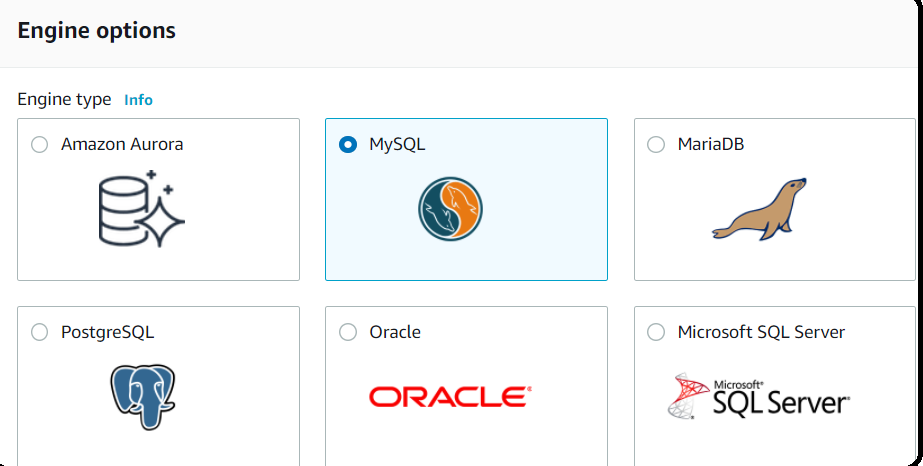
Relation Database Service (RDS)

(Managed Relational Database Service)

RDS is a web service or managed DB service used to set up, operate, and scale a relational database on AWS

Database engines supported by Amazon RDS

* Microsoft SQL Server
* Oracle
* MySQL Server
* PostgreSQL
* Amazon Aurora
* MariaDB



Benefits of AWS RDS managed service

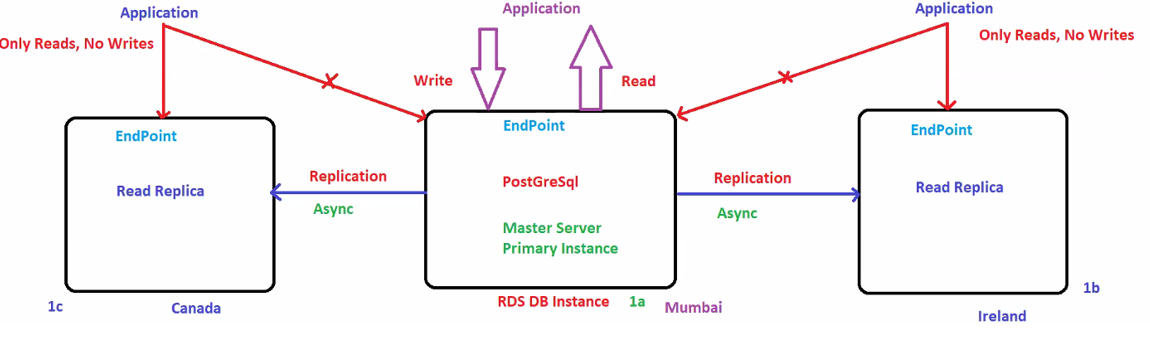
* Scaling capability
* Manage backups
* Software patching
* Automatic failover
* Read replicas
* High Availability (Multi-AZ deployment)
* Automated backups
* No shell access to the underlying OS

