Experiment 1.1

Student Name: Aradhya Sharma **UID: 25MCC20042**

Branch: MCA (CCD) **Section/Group: 25MCC-101A**

Semester: 2 **Date of Performance: 10 JAN 2026**

Subject Name: Technical training **Subject Code: 25CAP-652**

Experiment 1.1: Database System Implementation using DDL, DML, and DCL Commands

---

1. AIM

To design and implement a sample database system using DDL, DML, and DCL commands, ensuring data integrity through constraints and providing secure, read-only access using role-based privileges.

---

2. OBJECTIVE

- To create database tables using DDL commands with appropriate constraints
- To insert, update, and delete records using DML commands
- To modify database schema using ALTER statements
- To implement role-based access control using DCL commands

---

3. SOFTWARE REQUIREMENTS

To perform this experiment, the following software is required:

1. **Operating System**
    - Windows 10 / 11, Linux, or macOS
2. **Database Management System (DBMS)**
    - PostgreSQL (version 12 or higher)
3. **SQL Client / Interface**
    - pgAdmin 4 (for PostgreSQL)

      OR
    - Command Line Interface (psql)

4. PRACTICAL/ EXPERIMENT STEPS:

- Design the database schema for Department, Employee and Project tables.
- Create tables using appropriate constraints.
- Insert sample records into tables.
- Perform update and delete operations.
- Retrieve data using SELECT queries.
- Create a role and grant/remove privileges.
- Alter and Drop database objects.

5. PROCEDURE FOR EXPERIMENT

1. Table Creation (DDL Commands)

```
CREATE TABLE department (
   id INT PRIMARY KEY,
   name VARCHAR(50) NOT NULL UNIQUE,
   description VARCHAR(100),
   created_at DATE DEFAULT CURRENT_DATE
);

CREATE TABLE project (
   id INT PRIMARY KEY,
   name VARCHAR(50) NOT NULL,
   due_date DATE NOT NULL,
   budget DECIMAL(10,2),
   status VARCHAR(20) CHECK (status IN ('Planned', 'Ongoing', 'Completed')),
   dept_id INT,
   FOREIGN KEY (dept_id) REFERENCES department(id)
);
```

```
CREATE TABLE employee (
    id INT PRIMARY KEY,
    name VARCHAR(50) NOT NULL,
    age INT CHECK (age >= 18),
    salary DECIMAL(10,2),
    mobile VARCHAR(20) UNIQUE NOT NULL,
    email VARCHAR(50) UNIQUE,
    designation VARCHAR(50),
    hire_date DATE DEFAULT CURRENT_DATE,
    dept_id INT,
    FOREIGN KEY (dept_id) REFERENCES department(id)
);
```

	id [PK] integer	name character varying (50)	description character varying (100)	created_at date	total_employees integer
1	1	HR	Human Resources	2026-01-12	[null]
2	2	IT	Information Technology	2026-01-12	[null]
3	3	Finance	Finance and Accounting	2026-01-12	[null]

	id [PK] integer	name character varying (50)	due_date date	budget numeric (10,2)	status character varying (20)	dept_id integer
1	101	Recruitment Drive	2026-03-31	50000.00	Ongoing	1
2	102	ERP System	2026-06-30	250000.00	Planned	2
3	103	Annual Audit	2026-01-31	75000.00	Completed	3

	id [PK] integer	name character varying (50)	age integer	salary numeric (10,2)	mobile character varying (20)	email character varying (50)	designation character varying (50)
1	3	Charlie Brown	40	70000.00	9876543212	charlie@company.com	Accountant
2	2	Bob	32	65000.00	9876543211	bob@company.com	Software Engineer

