



## Worksheet 9

**Student Name:** Sumedh Vats  
**Branch:** CSE  
**Semester:** 5th  
**Subject Name:** ADBMS

**UID:** 23BCS11261  
**Section/Group:** KRG 1-B  
**Date of Performance:** 30/10/2025  
**Subject Code:** 23CSP-333

**1. Aim:** To understand and implement the setup of Amazon Relational Database Service (AWS RDS) by creating a database instance, configuring security groups, and establishing a secure connection between the local pgAdmin tool and the RDS instance hosted on the AWS Cloud.

### 2. Objective:

- To learn the basic concepts and features of Amazon Relational Database Service (AWS RDS).
- To create and configure a new RDS database instance on the AWS Management Console.
- To understand the role and configuration of security groups for controlling database access.
- To connect a local pgAdmin client to the AWS RDS instance securely using proper credentials and endpoint details.
- To verify successful database connectivity and perform basic operations through pgAdmin.

### 3. Code & Output:

#### 1. Sign-in

aws

Root user sign in ⓘ

Enter the password for  
sumedhvats2004@gmail.com (not you?)

Password  
.....

☐ Show password [Forgot password?](#)

[Sign in](#)

[Sign in to a different account](#)

[Create a new AWS account](#)

**Amazon Lightsail**

Lightsail is the easiest way  
to get started on AWS

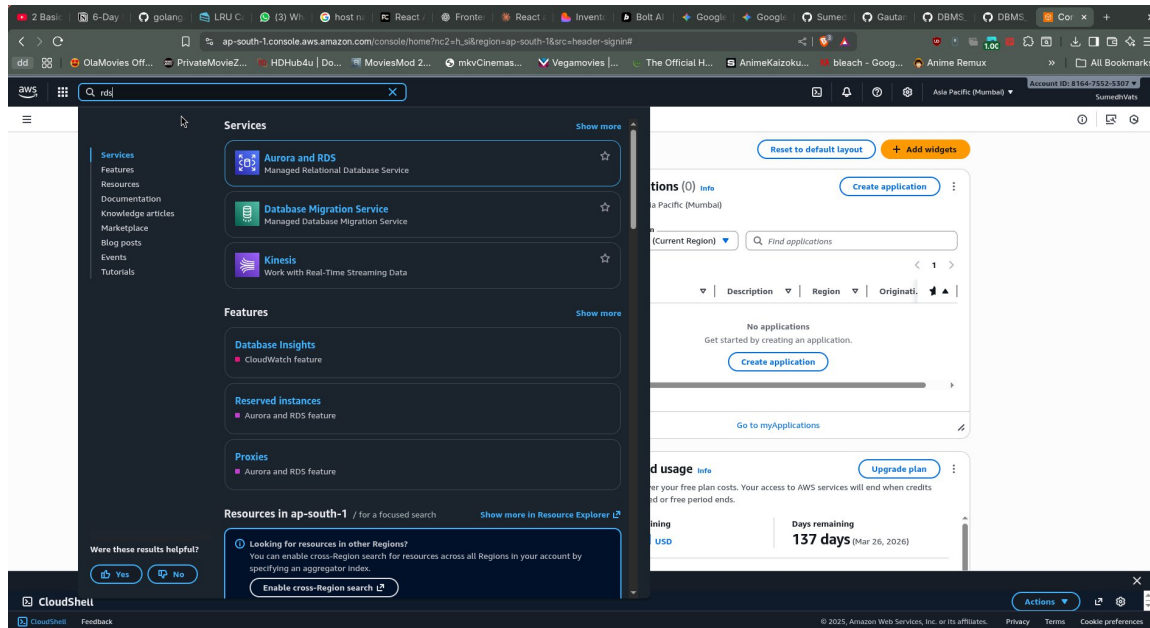
[Learn more »](#)



