



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Worksheet 9

Student Name: Akshay Guleria

Branch: CSE

Semester: 5th

Subject Name: ADBMS

UID: 23BCS10517

Section/Group: KRG 3-A

Date of Performance: 30/10/2025

Subject Code: 23CSP-333

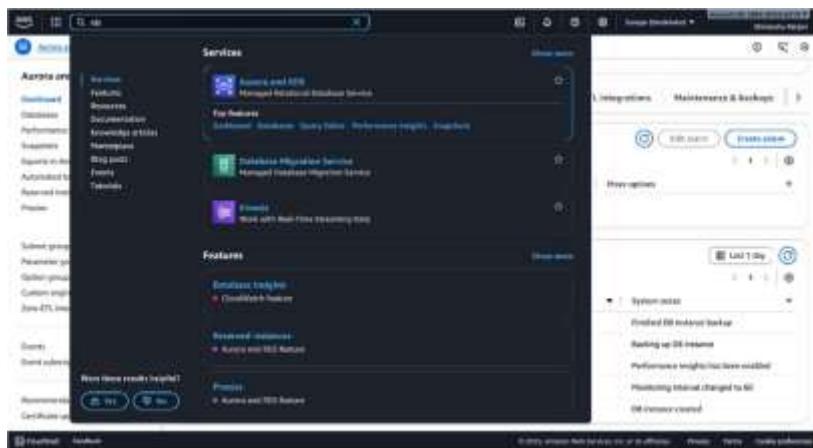
1. Aim: To understand and implement the setup of Amazon Relational Database Service (AWS RDS) by creating a database instance, configuring security groups, and establishing a secure connection between the local pgAdmin tool and the RDS instance hosted on the AWS Cloud.

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





DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

2. Navigating to RDS Service

The screenshot shows the AWS Aurora and RDS service dashboard. The left sidebar includes links for 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 displays a search bar, a 'Databases (0)' section with a 'Create database' button, and a central illustration of a robot standing next to a cloud. Below the illustration, it says 'No resources' and 'No resources to display'. At the bottom, there are links for Classified, Feedback, © 2025 Amazon Web Services, Inc. or its affiliates, Privacy, Terms, and Cookie preferences.

3. Amazon RDS Dashboard Overview

The screenshot shows the Amazon RDS Dashboard. The left sidebar includes links for 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 divided into sections: 'Resources' (listing 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), and Event subscriptions (0/20)), 'Explore RDS' (activity status, Start tutorial, Status Not started, Complete by April 30, 2026, Reward value USD 20.00, Estimated duration 2-5 minutes, Start tutorial), 'Create a database' (description of using a backup from Amazon S3 to restore and create a new Aurora MySQL and MySQL database, Create a database and Restore from S3 buttons), and 'Recommended services' (No recommendations yet). The footer includes links for Classified, Feedback, © 2025 Amazon Web Services, Inc. or its affiliates, Privacy, Terms, and Cookie preferences.



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

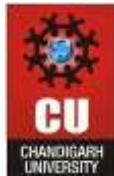
Discover. Learn. Empower.

4. Creating a New Database Instance

The screenshot shows the 'Create database' page in the AWS RDS console. At the top, there's a note about the free plan having limited features and an option to 'Upgrade plan'. Below that, under 'Choose a database creation method', the 'Standard create' option is selected, with a note that it provides full configuration options. The 'Easy create' option is also shown, which uses recommended best-practice configurations. In the 'Configuration' section, 'Engine type' is set to 'PostgreSQL'. Other options like Aurora (MySQL-Compatible), MySQL, MariaDB, Oracle, and Amazon Neptune are also listed with their respective icons. At the bottom, there are links for CloudShell, Feedback, and various AWS links.

5. Selecting PostgreSQL as Database Engine

This screenshot continues from the previous one, showing the 'Create database' page. The 'Engine type' dropdown is now explicitly set to 'PostgreSQL'. The 'DB instance identifier' field contains 'shivanshu-DB', which is highlighted in blue. The 'Master username' field is set to 'postgres'. Under 'Credentials management', the 'Self managed' option is selected, indicating that the user will create their own password. The 'Master password' field is filled with a long string of characters. The 'Password strength' bar is at 'Medium'. At the bottom, there are links for CloudShell, Feedback, and various AWS links.



