DEPARTMENTOF
COMPUTERSCIENCE&ENGINEERING
Discover. Learn. Empower.

WORKSHEET 9

Student Name: Omprakash Reddy **UID: 23BAI70296**

Branch: B.E CSE-H (AIML) **Section: 23-AIT_KRG _G2**

Semester: 5th **Date of Performance: 27-10-2025**

Subject Name: ADBMS **Subject Code: 23CSP-333**

1. AIM:

To understand and implement Amazon Web Services (AWS) Relational Database Service (RDS). This experiment focuses on creating and managing a database instance on AWS RDS, configuring security groups, and connecting local pgAdmin to the cloud-hosted RDS instance.

2. Tools Used:

AWS Console, RDS (Relational Database Service), EC2, PostgreSQL, and pgAdmin.

3. Procedure / Steps:

1. Login to AWS Management Console.
   - Navigate to https://aws.amazon.com/ and click on “Sign In”.

2. Search for “RDS” in the AWS search bar and open the RDS dashboard.

3. Create a New Database Instance:
   - Click on “Create Database”.
   - Choose a database engine (e.g., PostgreSQL or MySQL).
   - Choose “Standard Create” for full configuration control.
   - Specify DB instance identifier, username, and password.
   - Configure instance size, storage type, and enable auto-scaling.

4. Configure Connectivity:
- Select the correct Virtual Private Cloud (VPC).
- Set Public Access to “Yes” if you want to connect from outside AWS.
- Create or select an existing security group.
- Adjust inbound rules to allow connections from your IP address (default port 5432 for PostgreSQL).

5. Database Creation:
- Click “Create Database” and wait for the status to become “Available”.

6. Connect to RDS from Local pgAdmin:
- Open pgAdmin → Add New Server.
- Enter the RDS endpoint as the host name (found in RDS dashboard).
- Provide username and password created during setup.
- Test the connection and save the configuration.

7. (Optional) Launch EC2 Instance for Access:
- Create an EC2 instance with PostgreSQL/MySQL client pre-installed.
- Use SSH to connect and access the RDS database through command-line clients.

8. Verify Connection:
- Run SQL queries from pgAdmin or EC2 to ensure the RDS instance is working.

4. Output:

The AWS RDS database instance was successfully created and connected through pgAdmin. Security groups were configured to allow inbound connections, and the database was accessed using both pgAdmin and EC2 instances. The process demonstrated how cloud-based relational databases can be managed efficiently with high availability and security.

5. Learning Outcomes:

- Learned about AWS RDS and its advantages over EC2-hosted databases.
- Understood how to create and configure a cloud-based database instance.
- Gained practical knowledge of managing inbound rules and security groups.
- Learned to connect local clients (pgAdmin) to RDS instances.
- Observed key AWS RDS features such as backups, monitoring, and scalability.