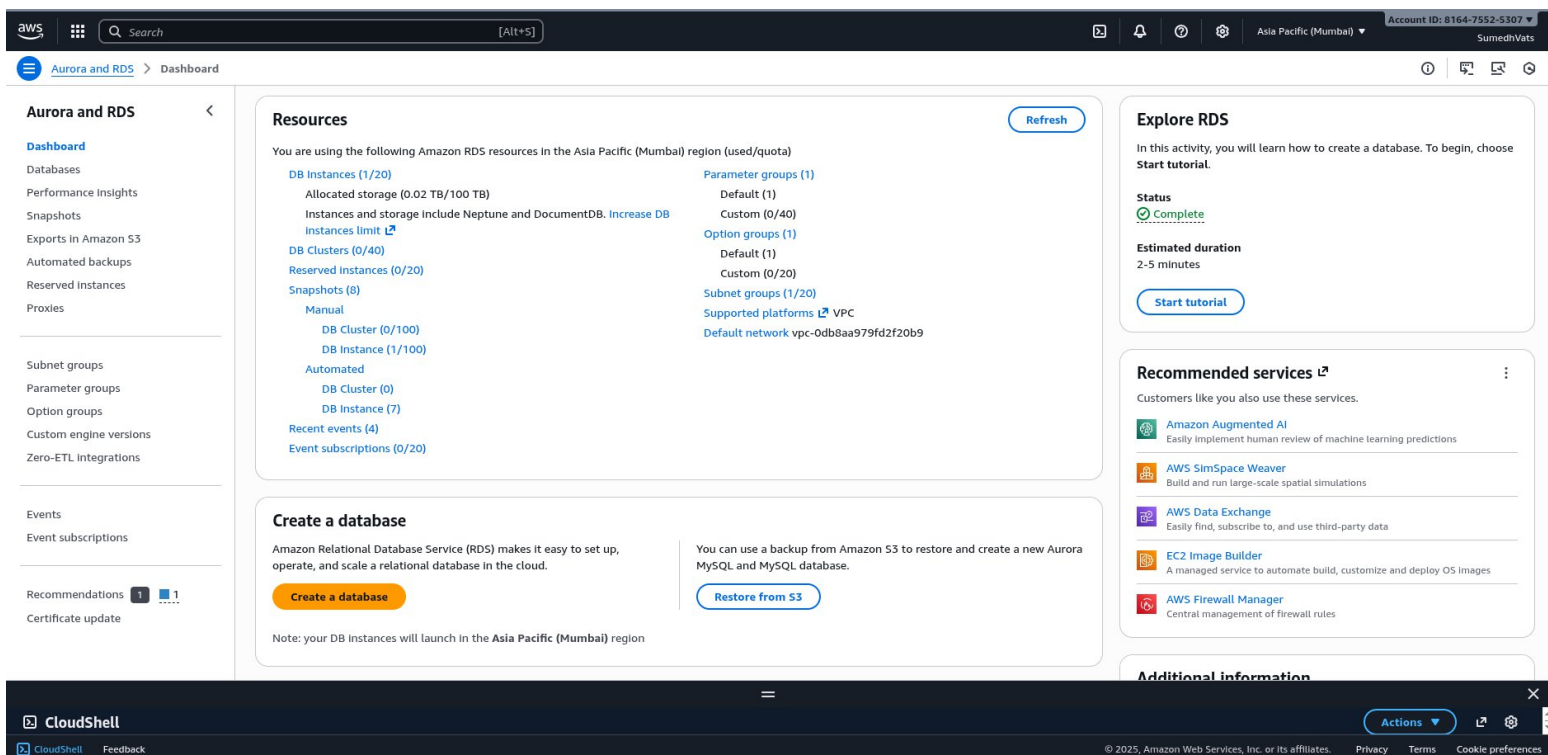
# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

