**Lab Exercise 21- Using Tools like pgAdmin and pg\_stat\_statements in PostgreSQL**

**Objective**

* Learn to use **pgAdmin** for GUI-based PostgreSQL management
* Enable and use **pg\_stat\_statements** to analyze query performance
* Interpret performance metrics and optimize SQL

**Prerequisites**

* PostgreSQL installed
* pgAdmin installed
* Superuser access to PostgreSQL
* Basic SQL knowledge

**Part 1: Using pgAdmin**

**Step 1: Open pgAdmin**

* Launch **pgAdmin** from your system menu.
* Connect to your PostgreSQL server using the superuser (e.g., postgres).

**Step 2: Explore Server Dashboard**

In pgAdmin:

* Click the server > Databases > your database
* Go to **Dashboard** to see:
  + Server load
  + Sessions
  + Transactions
  + Tuples read/returned/fetched
  + Cache hit ratio

**Step 3: Use Query Tool**

* Right-click your database > **Query Tool**
* Run sample queries:

SELECT \* FROM pg\_user;

SELECT \* FROM employees WHERE salary > 50000;

* Click the **Execution Plan** tab to view visual explain plans.

**Part 2: Using pg\_stat\_statements**

**Step 1: Enable the Extension**

In postgresql.conf (edit and restart PostgreSQL):

shared\_preload\_libraries = 'pg\_stat\_statements'

Then restart the server.

**Step 2: Create the Extension**

Log into the database as superuser and run:

CREATE EXTENSION pg\_stat\_statements;

**Step 3: Query the Collected Data**

Run a few sample queries:

SELECT \* FROM employees WHERE name = 'John';

SELECT \* FROM employees WHERE salary > 40000;

Now view statistics:

SELECT query,

calls,

total\_time,

rows,

mean\_time,

stddev\_time

FROM pg\_stat\_statements

ORDER BY total\_time DESC

LIMIT 5;

This will show the slowest and most frequent queries.

**Step 4: Reset Statistics**

You can reset statistics using:

SELECT pg\_stat\_statements\_reset();

**Additional Exercises**

* **Modify and rerun** a query to see how index use affects performance
* Use EXPLAIN (ANALYZE, BUFFERS) to inspect how PostgreSQL executes a query
* Use pgAdmin’s **Query Tool** to compare run-time metrics

**Summary**

In this lab, you:

* Explored system metrics and dashboards with **pgAdmin**
* Enabled and used **pg\_stat\_statements** to track slow and frequent queries
* Learned how to reset and interpret query statistics