

WORKSHEET - 9

Name: Sohrit Kumar

UID: 23BCS12492

Branch: CSE

Section/Group: KRG 3-A

Semester: 5th

Date of Performance: 30/10/2025

Subject Name: ADBMS

Subject Code: 23CSP-333

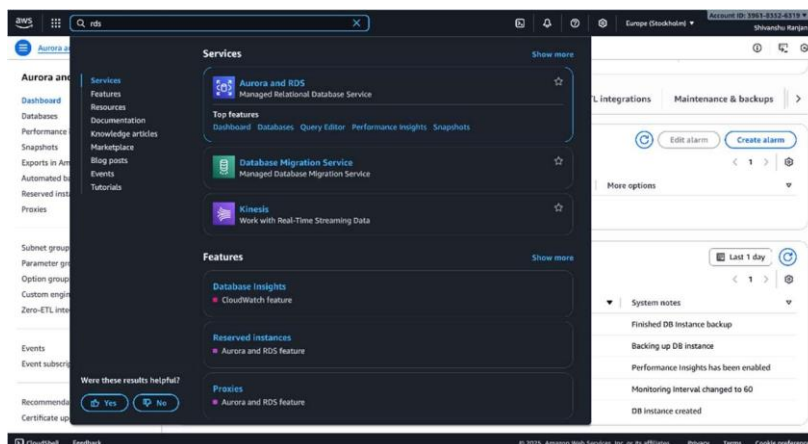
1. Aim: To understand and implement the setup of Amazon Relational Database Service (AWS RDS) by creating a database instance, configuring security groups, and establishing a secure connection between the local pgAdmin tool and the RDS instance hosted on the AWS Cloud.

2. Objective:

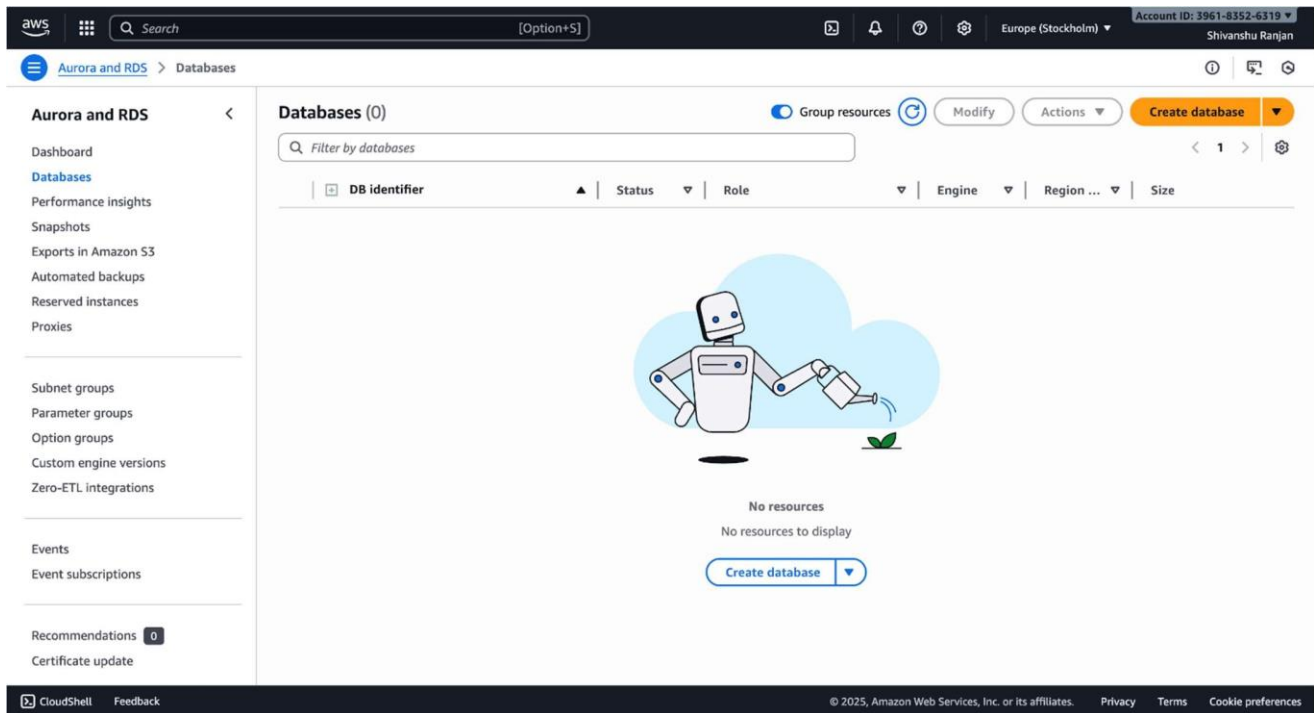
- To learn the basic concepts and features of Amazon Relational Database Service (AWS RDS).
- To create and configure a new RDS database instance on the AWS Management Console.
- To understand the role and configuration of security groups for controlling database access.
- To connect a local pgAdmin 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

