Experiment 7 Output

Aim: This experiment covers setting up and managing cloud-based relational databases using **PostgreSQL, Docker,** and **LocalStack**. The setup replicates real-world cloud database operations using containerized solutions.

Creating Postgres Container

```
C:\Users\rawat>docker run --name my-new-postgres -e POSTGRES_USER=
b2efdca3c6f0af6cf4154fce236f0b66b5efba0f4f9e14972c94b3e0a5afa9de

C:\Users\rawat>docker ps
CONTAINER ID   IMAGE                 COMMAND                  CR
      NAMES
b2efdca3c6f0   postgres:15           "docker-entrypoint.s…"   42
      my-new-postgres
a10c5a71f625   localstack/localstack "docker-entrypoint.sh"   3
/tcp   localstack-main
6c3f6fc35cc4   hackvortex-backend    "gunicorn --bind 0.0…"   21
      backend_app
dedb2d1ac081   postgres:15           "docker-entrypoint.s…"   21
      postgres_db
```

Fig 1: Postgres Setup

Accessing via Interactive shell

```
C:\Users\rawat>docker exec -it my-new-postgre
psql (15.12 (Debian 15.12-1.pgdg120+1))
Type "help" for help.

mydb=# CREATE TABLE students (
    id SERIAL PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    email VARCHAR(100) UNIQUE NOT NULL
);
INSERT INTO students (name, email) VALUES
('Alice Johnson', 'alice@example.com'),
('Bob Smith', 'bob@example.com'),
('Charlie Brown', 'charlie@example.com');
CREATE TABLE
INSERT 0 3
mydb=# SELECT * FROM students;
 id |     name      |        email
----+---------------+---------------------
  1 | Alice Johnson | alice@example.com
  2 | Bob Smith     | bob@example.com
  3 | Charlie Brown | charlie@example.com
(3 rows)
```

Fig 2: Creating Tables and Data insertion

Data Access & Queries

```
CREATE TABLE
INSERT 0 3
mydb=# SELECT * FROM students;
 id |     name      |        email
----+---------------+---------------------
  1 | Alice Johnson | alice@example.com
  2 | Bob Smith     | bob@example.com
  3 | Charlie Brown | charlie@example.com
(3 rows)

mydb=# UPDATE students SET email = 'bob.smith@example.com' WHERE name = 'Bob Smith';
UPDATE 1
mydb=# DELETE FROM students WHERE name = 'Charlie Brown';
DELETE 1
mydb=# SELECT * FROM students WHERE name LIKE 'A%';
 id |     name      |       email
----+---------------+-------------------
  1 | Alice Johnson | alice@example.com
(1 row)

mydb=# \q
```

Fig 3: Postgres Queries and Output

Madhurima Rawat DS 42