Worksheet 1.1

Student Name:Shubham Agarwal	**UID:25MCI10091**
Branch:MCA (AI&ML)	**Section/Group:1/A**
Semester:2nd	**Date of Performance:13/01/2026**
Subject Name:- Technical Training	**Subject Code:25CAP-652**

1. Aim of the Session

To design and implement a sample database system using DDL, DML, and DCL commands for managing departments, employees, and projects, and to apply role-based access control for secure data handling.

2. Software Requirements

- PostgreSQL (Database Server)
- pgAdmin
- Windows Operating System

3. Objective of the Session

After completing this practical, the student will be able to:

- Understand the use of DDL commands to create and modify database structures.
- Perform DML operations such as INSERT, UPDATE, DELETE, and SELECT.
- Implement relationships using primary and foreign keys.
- Apply DCL commands to manage roles and privileges.
- Analyze input and output of SQL queries in a real database environment.

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/revoke privileges.

Alter and drop database objects.

5. Procedure of the Practical

(i) Start the system and log in to the computer.

(ii) Open PostgreSQL software.

(iii) create database CompanyDB;

(iv) Create tables using DDL commands.

(i) create table Department

```
CREATE TABLE Department (
   dept_id SERIAL PRIMARY KEY,
   dept_name VARCHAR(50) NOT NULL UNIQUE,
   loc_name VARCHAR(50) NOT NULL
);
```

(ii) create table Employee

```
CREATE TABLE Employee (
   emp_id SERIAL PRIMARY KEY,
   emp_name VARCHAR(50) NOT NULL,
   email VARCHAR(100) UNIQUE NOT NULL,
   salary NUMERIC(10,2) CHECK (salary > 0),
   dept_id INT NOT NULL,
   FOREIGN KEY (dept_id) REFERENCES Department(dept_id)
      ON DELETE RESTRICT
);
```

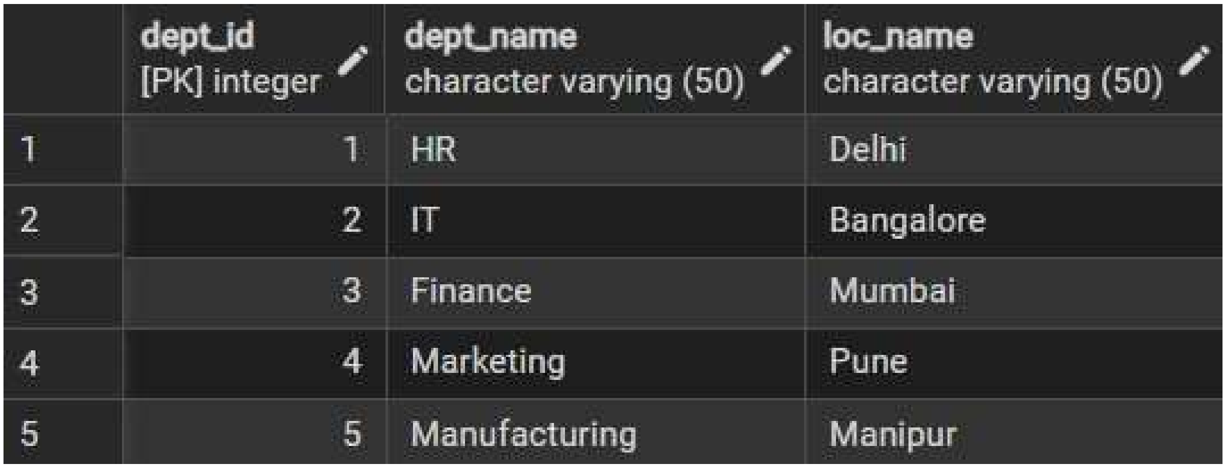
(iii) create table Project

```
CREATE TABLE Project (
    project_id SERIAL PRIMARY KEY,
    project_name VARCHAR(50) NOT NULL UNIQUE,
    budget NUMERIC(12,2) CHECK (budget >= 10000),
    dept_id INT NOT NULL,
    FOREIGN KEY (dept_id) REFERENCES Department(dept_id)
        ON DELETE CASCADE
);
```

(iv) Insert records using DML commands.

insert into Department values

```
INSERT INTO Department (dept_name, loc_name) VALUES
('HR', 'Delhi'),
('IT', 'Bangalore'),
('Finance', 'Mumbai'),
('Marketing', 'Pune'),
('Manufacturing', 'Manipur');
```



	dept_id [PK] integer	dept_name character varying (50)	loc_name character varying (50)
1	1	HR	Delhi
2	2	IT	Bangalore
3	3	Finance	Mumbai
4	4	Marketing	Pune
5	5	Manufacturing	Manipur