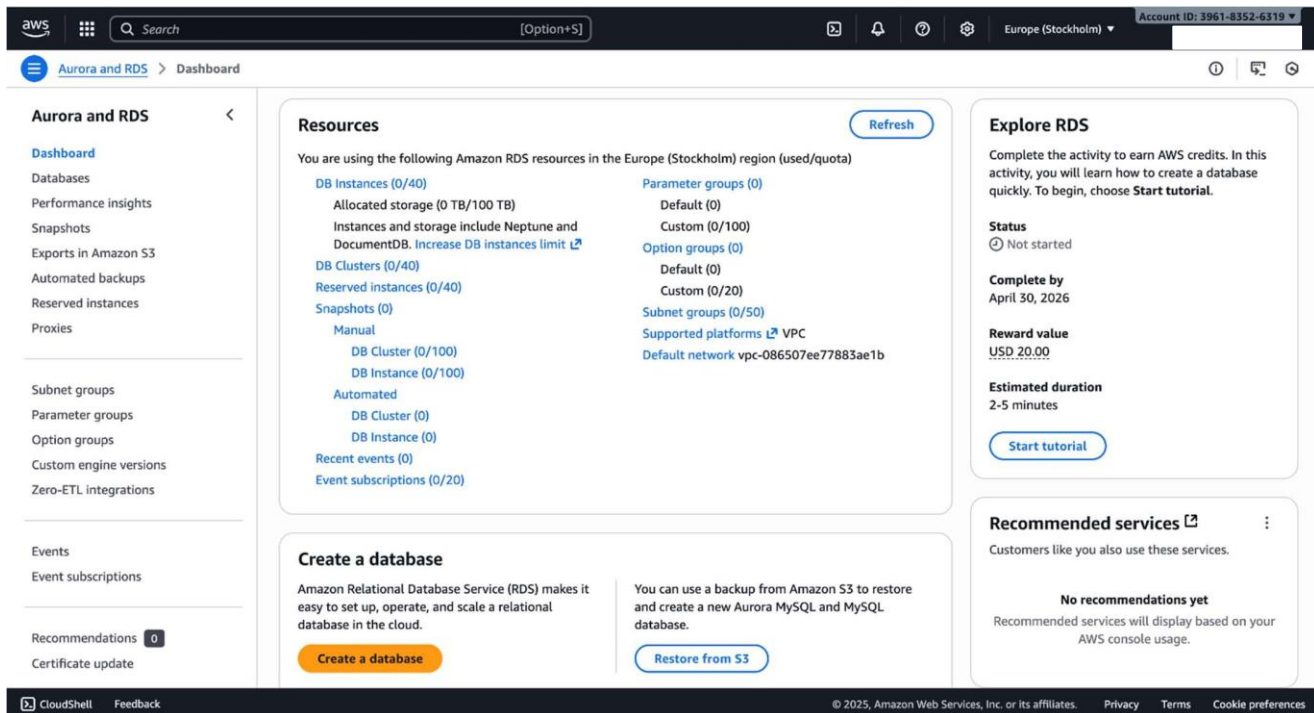


2. Navigating to RDS Service



The screenshot shows the AWS Aurora and RDS console. The left sidebar contains navigation links: 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 (0), and Certificate update. The main content area is titled 'Databases (0)' and features a search bar, a table with columns (DB identifier, Status, Role, Engine, Region, Size), and a large illustration of a robot watering a plant. Below the illustration, it states 'No resources' and 'No resources to display', with a 'Create database' button.

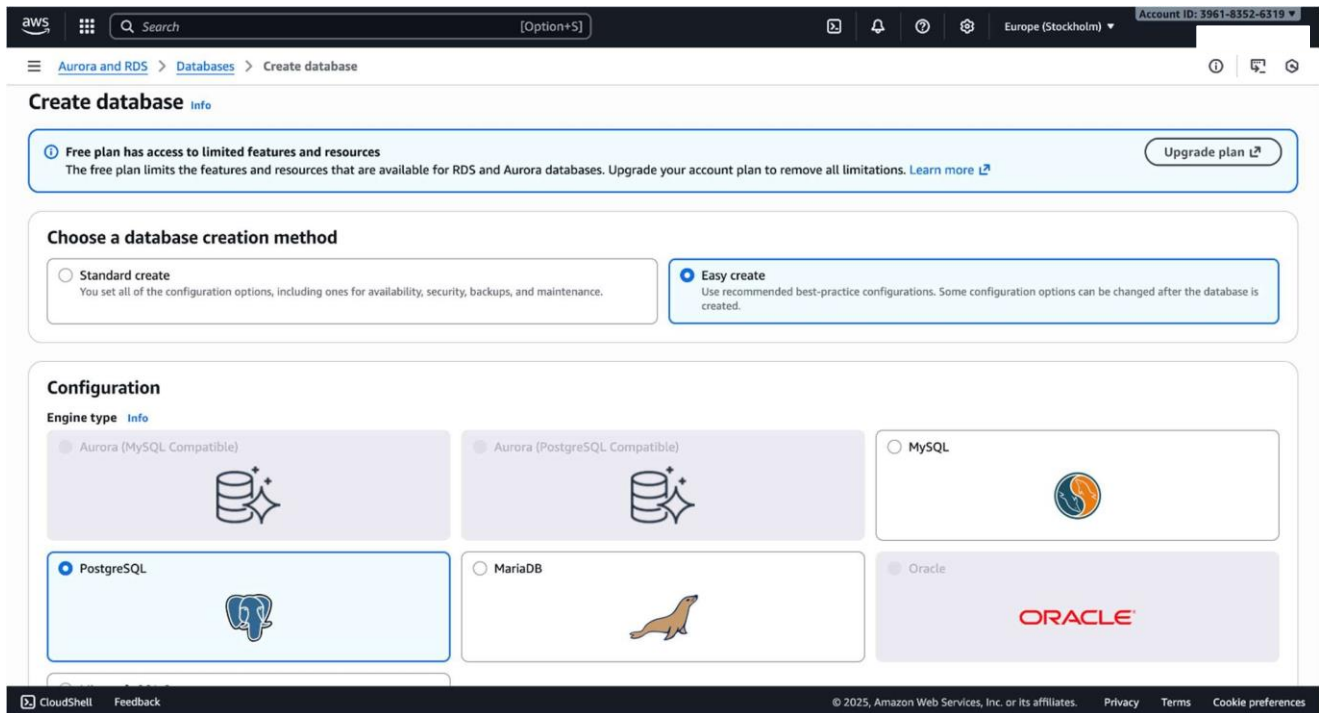
3. Amazon RDS Dashboard Overview



The screenshot shows the AWS Aurora and RDS Dashboard. The left sidebar is identical to the previous screenshot. The main content area is divided into several sections:

- Resources:** A summary of RDS resources in the Europe (Stockholm) region. It includes:
 - DB Instances (0/40): Allocated storage (0 TB/100 TB), Instances and storage include Neptune and DocumentDB. [Increase DB instances limit](#)
 - DB Clusters (0/40): Reserved instances (0/40), Snapshots (0)
 - Manual: DB Cluster (0/100), DB Instance (0/100)
 - Automated: DB Cluster (0), DB Instance (0)
 - Recent events (0), Event subscriptions (0/20)
 - Parameter groups (0): Default (0), Custom (0/100)
 - Option groups (0): Default (0), Custom (0/20)
 - Subnet groups (0/50)
 - Supported platforms [VPC](#): Default network vpc-086507ee77883ae1b
- Explore RDS:** A section for completing the activity to earn AWS credits. It includes a 'Start tutorial' button.
- Recommended services:** A section for recommended services. It states 'No recommendations yet' and 'Recommended services will display based on your AWS console usage.'
- Create a database:** A section for creating a new database. It includes a 'Create a database' button and a 'Restore from S3' button.

4. Creating a New Database Instance



Create database [info](#)

Free plan has access to limited features and resources
The free plan limits the features and resources that are available for RDS and Aurora databases. Upgrade your account plan to remove all limitations. [Learn more](#)

[Upgrade plan](#)

Choose a database creation method

☐ **Standard create**
You set all of the configuration options, including ones for availability, security, backups, and maintenance.

☒ **Easy create**
Use recommended best-practice configurations. Some configuration options can be changed after the database is created.

