



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Worksheet 9

Student Name: Aayush Aman

Branch: CSE

Semester: 5th

Subject Name: ADBMS

UID: 23BCS14046

Section/Group: KRG 3-B

Date of Performance: 31/10/2025

Subject Code: 23CSP-333

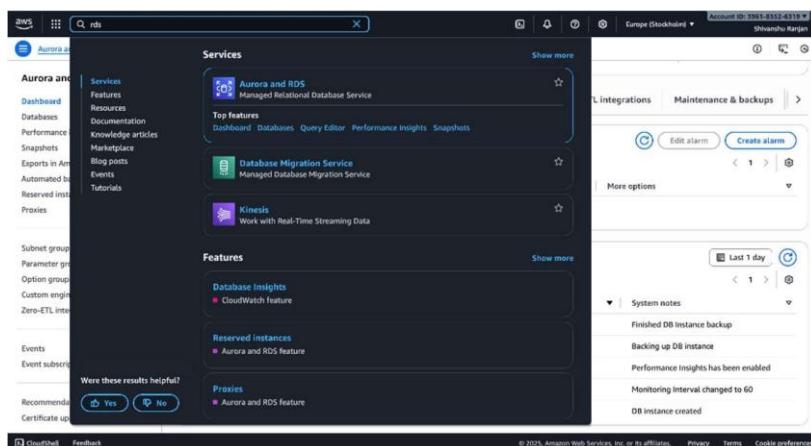
1. Aim: To understand and implement the setup of Amazon Relational Database Service (AWS RDS) by creating a database instance, configuring security groups, and establishing a secure connection between the local pgAdmin tool and the RDS instance hosted on the AWS Cloud.

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 Databases page. On the left, there's a sidebar with various navigation options like Dashboard, Databases, Performance insights, etc. The main area is titled 'Databases (0)' and features a search bar and filters for DB identifier, Status, Role, Engine, Region, and Size. A large, friendly-looking robot icon is centered above the message 'No resources'. Below it, a button says 'Create database'. At the bottom, there's a footer with links to CloudShell, Feedback, and legal notices.

3. Amazon RDS Dashboard Overview

The screenshot shows the AWS RDS Dashboard. The sidebar includes options like Dashboard, Databases, and Events. The main dashboard has a 'Resources' section showing usage in the Europe (Stockholm) region, including DB Instances, Parameter groups, DB Clusters, Option groups, Subnet groups, and more. It also displays Recent events and Event subscriptions. A 'Create a database' button is prominent. To the right, there's a 'Explore RDS' section with a tutorial, status, complete by date, reward value, and estimated duration. Another section for Recommended services shows no results yet. The footer includes CloudShell, Feedback, and legal notices.



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 a free plan having limited features. Below it, under 'Choose a database creation method', the 'Easy create' option is selected. In the 'Configuration' section, 'PostgreSQL' is chosen as the engine type. Other options like Aurora (MySQL Compatible), Aurora (PostgreSQL Compatible), MySQL, MariaDB, and Oracle are also listed with their respective icons. At the bottom, there are links for CloudShell, Feedback, and various AWS terms like Privacy, Terms, and Cookie preferences.

5. Selecting PostgreSQL as Database Engine

This screenshot continues from the previous one, showing the 'Create database' page with PostgreSQL selected. It displays three configuration options for the DB instance identifier: 'shivanush-DB'. Under 'Master username', 'postgres' is entered. For 'Credentials management', the 'Self managed' option is selected. A password field contains '*****'. The 'Password strength' bar is at 'Neutral'. At the bottom, there are links for CloudShell, Feedback, and various AWS terms like Privacy, Terms, and Cookie preferences.



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 configuration details are as follows:

Setting	Value	Status
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

A note at the bottom states: "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."

Buttons at the bottom right include 'Cancel' and 'Create database'.

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

The screenshot shows the 'Databases' page in the AWS RDS console. A blue banner indicates that the database 'shivanshu-db' is 'Creating'. The table lists the database details:

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

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.

