Salary number(6));

ALTER DATABASE TABLE

ALTER - Add a new attribute or Modify the characteristics of some existing attribute.

• **ALTER TABLE** table_name ADD (column_name1 data_type (size), column_name2 data_type (size),….., column_nameN data_type (size));

E.g.:

ALTER TABLE Employee ADD (Address varchar2(20));

ALTER TABLE Employee ADD (Designation varchar2(20), Dept varchar2(3));

ALTER DATABASE TABLE (CONTD.)

ALTER TABLE table_name **MODIFY** (column_name data_type(new_size));

E.g.:

ALTER TABLE Employee MODIFY (Name varchar2(30));

ALTER - dropping a column from the table

- **ALTER TABLE** table_name **DROP COLUMN** column_name;

E.g.:

ALTER TABLE Student DROP COLUMN Age;

DROP AND RENAME DATABASE TABLE

DROP - Deleting an entire table from the database.

DROP TABLE table_name;

E.g.:

DROP TABLE Employee ;

RENAME – Renaming the table RENAME old_table_name TO new_table_name;

E.g.:

RENAME Employee TO Employee_details;

TRUNCATE DATABASE TABLE

TRUNCATE – deleting all rows from a table and free the space containing the table.

TRUNCATE TABLE table_name;

E.g.:

TRUNCATE TABLE Employee_details;

DATA MANIPULATION LANGUAGE

DML is Data Manipulation Language statements

Some examples:

- SELECT - retrieve data from the a database
- INSERT - insert data into a table
- UPDATE - updates existing data within a table
- DELETE - deletes all records from a table, the space for the records remain
- CALL - call a PL/SQL or Java subprogram
- EXPLAIN PLAN - explain access path to data
- LOCK TABLE - control concurrency

DATA MANIPULATION LANGUAGE (CONTD.)

Add new rows to a table by using the INSERT statement.

1. INSERT INTO table VALUES(value1, value2,..);

- Only one row is inserted at a time with this syntax.
- List values in the default order of the columns in the table
- Enclose character and date values within single quotation marks.
- Insert a new row containing values for each column.

E.g.:

INSERT INTO Employee VALUES ('ashok', '16-mar-1998', 30000);

DATA MANIPULATION LANGUAGE (CONTD.)

2. **INSERT INTO** table(column1, column2,..)VALUES(value1, value2,..);

- Rows can be inserted with NULL values either
- by omitting column from the column list or
- by specifying NULL in the value field.

E.g.:

INSERT INTO Employee (name, dob, salary) VALUES ('ashok', '16-mar-1998', 30000);

3**. INSERT INTO** table_name1 SELECT column_name1, column_name2,….,column_nameN FROM table_name2;

INSERT INTO Employee_details SELECT name, dob FROM Exmployee;

DATA CONTROL LANGUAGE

DCL is Data Control Language statements

- Some examples:
- GRANT - gives user's access privileges to database
- REVOKE - withdraw access privileges given with the GRANT command

TRANSACTION CONTROL LANGUAGE

TCL(Transaction Control Language) is a DML

- COMMIT - save work done
- SAVEPOINT - identify a point in a transaction to which you can later roll back
- ROLLBACK - restore database to original since the last COMMIT
- SET TRANSACTION - Change transaction options like what rollback segment to use