Configuration

Engine type [info](#)

☐ Aurora (MySQL Compatible)

☐ Aurora (PostgreSQL Compatible)

☐ MySQL

☒ PostgreSQL

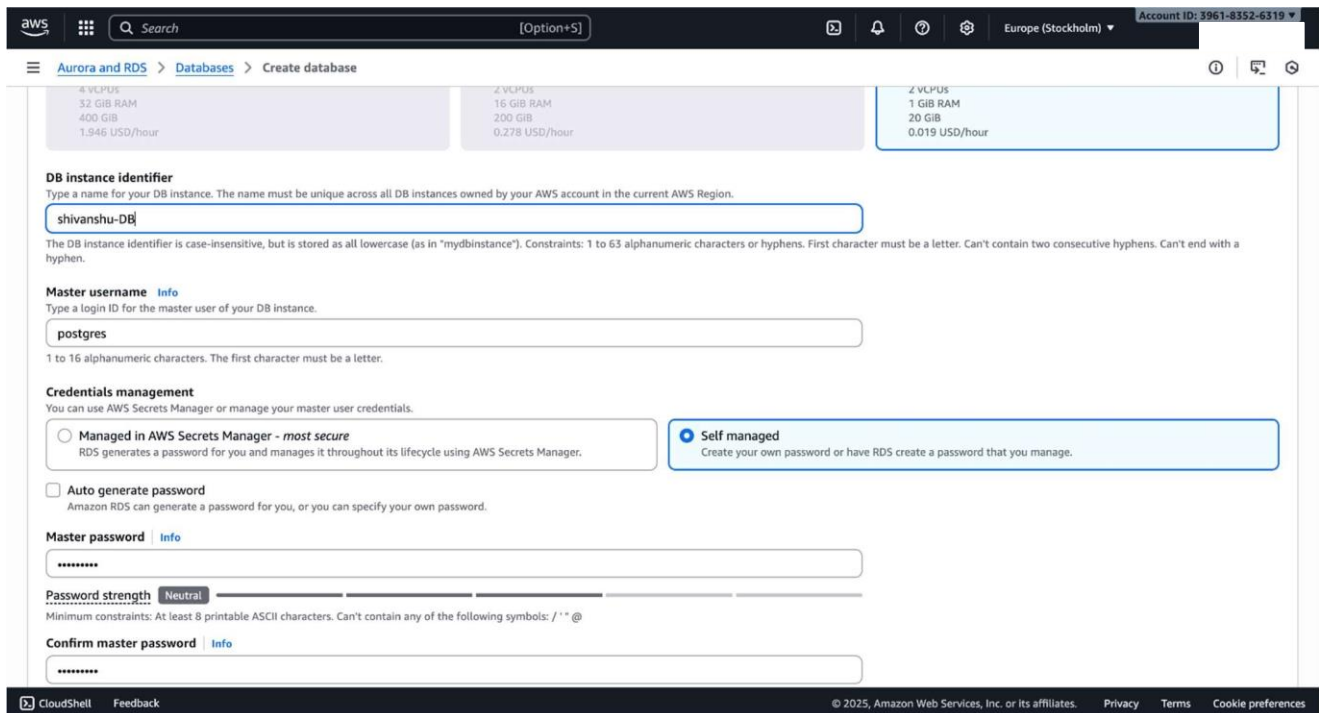
☐ MariaDB

☐ Oracle

CloudShell Feedback

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

5. Selecting PostgreSQL as Database Engine



4 VCPUS
32 GiB RAM
400 GiB
1.946 USD/hour

2 VCPUS
16 GiB RAM
200 GiB
0.278 USD/hour

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.

shivanshu-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.

☐ **Managed in AWS Secrets Manager - most secure**
RDS generates a password for you and manages it throughout its lifecycle using AWS Secrets Manager.

☒ **Self managed**
Create your own password or have RDS create a password that you manage.

☐ **Auto generate password**
Amazon RDS can generate a password for you, or you can specify your own password.

Master password [info](#)

Password strength [Neutral](#)

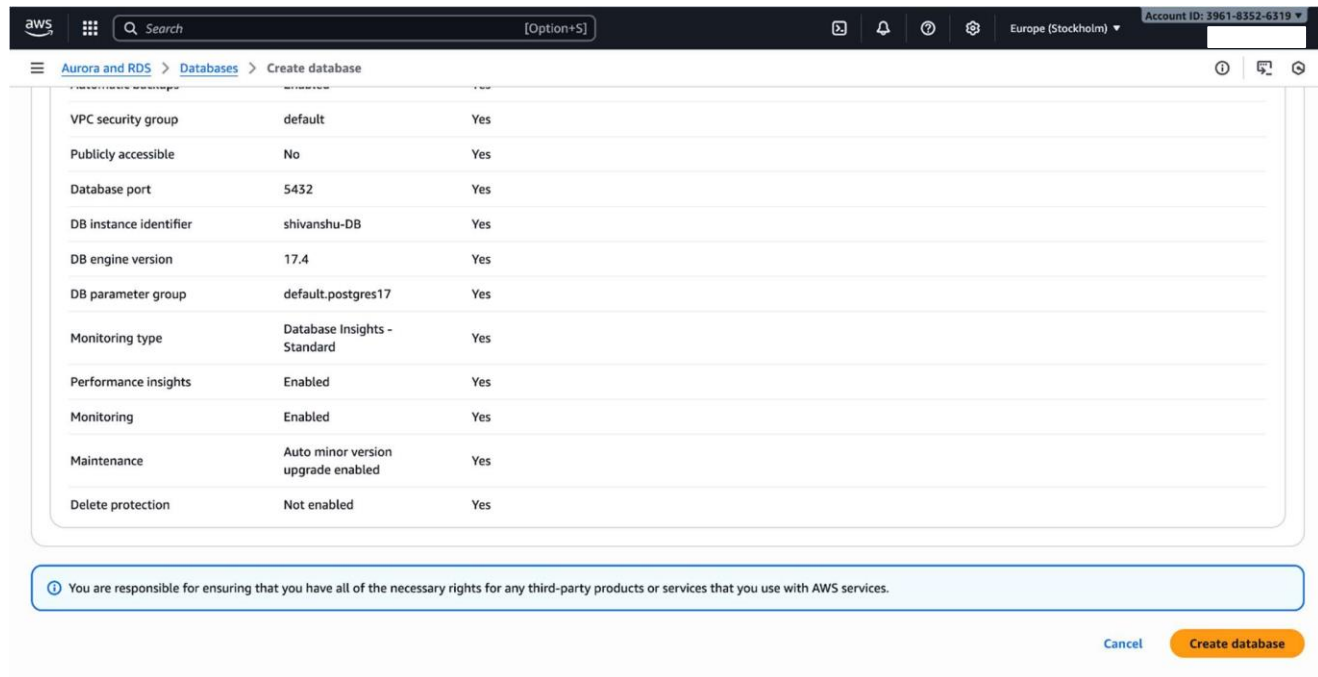
Minimum constraints: At least 8 printable ASCII characters. Can't contain any of the following symbols: / * @

Confirm master password [info](#)