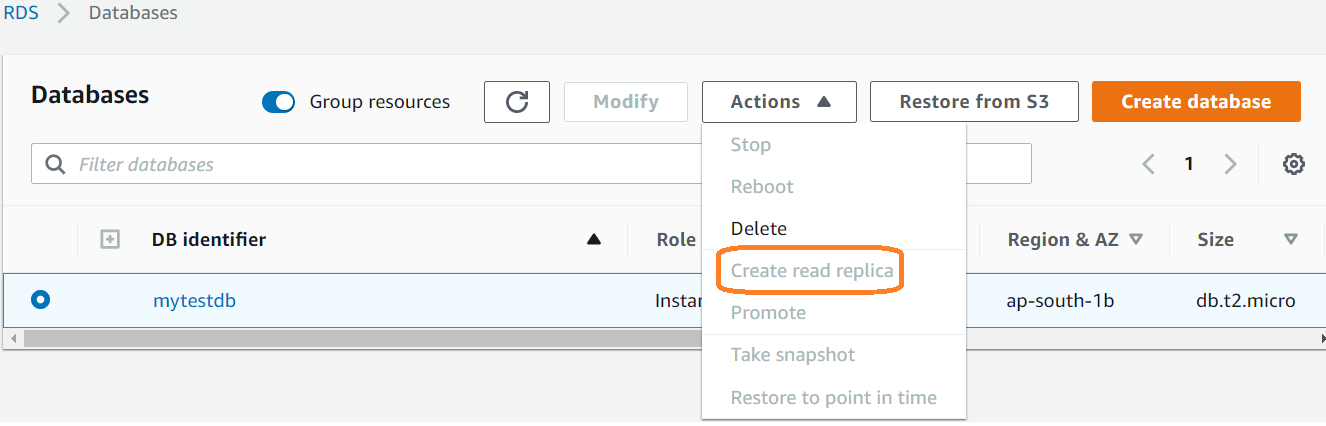
DB Instances

* An isolated database environment in AWS
* A DB instance can contain multiple databases
* Each DB instance runs a DB engine
* Compute and memory capacity determined DB instance class (example: db.m5, db.m4, etc)
* DB instance storage types support: Magnetic, General-Purpose SSD, Provisioned IOPS(i01)

Read Replica

* Reduce the load on source DB instance using Read Replica DB instance
* Read Replicas are used to increase the performance
* Read Replicas can only be used for **read not for write purpose**
* Read replicas allows you to have a read-only copy of your production database. This is achieved by using **Asynchronous replication** from the primary RDS instance to the read replica.
* You use read replicas primary for very read-heavy database workloads
* **You can have up to 5RR copies of any database**
* Each RR will have its own DNS end point
* Application must update the DB connection string to use Read Replica for reads
* Read Replica can be **Cross-Zone, Cross-Region**
* You can promote a Read Replica into a standalone DB instance





RDS Back-ups

Two types of Backups

* Automated Backups
* DB Snapshots

Automated Backups

* Automated backups allow you to recover your database to any point in time within a "**retention period**". **The retention period can be between one and 35 days.**
* Incremental backup is called transactional logs.
* Automated Backups will take a full daily snapshot and will also store transaction logs throughout the day. When you do a recovery. AWS will first choose the most recent daily backup, and then apply transaction logs relevant to the day.
* **Automated Backups are enabled by default**
* DB instance must be in ACTIVE state for automated backups to occur
* For example, for STORAGE\_FULL state, automated backups won't occur
* Automated backups occur during the preferred backup window
* If we intentionally delete the database, **backup will also be deleted.**

Snapshots

* DB Snapshots are done manually (i.e they are user initiated).They are stored even after you delete the original RDS instance, unlike automated backups.
* Manual backups by manually creating DB snapshots
* First snapshot is full DB backup, subsequent backups are incremental, which means only the data that has changed after most recent snapshot
* Default backup retention period:

1 Day 🡪 if DB instance created using AWS RDS API or AWS CLI

7 Days 🡪 Default retention period from console

* Manual snapshot limit 🡪 100 per region
* **AWS will take incremental backup in every 1 min.**
* **Manual backups are user initiated.**
* **Even if you delete the database intentionally, we will have manual backups.**
* **No retention period applicable for manual backup**

To copy snapshot to another region

**Actions 🡪 copy snapshot 🡪 select the destination region**

Restoring Backups

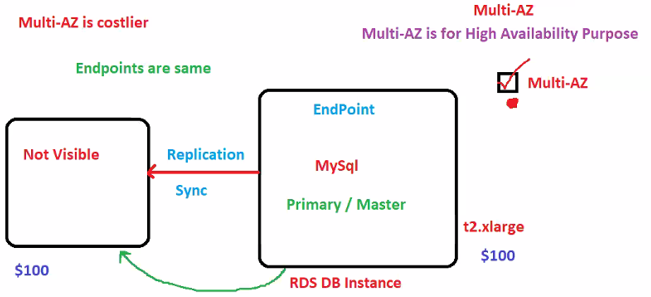
* Whenever you restore either an Automatic Backup or a manual Snapshot, the restored version of the database will be a new RDS instance with a new DNS endpoint.

