### **Lab 24- Grant & Revoke in PostgreSQL**

Here's a lab exercise to demonstrate the usage of GRANT and REVOKE in PostgreSQL:

**Objective:**

To understand and practice granting and revoking privileges in PostgreSQL.

**Requirements:**

* PostgreSQL installed on your system.
* Basic knowledge of SQL queries.

**Exercise:**

1. **Setup Database:**

If you haven't already, create a sample database and table. For this exercise, let's create a table named employees with the following schema:

CREATE TABLE employees (

id SERIAL PRIMARY KEY,

name VARCHAR(100),

department VARCHAR(100)

);

1. **Insert Sample Data:**

Insert some sample data into the employees table:

INSERT INTO employees (name, department) VALUES

('John Doe', 'HR'),

('Jane Smith', 'IT'),

('Michael Johnson', 'Finance');

1. **Grant Privileges:**

Grant SELECT privilege on the employees table to a new user named analyst.

GRANT SELECT ON TABLE employees TO analyst;

1. **Verify Grant:**

Connect to the database with the analyst user and try to select data from the employees table:

SELECT \* FROM employees;

You should be able to retrieve data successfully.

1. **Revoke Privileges:**

Revoke the previously granted SELECT privilege from the analyst user.

REVOKE SELECT ON TABLE employees FROM analyst;

1. **Verify Revoke:**

Try to select data from the employees table again with the analyst user:

SELECT \* FROM employees;

You should receive an error message indicating that permission is denied.

1. **Experimentation:**

Grant and revoke other privileges such as INSERT, UPDATE, DELETE to different users and roles.

Grant privileges on specific columns instead of the entire table.

Experiment with granting privileges on multiple tables or schemas at once.

1. **Cleanup:**

Don't forget to clean up your database after you're done with the exercise:

DROP TABLE employees;

**Conclusion:**

Understanding how to grant and revoke privileges in PostgreSQL is essential for managing access control and maintaining data security within your database environment. Regular practice with GRANT and REVOKE statements will help you effectively control user permissions and ensure that only authorized users can access and manipulate data as needed.