CloudShell Feedback

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

6. Choosing Deployment Option and Template

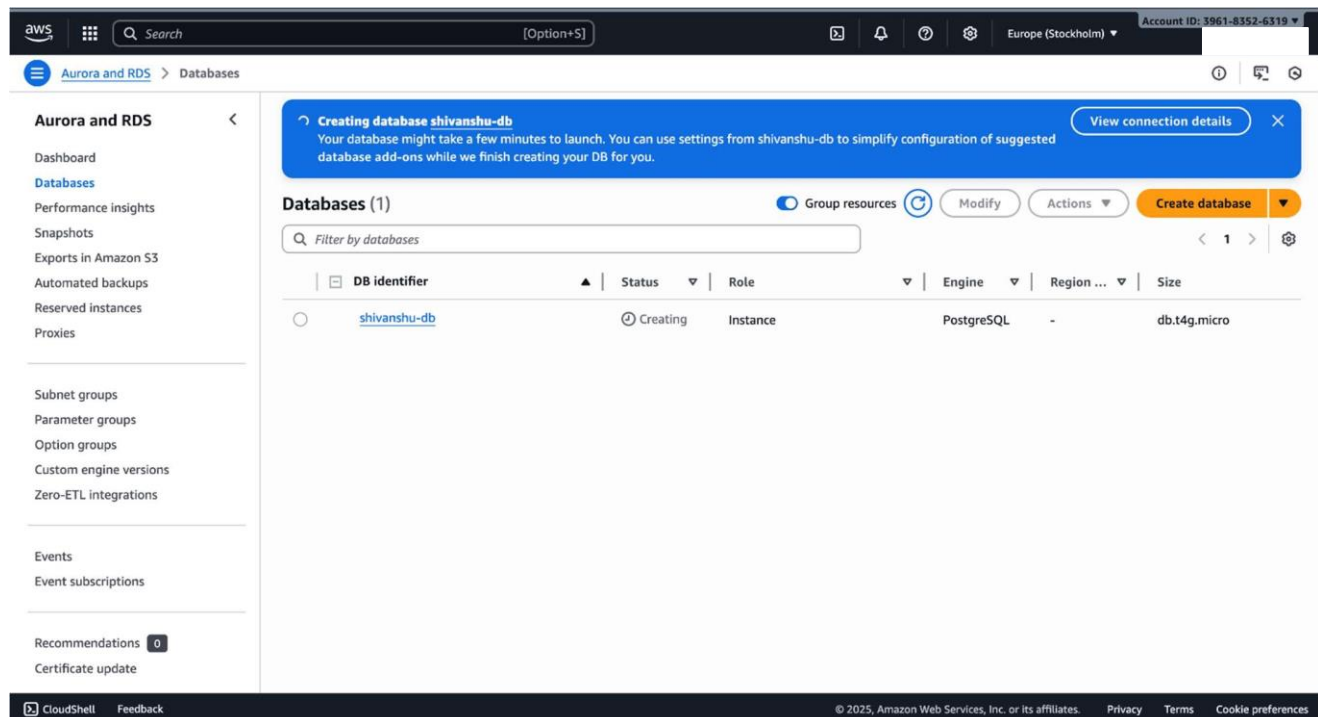


Configuration Option	Value	Required
VPC security group	default	Yes
Publicly accessible	No	Yes
Database port	5432	Yes
DB instance identifier	shivanshu-DB	Yes
DB engine version	17.4	Yes
DB parameter group	default.postgres17	Yes
Monitoring type	Database Insights - Standard	Yes
Performance insights	Enabled	Yes
Monitoring	Enabled	Yes
Maintenance	Auto minor version upgrade enabled	Yes
Delete protection	Not enabled	Yes

You are responsible for ensuring that you have all of the necessary rights for any third-party products or services that you use with AWS services.

[Cancel](#) [Create database](#)

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



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

[View connection details](#)

Databases (1) [Group resources](#) [Modify](#) [Actions](#) [Create database](#)

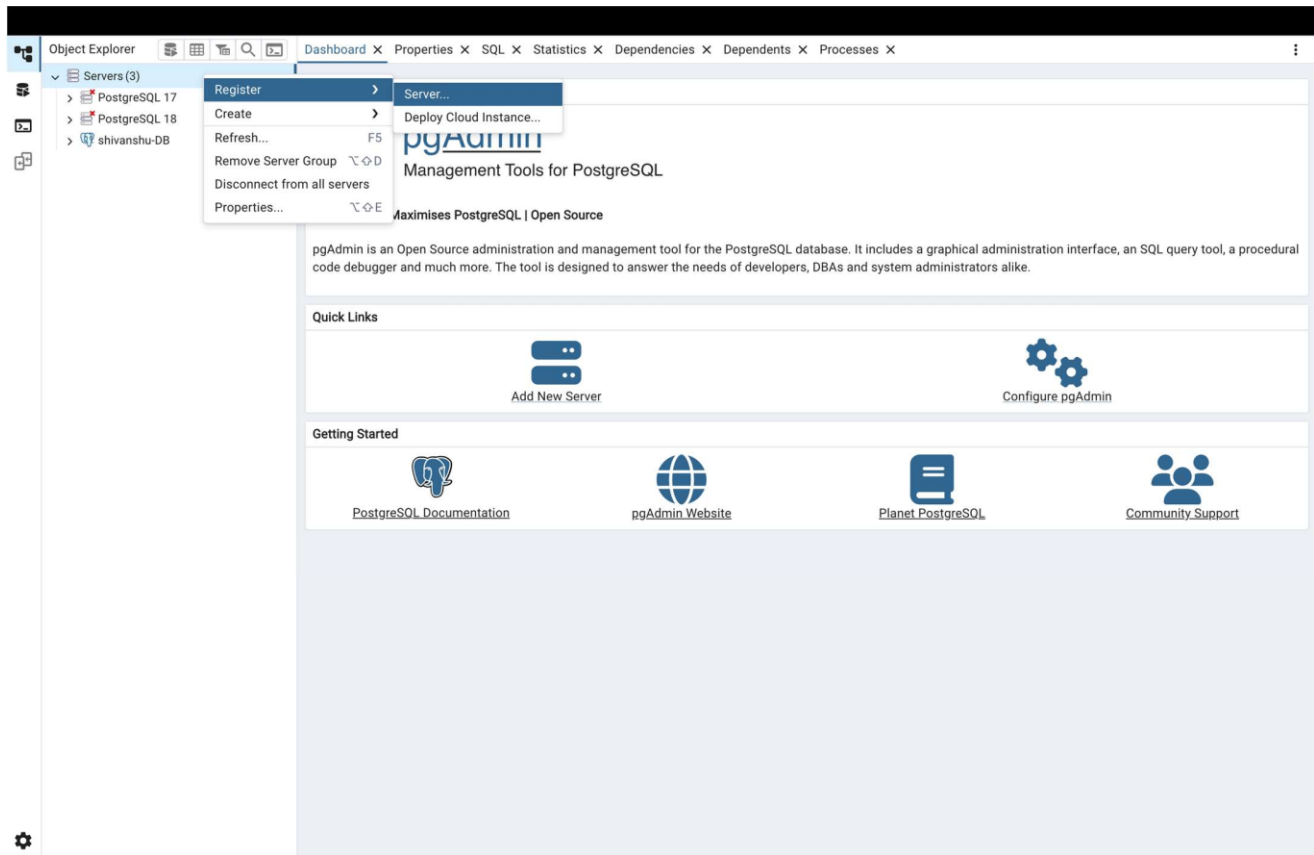
Filter by databases

DB identifier	Status	Role	Engine	Region	Size
shivanshu-db	Creating	Instance	PostgreSQL	-	db.t4g.micro