(v) insert into Employee values

INSERT INTO Employee (emp_name, email, salary, dept_id) VALUES

('Amit', 'amit@org.com', 50000, 1);,

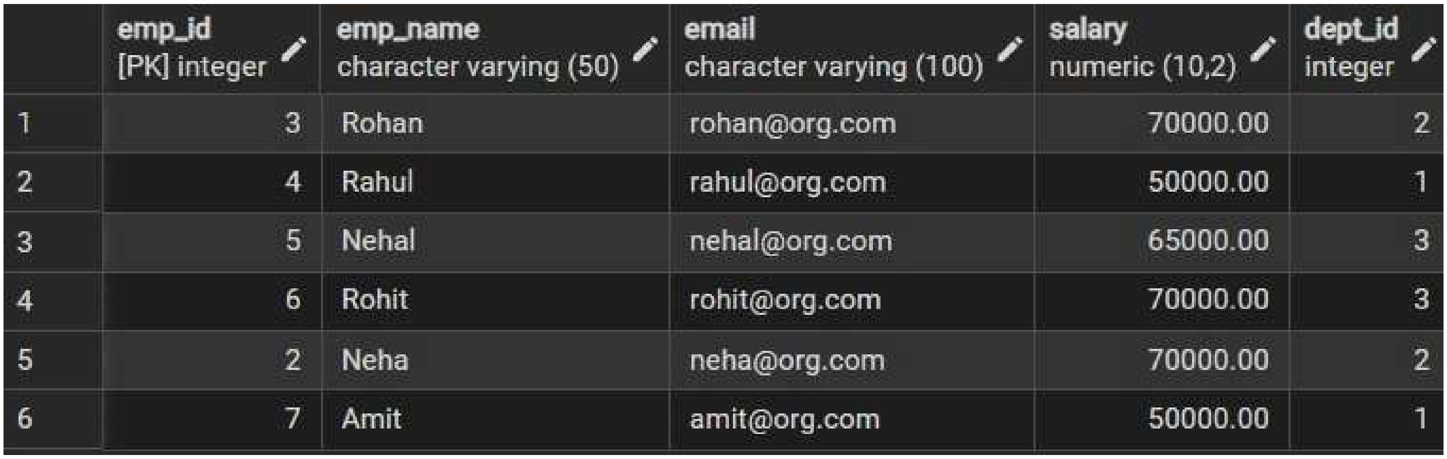
('Neha', 'neha@org.com', 70000, 2),

('Rohan', 'rohan@org.com', 70000, 2),

('Rahul', 'rahul@org.com', 50000, 1),

('Nehal', 'nehal@org.com', 65000, 3),

('Rohit', 'rohit@org.com', 70000, 3);



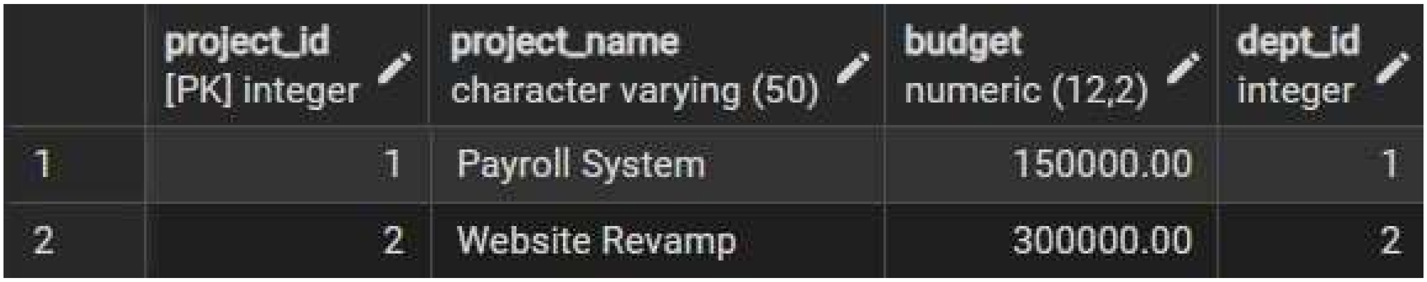
	emp_id [PK] integer	emp_name character varying (50)	email character varying (100)	salary numeric (10,2)	dept_id integer
1	3	Rohan	rohan@org.com	70000.00	2
2	4	Rahul	rahul@org.com	50000.00	1
3	5	Nehal	nehal@org.com	65000.00	3
4	6	Rohit	rohit@org.com	70000.00	3
5	2	Neha	neha@org.com	70000.00	2
6	7	Amit	amit@org.com	50000.00	1