At the bottom, there are links for CloudShell, Feedback, and footer text: © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences.



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

8. Setting Up Instance Size and Storage

The screenshot shows the pgAdmin 4 interface. In the top navigation bar, 'Dashboard' is selected. The left sidebar shows 'Object Explorer' with a 'Servers' section containing three entries: 'PostgreSQL 17', 'PostgreSQL 18', and 'shivanshu-DB'. A context menu is open over the 'shivanshu-DB' entry, with 'Server...' highlighted. Below the menu, a tooltip for 'pyAUMITI' provides information about pgAdmin as an open-source administration tool. The main pane displays 'Quick Links' for 'Add New Server', 'Configure pgAdmin', 'PostgreSQL Documentation', 'pgAdmin Website', 'Planet PostgreSQL', and 'Community Support'.

9. Configuring Connectivity and VPC Settings

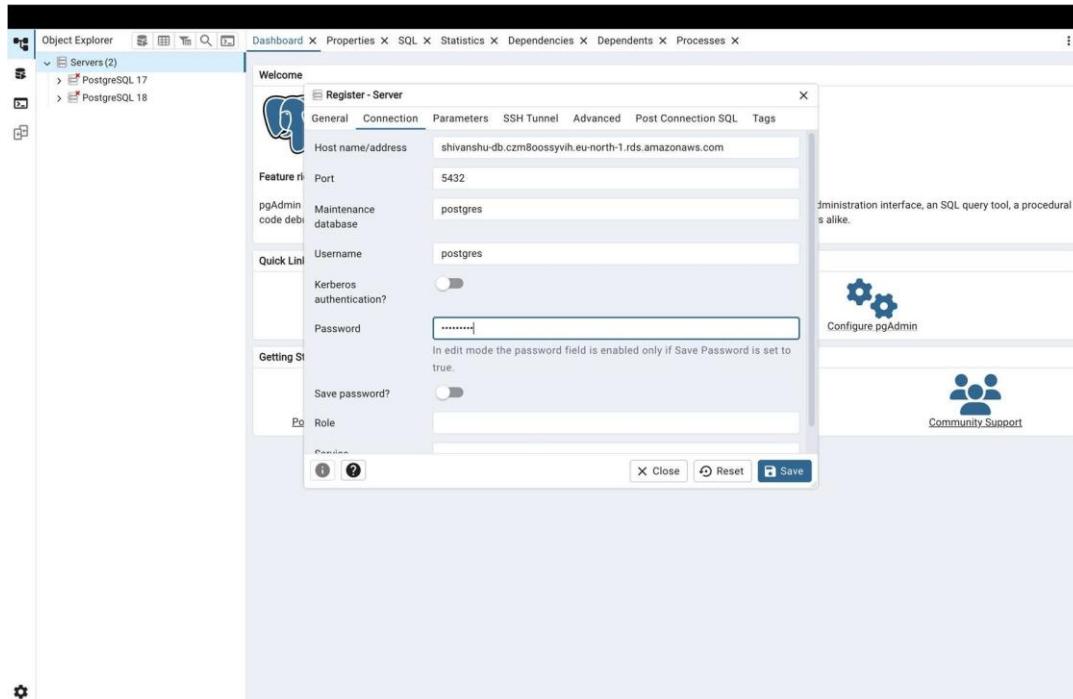
The screenshot shows the 'Register - Server' dialog box in pgAdmin 4. The 'General' tab is selected. The 'Name' field contains 'shivanshu-DB'. Under the 'Feature' section, 'Background' and 'Foreground' are set to 'Servers'. The 'Comments' field is empty. At the bottom, there are 'Close', 'Reset', and 'Save' buttons.