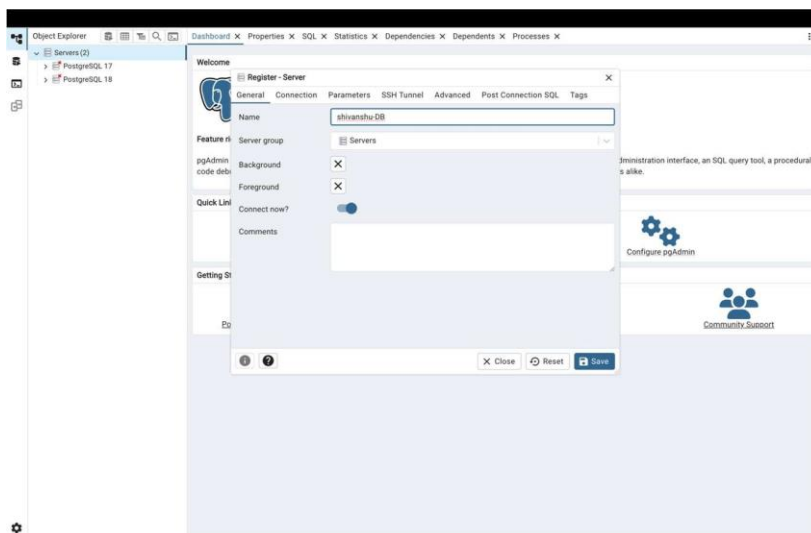
CloudShell Feedback

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

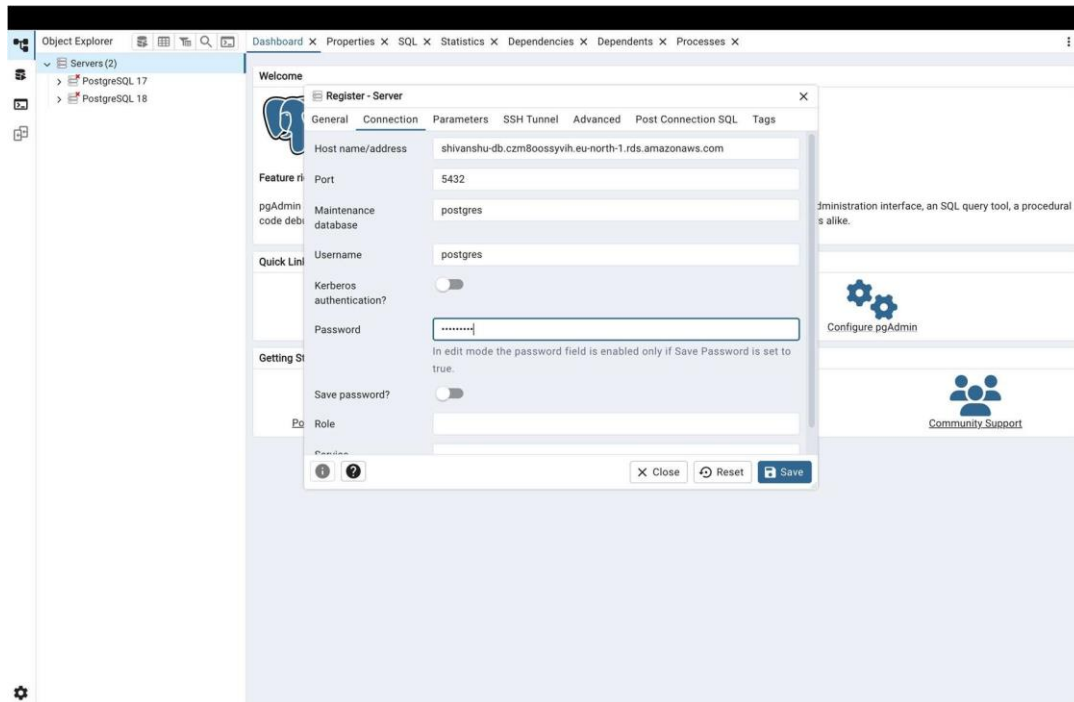
8. Setting Up Instance Size and Storage



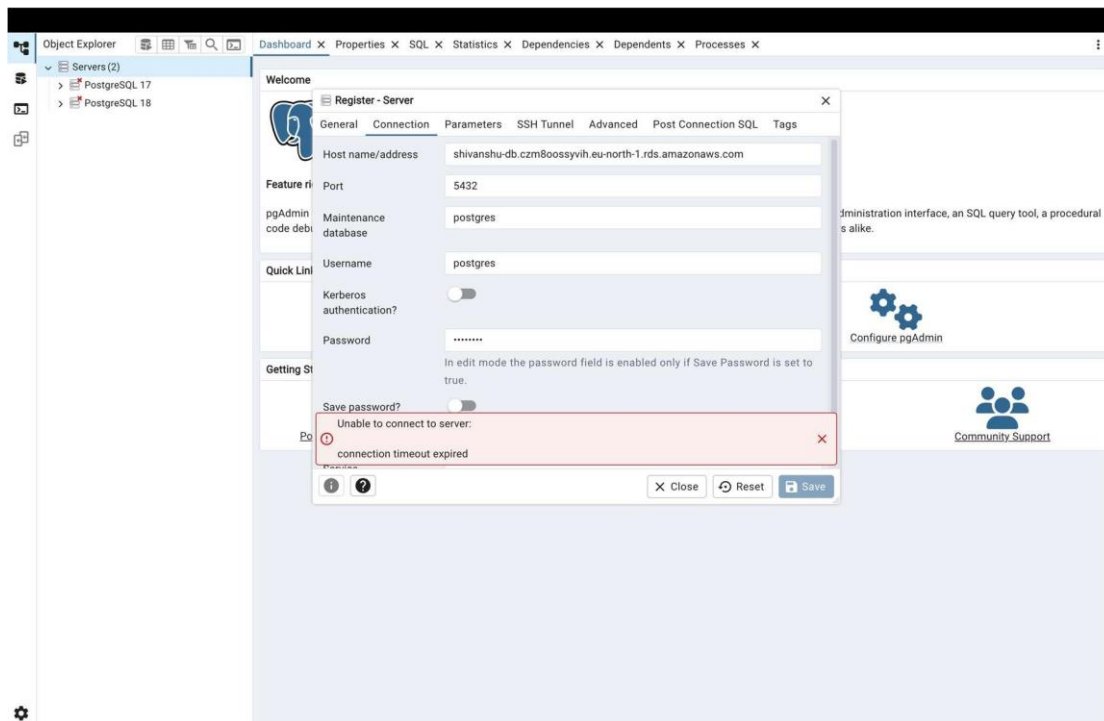
9. Configuring Connectivity and VPC Settings