## 2. Navigating to RDS Service



## 3. Amazon RDS Dashboard Overview

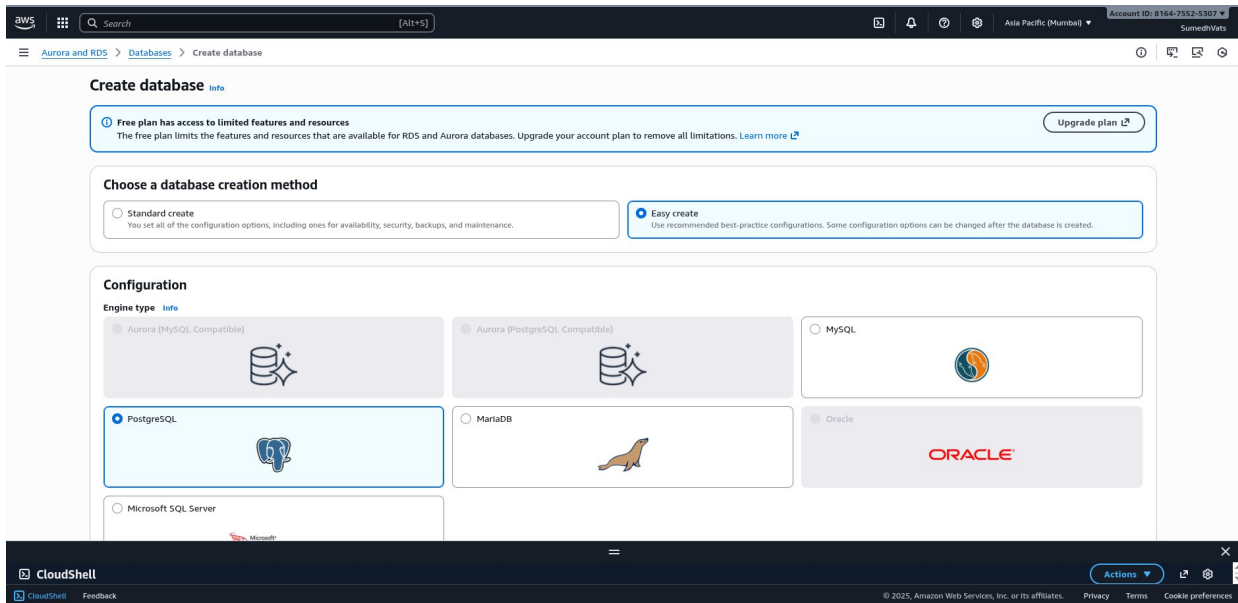




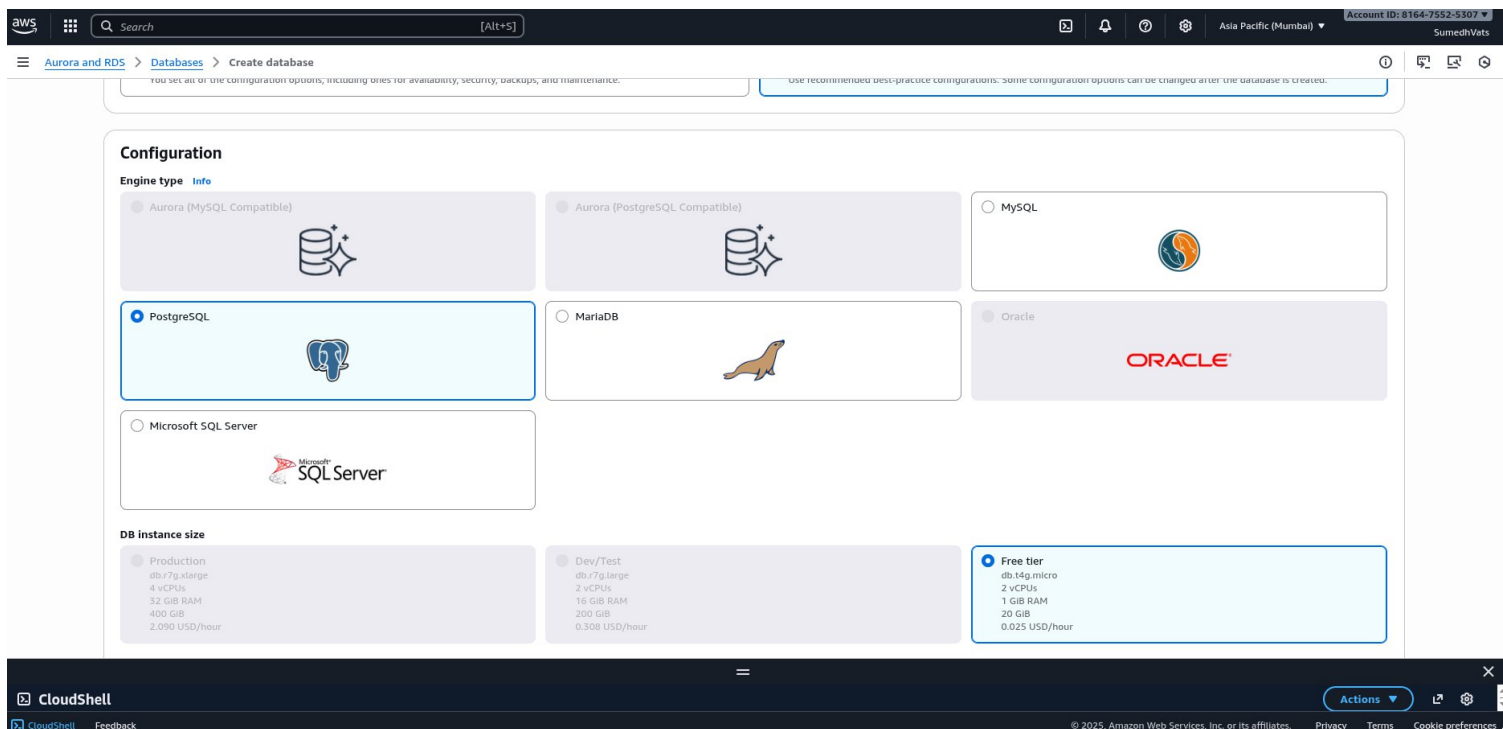
# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

