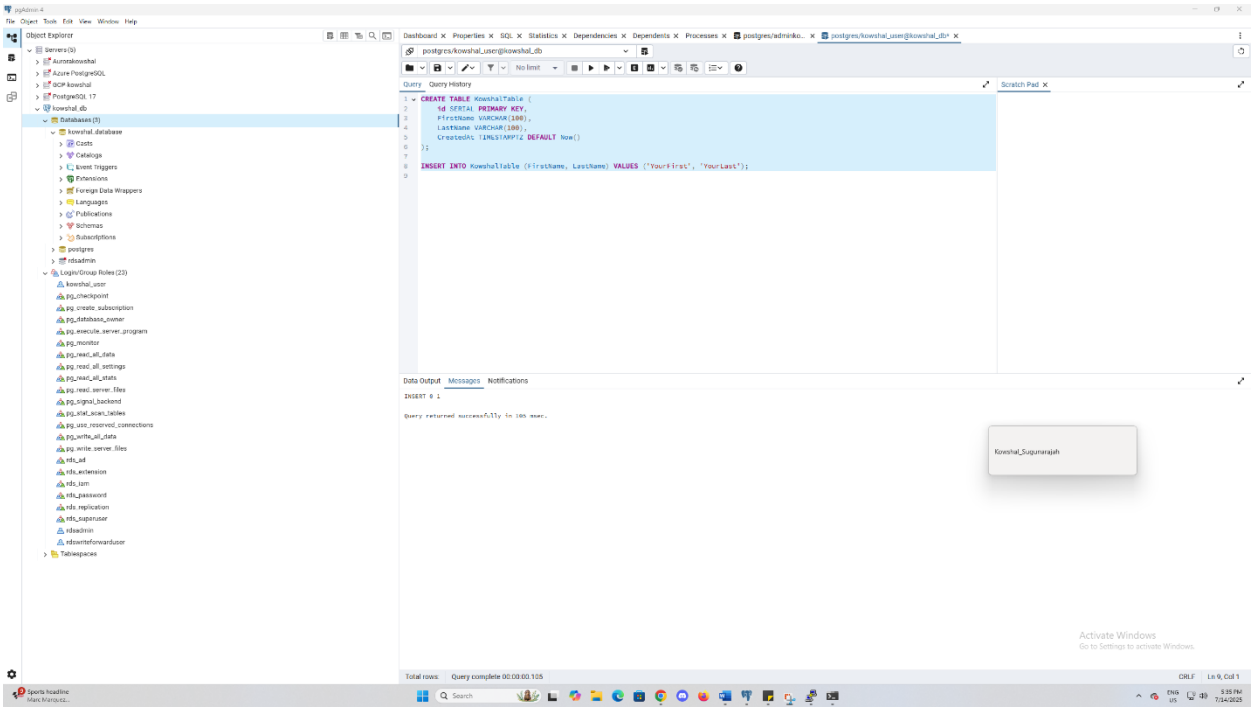
Cloud Project

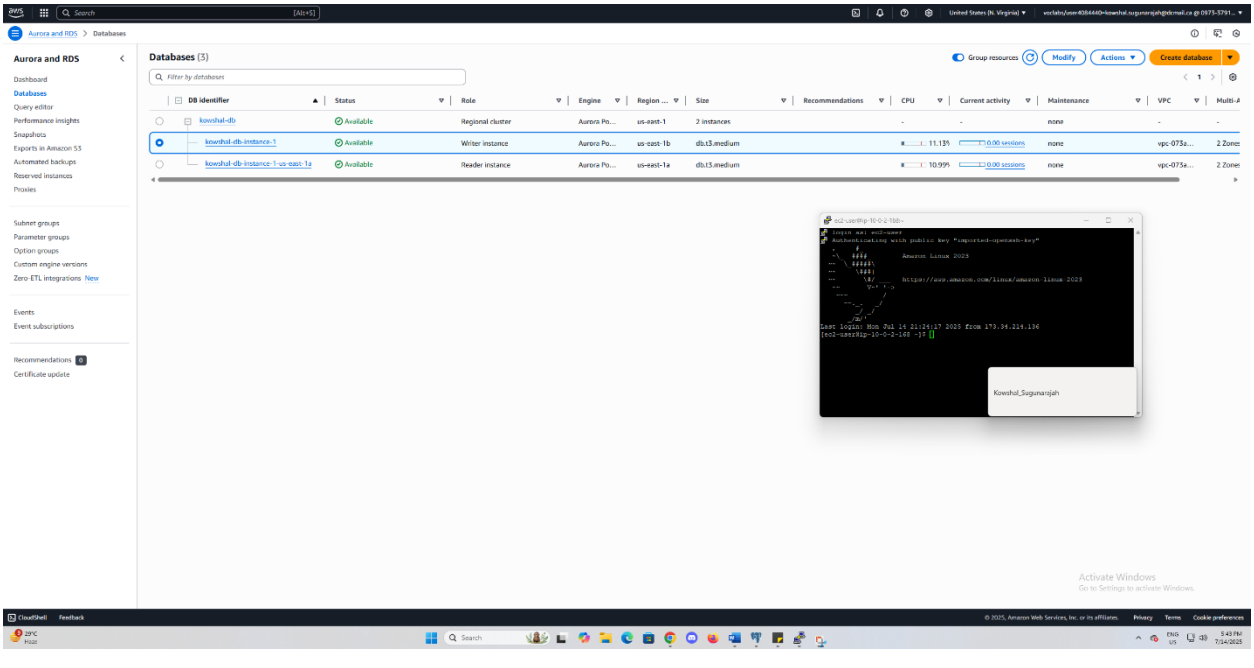
Connection Postgre with AWS, GCP and AZURE