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

6. Choosing Deployment Option and Template

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

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

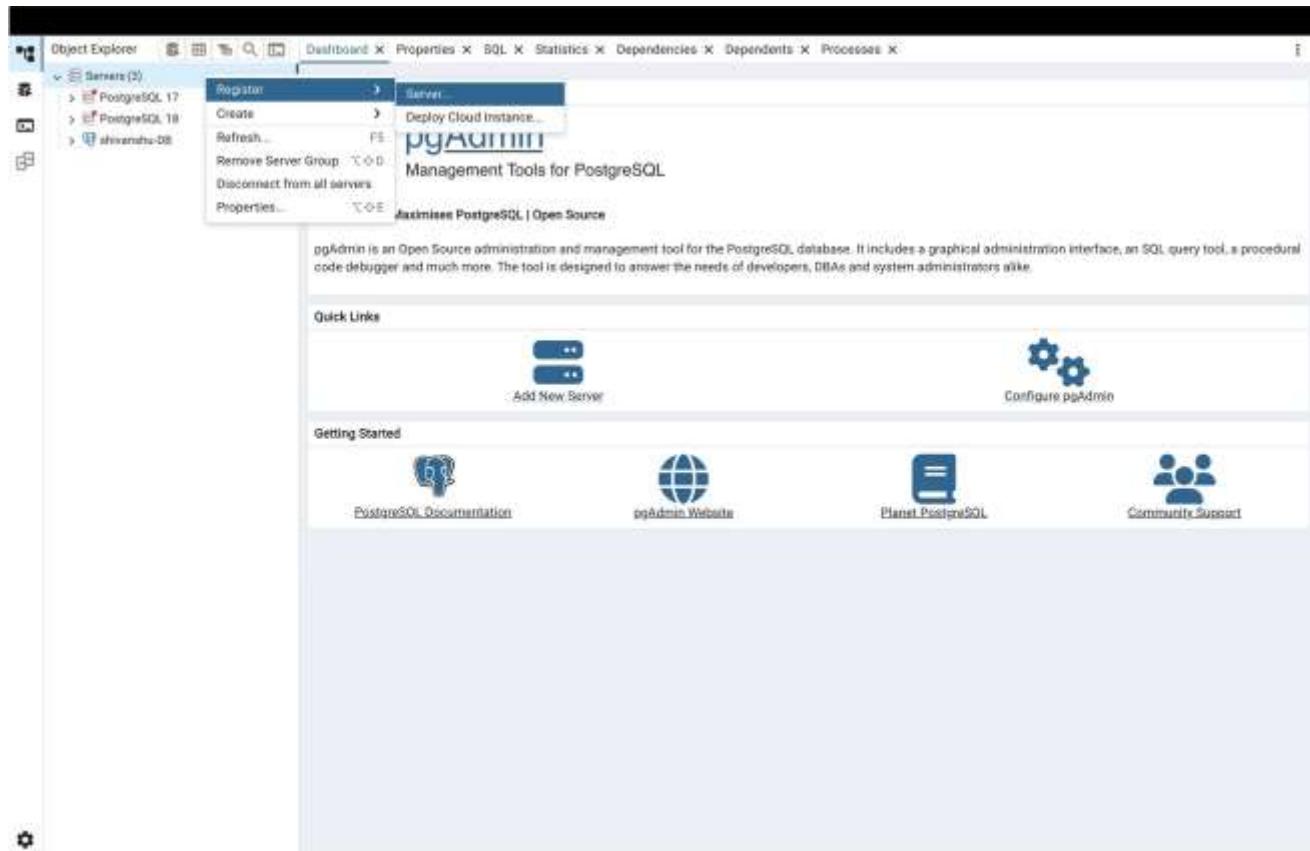
DB identifier	Status	Role	Engine	Region...	Size
shivanshu-db	Creating	Instance	PostgreSQL	Europe (Stockholm)	db.t4g.micro