## 4. Creating a New Database Instance



## 5. Selecting PostgreSQL as Database Engine





# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

## 6. Choosing Deployment Option and Template

The screenshot shows the 'Create database' page in the AWS Aurora and RDS console. The 'DB instance size' section offers three options: Production (db.t4g.large), Dev/Test (db.t4g.micro), and Free tier (db.t4g.micro). The 'DB instance identifier' field is set to 'database-2'. The 'Master username' field is set to 'postgres'. The 'Credentials management' section has two options: 'Managed in AWS Secrets Manager - most secure' and 'Self managed'. The 'Master password' field is currently empty, with a message indicating it is invalid. The 'Confirm master password' field is also empty. The 'Set up EC2 connection - optional' section is visible at the bottom.

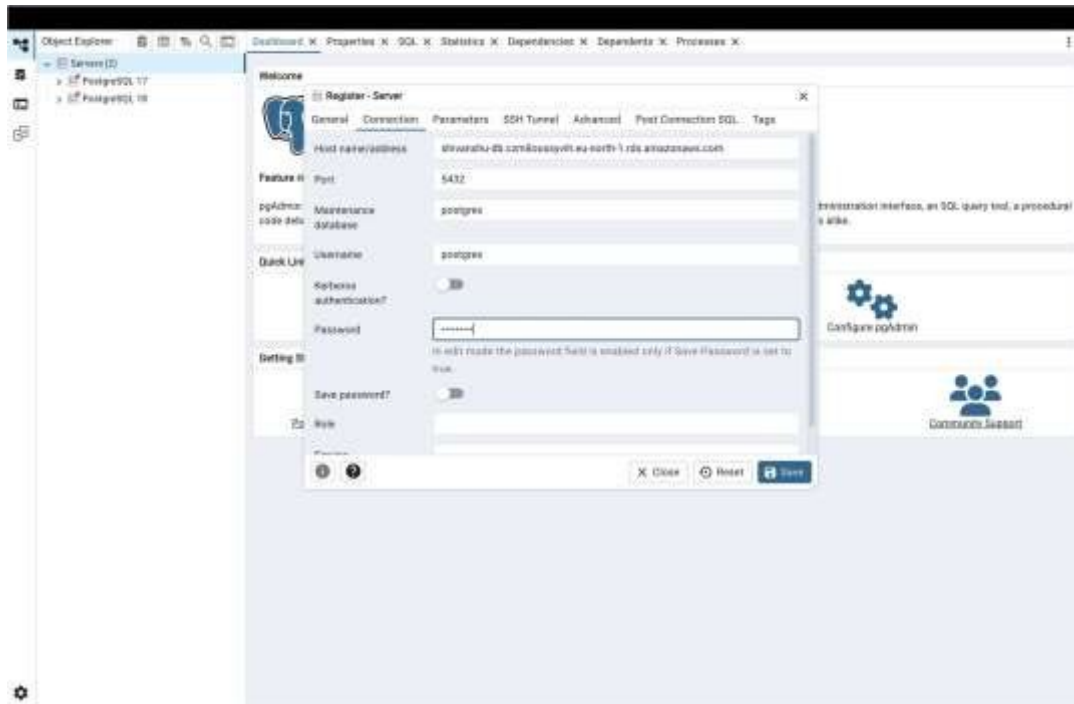
## 7. Configuring Database Settings (Name, Username, Password)

