



PARSHVANATH CHARITABLE TRUST'S

**A. P. SHAH INSTITUTE OF TECHNOLOGY**

**Department of Information Technology**

**(NBA Accredited)**



**Academic Year: 2022-23**

**Semester: VIII**

**Class / Branch: BE- IT**

**Subject: Cloud Computing Lab**

**Name of Instructor: Yaminee Patil**

**Name of Student:**

**Student ID:**

**Roll No.:**

**Date of Submission:**

---

## **Experiment No.:08**

**Aim:** To implement Database as a Service on SQL/NOSQL databases like AWS RDS.

**Problem Statement:** To build your DB server and interact with your DB using an app

**Software Used:** AWS Console

### **Theory:**

Amazon Relational Database Service (Amazon RDS) is a web service that makes it easier to set up, operate, and scale a relational database in the AWS Cloud. It provides cost-efficient, resizable capacity for an industry-standard relational database and manages common database administration tasks.

Amazon RDS is a managed database service. It's responsible for most management tasks. By eliminating tedious manual tasks, Amazon RDS frees you to focus on your application and your users. We recommend Amazon RDS over Amazon EC2 as your default choice for most database deployments.

Amazon RDS provides the following specific advantages over database deployments that aren't fully managed:

- You can use the database products you are already familiar with: MariaDB, Microsoft SQL Server, MySQL, Oracle, and PostgreSQL.
- Amazon RDS manages backups, software patching, automatic failure detection, and recovery.
- You can turn on automated backups, or manually create your own backup snapshots. You can use these backups to restore a database. The Amazon RDS restore process works reliably and efficiently.
- You can get high availability with a primary instance and a synchronous secondary instance that you can fail over to when problems occur. You can also use read replicas to increase read scaling.
- In addition to the security in your database package, you can help control who can access your RDS databases. To do so, you can use AWS Identity and Access



PARSHVANATH CHARITABLE TRUST'S

**A. P. SHAH INSTITUTE OF TECHNOLOGY**

**Department of Information Technology**

**(NBA Accredited)**



---

Management (IAM) to define users and permissions. You can also help protect your databases by putting them in a virtual private cloud (VPC).

### **Amazon RDS Custom for Oracle and Microsoft SQL Server**

Amazon RDS Custom is an RDS management type that gives you full access to your database and operating system.

You can use the control capabilities of RDS Custom to access and customize the database environment and operating system for legacy and packaged business applications. Meanwhile, Amazon RDS automates database administration tasks and operations.

In this deployment model, you can install applications and change configuration settings to suit your applications. At the same time, you can offload database administration tasks such as provisioning, scaling, upgrading, and backup to AWS. You can take advantage of the database management benefits of Amazon RDS, with more control and flexibility.

For Oracle Database and Microsoft SQL Server, RDS Custom combines the automation of Amazon RDS with the flexibility of Amazon EC2.

With the shared responsibility model of RDS Custom, you get more control than in Amazon RDS, but also more responsibility.

### **Amazon RDS on AWS Outposts**

Amazon RDS on AWS Outposts extends RDS for SQL Server, RDS for MySQL, and RDS for PostgreSQL databases to AWS Outposts environments. AWS Outposts uses the same hardware as in public AWS Regions to bring AWS services, infrastructure, and operation models on-premises. With RDS on Outposts, you can provision managed DB instances close to the business applications that must run on-premises.

### **DB instances**

A *DB instance* is an isolated database environment in the AWS Cloud. The basic building block of Amazon RDS is the DB instance.

Your DB instance can contain one or more user-created databases. You can access your DB instance by using the same tools and applications that you use with a standalone database



PARSHVANATH CHARITABLE TRUST'S

**A. P. SHAH INSTITUTE OF TECHNOLOGY**

**Department of Information Technology**

**(NBA Accredited)**



---

instance. You can create and modify a DB instance by using the AWS Command Line Interface (AWS CLI), the Amazon RDS API, or the AWS Management Console.

## **DB engines**

A *DB engine* is the specific relational database software that runs on your DB instance. Amazon RDS currently supports the following engines:

- MariaDB
- Microsoft SQL Server
- MySQL
- Oracle
- PostgreSQL

Each DB engine has its own supported features, and each version of a DB engine can include specific features. Support for Amazon RDS features varies across AWS Regions and specific versions of each DB engine. To check feature support in different engine versions and Regions, see [Supported features in Amazon RDS by AWS Region and DB engine](#).

Additionally, each DB engine has a set of parameters in a DB parameter group that control the behaviour of the databases that it manages.

## **DB instance classes**

A *DB instance class* determines the computation and memory capacity of a DB instance. A DB instance class consists of both the DB instance type and the size. Each instance type offers different compute, memory, and storage capabilities. For example, db.m6g is a general-purpose DB instance type powered by AWS Graviton2 processors. Within the db.m6g instance type, db.m6g.2xlarge is a DB instance class.

You can select the DB instance that best meets your needs. If your needs change over time, you can change DB instances.

## **DB instance storage**

Amazon EBS provides durable, block-level storage volumes that you can attach to a running instance. DB instance storage comes in the following types:

- General Purpose (SSD)



PARSHVANATH CHARITABLE TRUST'S

**A. P. SHAH INSTITUTE OF TECHNOLOGY**

**Department of Information Technology**

**(NBA Accredited)**



- 
- Provisioned IOPS (PIOPS)
  - Magnetic

The storage types differ in performance characteristics and price. You can tailor your storage performance and cost to the needs of your database.

Each DB instance has minimum and maximum storage requirements depending on the storage type and the database engine it supports. It's important to have sufficient storage so that your databases have room to grow. Also, sufficient storage makes sure that features for the DB engine have room to write content or log entries.

**Conclusion:** Thus, we have successfully deployed versioning on S3 objects using Database as-a-Service on AWS.