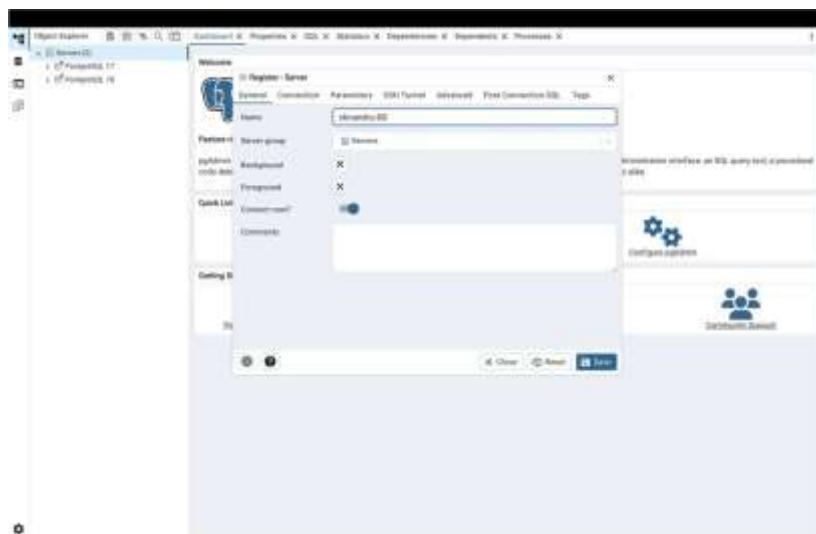
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