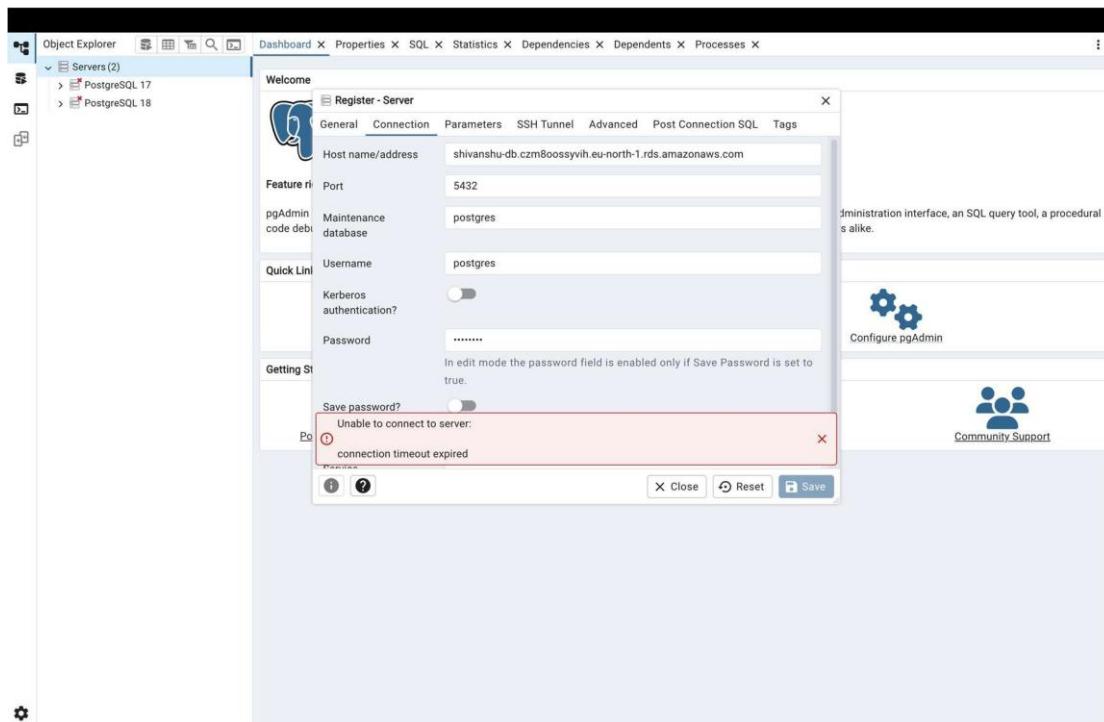
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

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. The 'Summary' tab is selected, displaying basic information like DB identifier, status (Available), role (Instance), engine (PostgreSQL), and region (eu-north-1a). Below the summary, there are tabs for Connectivity & security, Monitoring, Logs & events, Configuration, Zero-ETL integrations, and Maintenance & backups. The 'Connectivity & security' tab is active, showing details such as endpoint, port, networking (availability zone eu-north-1a, VPC vpc-086507ee77883ae1b, subnet group default-vpc-086507ee77883ae1b, subnets subnet-0db6b45e321b7000a, subnet-087377db566f545dc, subnet-0bac42bdab1e990c5), and security (VPC security groups default sg-0b4c8dc4647072099, Active). The bottom of the page includes CloudShell, Feedback, and copyright information.

13. RDS Instance Creation in Progress

The screenshot shows the AWS EC2 Security Groups console for the security group 'sg-0b4c8dc4647072099 - default'. An inbound rule is being edited for a PostgreSQL instance. The rule configuration includes: Type: All traffic, Protocol: TCP, Port range: 5432, Source: My IP, and Description: optional. A new rule is being added with: Type: PostgreSQL, Protocol: TCP, Port range: 5432, Source: 47.247.118.30/32, and Description: optional. At the bottom, there are buttons for Add rule, Preview changes, and Save rules.