(vi) insert into Project values

INSERT INTO Project (project_name, budget, dept_id) VALUES

('Payroll System', 150000, 1),

('Website Revamp', 300000, 2);



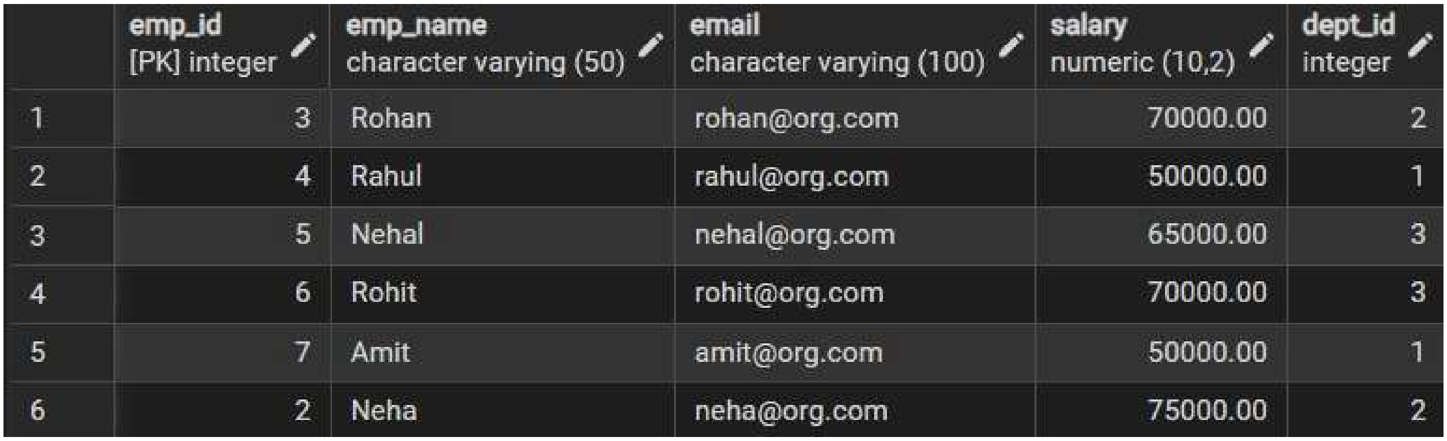
	project_id [PK] integer	project_name character varying (50)	budget numeric (12,2)	dept_id integer
1	1	Payroll System	150000.00	1
2	2	Website Revamp	300000.00	2

(vii) Update and delete records.

UPDATE Employee

SET salary = salary + 5000

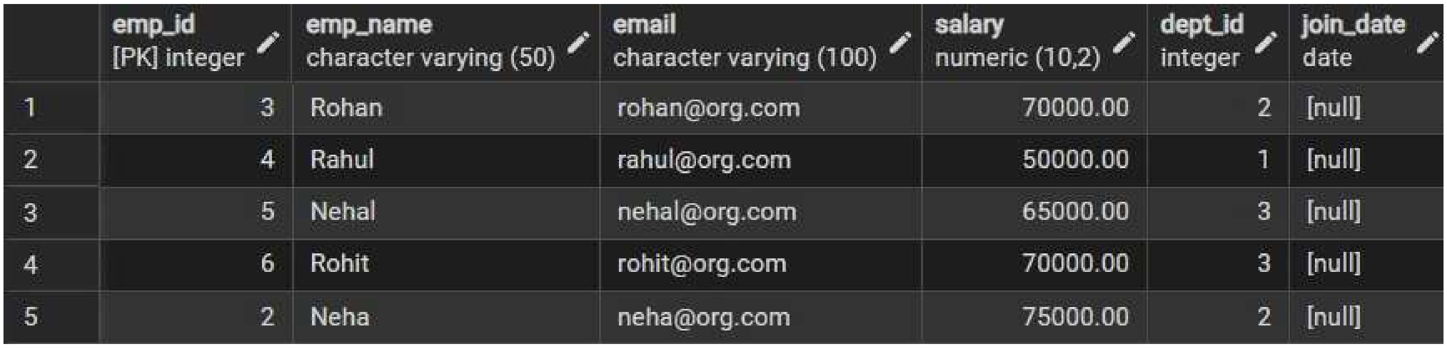
WHERE emp_name = 'Neha';



	emp_id [PK] integer	emp_name character varying (50)	email character varying (100)	salary numeric (10,2)	dept_id integer
1	3	Rohan	rohan@org.com	70000.00	2
2	4	Rahul	rahul@org.com	50000.00	1
3	5	Nehal	nehal@org.com	65000.00	3
4	6	Rohit	rohit@org.com	70000.00	3
5	7	Amit	amit@org.com	50000.00	1
6	2	Neha	neha@org.com	75000.00	2