8. Setting Up Instance Size and Storage



9. Configuring Connectivity and VPC Settings

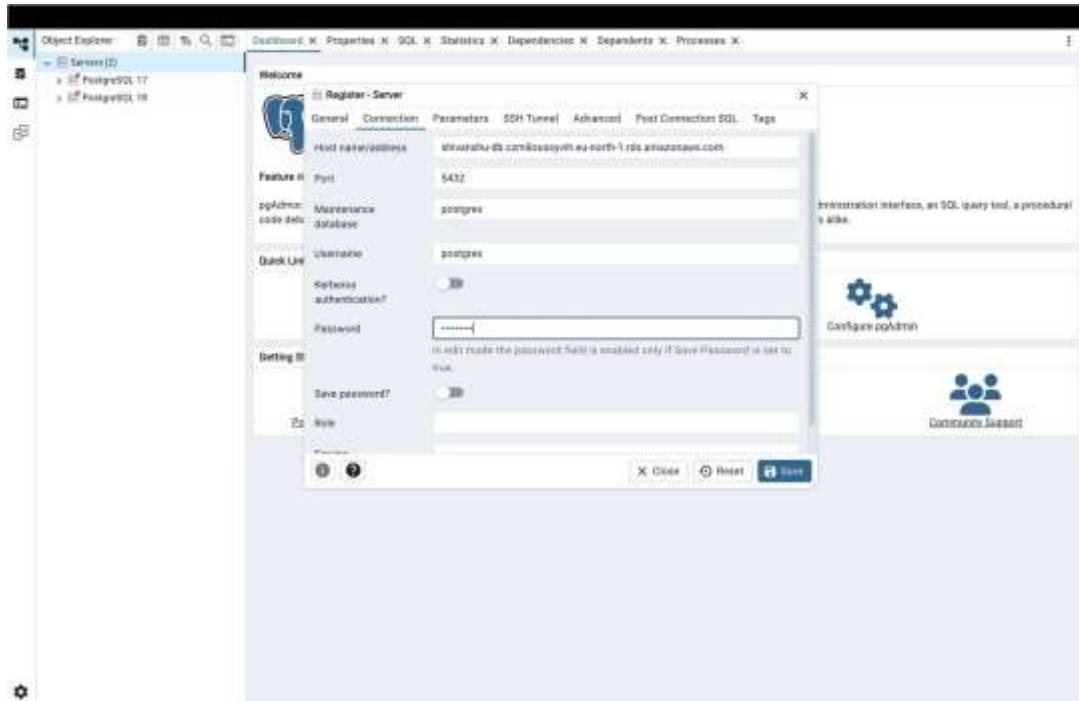




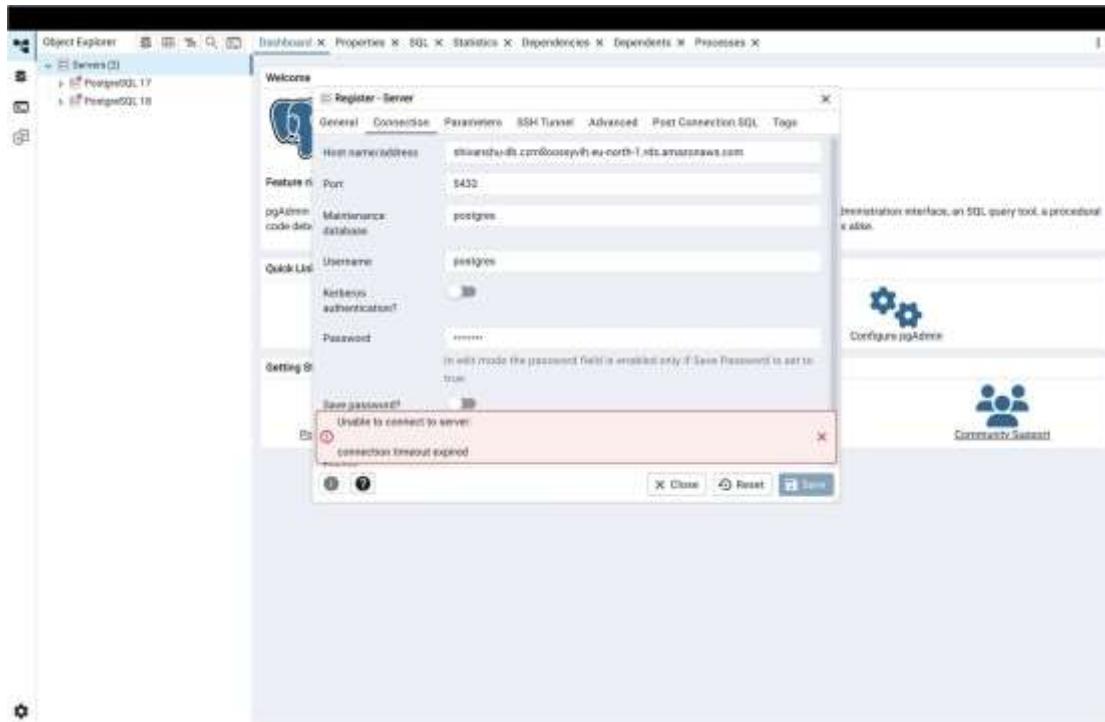
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

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 shows the AWS RDS console for the 'shivanshu-db' database instance. The left sidebar includes links for Aurora and RDS, 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 summary panel displays the DB identifier (shivanshu-db), Status (Available), Role (Instance), Engine (PostgreSQL), and Region & AZ (eu-north-1a). Below the summary, tabs for Connectivity & security, Monitoring, Logs & events, Configuration, Zero-ETL integrations, and Maintenance & backups are visible. The Connectivity & security tab is selected, showing details like Endpoint (shivanshu-db.cmboosyyvh.eu-north-1.rds.amazonaws.com), Port (5432), VPC (vpc-086507ee77883ae1b), Availability Zone (eu-north-1a), Subnet group (default-vpc-086507ee77883ae1b), and Subnets (subnet-0ff66b45a321b7000a, subnet-087377d566f545dc, subnet-0bac42bdab1e990c). Security details include VPC security groups (default sg-0b4c8dc4647072099) and Publicly accessible (No). Certificate authority (rds-ca-rs42048-g1) and certificate authority date (May 25, 2061, 03:29 (UTC+05:30)) are also listed. The DB instance certificate expiration is shown as May 25, 2061, 03:29 (UTC+05:30).

13. RDS Instance Creation in Progress

The screenshot shows the AWS EC2 Security Groups console for the 'sg-0b4c8dc4647072099 - default' security group. The left sidebar includes links for EC2, Security Groups, and Edit inbound rules. The main page displays the 'Edit inbound rules' section, which lists an existing rule for All traffic (Protocol: TCP, Port range: 5432, Source: My IP) and a new rule for PostgreSQL (Protocol: TCP, Port range: 5432, Source: 47.247.118.50/32). Buttons for Add rule, Preview changes, and Save rules are at the bottom. The bottom of the screen shows standard AWS navigation links for CloudShell, Feedback, Privacy, Terms, and Cookie preferences.



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

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](#)