Multi-AZ

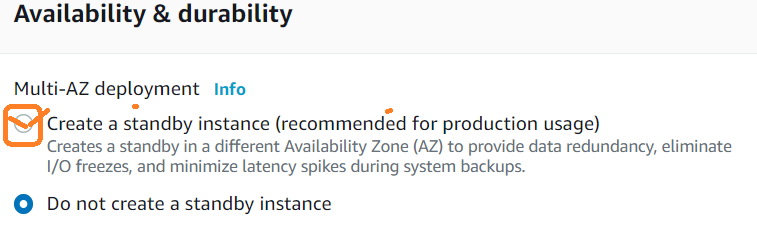
* Multi-AZ allows you to have an exact copy of your production database in another AZ.AWS handles the replication for you, so when your production database is written to, this write will automatically be synchronized to the stand by database
* In the event of planned database maintenance, DB Instance failure, or an AZ failure, Amazon RDS will automatically failover to the standby so that database operations can resume quickly without administrative intervention
* Both DB servers have same DNS endpoints.

**Sample End point url:**

* **mytestdb.cgpudzjguigp.ap-south-1.rds.amazonaws.com**
* instance endpoint which is used to connect the database and my sql instance will be running 3306 port by default.



**Select your database🡪modify🡪Multi AZ Deployment🡪yes (it just matters of selecting radio button) 🡪 Continue**



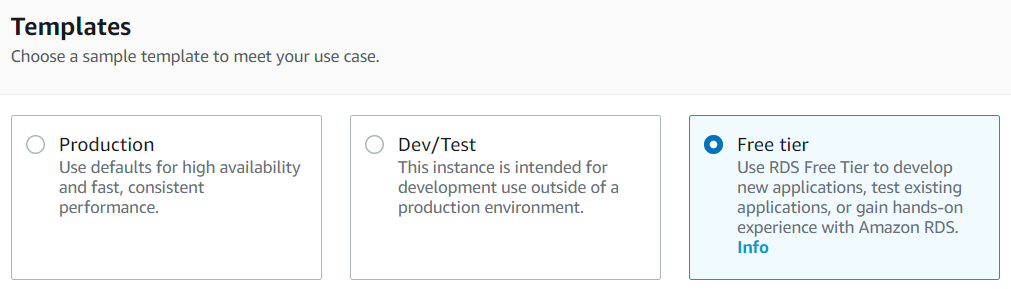
Encryption

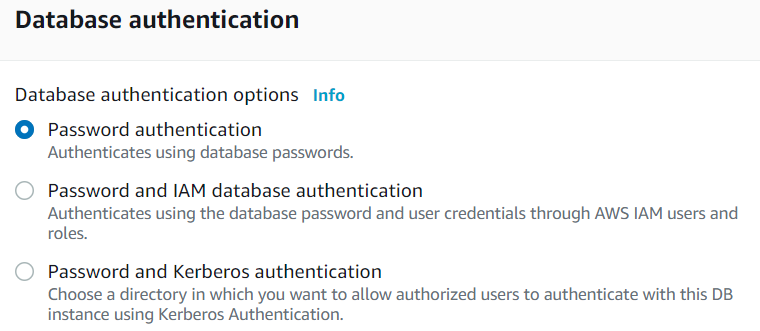
* Encryption at rest (AES-256) by enabling encryption option for your DB instance
* Data that is encrypted at rest includes storage for DB instances, automated backups, Read Replicas, and snapshots
* Encryption in transit using SLL
* You can use SSL from your application to encrypt a connetion to your DB instance