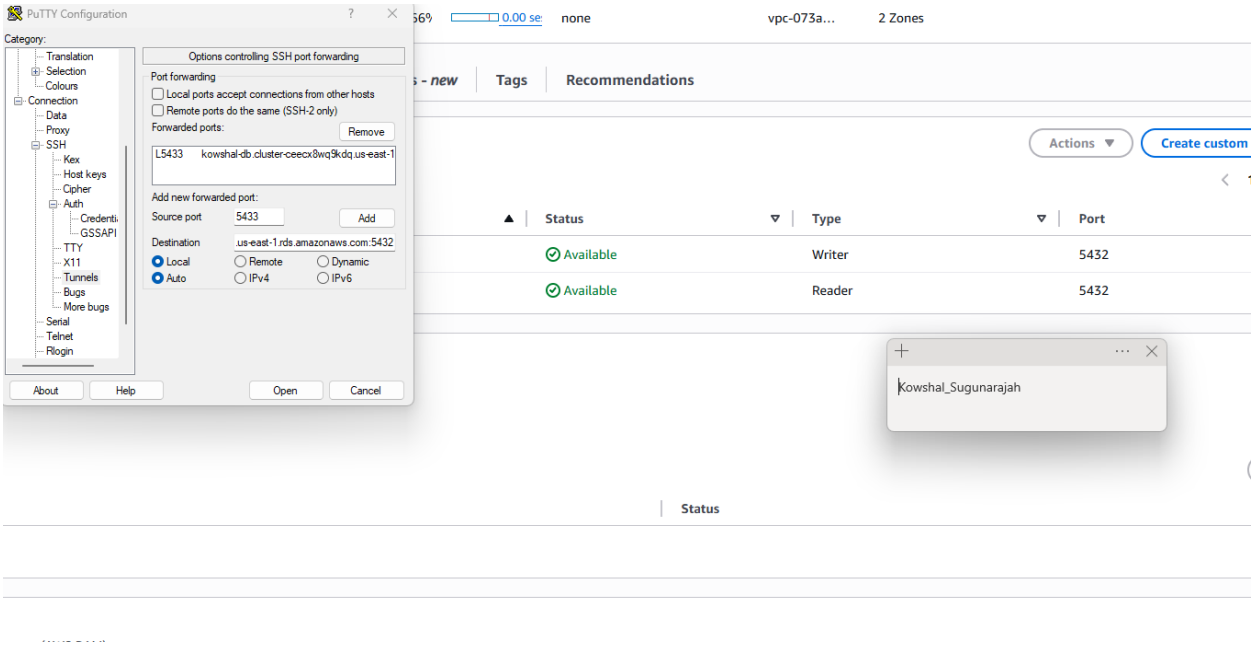
AWS

This project provided practical and in-depth experience in deploying a cloud-based database system using AWS. By building everything from scratch, including a new VPC, subnets, security groups, and a private Aurora PostgreSQL instance—I developed a solid understanding of secure cloud architecture and networking. Setting up an EC2 bastion host and using SSH tunneling to connect to a private database through pgAdmin helped reinforce concepts of secure access and private resource isolation. Additionally, performing database snapshot and recovery tasks demonstrated essential