



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

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

The screenshot shows two adjacent web pages. On the left is the AWS Root user sign-in page. It features the AWS logo at the top, followed by a 'Root user sign in' button. Below it, a message says 'Enter the password for sumedhvats2004@gmail.com (not you?)'. There is a password input field containing '*****', a 'Show password' checkbox, a 'Forgot password?' link, and a large orange 'Sign in' button. Below the sign-in form are links for 'Sign in to a different account' and 'Create a new AWS account'. On the right is the Amazon Lightsail landing page. It has a dark background with a bright orange and yellow light streak effect. The 'Amazon Lightsail' logo is at the top, followed by the text 'Lightsail is the easiest way to get started on AWS'. There is a 'Learn more »' button and a cartoon robot character giving a thumbs-up.



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

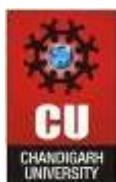
Discover. Learn. Empower.

2. Navigating to RDS Service

The screenshot shows the AWS CloudShell search interface. The search term 'rds' has been entered into the search bar at the top. The results are displayed in two main sections: 'Services' and 'Features'. Under 'Services', there are three items: 'Aurora and RDS' (Managed Relational Database Service), 'Database Migration Service' (Managed database migration service), and 'Kinesis' (Work with Real-Time Streaming Data). Under 'Features', there are three items: 'Database Insights' (CloudWatch feature), 'Reserved instances' (Aurora and RDS feature), and 'Proxies' (Aurora and RDS feature). Below these sections, there is a 'Resources in ap-south-1' section with a note about cross-Region search. On the right side of the screen, there is a separate window showing the Amazon RDS Dashboard for the 'ap-south-1' region, which includes tabs for 'Applications', 'Usage', and 'Billing'.

3. Amazon RDS Dashboard Overview

The screenshot shows the Amazon RDS Dashboard for the 'ap-south-1' region. The left sidebar contains navigation links for 'Aurora and RDS' (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 is divided into several sections: 'Resources' (Allocated storage (0.02 TB/100 TB), Instances and storage include Neptune and DocumentDB. Increase DB instances limit (2), DB Clusters (0/40), Reserved instances (0/20), Snapshots (8), and Recent events (4)), 'Parameter groups (1)', 'Option groups (1)', 'Subnet groups (1/20)', 'Supported platforms (VPC)', and 'Default network vpc-0db8aa979fd2f20b9'. There is also a 'Create a database' section with a note about using a backup from Amazon S3 to restore and create a new Aurora MySQL and MySQL database, and a 'Restore from S3' button. The right sidebar includes sections for 'Explore RDS' (Status: Complete, Estimated duration: 2-5 minutes, Start tutorial), 'Recommended services' (Amazon Augmented AI, AWS SimSpace Weaver, AWS Data Exchange, EC2 Image Builder, and AWS Firewall Manager), and 'Additional information'.



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

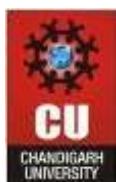
Discover. Learn. Empower.

4. Creating a New Database Instance

The screenshot shows the 'Create database' wizard in the AWS RDS console. At the top, there's a message about a free plan having limited features. Below it, two creation methods are shown: 'Standard create' (selected) and 'Easy create'. The 'Configuration' section allows choosing between various database engines: Aurora (MySQL Compatible), Aurora (PostgreSQL Compatible), MySQL, Oracle, PostgreSQL (selected), MariaDB, Microsoft SQL Server, and Redshift. The PostgreSQL option is highlighted with a blue border. At the bottom, there are tabs for 'CloudShell' and 'Feedback', along with standard AWS navigation icons.

5. Selecting PostgreSQL as Database Engine

This screenshot continues from the previous one, showing the 'Create database' wizard. In the 'Configuration' section, 'PostgreSQL' is selected as the engine type. The 'DB instance size' section shows several options: 'Production' (db.r7g.xlarge, 4 vCPUs, 32 GB RAM, 400 GB, 2.090 USD/hour), 'Dev/Test' (db.t4g.micro, 2 vCPUs, 16 GB RAM, 200 GB, 0.308 USD/hour), and 'Free tier' (db.t4g.micro, 2 vCPUs, 1 GB RAM, 20 GB, 0.025 USD/hour). The 'Free tier' option is selected and highlighted with a blue border. The bottom of the screen shows the 'CloudShell' tab and standard AWS footer links.