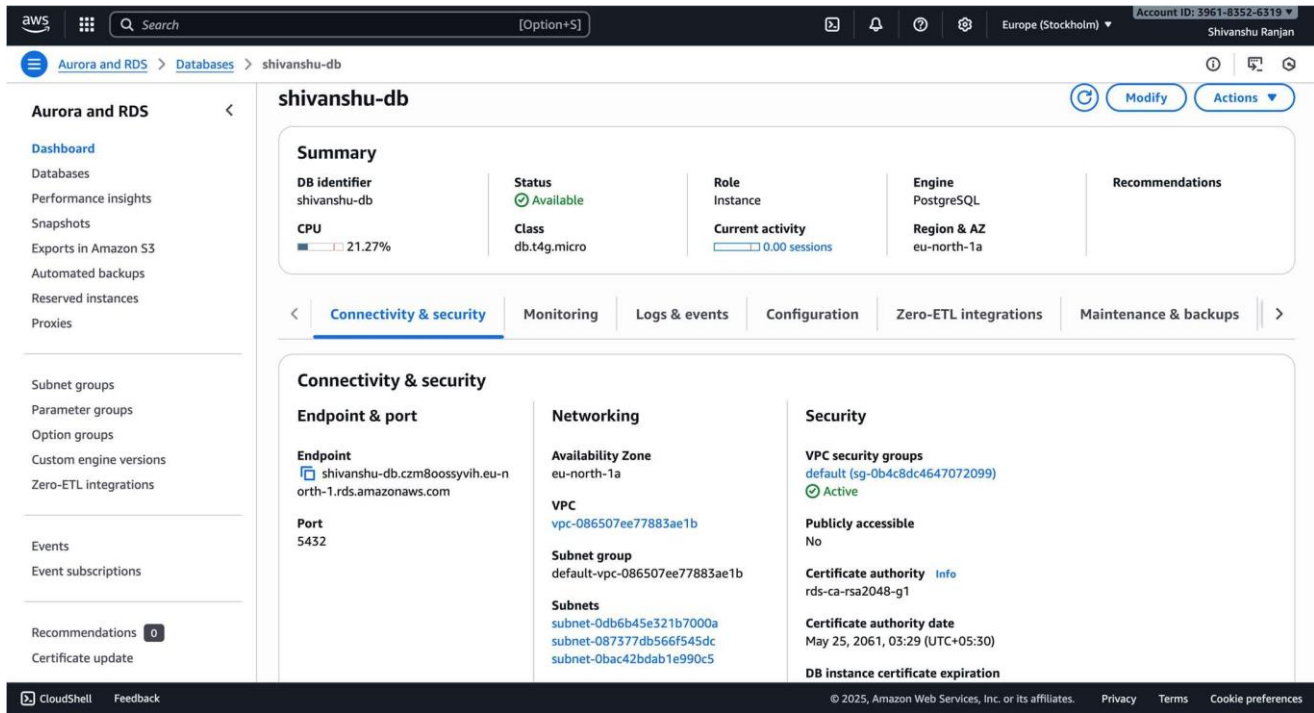
10. Gr Setting Up Security Groups for RDS Access



11. Additional Database Configuration Options

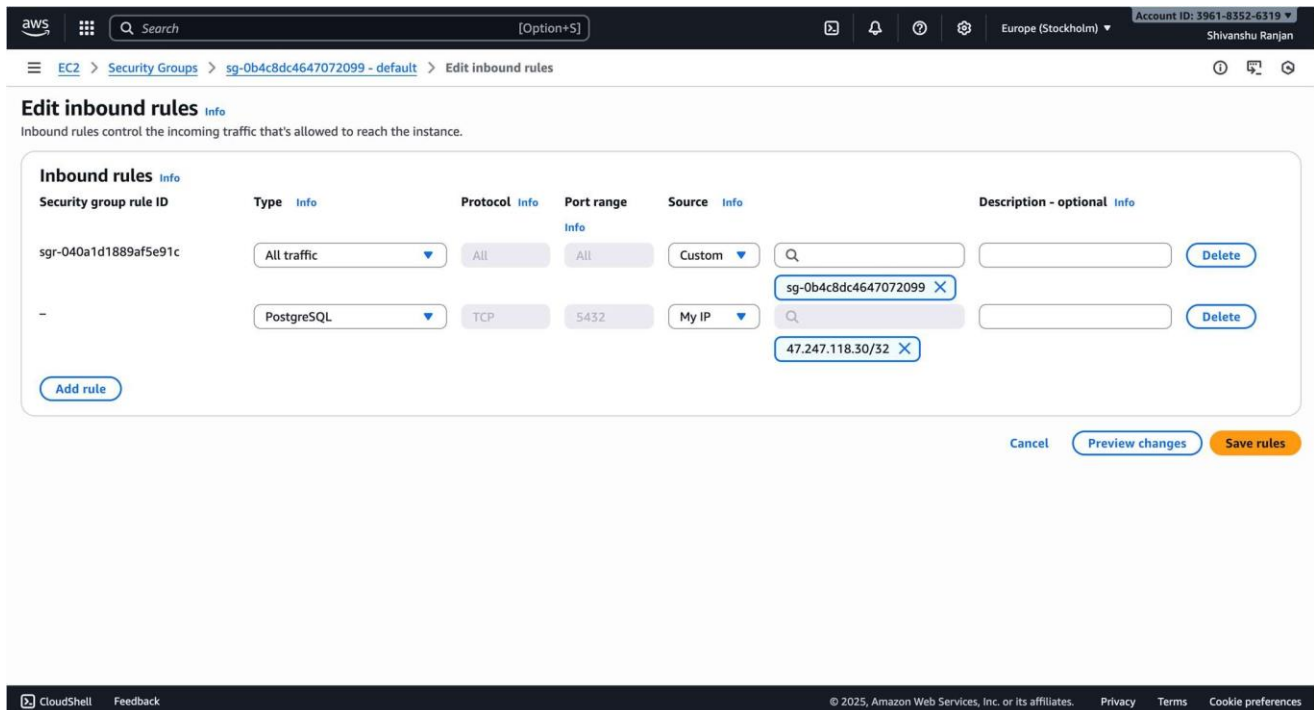


12. Reviewing and Creating the Database Instance



The screenshot shows the AWS Management Console for an Amazon RDS instance named 'shivanshu-db'. The instance is in the 'Available' state. The summary shows the DB identifier, CPU usage (21.27%), Status (Available), Role (Instance), Engine (PostgreSQL), Region & AZ (eu-north-1a), and Recommendations. The 'Connectivity & security' tab is selected, showing the Endpoint & port, Networking (Availability Zone, VPC, Subnet group, Subnets), and Security (VPC security groups, Publicly accessible, Certificate authority, Certificate authority date, DB instance certificate expiration).

