

Cycle 08 AWS Homework

1. Create a Multi-AZ MySQL Database Instance

Objective: Deploy a MySQL RDS instance with Multi-AZ to ensure high availability.

Steps:

- In the AWS Console, navigate to RDS → Create Database.
- Select MySQL as the engine.
- Choose "Production" template.
- Under "Availability & durability," select "Multi-AZ DB instance" deployment.
- Specify instance details (DB size, instance class, storage).
- Proceed with default security groups, or configure as required.
- Launch the instance; AWS automatically provisions a standby replica in a different Availability Zone.

Verification (Failover Testing):

- Connect to the DB using endpoint details.
- In the RDS Console, perform a manual failover (Actions → Failover).
- Observe brief downtime during automated failover; new primary assumes workload.
- Applications using the endpoint experience minimal impact since DNS record points to the new primary.

Takeaway: Multi-AZ deployments improve resilience, ensuring automated failover and minimal downtime for the application.

Screenshot of the "Create database" wizard in the AWS RDS console. The current step is "Choose a database creation method".

Choose a database creation method

- Standard create: You set all of the configuration options, including ones for availability, security, backups, and maintenance.
- Easy create: Use recommended best-practice configurations. Some configuration options can be changed after the database is created.

Engine options

Engine type: [Info](#)

<input type="radio"/> Aurora (MySQL Compatible)	<input type="radio"/> Aurora (PostgreSQL Compatible)	<input checked="" type="radio"/> MySQL
<input type="radio"/> PostgreSQL	<input type="radio"/> MariaDB	<input type="radio"/> Oracle
<input type="radio"/> Microsoft SQL Server	<input type="radio"/> IBM Db2	

IBM Db2

CloudShell Feedback Rain warning In effect © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 20:58 16-08-2025

Screenshot of the "Create database" wizard in the AWS RDS console. The current step is "Templates".

Templates

Choose a sample template to meet your use case.

- Production: Use defaults for high availability and fast, consistent performance.
- Dev/Test: This instance is intended for development use outside of a production environment.
- Free tier: Use RDS Free Tier to develop new applications, test existing applications, or gain hands-on experience with Amazon RDS. [Info](#)

Availability and durability

Deployment options: [Info](#)

Choose the deployment option that provides the availability and durability needed for your use case. AWS is committed to a certain level of uptime depending on the deployment option you choose. Learn more in the [Amazon RDS service level agreement \(SLA\)](#).

- Multi-AZ DB cluster deployment (3 instances): Creates a primary DB instance with two readable standbys in separate Availability Zones. This setup provides:
 - 99.95% uptime
 - Redundancy across Availability Zones
 - Increased read capacity
 - Reduced write latency
- Multi-AZ DB instance deployment (2 instances): Creates a primary DB instance with a non-readable standby instance in a separate Availability Zone. This setup provides:
 - 99.95% uptime
 - Redundancy across Availability Zones
- Single-AZ DB instance deployment (1 instance): Creates a single DB instance without standby instances. This setup provides:
 - 99.5% uptime
 - No data redundancy

CloudShell Feedback Rain warning In effect © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 20:58 16-08-2025

Aurora and RDS > Create database

DB instance identifier [Info](#)
Type a name for your DB instance. The name must be unique across all DB instances owned by your AWS account in the current AWS Region.
chetandb1
The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 63 alphanumeric characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

Credentials Settings

Master username [Info](#)
Type a login ID for the master user of your DB instance.
chetandb1
1 to 16 alphanumeric characters. The first character must be a letter.

Credentials management
You can use AWS Secrets Manager or manage your master user credentials.

Managed in AWS Secrets Manager - most secure
RDS generates a password for you and manages it throughout its lifecycle using AWS Secrets Manager.

Self managed
Create your own password or have RDS create a password that you manage.

Auto generate password
Amazon RDS can generate a password for you, or you can specify your own password.