real-world backup strategies. Overall, this lab not only strengthened my cloud deployment skills but also deepened my confidence in managing production-grade infrastructure securely and efficiently.

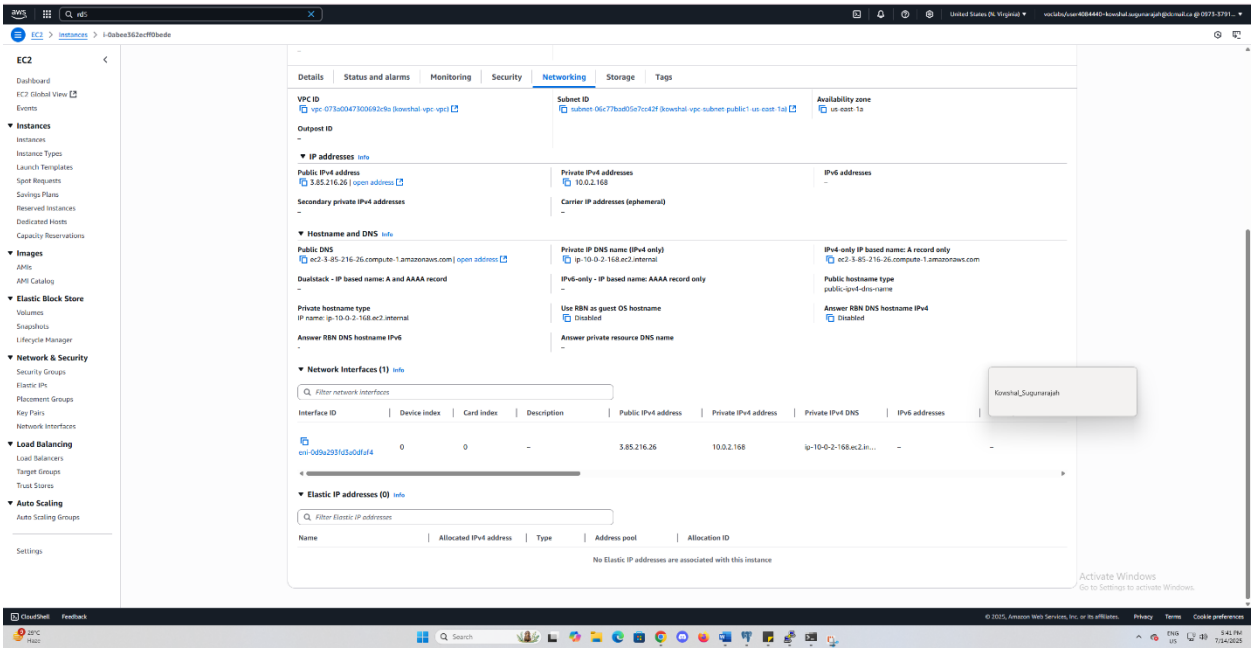
- **Created a new VPC (Virtual Private Network)**
- **Created 2 subnets with both public and private**
- **Created new security group with Postgre access**
- **Created subnet group**
- **Created Aurora Postgre database with public access**
- **Created a new bastion server with a new security group**
- **With SSH made a connection with AWS and Postgre server using a Tunnel**