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

Endpoint
 [shivanshu-db.czr8oossyvih.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

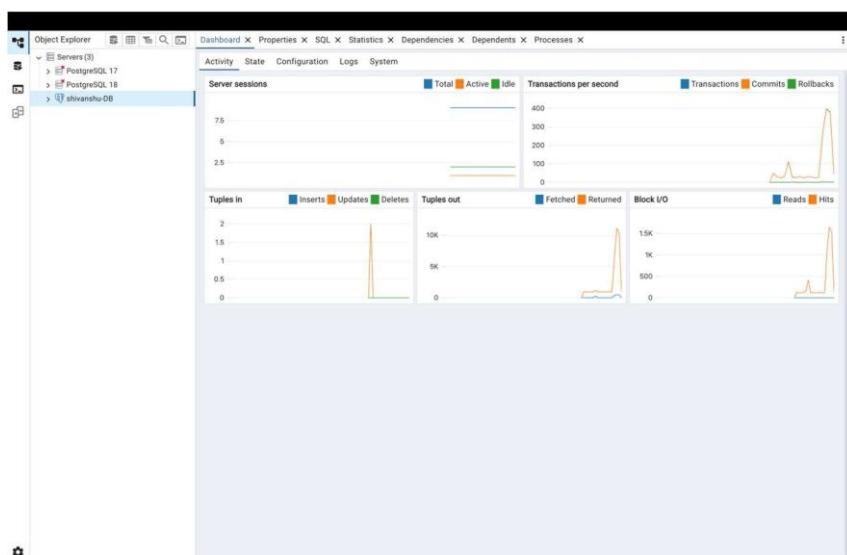
Publicly accessible
Yes

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

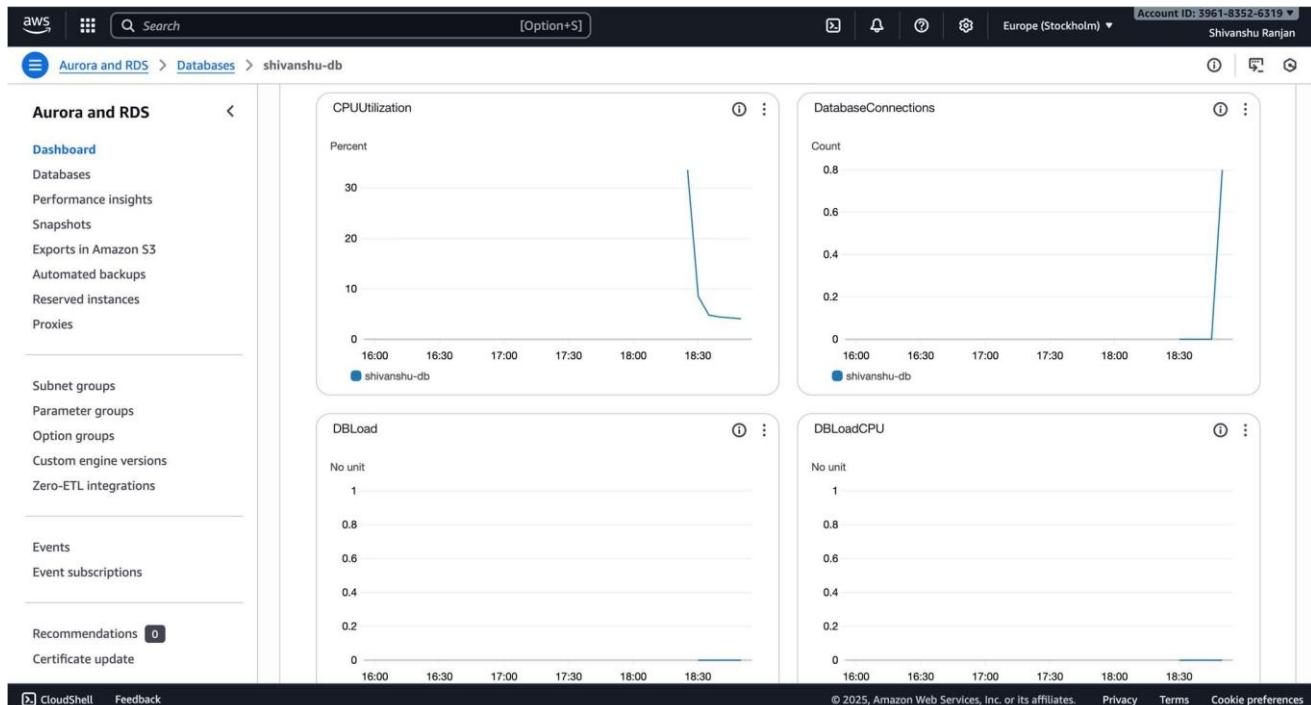




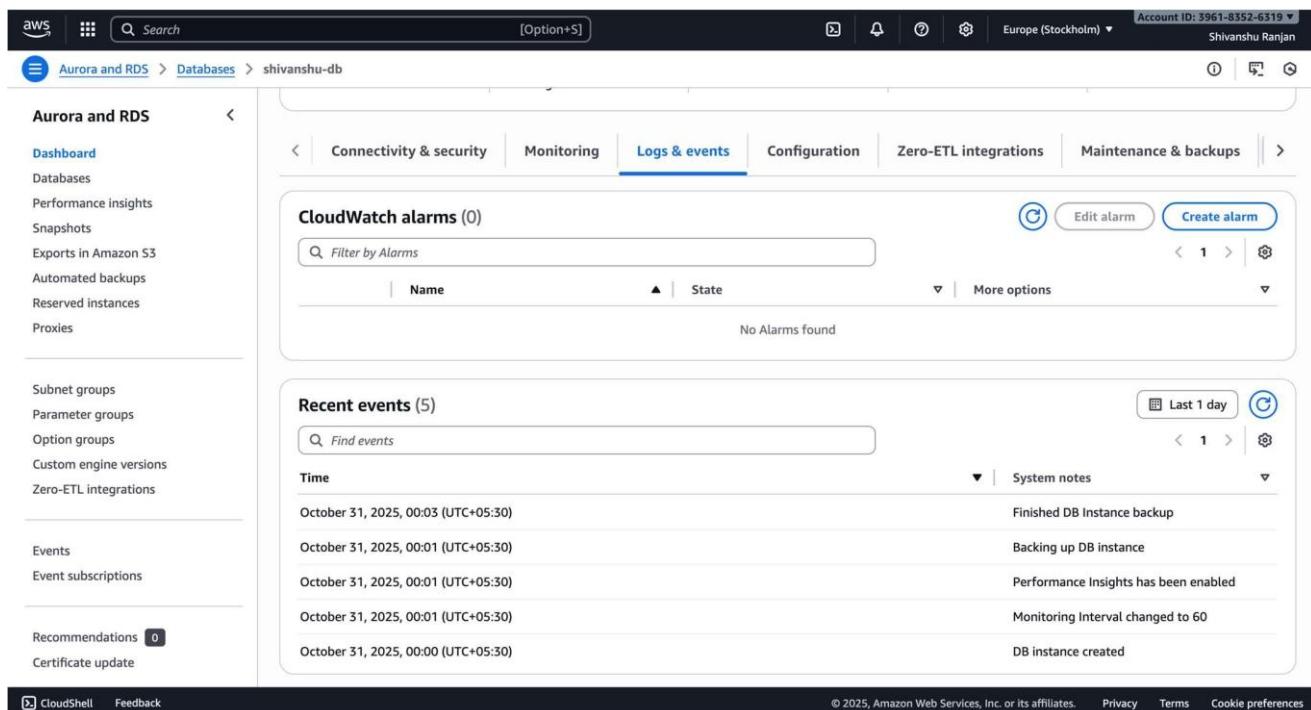
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

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 under the 'Aurora and RDS' section. A modal window titled 'Deleting DB instance shivanshu-db' is open, indicating the current status of the database instance. The main table lists one database entry:

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

The left sidebar contains navigation 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. At the bottom, there are links for CloudShell, Feedback, and a footer with copyright information and links to 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.