Master password [Info](#)

Minimum constraints: At least 8 printable ASCII characters. Can't contain any of the following symbols: / * @

Confirm master password [Info](#)

Actions ▾

- Group resources
- Modify**
- Actions ▾
- Create database ▾

Stop temporarily
Reboot
Delete
Set up EC2 connection
Set up Lambda connection
Migrate data from EC2 database - new
Create read replica
Create Aurora read replica
Create blue/green deployment
Promote
Convert to Multi-AZ deployment
Take snapshot
Restore to point in time
Migrate snapshot
Create zero-ETL integration
Create RDS Proxy
Create ElastiCache cluster

Aurora and RDS > Databases

Databases (1)

DB identifier	Status	Role
chetandb1	Available	Instance

Actions ▾

- Group resources
- Modify**
- Actions ▾
- Create database** ▾

Stop temporarily
Reboot
Delete
Set up EC2 connection
Set up Lambda connection
Migrate data from EC2 database - new
Create read replica
Create Aurora read replica
Create blue/green deployment
Promote
Convert to Multi-AZ deployment
Take snapshot
Restore to point in time
Migrate snapshot
Create zero-ETL integration
Create RDS Proxy
Create ElastiCache cluster

2. Cost Optimization Exercise

Objective: Reduce RDS costs by using a T3.micro instance class and gp3 storage.

Steps:

- Identify an existing RDS instance.
- Modify the instance: change the class to **db.t3.micro** (burstable, low-cost for dev/test).
- Change storage type to **gp3** SSD, offering lower price per GiB and flexible IOPS configuration.
- Apply changes during the next maintenance window or immediately if downtime is acceptable.
- Use the AWS Pricing Calculator to compare costs between gp3 and Provisioned IOPS (io2).

Findings:

- T3.micro + gp3 significantly lower monthly costs for small or development workloads.
- For I/O-intensive production workloads, io2 may be needed despite higher costs.

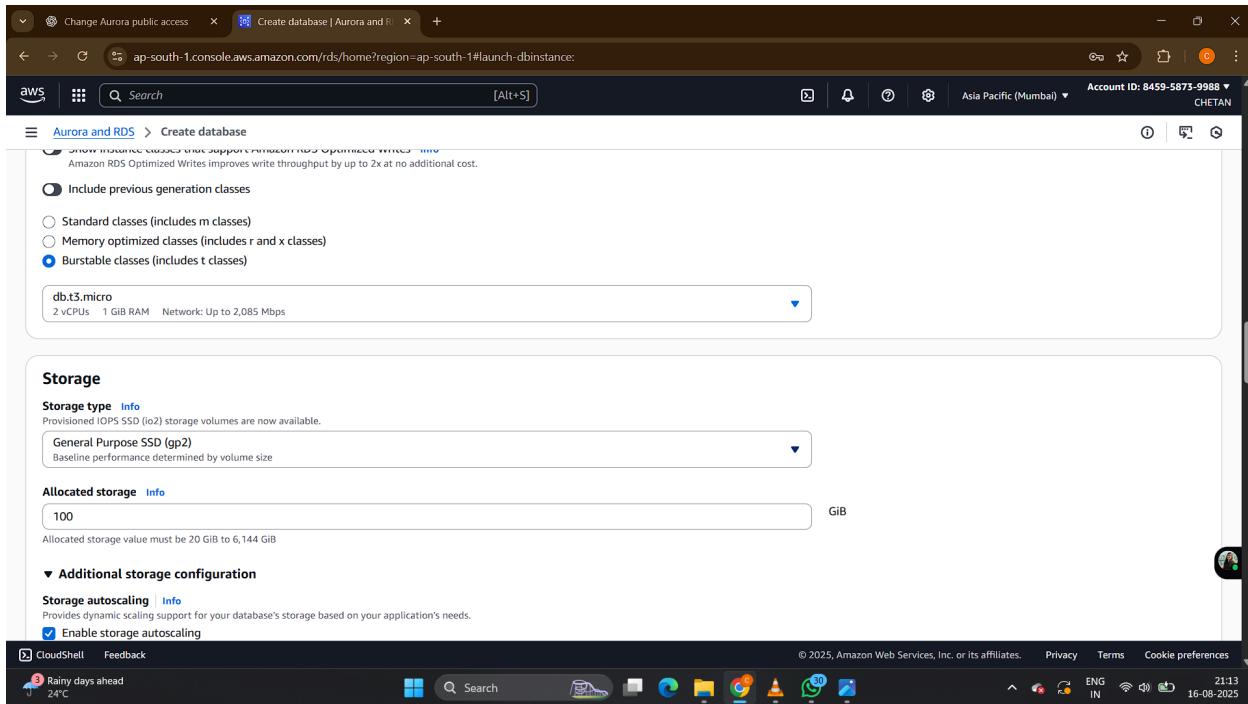
The screenshot shows the 'Create database' wizard in the AWS RDS console. In the 'DB instance class' section, 'db.m7g.large (supports Amazon RDS Optimized Writes)' is selected. Under 'Storage', 'Provisioned IOPS SSD (io2)' is chosen as the storage type. The 'Allocated storage' field is set to 400 GB. The top navigation bar shows the URL as 'ap-south-1.console.aws.amazon.com/rds/home?region=ap-south-1#launch-dbinstance:' and the account information as 'Account ID: 8459-5873-9988 CHETAN'.