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

6. Choosing Deployment Option and Template

The screenshot shows the 'Create database' wizard in the AWS RDS console. The current step is 'DB instance identifier'. It displays three DB instance size options: 'Production' (db.t2.micro), 'Dev/Test' (db.t2.large), and 'Free tier' (db.t4g.micro). The 'Free tier' option is selected. Below the sizes, there's a field for 'DB instance identifier' with the placeholder 'database-2'. Under 'Master username', it says 'postgres' with the note '1 to 16 alphanumeric characters. The first character must be a letter.' There are two credential management options: 'Managed in AWS Secrets Manager - most secure' (selected) and 'Self managed'. The 'Self managed' option includes a note: 'Amazon RDS generates a password for you and manages it throughout its lifecycle using AWS Secrets Manager.' Below these, there's a 'Master password' field containing 'postgres' with validation notes: 'The Master password is invalid. Minimum constraints: At least 8 printable ASCII characters. Can't contain any of the following symbols: / \ ? @'. A 'Confirm master password' field is also present. At the bottom, there's a note: '▶ Set up EC2 connection - optional'.

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

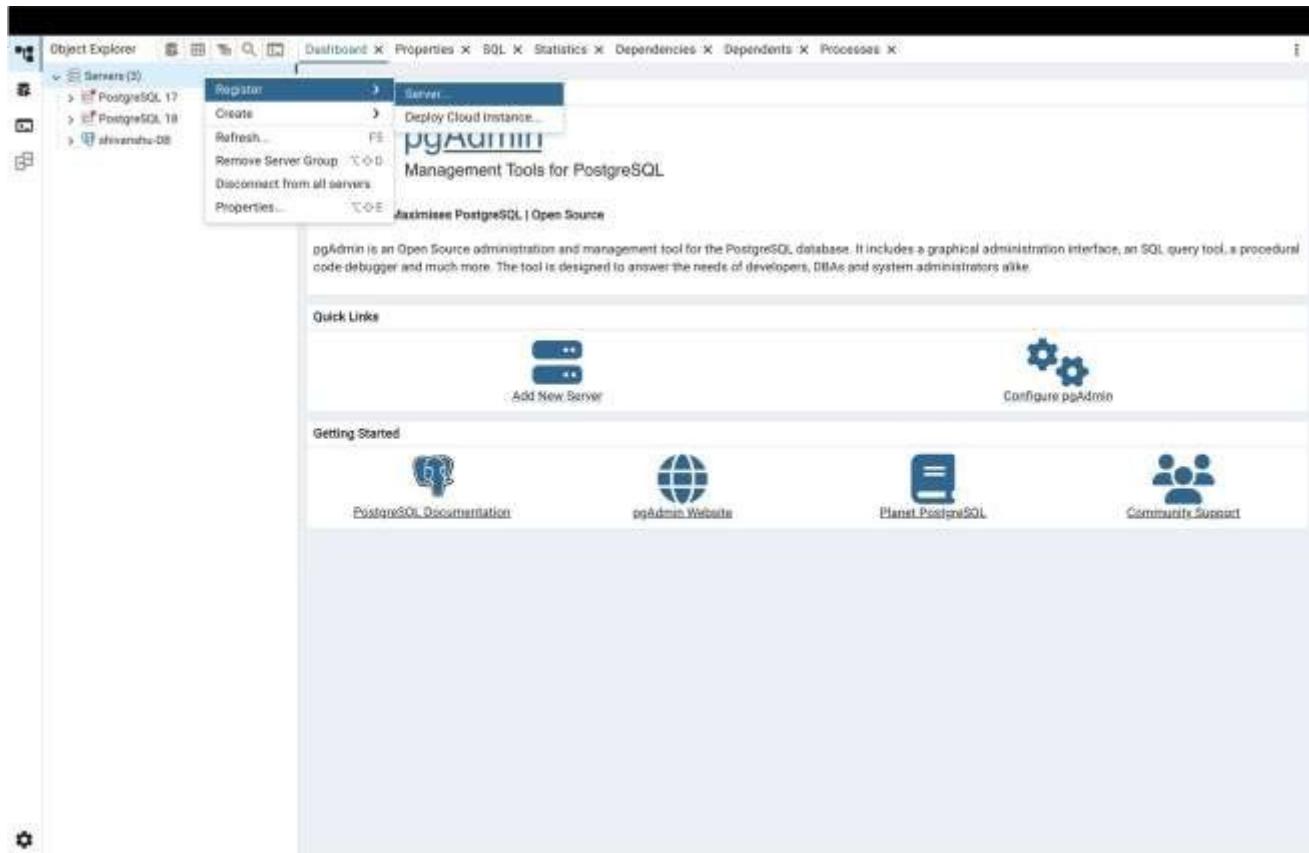
The screenshot shows the 'Databases' page in the AWS RDS console. On the left, there's a sidebar with 'Aurora and RDS' navigation and various settings like 'Subnet groups', 'Parameter groups', and 'Custom engine versions'. The main area shows a table titled 'Databases (1)'. The table has columns: DB identifier, Status, Role, Engine, Region..., Size, Recommendations, CPU, Current activity, and Maintenance. One row is listed: 'database-1' (Status: Available, Instance: PostgreSQL, Region: ap-south-1a, Size: db.t4g.micro, Recommendations: 1 Informational, CPU: 3.63%, Current activity: 0.00 sessions, Maintenance: none).