13. RDS Instance Creation in Progress



The screenshot shows the AWS Management Console for the 'Edit inbound rules' page of a Security Group. The page displays a table of inbound rules with columns: Security group rule ID, Type, Protocol, Port range, Source, and Description - optional. There are two rules listed: one for 'All traffic' and one for 'PostgreSQL'. The 'PostgreSQL' rule is selected, and its details are shown in the right-hand pane, including the Source (My IP) and the Port range (5432). The 'Add rule' button is visible at the bottom left, and the 'Preview changes' and 'Save rules' buttons are 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

Endpoint & port

Endpoint

 shivanshu-db.czm8oossyviu.eu-north-1.rds.amazonaws.com

Port

5432

Networking

Availability Zone

eu-north-1a

VPC

[vpc-086507ee77883ae1b](#)

Subnet group

default-vpc-086507ee77883ae1b

Subnets

[subnet-0db6b45e321b7000a](#)

[subnet-087377db566f545dc](#)

[subnet-0bac42bdab1e990c5](#)

Network type

IPv4

Security

VPC security groups

[default \(sg-0b4c8dc4647072099\)](#)

 Active

Publicly accessible

Yes

Certificate authority [Info](#)

rds-ca-rsa2048-g1

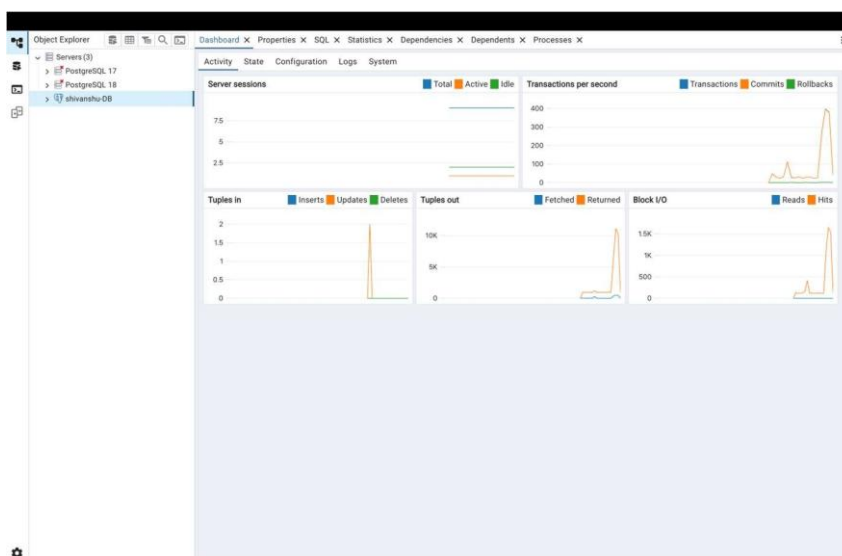
Certificate authority date

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

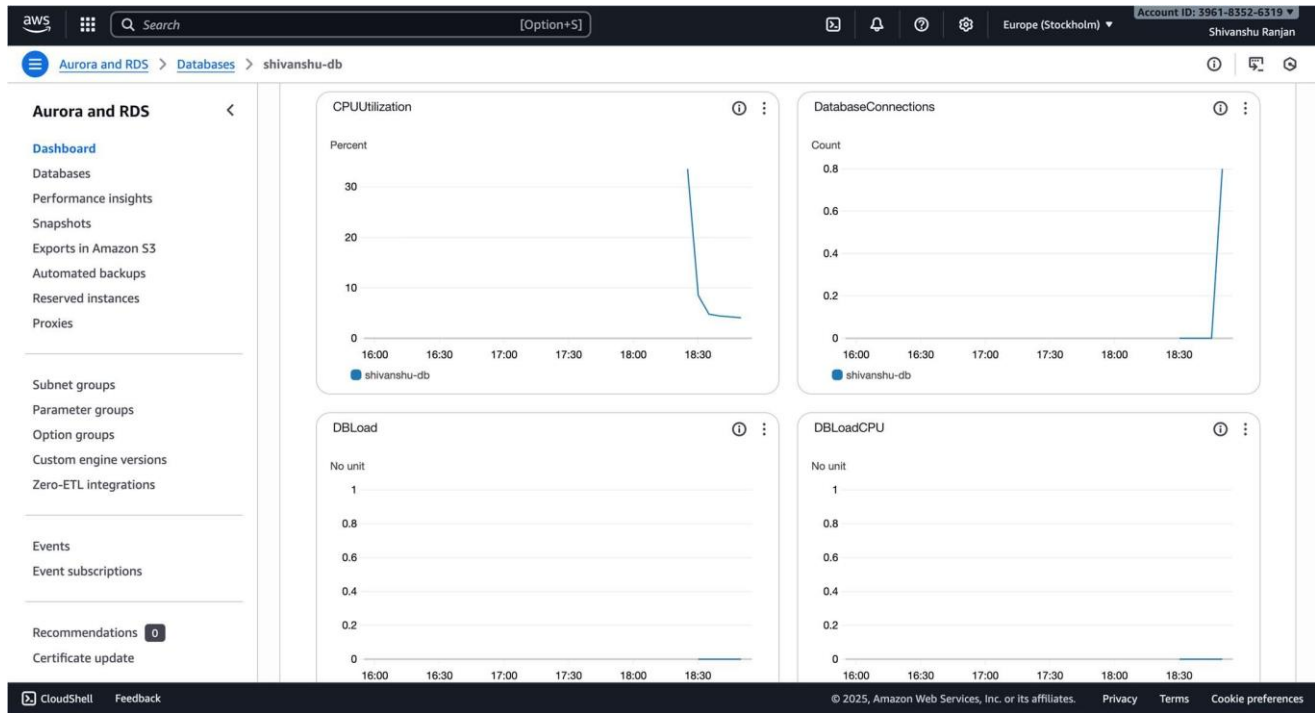
DB instance certificate expiration date

October 30, 2026, 23:59 (UTC+05:30)

