



EXPERIMENT- 09

Student Name: Ashutosh Yadav
Branch: BE-CSE
Semester: 05
Subject Name: ADBMS

UID: 23BCS11023
Section/Group: KRG-1-B
Date of Performance: 28/10/25
Subject Code: 23CSP-333

1. Aim: To create and connect a PostgreSQL database instance on **Amazon RDS**
(Relational Database Service)

2. 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).

3. Tools / Software

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

4. Program:

Step 1: Create and Configure Database Instance

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

DB instance size

- Production db.r7g.xlarge
4 vCPUs
32 GiB RAM
400 GiB
1.946 USD/hour
- Dev/Test db.t4g.large
2 vCPUs
16 GiB RAM
200 GiB
0.278 USD/hour
- Free tier db.t4g.micro
2 vCPUs
1 GiB RAM
20 GiB
0.019 USD/hour

DB instance identifier
Type a name for your DB instance. The name must be unique across all DB instances owned by your AWS account in the current AWS Region.
`ashutosh-db`

The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 63 alphanumeric characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

Master username [Info](#)
Type a login ID for the master user of your DB instance.
`postgres`

1 to 16 alphanumeric characters. The first character must be a letter.

Credentials management
You can use AWS Secrets Manager or manage your master user credentials.

Aurora and RDS > Databases

Creating database ashutosh-db
Your database might take a few minutes to launch. You can use settings from ashutosh-db to simplify configuration of suggested database add-ons while we finish creating your DB for you.

Databases (2)

DB identifier	Status	Role	Engine	Region ...	Size
ashutosh-db	Creating	Instance	PostgreSQL	-	db.t4g.micro
database-1	Available	Instance	PostgreSQL	eu-north-1a	db.t4g.micro

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

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

1. In AWS Console → go to RDS → Databases → click your DB .
2. Open the Connectivity & Security tab.
3. Under VPC security groups, click the linked group name (it opens EC2 security groups).
4. Click Edit inbound rules → Add rule

Type:

- ② PostgreSQL
- ② Protocol: TCP ②
- Port: 5432
- ② Source: My IP

5. ClickSaverules.

Inbound rules [Info](#)

Security group rule ID	Type Info	Protocol Info	Port range	Source Info	Description - optional Info
sgr-038480f94d3b4cd53	All traffic	All	All	Custom	<input type="text"/> sg-08ff1930ac7edd3aa X
-	PostgreSQL	TCP	5432	My IP	<input type="text"/> 112.196.95.2/32 X

[Add rule](#)

Cancel [Preview changes](#) [Save rules](#)

CloudShell Feedback [Console Mobile App](#) © 2025, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

Aurora and RDS [Edit](#) [Details](#) [Logs](#) [Metrics](#) [Actions](#) [Copy link](#)

DB identifier ashutosh-db

Status [Edit](#) Backing-up

Role Instance

Engine PostgreSQL

Region & AZ eu-north-1b

Connectivity & security

Endpoint [Edit](#) ashutosh-db.cf8seu2mkh0r.eu-north-1.rds.amazonaws.com

Port [Edit](#) 5432

Networking

Availability Zone eu-north-1b

VPC vpc-0b1cd22aeef338973

Subnet group default-vpc-0b1cd22aeef338973

Subnets subnet-0c72da932746f384a
subnet-07ba54860723c3b0
subnet-0aab8c5189c3b5407

Network type IPv4

Security

VPC security groups default (sg-08ff1930ac7edd3aa) [Edit](#) Active

Publicly accessible No

Certificate authority [Edit](#) rds-ca-rsa2048-g1

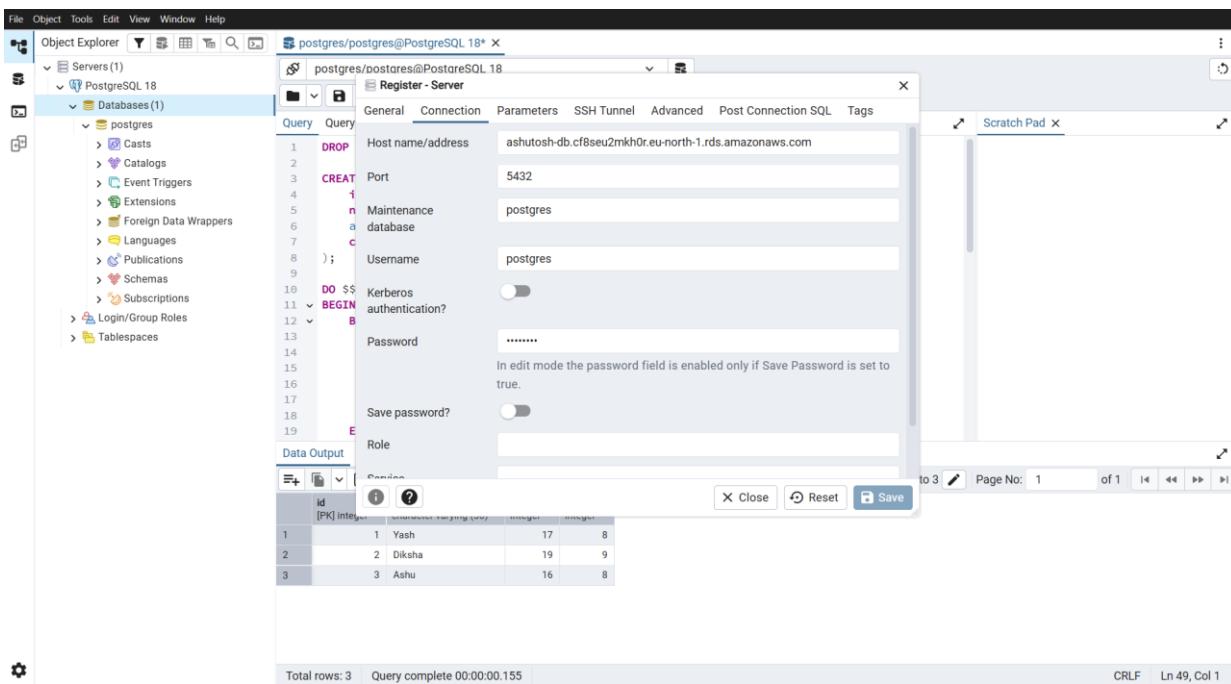
Certificate authority date May 25, 2061, 03:29 (UTC+05:30)

DB instance certificate expiration date November 04, 2026, 12:59 (UTC+05:30)

Step3: Connect Database Using pgAdmin

1. Open pgAdmin 4 on your local system.
2. Right-click Servers → Create → Server.
3. Under the General tab, enter the name: postgres.
4. Under the Connection tab, fill in the following details:
 - Host name/address from instance connectivity & security
 - Port: 5432 Username and password as while creating the instance.

5. Click Save to connect your RDS PostgreSQL database.



Learning Outcomes:

1. Understand and configure RDS instances:

Learners will be able to create and configure a PostgreSQL database instance on Amazon RDS by selecting appropriate instance types, storage options, and security groups.

2. Establish secure database connections:

Learners will be able to connect to the PostgreSQL RDS instance using database clients or application code through proper authentication, networking, and endpoint management.

3. Manage and monitor RDS resources:

Learners will be able to manage database parameters, perform basic maintenance tasks, and use AWS monitoring tools (like CloudWatch) to ensure database performance and availability.