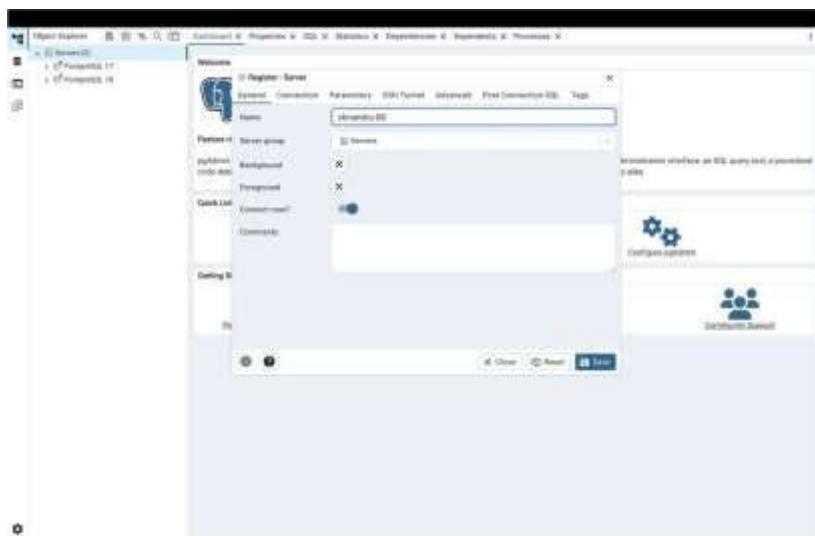
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