CREATE TABLE KowshalTable (
Id SERIAL PRIMARY KEY,
Firstname VARCHAR(100),
LastName VARCHAR(100),
CreatedAt TIMESTAMPTZ DEFAULT Now()
);
INSERT INTO KowshalTable (Firstname, LastName) VALUES ('YourFirst', 'YourLast');
Kowshal_Sugunarajah

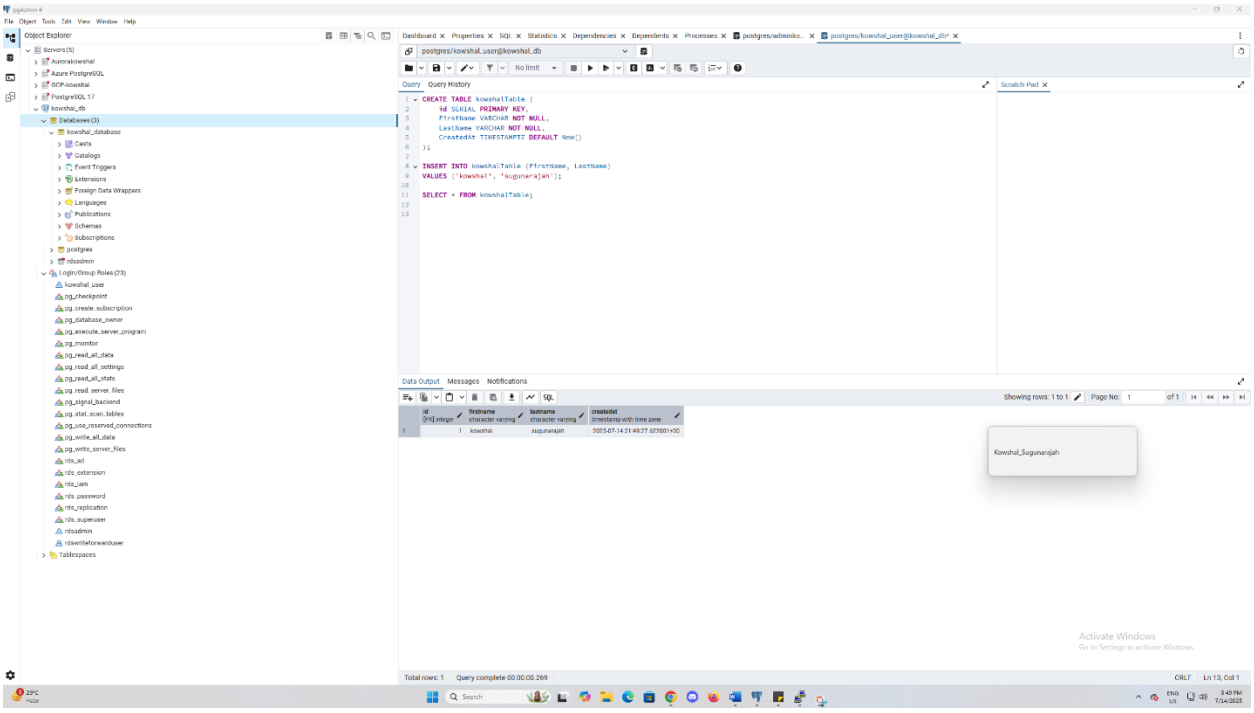

Aurora and RDS
Databases (3)
kowshal-db
kowshal-db-instance-1
kowshal-db-instance-1-us-east-1a
Kowshal_Sugunarajah