DELETE FROM Employee

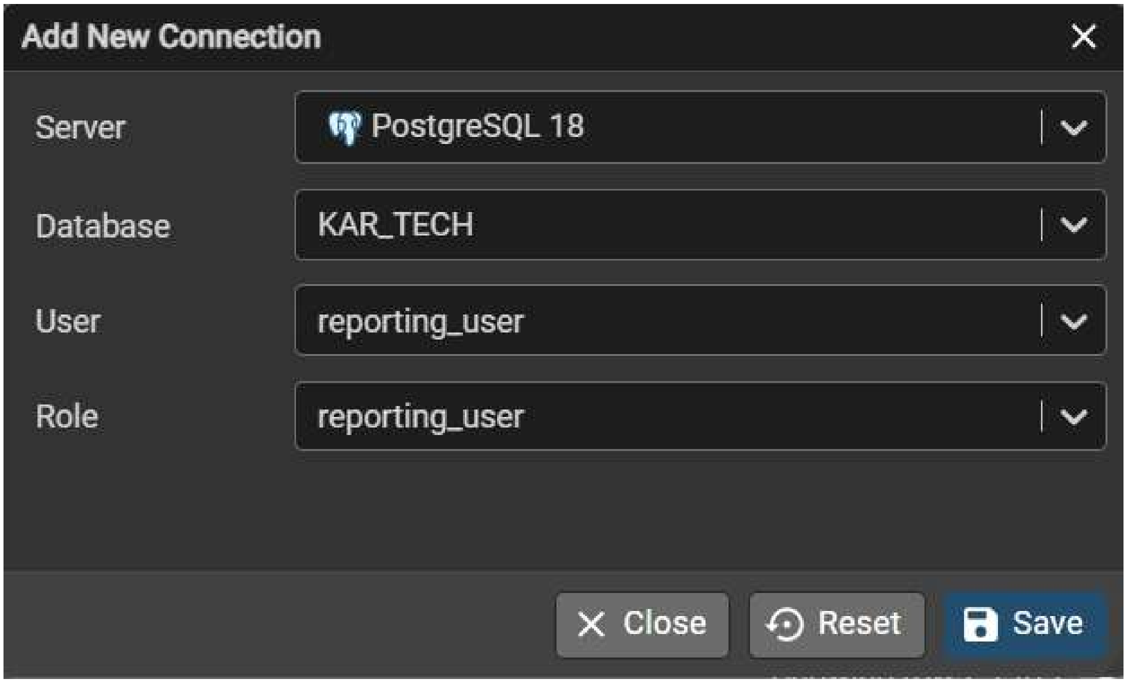
WHERE emp_name = 'Amit';



	emp_id [PK] integer	emp_name character varying (50)	email character varying (100)	salary numeric (10,2)	dept_id integer	join_date date
1	3	Rohan	rohan@org.com	70000.00	2	[null]
2	4	Rahul	rahul@org.com	50000.00	1	[null]
3	5	Nehal	nehal@org.com	65000.00	3	[null]
4	6	Rohit	rohit@org.com	70000.00	3	[null]
5	2	Neha	neha@org.com	75000.00	2	[null]

(viii) Create role and assign privileges.

CREATE USER reporting_user WITH PASSWORD 'report@123';



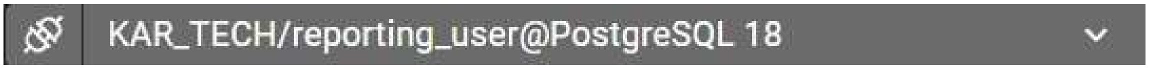
Add New Connection

Server: PostgreSQL 18

Database: KAR_TECH

User: reporting_user

Role: reporting_user



KAR_TECH/reporting_user@PostgreSQL 18

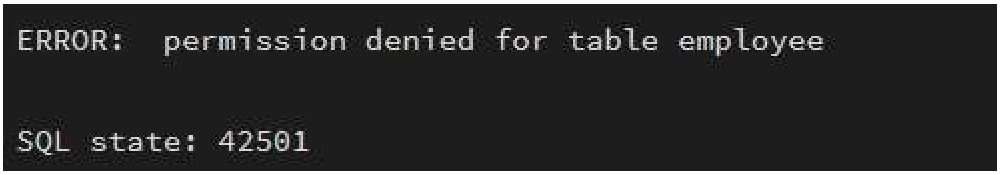
GRANT SELECT ON Department TO reporting_user;