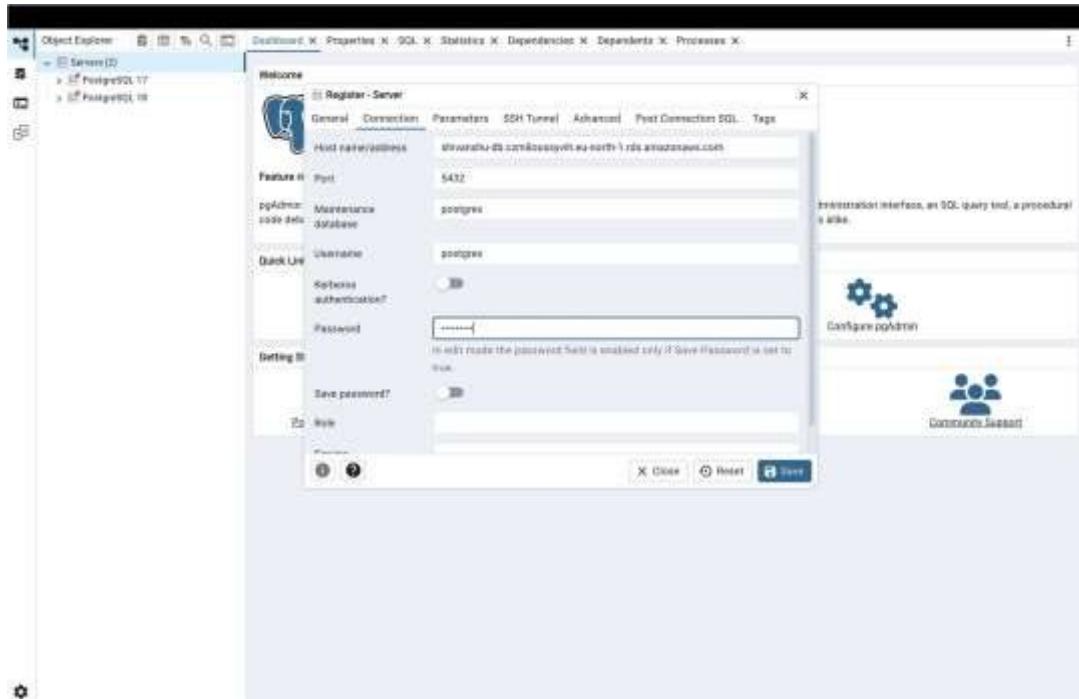
8. Setting Up Instance Size and Storage



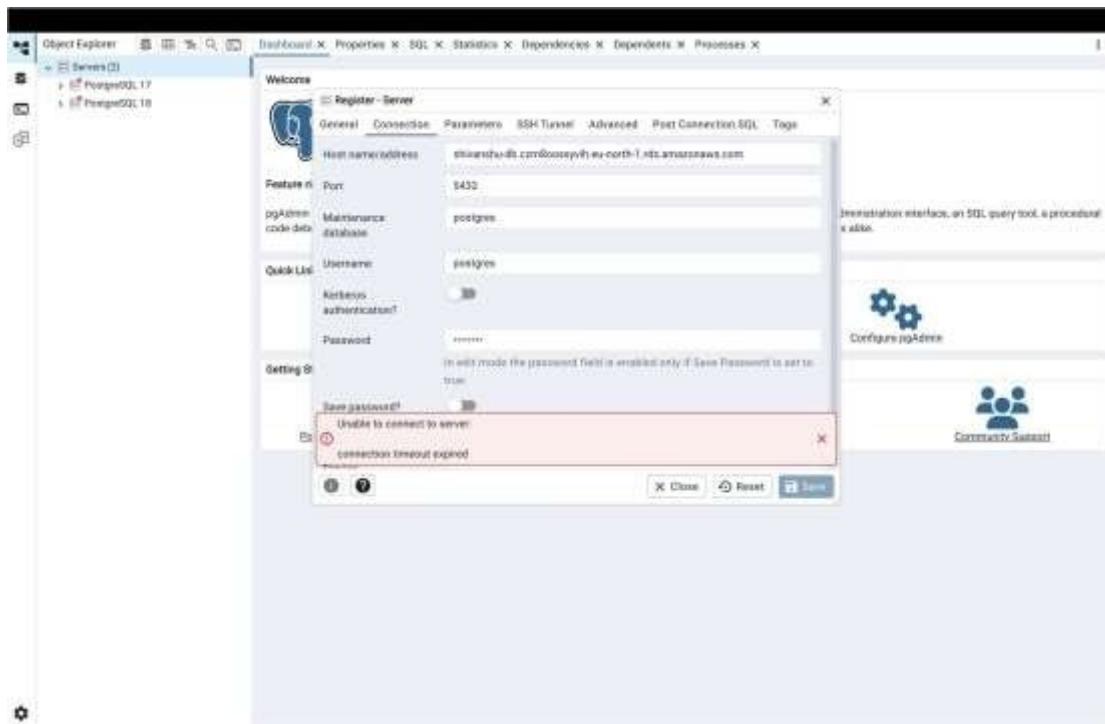
9. Configuring Connectivity and VPC Settings



10. Gr Setting Up Security Groups for RDS Access



11. Additional Database Configuration Options





DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

12. Reviewing and Creating the Database Instance

The screenshot shows the AWS RDS console with the database instance 'database-1'. The 'Summary' tab is selected, displaying details like DB identifier, Status (Available), Role (Instance), Engine (PostgreSQL), and Region & AZ (ap-south-1a). The 'Connectivity & security' tab is active, showing the endpoint (database-1.caalogq22pxe.ap-south-1.rds.amazonaws.com), port (5432), VPC (vpc-0dbbaa979fd2f20b9), Subnet group (default-vpc-0dbbaa979fd2f20b9), and Subnets (subnet-0370a3843d47fb043, subnet-0236a3778bcf9c92f, subnet-00cce0a66f1d6e78d). It also lists the Network type (IPv4) and Security settings, including VPC security groups (default sg-05c3b0fb496219656) and Certificate authority (rds-ca-rsa2048-g1).

13. RDS Instance Creation in Progress

The screenshot shows the AWS EC2 Security Groups console for the security group 'sg-05c3b0fb496219656 - default'. The 'Edit inbound rules' section is open, showing two existing rules: one for 'All traffic' from 'sg-05c3b0fb496219656' and another for 'All traffic' from '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.' Below the rules are 'Add rule' and 'Save rules' buttons.