The screenshot shows the 'Create database' wizard in the AWS RDS console. The current step is 'DB subnet group'. It displays a single available VPC subnet group named 'default-vpc-00dcf4d69d6b4f5d' which contains two subnets across two availability zones. The 'Public access' section is set to 'Yes', allowing public IP access. Under 'VPC security group (firewall)', the 'Choose existing' option is selected, showing the 'default' security group. The 'RDS Proxy' section indicates it is 'In effect'. The top navigation bar shows the URL as 'ap-south-1.console.aws.amazon.com/rds/home?region=ap-south-1#launch-dbinstance:'.

The screenshot continues through the 'Create database' wizard. The next step is 'IAM role', where it's noted that a service-linked role is used for publishing logs to CloudWatch Logs. The 'Additional configuration' step follows, mentioning options like encryption, backup, and CloudWatch Logs. Below this, the 'Estimated monthly costs' table provides a breakdown of charges:

DB instance	349.67 USD
Storage	104.80 USD
Provisioned IOPS	62.00 USD
Total	1081.47 USD

A note states that the billing estimate is based on on-demand usage and does not include costs for backup storage or data transfer. The user is encouraged to use the AWS Simple Monthly Calculator. A reminder at the bottom says they are responsible for third-party products. The 'Create database' button is visible in the bottom right corner. The top navigation bar remains consistent with the previous screenshot.



3. Proper Database Deletion

Objective: Safely delete a DB instance without incurring snapshot storage costs.

Steps:

- In the RDS Console, select the test instance.
- Choose "Delete."
- When prompted, **uncheck** the option to create a final snapshot before deletion.
- Confirm deletion after acknowledging the warning.

Best Practice: Only skip the final snapshot if the data is not needed. For production/critical data, always take a final snapshot for disaster recovery.

RDS is creating blue/green deployment bg-deployment-1.
Set up in progress, click on view details to see all details

Delete blue/green deployment?

Are you sure you want to delete the **bg-deployment-1**?

Delete the green databases in this blue/green deployment
Select to delete the blue/green deployment and the databases in the green environment. The databases in the blue environment aren't changed or deleted.

Type **delete me** to permanently delete this blue/green deployment
delete me

Cancel **Delete**

Region ...	Size	Recom...	CPU
ap-south-1a	db.t3.micro	5.3	
ap-south-1b	db.t3.micro	3.0	

Summary of modifications
You are about to submit the following modifications. Only values that will change are displayed. Carefully verify your changes and click Modify DB Instance.