The screenshot shows the 'Databases (1)' page in the AWS Aurora and RDS console. The table lists the following database instance:

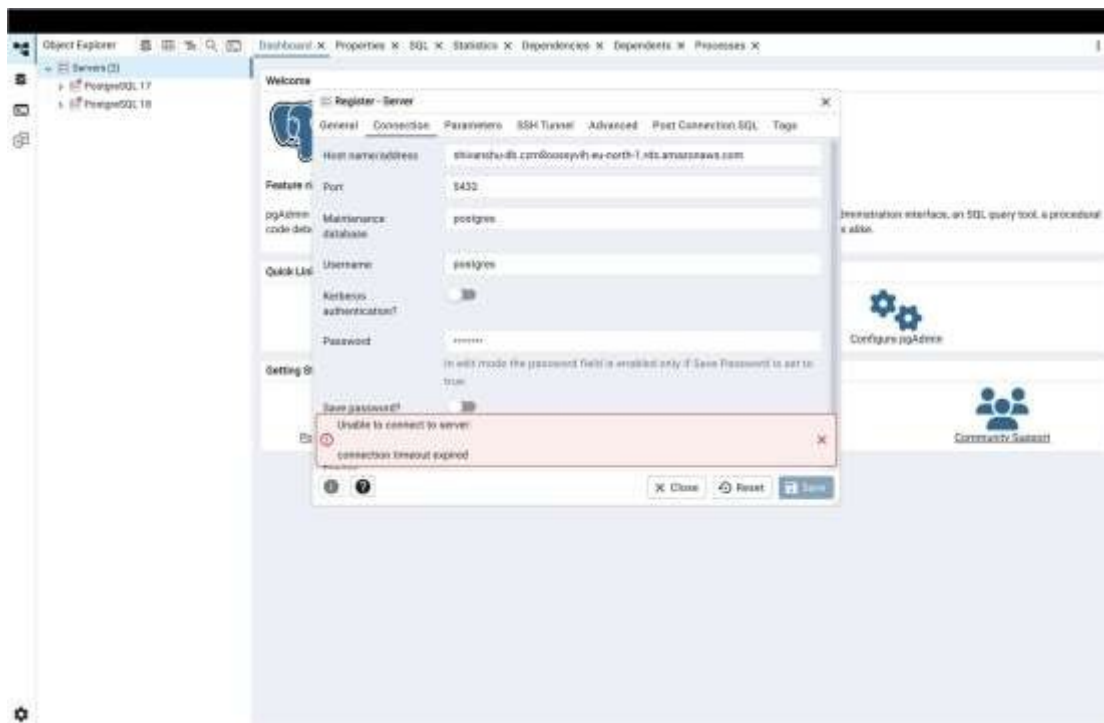
DB identifier	Status	Role	Engine	Region	Size	Recommendations	CPU	Current activity	Maintenance
database-1	Available	Instance	PostgreSQL	ap-south-1a	db.t4g.micro	1 informational	3.63%	0.00 sessions	none



## 10. Gr Setting Up Security Groups for RDS Access



## 11. Additional Database Configuration Options





# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

## 12. Reviewing and Creating the Database Instance

The screenshot displays the AWS Aurora and RDS console for an instance named 'database-1'. The interface includes a left-hand navigation menu with options like Dashboard, Performance Insights, Snapshots, and Subnet groups. The main content area shows the 'Summary' tab, which provides an overview of the instance's status (Available), CPU usage (3.63%), and engine (PostgreSQL). Below the summary, the 'Connectivity & security' tab is active, showing details for the endpoint (database-1.c2alogq22pxe.ap-south-1.rds.amazonaws.com), port (5432), and network configuration (VPC, Subnet group, Subnets, and Network type). The 'Security' section shows the VPC security groups and certificate authority information.