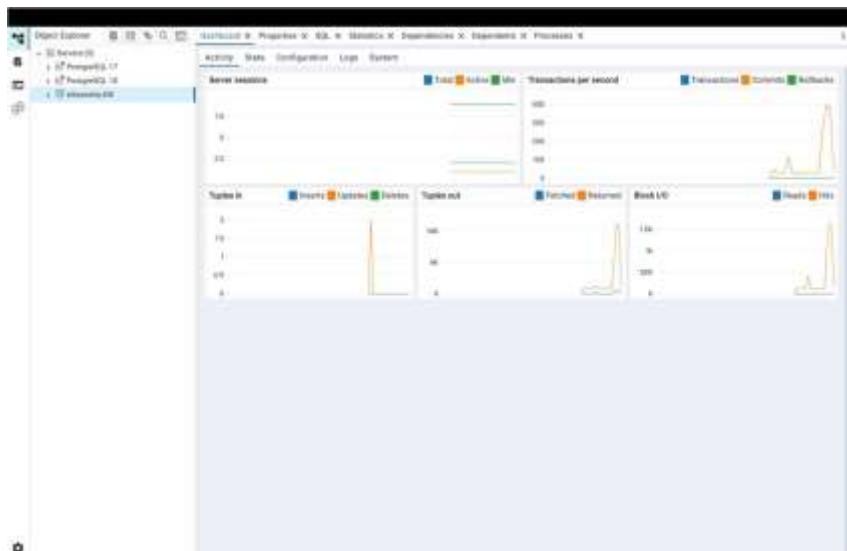
5432

15. Copying the RDS Endpoint for Connection

Connectivity & security

Endpoint & port	Networking	Security
Endpoint shivanshu-db.czr8oossyvih.eu-north-1.rds.amazonaws.com	Availability Zone eu-north-1a	VPC security groups default (sg-0b4c8dc4647072099) <input checked="" type="checkbox"/> Active
Port 5432	VPC vpc-086507ee77883ae1b	Publicly accessible Yes
	Subnet group default-vpc-086507ee77883ae1b	Certificate authority Info rds-ca-rsa2048-g1
	Subnets subnet-0db6b45e321b7000a subnet-087377db566f545dc subnet-0bac42bdab1e990c5	Certificate authority date May 25, 2061, 03:29 (UTC+05:30)
	Network type IPv4	DB instance certificate expiration date October 30, 2026, 23:59 (UTC+05:30)