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

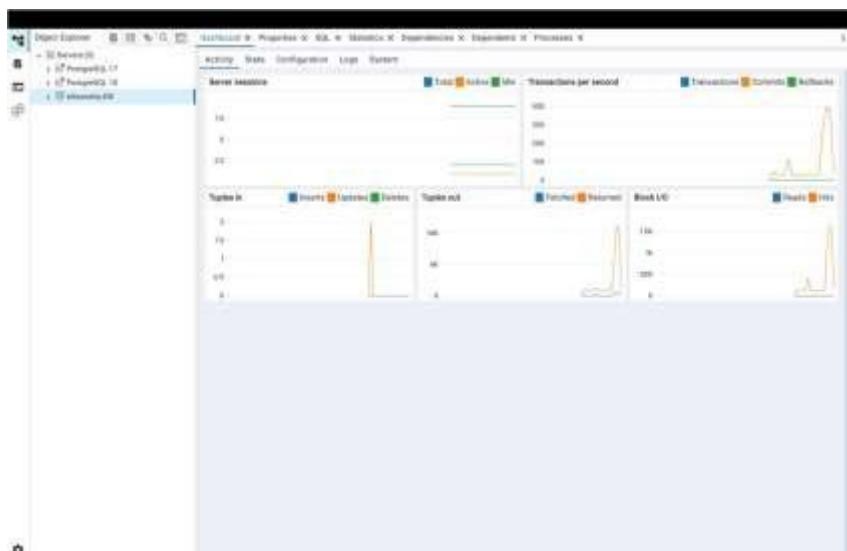
15. Copying the RDS Endpoint for Connection

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

Connectivity & security

Endpoint & port	Networking	Security
Endpoint database-1.czalogq2px.ap-south-1.rds.amazonaws.com	Availability Zone ap-south-1a	VPC security groups default (sg-05c3b0fb496219656)
Port 5432	VPC vpc-0db8aa979fd2f20b9	Publicly accessible No
	Subnet group default-vpc-0db8aa979fd2f20b9	Certificate authority Info rds-ca-rsa2048-g1
	Subnets subnet-0370a3843d47fb043 subnet-023643778bcf98c2f subnet-00cce0a66f1d6e78d	Certificate authority date May 20, 2061, 00:10 (UTC+05:30)
	Network type IPv4	DB instance certificate expiration date November 04, 2026, 09:43 (UTC+05:30)