## 13. RDS Instance Creation in Progress

The screenshot shows the AWS Security Groups console, specifically the 'Edit inbound rules' page for a security group. The page displays a table of inbound rules with columns for Type, Protocol, Port range, Source, and Description. Two rules are listed: one for 'All traffic' from 'All' sources, and another for 'All traffic' from 'All' sources. The 'Source' column for the second rule is highlighted, showing a dropdown menu with options like 'Custom' and '0.0.0.0/0'. A warning message at the bottom states: 'Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.' The 'Save rules' button is visible at the bottom right.





## 14. Viewing Database Instance Details

### Additional configuration

#### Public access

##### ☒ Publicly accessible

RDS assigns a public IP address to the database. Amazon EC2 instances and other resources outside of the VPC can connect to your database. Resources inside the VPC can also connect to the database. Choose one or more VPC security groups that specify which resources can connect to the database.

##### ☐ Not publicly accessible

No IP address is assigned to the DB instance. EC2 instances and devices outside the VPC can't connect.

#### Database port

Specify the TCP/IP port that the DB instance will use for application connections. The application connection string must specify the port number. The DB security group and your firewall must allow connections to the port. [Learn more](#)

5432

## 15. Copying the RDS Endpoint for Connection

Connectivity & security | Monitoring | Logs & events | Configuration | Zero-ETL integrations | Maintenance & backups | Data migrations | Tags | Recommendations