Attribute	Current value	New value
Delete protection	Enabled	Disabled

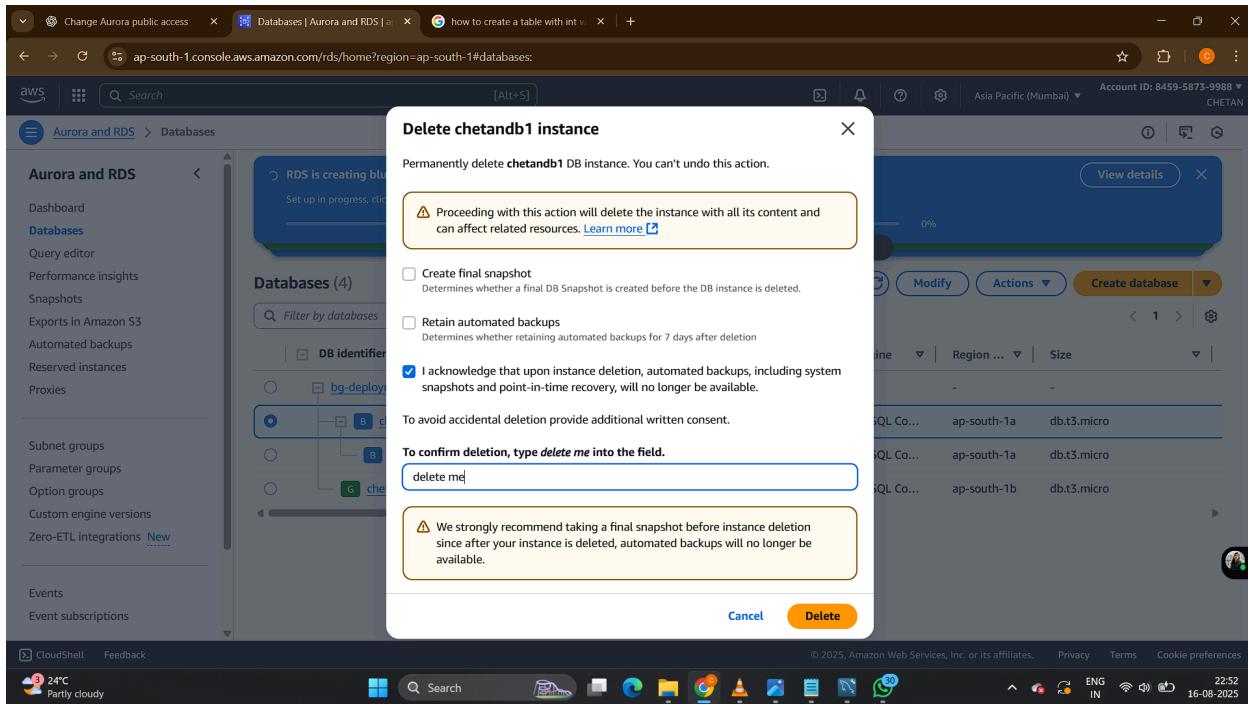
Schedule modifications

When to apply modifications

- Apply during the next scheduled maintenance window
Current maintenance window: August 18, 2025 17:52 - 18:22 (UTC+5:50)
- Apply immediately
The modifications in this request and any pending modifications will be asynchronously applied as soon as possible, regardless of the maintenance window setting for this database instance.

This modification won't be applied to the green database chetandb1-green-ppt0j associated with this blue database. To change chetandb1-green-ppt0j, modify it directly.

Cancel **Back** **Modify DB instance**



4. Read Replica Behavior Testing

Objective: Understand RDS read replicas' roles and restrictions.

Steps:

- Create a read replica for an existing MySQL RDS instance.
- Connect to the replica endpoint.
- Attempt to execute write operations (INSERT, UPDATE).
- Expected error: MySQL denies write access on a read replica, confirming read-only restrictions.

Findings: Read replicas offload read traffic, enhance scalability, and cannot be used for direct writes.

Create read replica

You are creating a replica DB instance from a source DB instance. This new DB instance will have the source DB instance's DB security groups and DB parameter groups.

Settings

Replica source
Source DB instance identifier
chetandb1
Role: Instance

DB instance identifier
This is the unique key that identifies a DB instance. This parameter is stored as a lowercase string (for example, mydbinstance).
chetadb1-readreplica

Instance configuration
The DB instance configuration options below are limited to those supported by the engine that you selected above.

DB instance class | [Info](#)

Hide filters

Include previous generation classes
 Standard classes (includes m classes)
 Memory optimized classes (includes r and x classes)
 Burstable classes (includes t classes)

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 24°C Partly cloudy ENG IN 21:36 16-08-2025

AWS KMS key | [Info](#)

Enter a key ARN

Amazon Resource Name (ARN)
arn:aws:kms:ap-south-1:845958739988:key/a7f7d5ec-eb61-4bd7-81ba-6a237f3241ba
Example: arn:aws:kms:<region>:<accountID>/key/<key-id>

Account
845958739988

KMS key ID
a7f7d5ec-eb61-4bd7-81ba-6a237f3241ba

Maintenance
Auto minor version upgrade [Info](#)

Enable auto minor version upgrade
Enabling auto minor version upgrade will automatically upgrade your database minor version. For limitations and more details, see Automatically upgrading the minor engine version [documentation](#).

Enable deletion protection
Protects the database from being deleted accidentally. While this option is enabled, you can't delete the database.

Cancel Create read replica

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 24°C Partly cloudy ENG IN 21:36 16-08-2025

Change Aurora public access Databases | Aurora and RDS | +

ap-south-1.console.aws.amazon.com/rds/home?region=ap-south-1#databases:

Aurora and RDS > Databases

Aurora and RDS

Dashboard
Databases
Query editor
Performance insights
Schemas
Exports in Amazon S3
Automated backups
Reserved instances
Proxies

Subnet groups
Parameter groups
Option groups
Custom engine versions
Zero-ETL integrations [New](#)

Events
Event subscriptions

CloudShell Feedback

24°C Partly cloudy

Creating replica chetadb1-readreplica in Asia Pacific (Mumbai)
Your database might take a few minutes to launch.

