

## Experiment No 9

**Student Name:** Bhavya Gupta  
**Branch:** BE CSE (IS)  
**Semester:** 5th  
**Subject:** ADBMS

**UID:** 23BIS70147  
**Section/Group:** 23AIT-KRG(2A)  
**Date of Performance:** 24-Oct-2025  
**Subject Code:** 23-CSP-333

### Aim:

To create and connect a PostgreSQL database instance on Amazon RDS (Relational Database Service)

### Objective:

- To understand the steps involved in launching a database instance using Amazon RDS.
- To configure a database for public access and connect it with a local client (pgAdmin).
- To perform basic SQL operations (CREATE, INSERT, SELECT).

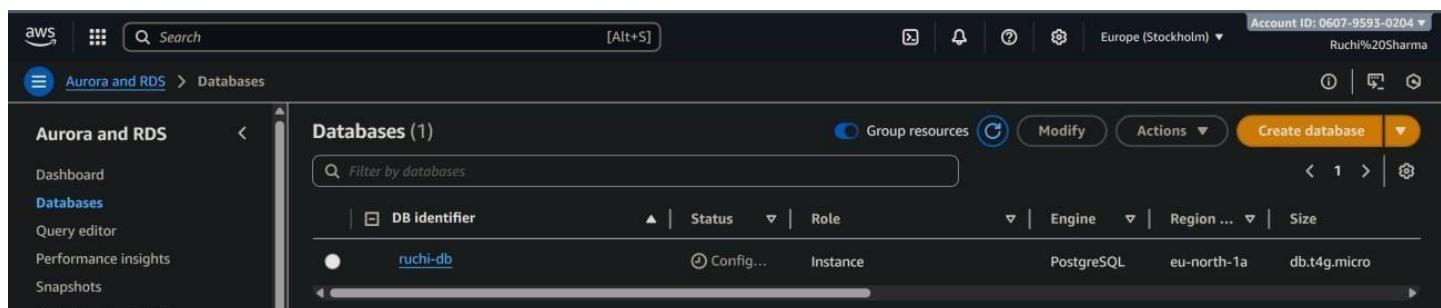
### Tools / Software

- Amazon Web Services (AWS)
- PostgreSQL
- pgAdmin 4
- RDS (Relational Database Service)

### Program:

#### 1. Step 1: Create and Configure Database Instance

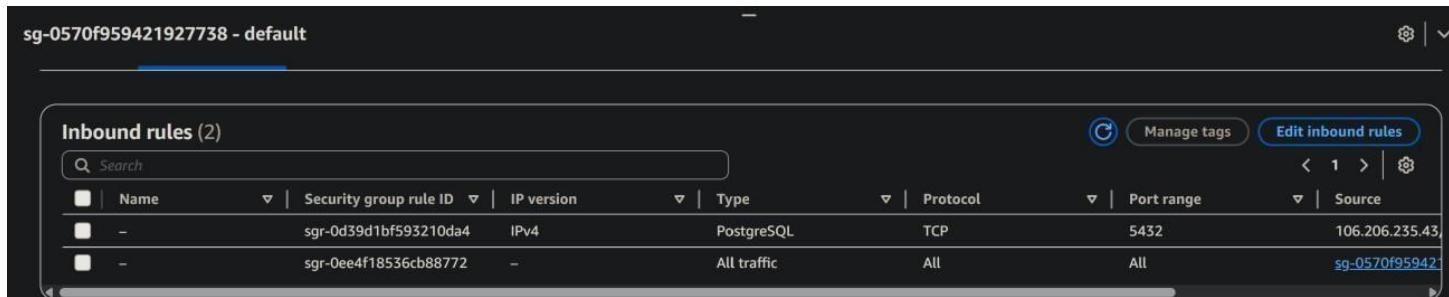
- Login to AWS Console → RDS → Create database, select Standard create and PostgreSQL under the Free Tier template.
- Set DB identifier: ruchi-db, Username: postgres, choose db.t3.micro, 20 GB gp2 storage, and enable Public access.
- Click Create database and wait until the status shows Available in the RDS dashboard.



The screenshot shows the AWS RDS Databases page. The left sidebar has links for Aurora and RDS, Dashboard, Databases (selected), Query editor, Performance insights, Snapshots, and Export to Amazon S3. The main area title is 'Databases (1)'. It shows a table with one row for 'ruchi-db'. The columns are DB identifier, Status, Role, Engine, Region ..., and Size. The status is 'Available', engine is 'PostgreSQL', region is 'eu-north-1a', and size is 'db.t4g.micro'. There are buttons for Group resources, Modify, Actions, and Create database.

## 2. Step 2: Configure Security Group (Allow Local Access Only)

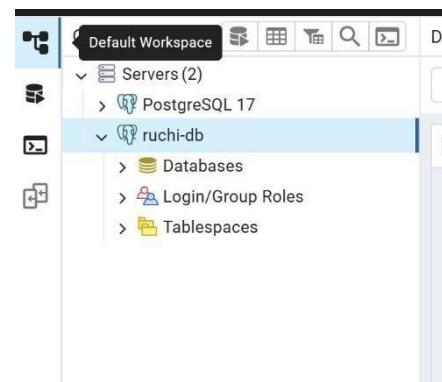
- In AWS Console → go to RDS → Databases → click your DB (ruchi-db).
- Open the Connectivity & Security tab.
- Under VPC security groups, click the linked group name (it opens EC2 security groups).
- Click Edit inbound rules → Add rule
- Type: PostgreSQL
- Protocol: TCP
- Port: 5432
- Source: My IP
- Click Save rules.



Inbound rules (2)						
	Name	Security group rule ID	IP version	Type	Protocol	Port range
<input checked="" type="checkbox"/>	-	sgr-0d39d1bf593210da4	IPv4	PostgreSQL	TCP	5432
<input checked="" type="checkbox"/>	-	sgr-0ee4f18536cb88772	-	All traffic	All	106.206.235.43

## Step 3: Connect Database Using pgAdmin

- Open pgAdmin 4 on your local system.
- Right-click Servers → Create → Server.
- Under the General tab, enter the name: poste.
- Under the Connection tab, fill in the following details:
- Host name/address: ruchi- db.xxxxxxx.rds.amazonaws.com
- Port: 5432
- Username: poste
- Check Save password.
- Click Save to connect your RDS PostgreSQL database.



## 5. Learning Outcomes:

- Understand the procedure to provision and configure a PostgreSQL instance using AWS RDS.
- Configure security groups and network access controls for secure database connectivity.
- Establish a remote database connection using pgAdmin and verify successful access.