16. Launching pgAdmin on Local Machine

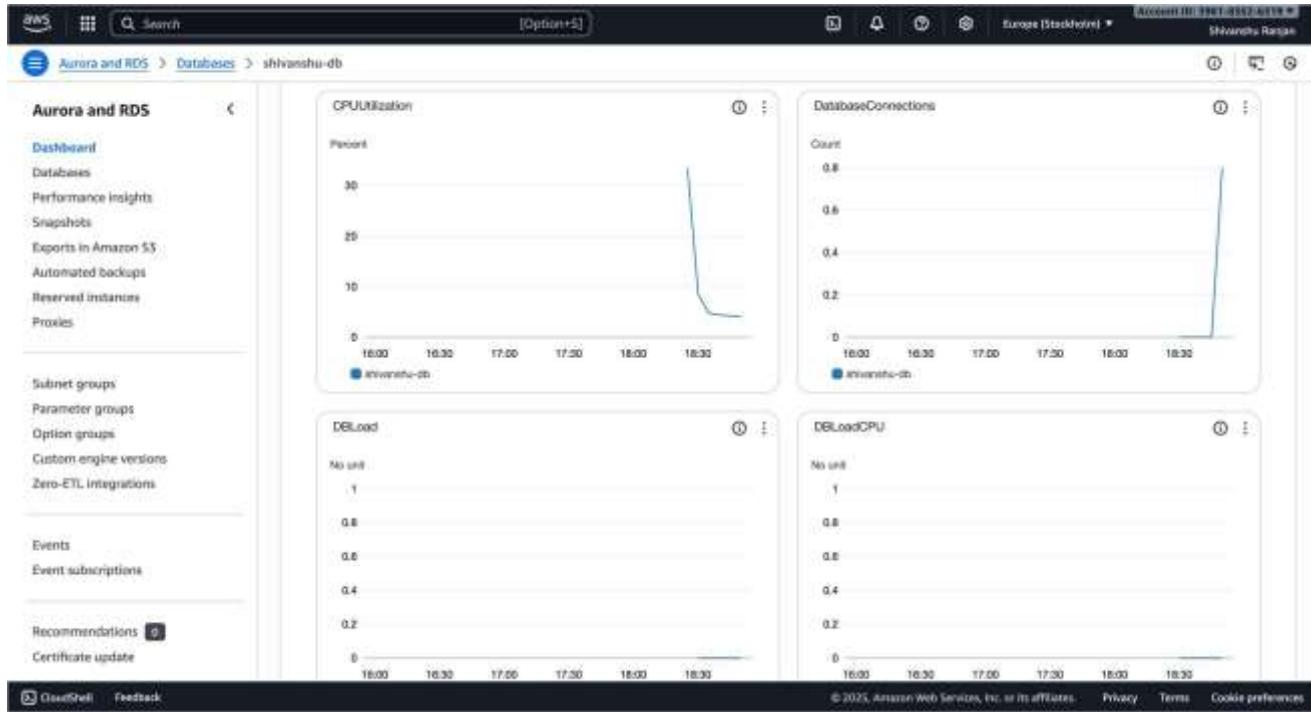




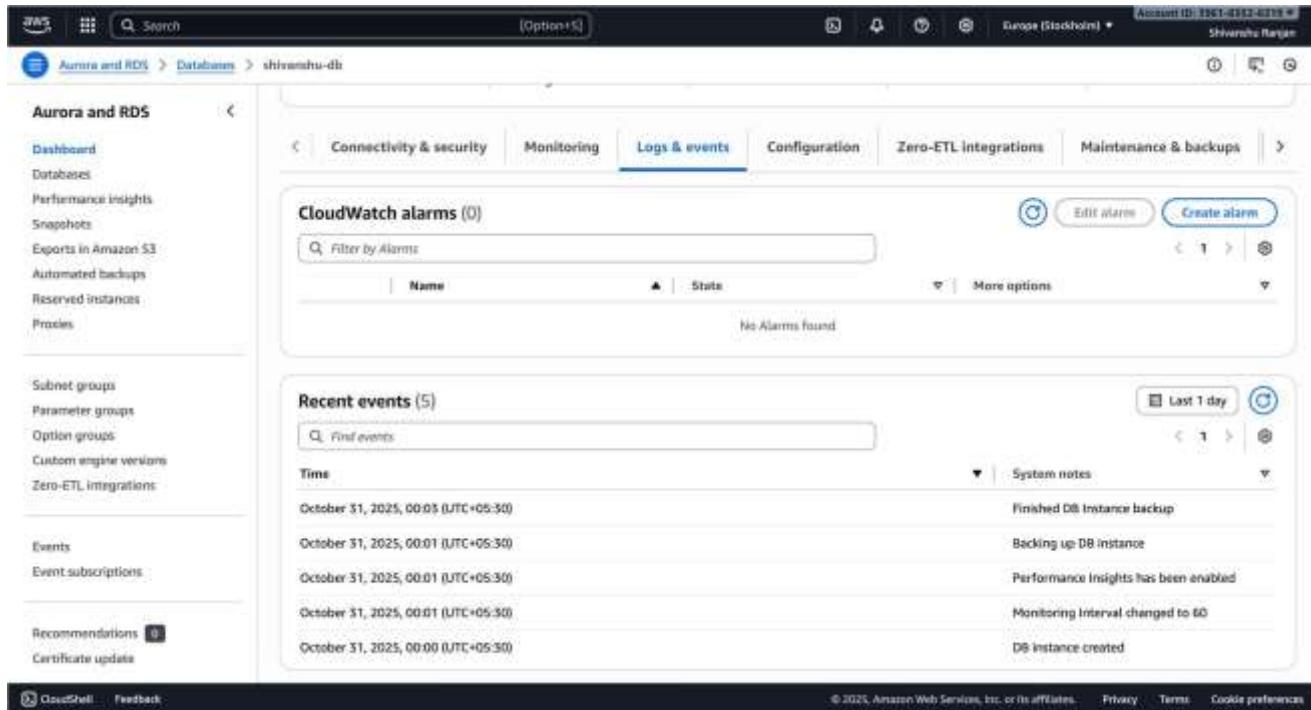
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

17. Adding a New Server in pgAdmin



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





DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

19. Successful Connection to AWS RDS Database via pgAdmin

The screenshot shows the AWS RDS (Amazon Relational Database Service) console. The left sidebar navigation includes: Dashboard, Databases (selected), Performance insights, Snapshots, Exports in Amazon S3, Automated backups, Reserved instances, Processes, Subnet groups, Parameter groups, Option groups, Custom engine versions, Zero-ETL integrations, Events, Event subscriptions, Recommendations (0), and Certificate update. The main content area displays a table titled 'Deleting DB Instance shivanshu-db'. The table has columns: DB identifier, Status, Role, Engine, Region ..., and Size. One row is listed: shivanshu-db (Deleting, Instance, PostgreSQL, eu-north-1a, db.t4g.micro). Above the table, a blue header bar indicates 'Deleting DB Instance shivanshu-db'.

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.