16. Launching pgAdmin on Local Machine

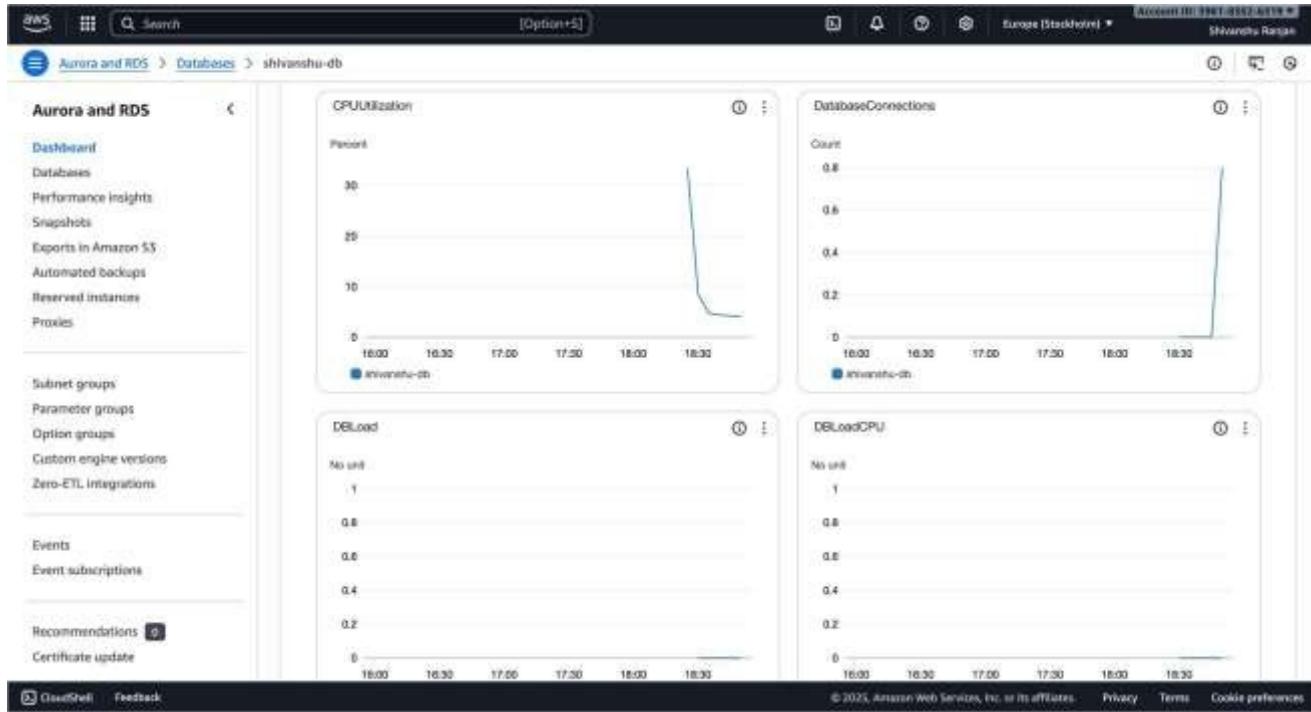




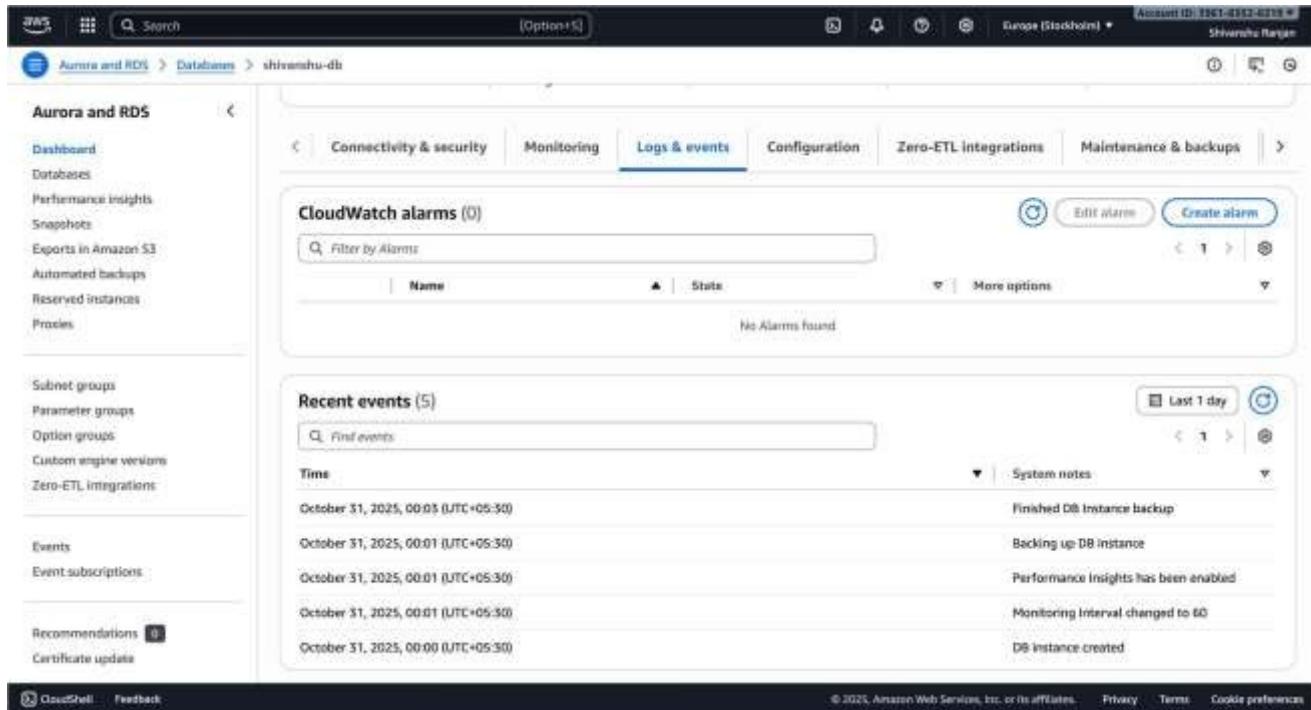
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

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 console for the Aurora and RDS service. On the left, there's a sidebar with navigation links like Dashboard, Databases, Performance insights, Snapshots, and more. The main area displays a log viewer titled 'Logs (73)'. It lists 73 log entries for 'error/postgresql.log' from November 8, 2025, at various times between 00:25 and 12:26 UTC+05:30. Each log entry is 4.4 kB in size. Above the log viewer, there are sections for 'Find events' and 'System notes' which mention a finished DB instance backup and backing up the DB instance. The top right corner shows account information and a user name.

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.