### Connectivity & security

#### Endpoint & port

**Endpoint**  
[database-1.czalogq22pxe.ap-south-1.rds.amazonaws.com](#)

**Port**  
5432

#### Networking

**Availability Zone**  
ap-south-1a

**VPC**  
[vpc-0db8aa979fd2f20b9](#)

**Subnet group**  
default-vpc-0db8aa979fd2f20b9

**Subnets**  
[subnet-0370a3843d47fb043](#)  
[subnet-023643778bcf98c2f](#)  
[subnet-00cce0a66f1d6e78d](#)

**Network type**  
IPv4

#### Security

**VPC security groups**  
[default \(sg-05c3b0fb496219656\)](#)  
Active

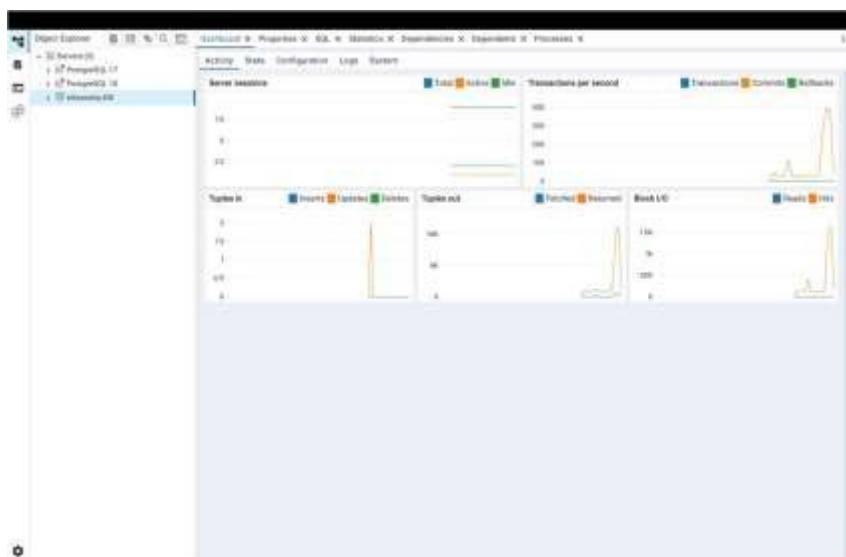
**Publicly accessible**  
No

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

**Certificate authority date**  
May 20, 2061, 00:10 (UTC+05:30)

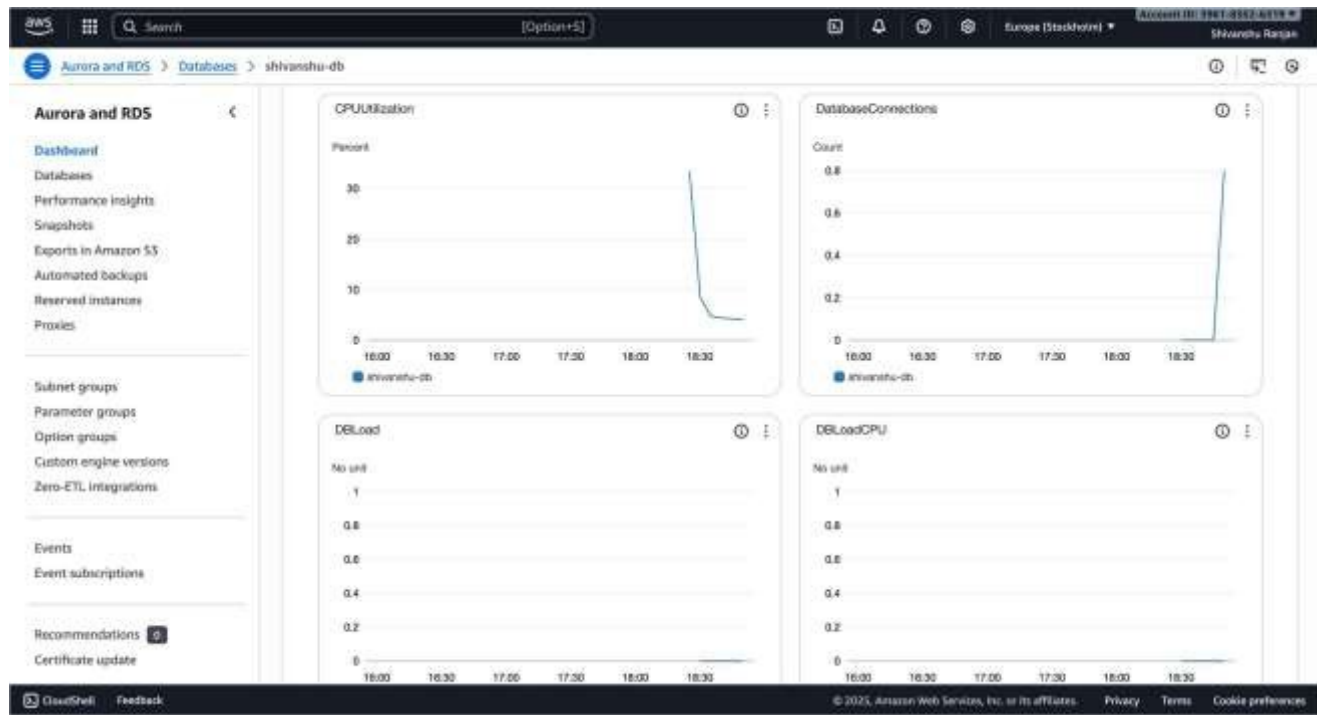
**DB instance certificate expiration date**  
November 04, 2026, 09:43 (UTC+05:30)

