



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

WORKSHEET 8

Student Name: Ashutosh
Yadav

UID: 23BCS11023

Branch: CSE(3rd Year)

Section/Group: Krg-1-B

Semester: 5th

Date of Performance: 09/10/25

Subject Name: ADBMS

Subject Code: 23CSP-333

1. AIM:

Design a robust PostgreSQL transaction system for the students table where multiple student records are inserted in a single transaction.

If any insert fails due to invalid data, only that insert should be rolled back while preserving the previous successful inserts using savepoints.

The system should provide clear messages for both successful and failed insertions, ensuring data integrity and controlled error handling.

2. Tools Used : Postgres

Solutions:

```
DROP TABLE IF EXISTS students;
```

```
CREATE TABLE students (
    id SERIAL PRIMARY KEY,
    name VARCHAR(50),
    age INT,
    class INT
);
```

```
DO $$  
BEGIN  
    BEGIN  
        INSERT INTO students(name, age, class) VALUES ('Yash',17,8);  
        INSERT INTO students(name, age, class) VALUES ('Diksha',19,9);  
        INSERT INTO students(name, age, class) VALUES ('Ashu',16,8);  
    END;  
END $$
```

```
RAISE NOTICE 'Transaction Successfully Done';

EXCEPTION
WHEN OTHERS THEN
    RAISE NOTICE 'Transaction Failed..! Rolling back changes.';
    RAISE;
END;
END;
$$;
```

```
SELECT * FROM students;
```

-----WRONG DATA TYPE SCENARIO-----

```
BEGIN; -- start transaction

SAVEPOINT sp1;
INSERT INTO students(name, age, class) VALUES ('Aman',16,8);

SAVEPOINT sp2;
BEGIN
    INSERT INTO students(name, age, class) VALUES ('Akash','wrong',9); --fails
EXCEPTION WHEN OTHERS THEN
    RAISE NOTICE 'Failed to insert Akash, rolling back to savepoint sp2';
    ROLLBACK TO SAVEPOINT sp2;
END;

-- Next insert
INSERT INTO students(name, age, class) VALUES ('Sita',17,10);

COMMIT; -- commit all successful inserts
```

Output:

The screenshot shows the pgAdmin 4 interface with the Object Explorer on the left and a query editor on the right. The query editor displays a DO block that inserts three rows into the 'students' table and then raises a notice. It includes an exception handling block for other errors. The 'Messages' tab shows a success message: 'NOTICE: Transaction Successfully Done'. A green success message box at the bottom right indicates 'Query returned successfully'.

```
9
10 DO $$$
11 BEGIN
12     INSERT INTO students(name, age, class) VALUES ('Yash',17,8);
13     INSERT INTO students(name, age, class) VALUES ('Diksha',19,9);
14     INSERT INTO students(name, age, class) VALUES ('Ashu',16,8);

15     RAISE NOTICE 'Transaction Successfully Done';

16     EXCEPTION
17         WHEN OTHERS THEN
18             RAISE NOTICE 'Transaction Failed..! Rolling back changes.';
19             RAISE;
20     END;
21 $$
22
23
24
25
26
27 SELECT * FROM students;
```

NOTICE: Transaction Successfully Done

DO

Query returned successfully in 100 msec.

Total rows: 0 Query complete 00:00:00.100 ✓ Query returned successfully

The screenshot shows the pgAdmin 4 interface with the Object Explorer on the left and a query editor on the right. The query editor demonstrates a transaction that starts with a savepoint 'sp1', inserts a row, creates another savepoint 'sp2', and then attempts to insert a row with a wrong data type ('wrong' instead of '16') which fails. An exception is caught, and the transaction is rolled back to savepoint 'sp2'. The 'Messages' tab shows an error message: 'ERROR: syntax error at or near "INSERT"'. A green success message box at the bottom right indicates 'Query returned successfully'.

```
29 -----WRONG DATA TYPE SCENARIO-----
30 BEGIN; -- start transaction
31
32 SAVEPOINT sp1;
33 INSERT INTO students(name, age, class) VALUES ('Ashu',16,8);
34
35 SAVEPOINT sp2;
36 BEGIN
37     INSERT INTO students(name, age, class) VALUES ('Akash','wrong',9); -- fails
38     EXCEPTION WHEN OTHERS THEN
39         RAISE NOTICE 'Failed to insert Akash, rolling back to savepoint sp2';
40         ROLLBACK TO SAVEPOINT sp2;
41     END;
42
43 -- Next insert
44 INSERT INTO students(name, age, class) VALUES ('Sita',17,10);
45
46 COMMIT; -- commit all successful inserts
47
```

ERROR: syntax error at or near "INSERT"

LINE 8: INSERT INTO students(name, age, class) VALUES ('Akash','...

SQL state: 42601

Character: 133

Total rows: 0 Query complete 00:00:00.112 ✓ Query returned successfully

File Object Tools Edit View Window Help

Object Explorer postgres/postgres@PostgreSQL 18*

Servers (1) PostgreSQL 18 Databases (1)

Databases (1) postgres

Casts Catalogs Event Triggers Extensions Foreign Data Wrappers Languages Publications Schemas Subscriptions

Login/Group Roles Tablespaces

Query Scratch Pad

```
31 SAVEPOINT sp1;
32 INSERT INTO students(name, age, class) VALUES ('Ashu',16,8);
33
34
35 SAVEPOINT sp2;
36 BEGIN
37     INSERT INTO students(name, age, class) VALUES ('Akash','wrong',9); -- fails
38 EXCEPTION WHEN OTHERS THEN
39     RAISE NOTICE 'Failed to insert Akash, rolling back to savepoint sp2';
40     ROLLBACK TO SAVEPOINT sp2;
41 END;
42
43 -- Next insert
44 INSERT INTO students(name, age, class) VALUES ('Sita',17,10);
45
46 COMMIT; -- commit all successful inserts
47
48
49 select * from students;
```

Data Output Messages Notifications

Showing rows: 1 to 3 Page No: 1 of 1

	id [PK] integer	name character varying (50)	age integer	class integer
1	1	Yash	17	8
2	2	Diksha	19	9
3	3	Ashu	16	8

Total rows: 3 Query complete 00:00:00.155 CRLF Ln 49, Col 1

3. Learning Outcomes:

- Understand the concept of PostgreSQL transactions and how to start, commit, and rollback.
- Learn how to use **SAVEPOINT** to handle partial rollbacks within a transaction.
- Practice controlled error handling for individual insert failures without affecting other successful operations.
- Gain experience in maintaining **data integrity** while performing multiple inserts.
- Learn to generate informative **NOTICES** to monitor transaction progress and errors.