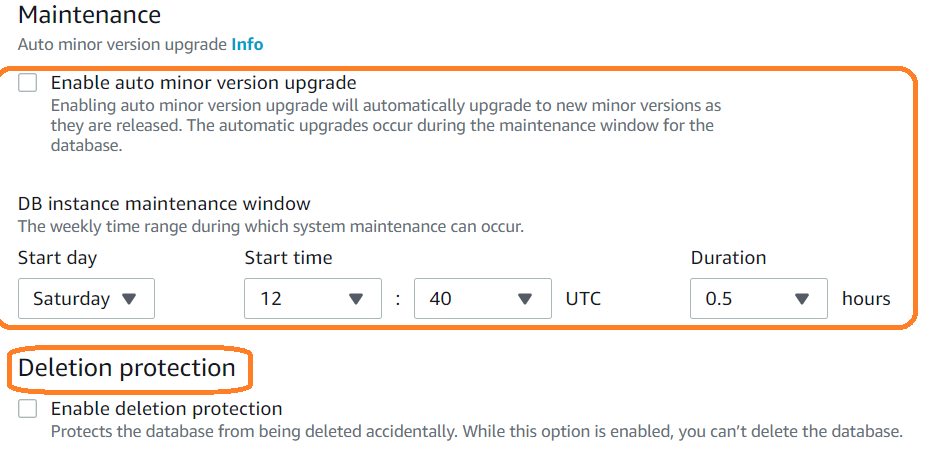
Security

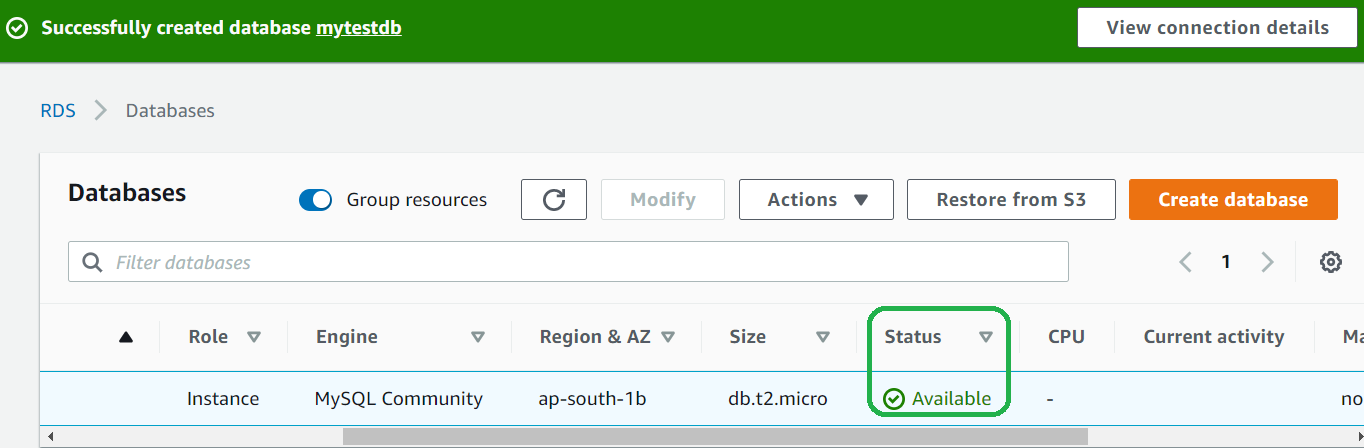
* Run DB instance in VPC
* Use IAM policies to assign permissions that determine who is allowed to manage RDS resources
* Use Security Groups to control what IP address can connect to RDS DB