## 16. Launching pgAdmin on Local Machine

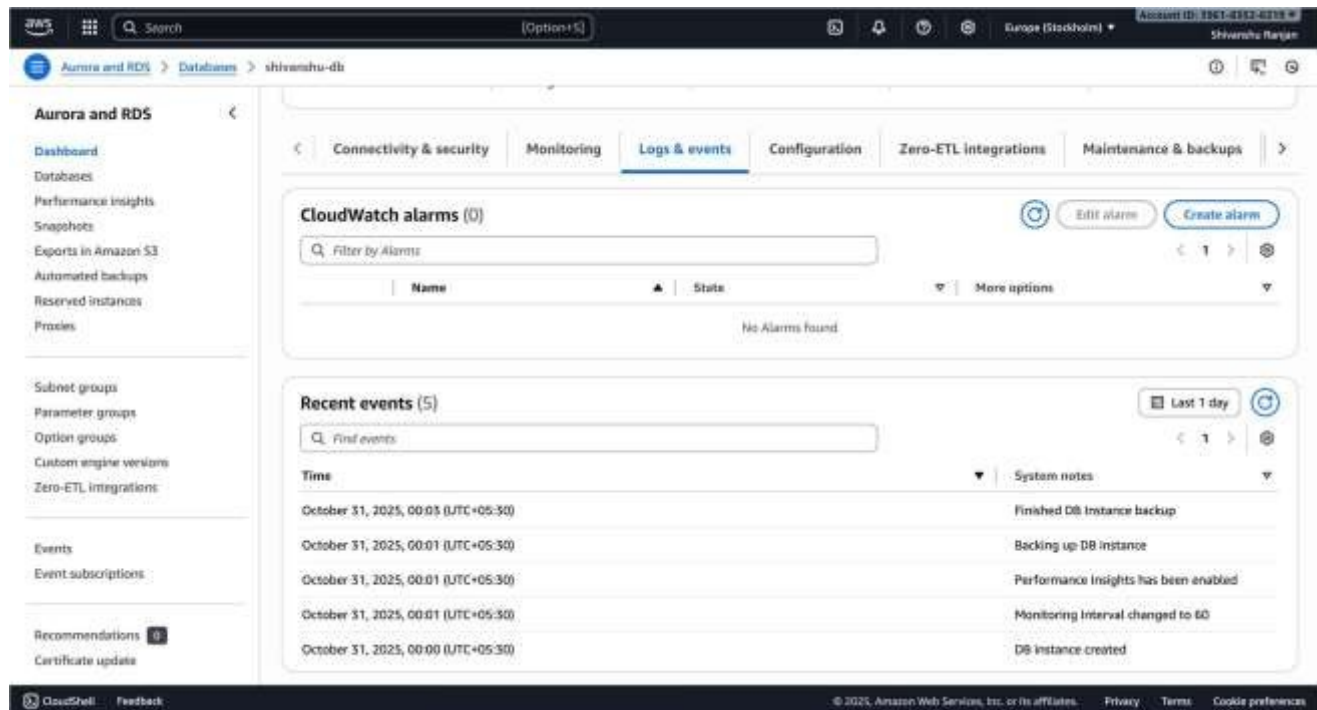




## 17. Adding a New Server in pgAdmin



## 18. Entering Connection Details (Endpoint, Username, Password)





## 19. Successful Connection to AWS RDS Database via pgAdmin

The screenshot displays the AWS Management Console for the 'Aurora and RDS' service. The left sidebar shows the navigation menu with options like 'Dashboard', 'Databases', 'Performance Insights', 'Snapshots', 'Exports in Amazon S3', 'Automated backups', 'Reserved instances', 'Proxies', 'Subnet groups', 'Parameter groups', 'Option groups', 'Custom engine versions', 'Zero-ETL integrations', 'Events', 'Event subscriptions', 'Recommendations', and 'Certificate update'. The main content area shows the 'Databases' page for 'database-1'. The 'Logs' section is expanded, showing a list of 73 logs. The logs are sorted by 'Last written' time. The 'Name' column shows the log file names, including 'error/postgresql.log.2025-11-07-18' through 'error/postgresql.log.2025-11-08-06'. The 'Time' column shows the timestamps, ranging from November 08, 2025, to November 10, 2025. The 'Size' column shows the size of each log file, mostly 4.4 kB. The 'Time' section shows two events: 'Finished DB Instance backup' and 'Backing up DB Instance'.

## 4. Learning Outcomes:

- Understand the fundamental concepts and benefits of using Amazon RDS for relational database management in the cloud.
- Gain practical knowledge of creating and configuring an RDS database instance on AWS.
- Learn how to manage and secure database access using AWS security groups.
- Develop skills to connect a local pgAdmin client to a cloud-hosted RDS instance.
- Be able to monitor, manage, and test database connectivity and performance in a cloud environment.