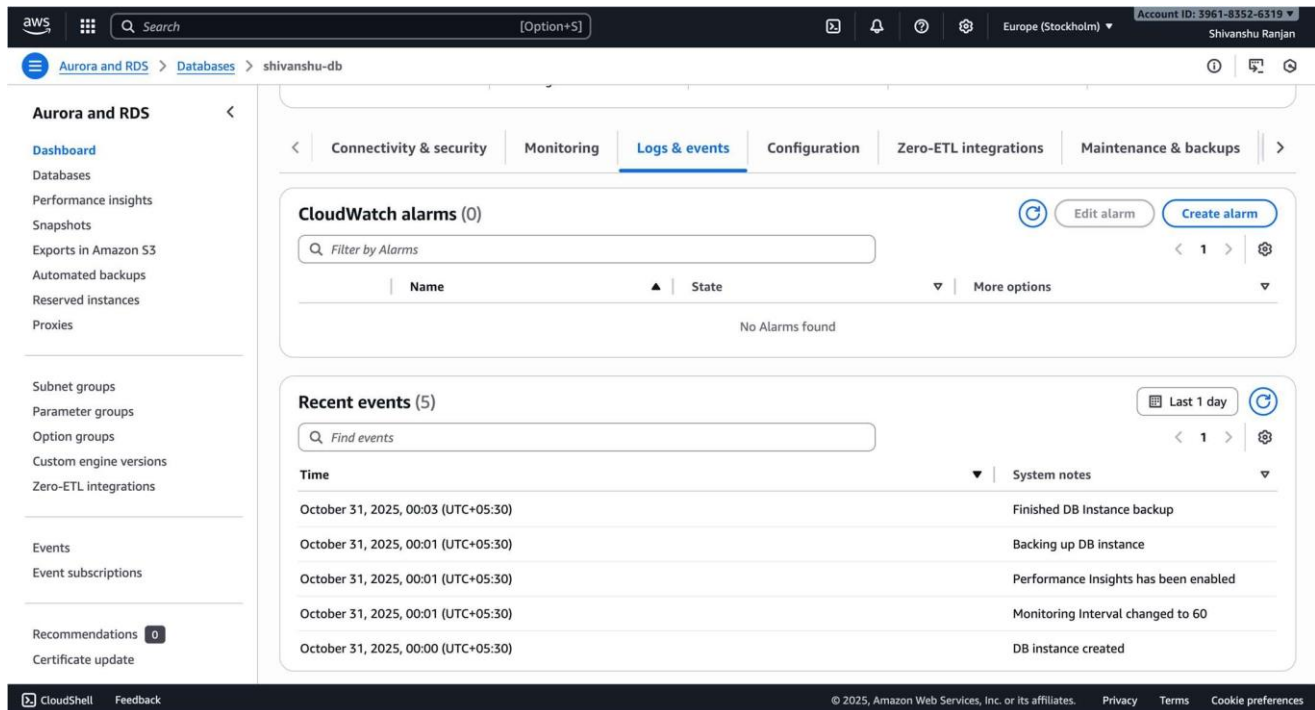
16. Launching pgAdmin on Local Machine



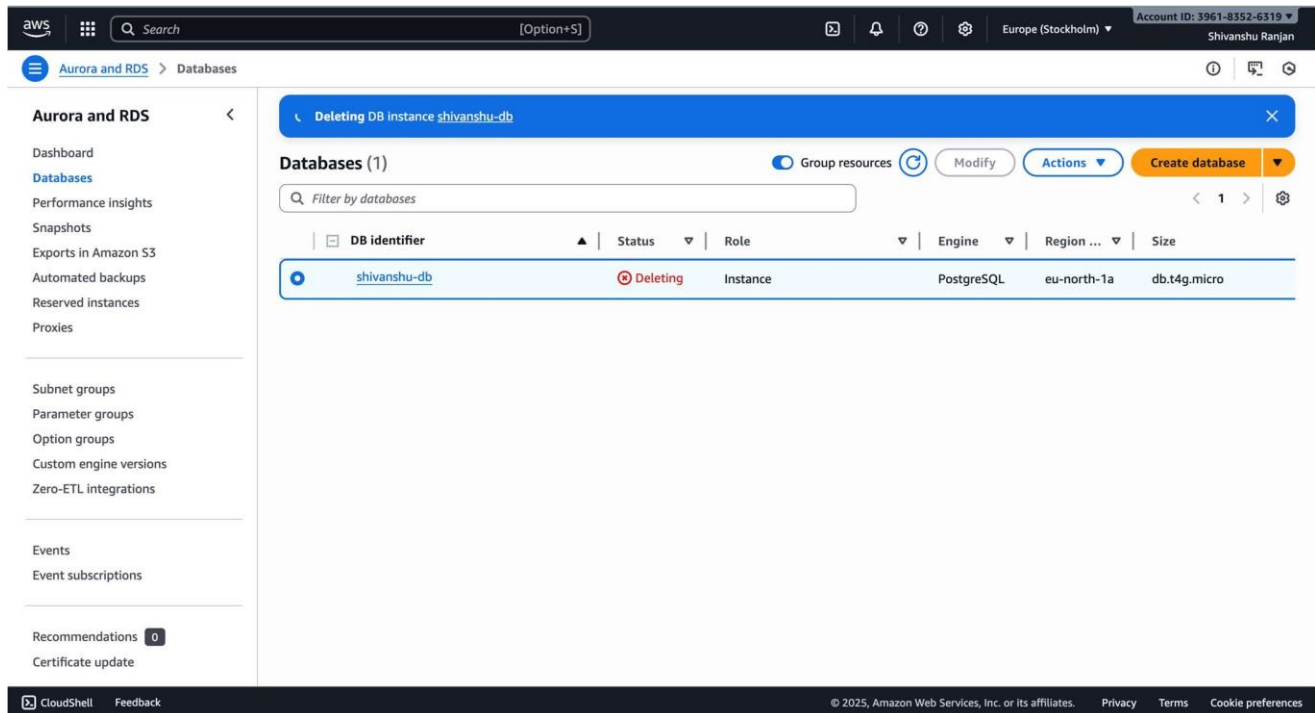
17. Adding a New Server in pgAdmin



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



19. Successful Connection to AWS RDS Database via pgAdmin



The screenshot displays the AWS Management Console interface. At the top, the AWS logo and a search bar are visible. The navigation pane on the left shows the 'Aurora and RDS' section expanded, with 'Databases' selected. The main content area shows a blue banner at the top indicating 'Deleting DB instance shivanshu-db'. Below this, the 'Databases (1)' section is visible, with a search filter 'Filter by databases'. A table lists the database instance:

DB identifier	Status	Role	Engine	Region	Size
shivanshu-db	Deleting	Instance	PostgreSQL	eu-north-1a	db.t4g.micro

The bottom of the console shows the 'CloudShell' button and 'Feedback' link. The footer contains the copyright notice '© 2025, Amazon Web Services, Inc. or its affiliates.' and links for 'Privacy', 'Terms', and 'Cookie preferences'.

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.