GRANT SELECT ON Employee TO reporting_user;

GRANT SELECT ON Project TO reporting_user;

REVOKE CREATE ON SCHEMA public FROM reporting_user;

REVOKE INSERT, UPDATE, DELETE ON ALL TABLES IN SCHEMA public FROM reporting_user;



```
ERROR:  permission denied for table employee


SQL state: 42501
```

SELECT * FROM Department;

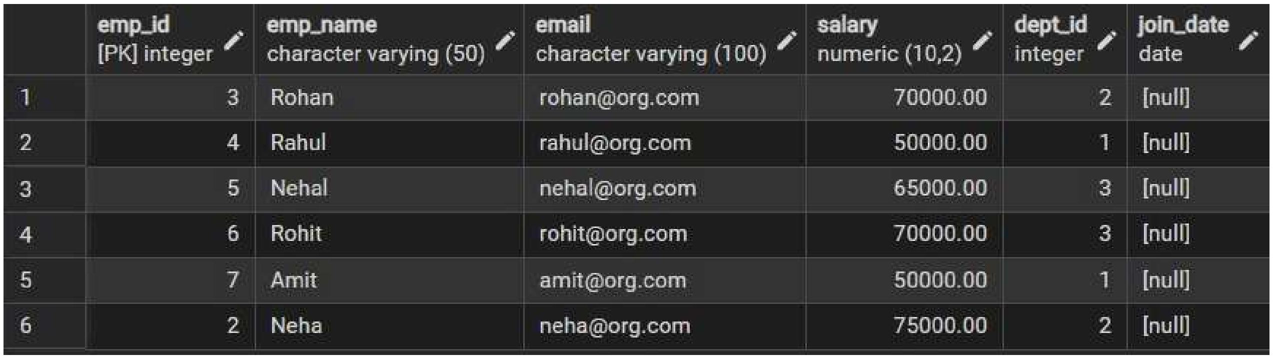
SELECT * FROM Employee;

SELECT * FROM Project;

(ix) Alter and drop table.

ALTER TABLE Employee

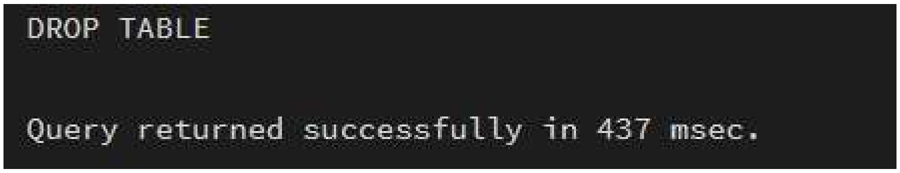
ADD COLUMN join_date DATE;



	emp_id [PK] integer	emp_name character varying (50)	email character varying (100)	salary numeric (10,2)	dept_id integer	join_date date
1	3	Rohan	rohan@org.com	70000.00	2	[null]
2	4	Rahul	rahul@org.com	50000.00	1	[null]
3	5	Nehal	nehal@org.com	65000.00	3	[null]
4	6	Rohit	rohit@org.com	70000.00	3	[null]
5	7	Amit	amit@org.com	50000.00	1	[null]
6	2	Neha	neha@org.com	75000.00	2	[null]

(x) drop table Project;

DROP TABLE Project



```
DROP TABLE

Query returned successfully in 437 msec.
```

6. I/O Analysis (Input / Output)

Input:

- Department, Employee, and Project table creation queries
- Records inserted into all tables using INSERT commands
- Update query to modify employee department
- Delete queries to remove project and employee records

- Role creation and privilege assignment queries
- ALTER and DROP table commands

Output:

- Department, Employee, and Project tables created successfully
- Records inserted, updated, and deleted correctly
- Referential integrity maintained between tables
- Data displayed correctly using SELECT queries
- Role-based access verified using GRANT and REVOKE
- Table structure modified and project table dropped successfully

Screenshots of execution and obtained results are attached.

7.Learning Outcomes

- Understood the basics of relational database design using tables, keys, and relationships.
- Learned to apply primary and foreign key constraints to maintain data integrity.
- Gained hands-on experience with INSERT, UPDATE, and DELETE operations.
- Understood role-based access control using GRANT and REVOKE.
- Learned how to create read-only users for secure data access.
- Practiced ALTER TABLE and DROP TABLE commands for schema management.