2. Data Manipulation (DML Commands)

```sql
INSERT INTO department (id, name, description) VALUES
(1, 'HR', 'Human Resources'),
(2, 'IT', 'Information Technology'),
(3, 'Finance', 'Finance and Accounting');

INSERT INTO project (id, name, due_date, budget, status, dept_id) VALUES
(101, 'Recruitment Drive', '2026-03-31', 50000, 'Ongoing', 1),
(102, 'ERP System', '2026-06-30', 250000, 'Planned', 2),
(103, 'Annual Audit', '2026-01-31', 75000, 'Completed', 3);

INSERT INTO employee (id, name, age, salary, mobile, email, designation, dept_id) VALUES
(1, 'Alice Johnson', 28, 45000, '9876543210', 'alice@company.com', 'HR Executive', 1),
(2, 'Bob Smith', 32, 65000, '9876543211', 'bob@company.com', 'Software Engineer', 2),
(3, 'Charlie Brown', 40, 70000, '9876543212', 'charlie@company.com', 'Accountant', 3);
```

DELETE FROM employee

WHERE id = 1;

UPDATE employee

SET name = 'Mary'

WHERE id = 2;

	id [PK] integer	name character varying (50)	age integer
1	3	Charlie Brown	40
2	2	Mary	32

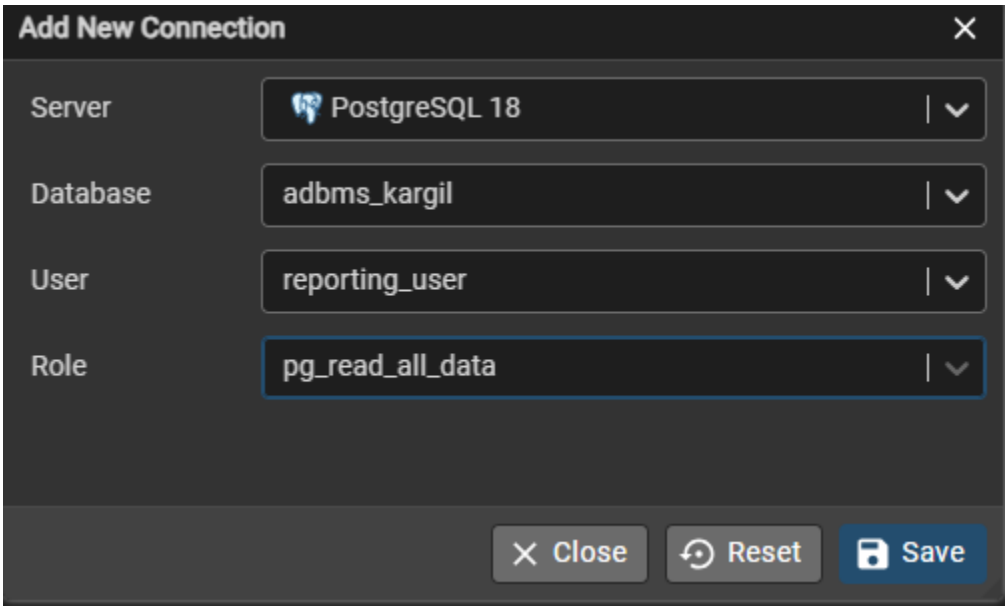
---

3. Schema Modification (DDL Command)

```sql

ALTER TABLE department ADD total_employees INT;

```

4. Access Control (DCL Commands)

```
CREATE ROLE reporting_user

LOGIN

PASSWORD 'report123';

GRANT SELECT ON department TO reporting_user;

GRANT SELECT ON project TO reporting_user;

GRANT SELECT ON employee TO reporting_user;

GRANT USAGE ON SCHEMA public TO reporting_user;

REVOKE INSERT, UPDATE, DELETE ON department FROM reporting_user;

REVOKE INSERT, UPDATE, DELETE ON employee FROM reporting_user;

REVOKE INSERT, UPDATE, DELETE ON project FROM reporting_user;
```



6. LEARNING OUTCOMES

- Understood the use of DDL commands to create and modify database schemas
- Gained experience with DML commands for inserting, updating, and deleting data
- Learned to enforce data integrity using primary key, foreign key, check, and unique constraints
- Implemented role-based security using DCL commands
- Ensured secure, read-only access for authorized users