Template



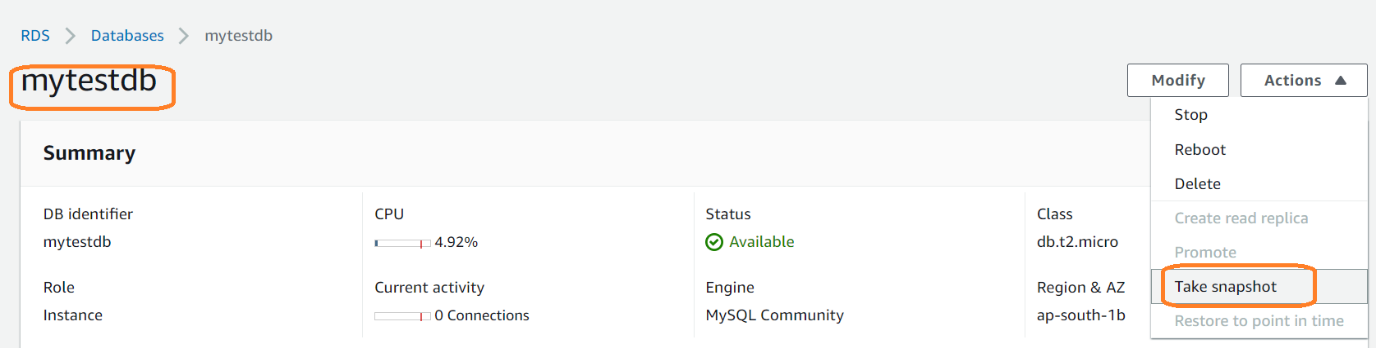




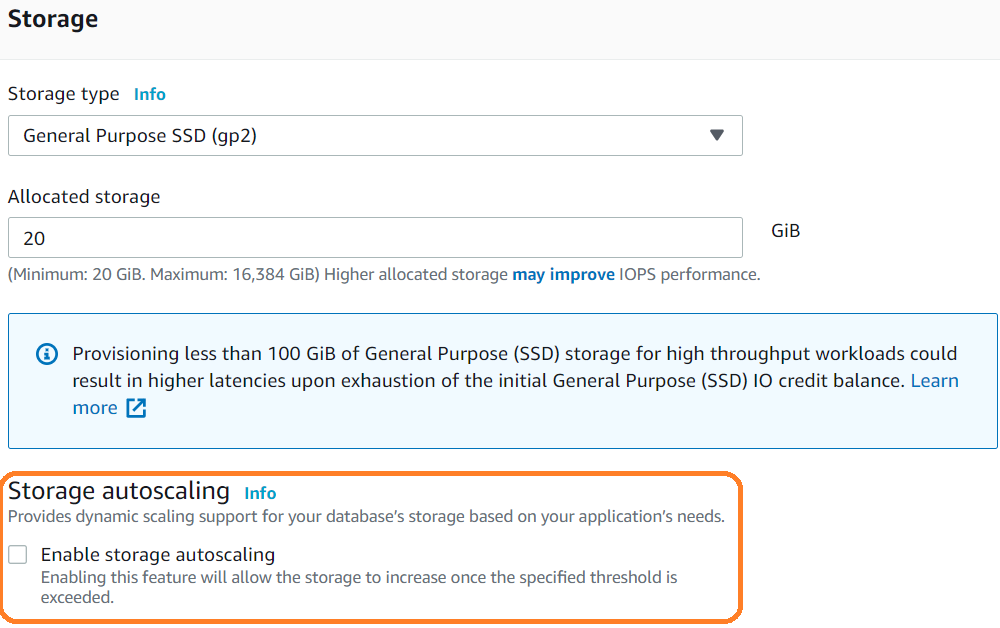


To take manual snapshot

Dashboard🡪databases🡪select your database🡪Actions🡪**take snapshot**



Storage type and **dynamic auto scaling**



Deletion process

**Note: You cannot delete directly.**

**Step 1:** Select your database 🡪 Modify 🡪 Scroll down uncheck Delete **protection** 🡪 continue 🡪 **Modify** DB Instance