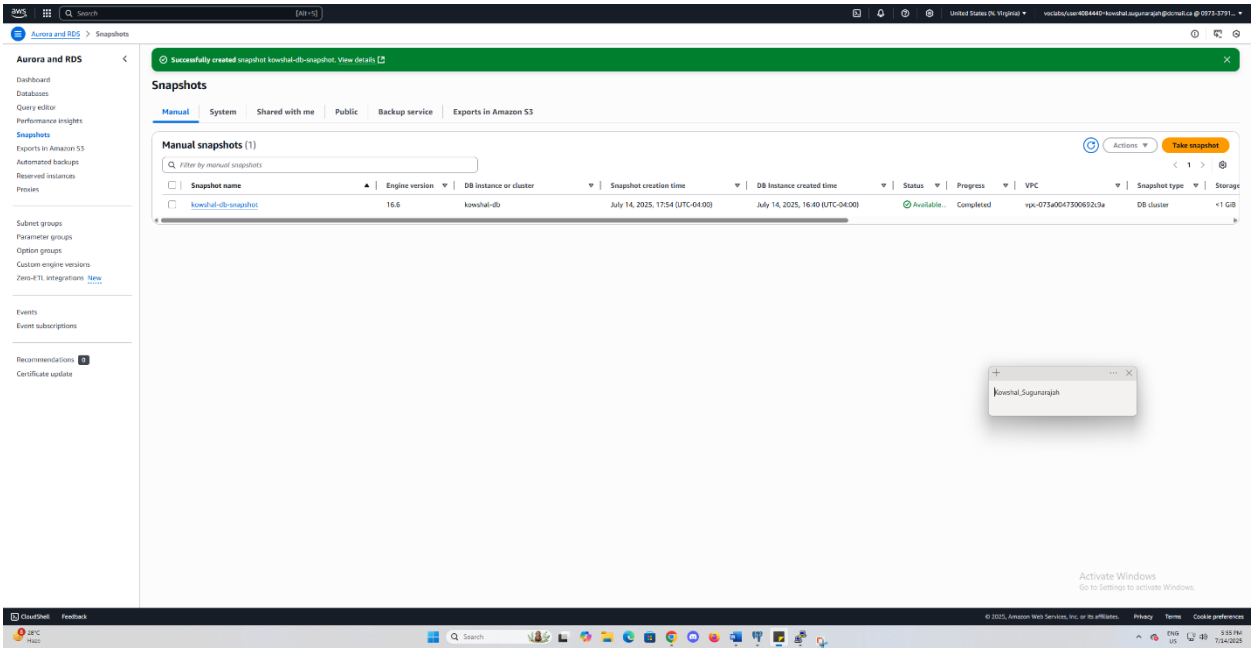
EC2 bastion server configurations:



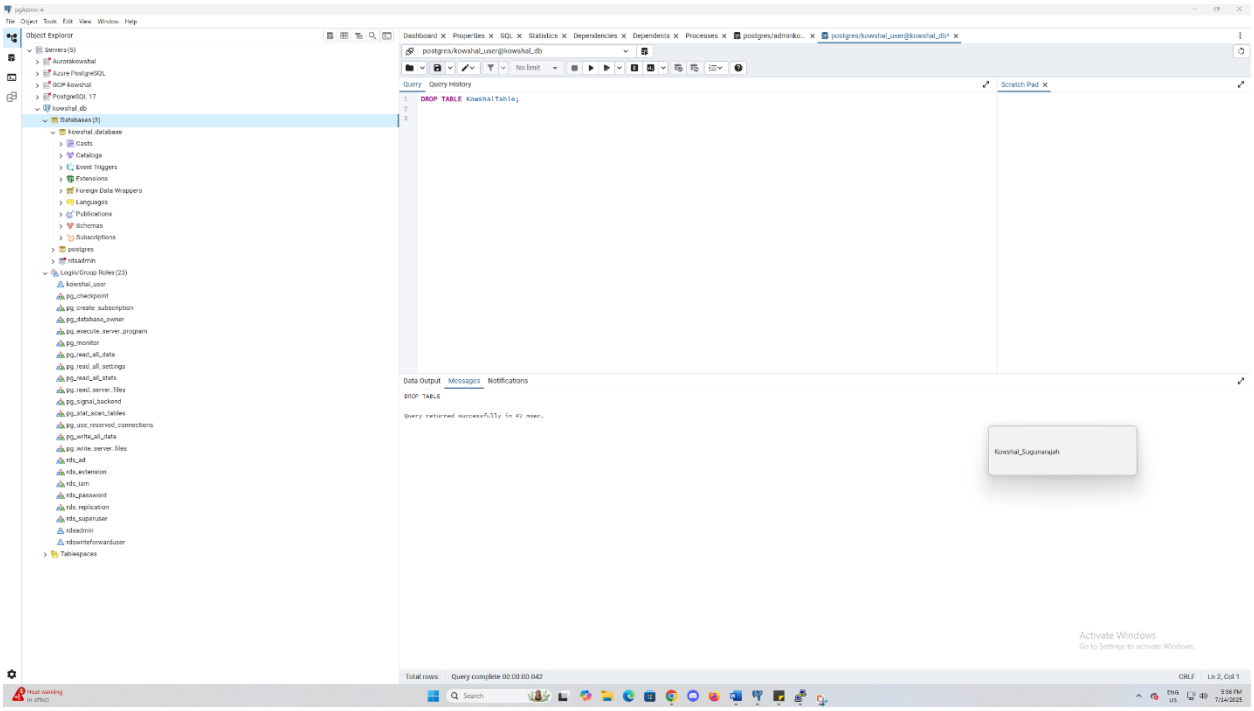
Sample set of data created inside database:



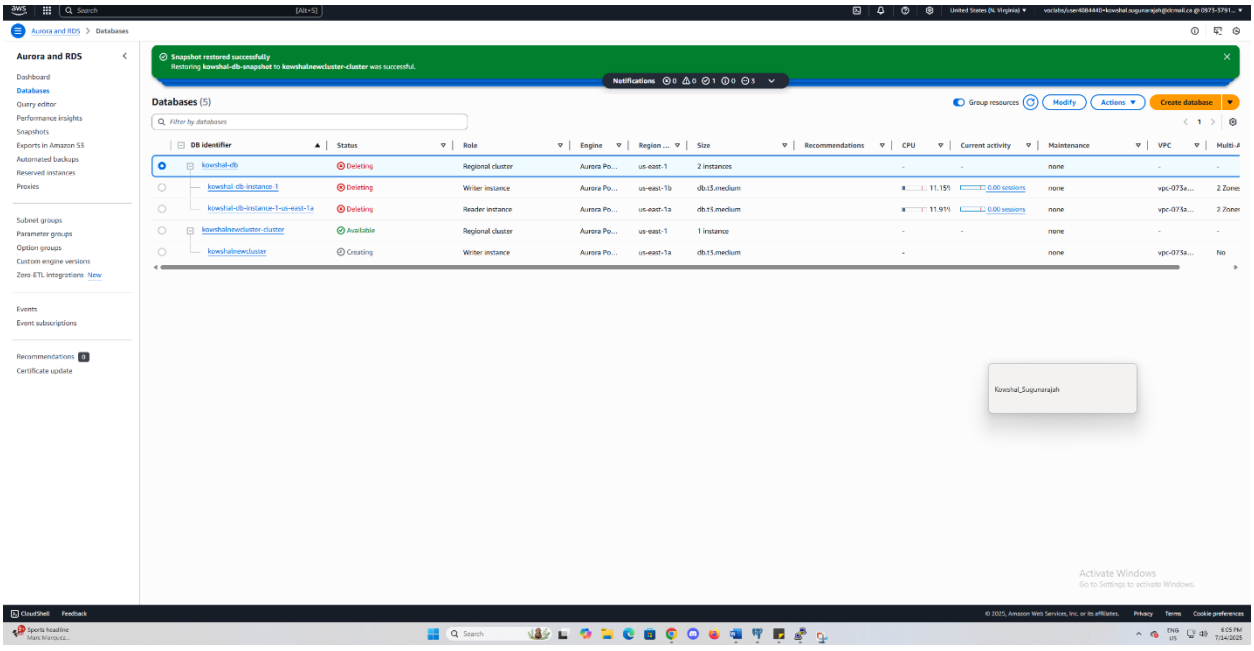
Took a snapshot of this database:



With query command deleted the data:

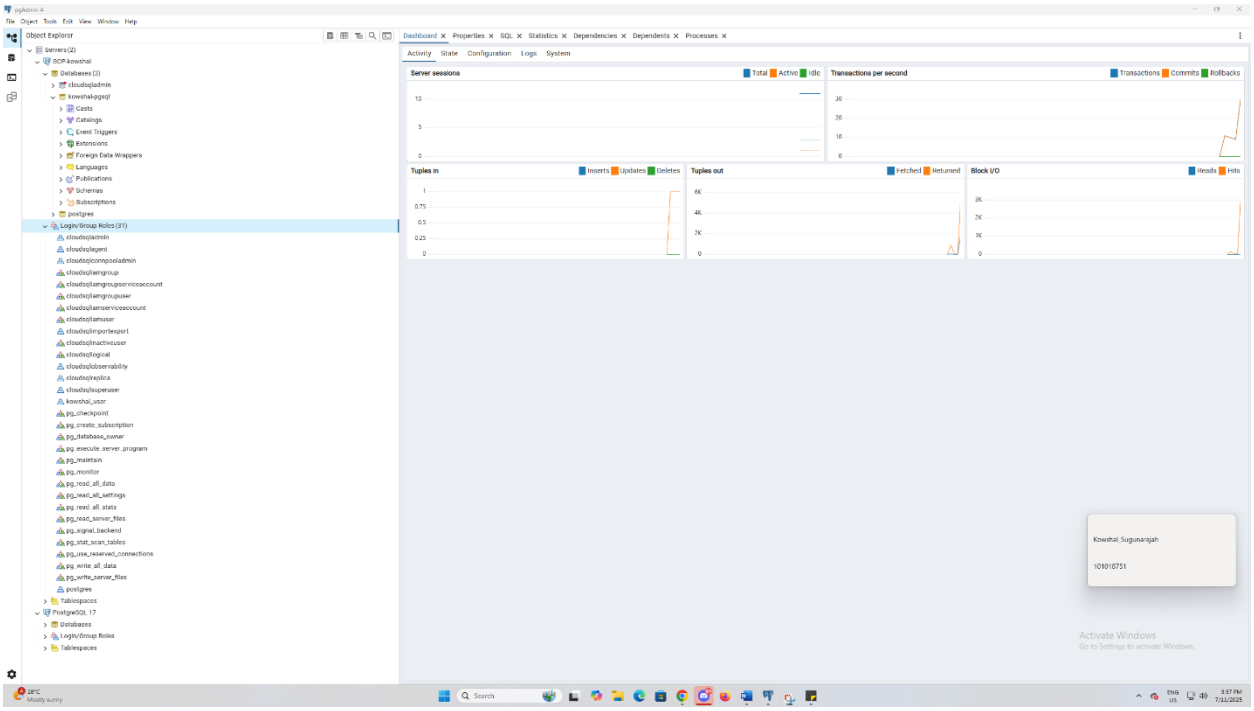


Recreated the data from the snapshot:



GCP

Database connection with GCP



Azure

Database connection with Azure