**Step** 2: Select your database 🡪 Actions 🡪 delete 🡪 **uncheck “create final snapshot”**

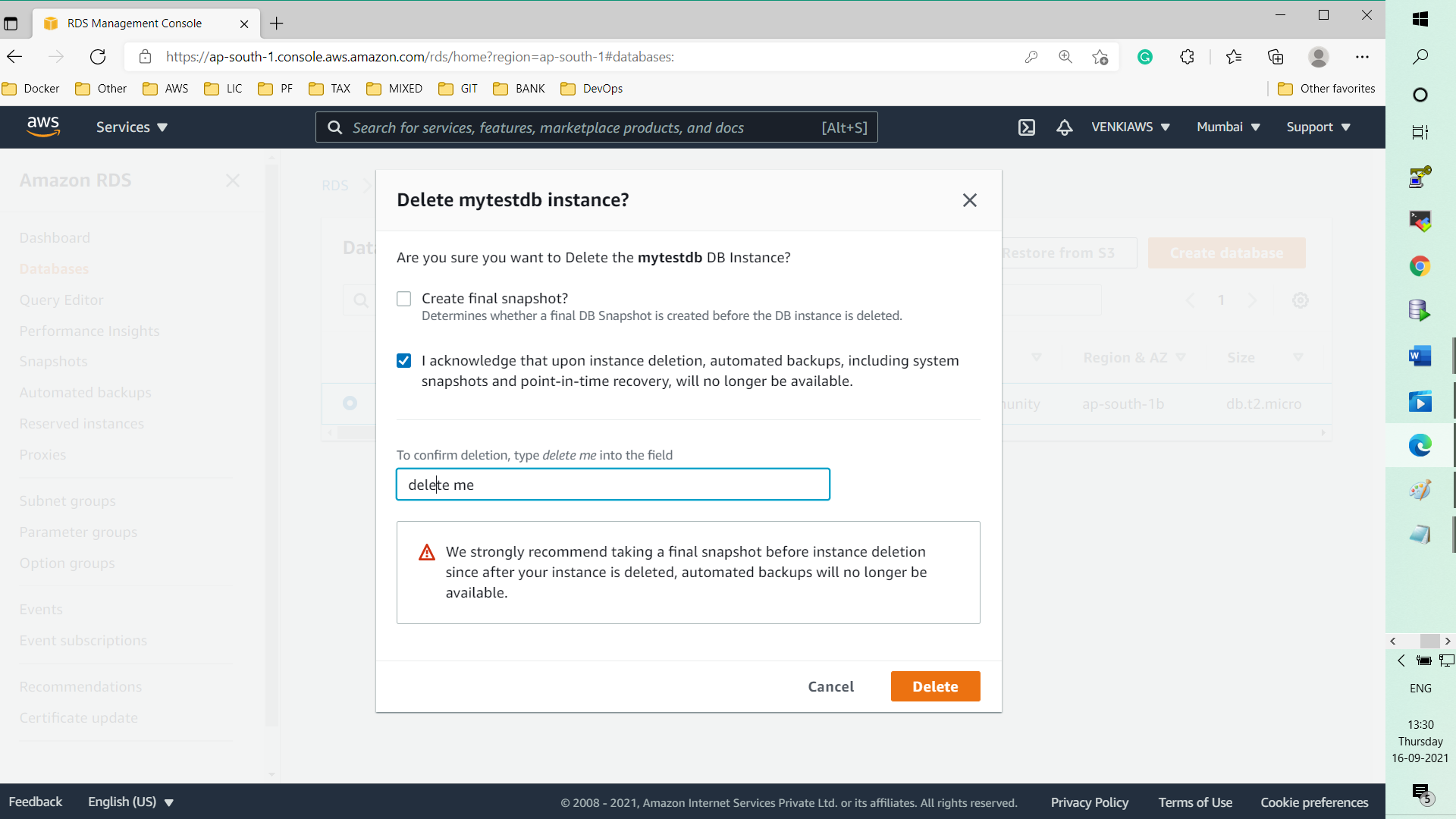
Select **🡪 I Acknowledge**

Type 🡪 delete me

**Delete**

**It takes 3-4 min to get delete from the dashboard.**

snapshot will also be deleted automatically.



Connect mysql from EC2 instance

* **Install mysql in EC2 isntance** 
  + **$ yum install mysql -y**
  + **$ mysql -h <<db-enpoint-url>> -P 3306 -u <<user-name>> -p**
  + **$ enter-password (enter)**

RDS Aurora

* Aurora is proprietary technology from AWS (not open sourced)
* Postgres and MYSQL are both supported as Aurora DB (that means your drivers will work as if Aurora was a Postgres or MYSQL database)
* Aurora is "AWS Cloud Optimized" and claims 5x performance improvement over MYSQL on RDS, over 3x the performance of Postgres on RDS
* Aurora can have 15 replicas while MYSQL has 5, and the replication process is faster
* Failover in Aurora is instantaneous. Its HA native
* Aurora costs more than RDS (20% more) -but is more efficient