Databases (2)

DB identifier	Status	Role	Engine	Region ...	Size
chetadb1	Available	Primary	MySQL Co...	ap-south-1a	db.t3.micro
chetadb1-readreplica	Creating	Replica	MySQL Co...	ap-south-1a	db.t3.micro

Group resources [Modify](#) Actions [Create database](#)

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

CloudShell Feedback

24°C Partly cloudy

21:41 16-08-2025

Change Aurora public access chetadb1 - Create Read Replica +

ap-south-1.console.aws.amazon.com/rds/home?region=ap-south-1#create-read-replica:id=chetadb1;is-cluster=false;isHermesDB=false

Aurora and RDS > Databases > Create read replica

Create read replica

You are creating a replica DB instance from a source DB instance. This new DB instance will have the source DB instance's DB security groups and DB parameter groups.

Settings

Replica source
Source DB instance identifier
chetadb1
Role: Primary

DB instance identifier
This is the unique key that identifies a DB instance. This parameter is stored as a lowercase string (for example, mydbinstance).
chetadb1-singlezone-readreplica

Instance configuration

The DB instance configuration options below are limited to those supported by the engine that you selected above.

DB instance class [Info](#)
▼ Hide filters
Include previous generation classes

CloudShell Feedback

24°C Partly cloudy

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

CloudShell Feedback

24°C Partly cloudy

21:41 16-08-2025

Change Aurora public access chetandb1 - Create Read Replica

ap-south-1.console.aws.amazon.com/rds/home?region=ap-south-1#create-read-replica:id=chetandb1;is-cluster=false;isHermesDB=false

Aurora and RDS > Databases > Create read replica

Additional storage configuration

Availability

Deployment options [Info](#)

The following deployment options are limited to those supported by the engine.

- Multi-AZ DB cluster deployment (3 instances)**
Creates a primary DB instance with two readable standbys in separate Availability Zones. This setup provides:
 - 99.95% uptime
 - Redundancy across Availability Zones
 - Increased read capacity
 - Reduced write latency
- Multi-AZ DB instance deployment (2 instances)**
Creates a primary DB instance with a non-readable standby instance in a separate Availability Zone. This setup provides:
 - 99.95% uptime
 - Redundancy across Availability Zones
- Single-AZ DB instance deployment (1 instance)**
Creates a single writer DB instance with no reader instances. This setup provides:
 - 99.5% uptime
 - No data redundancy

CloudShell Feedback 24°C Partly cloudy © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 21:41 16-08-2025

Change Aurora public access chetandb1 - Create Read Replica

ap-south-1.console.aws.amazon.com/rds/home?region=ap-south-1#create-read-replica:id=chetandb1;is-cluster=false;isHermesDB=false

Aurora and RDS > Databases > Create read replica

AWS KMS key [Info](#)

Enter a key ARN

Amazon Resource Name (ARN)
arn:aws:kms:ap-south-1:845958739988:key/a7f7d5ec-eb61-4bd7-81ba-6a237f3241ba
Example: arn:aws:kms:<region>:<accountID>/key/<key-id>

Account
845958739988

KMS key ID
a7f7d5ec-eb61-4bd7-81ba-6a237f3241ba

Maintenance

Auto minor version upgrade [Info](#)

Enable auto minor version upgrade
Enabling auto minor version upgrade will automatically upgrade your database minor version. For limitations and more details, see [Automatically upgrading the minor engine version documentation](#).

Enable deletion protection
Protects the database from being deleted accidentally. While this option is enabled, you can't delete the database.

[Cancel](#) [Create read replica](#)

CloudShell Feedback 24°C Partly cloudy © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 21:41 16-08-2025

Screenshot of the AWS RDS console showing the creation of a new database.

Creating replica chetandb1-singlezone-readreplica in Asia Pacific (Mumbai)
Your database might take a few minutes to launch.

Databases (3)

DB identifier	Status	Role	Engine	Region ...	Size
chetandb1	Available	Primary	MySQL Co...	ap-south-1a	db.t3.micro
chetandb1-readreplica	Creating	Replica	MySQL Co...	ap-south-1a	db.t3.micro
chetandb1-singlezone-readreplica	Creating	Replica	MySQL Co...	-	db.t3.micro

Screenshot of MySQL Workbench showing the creation of a database and table.

Query 1

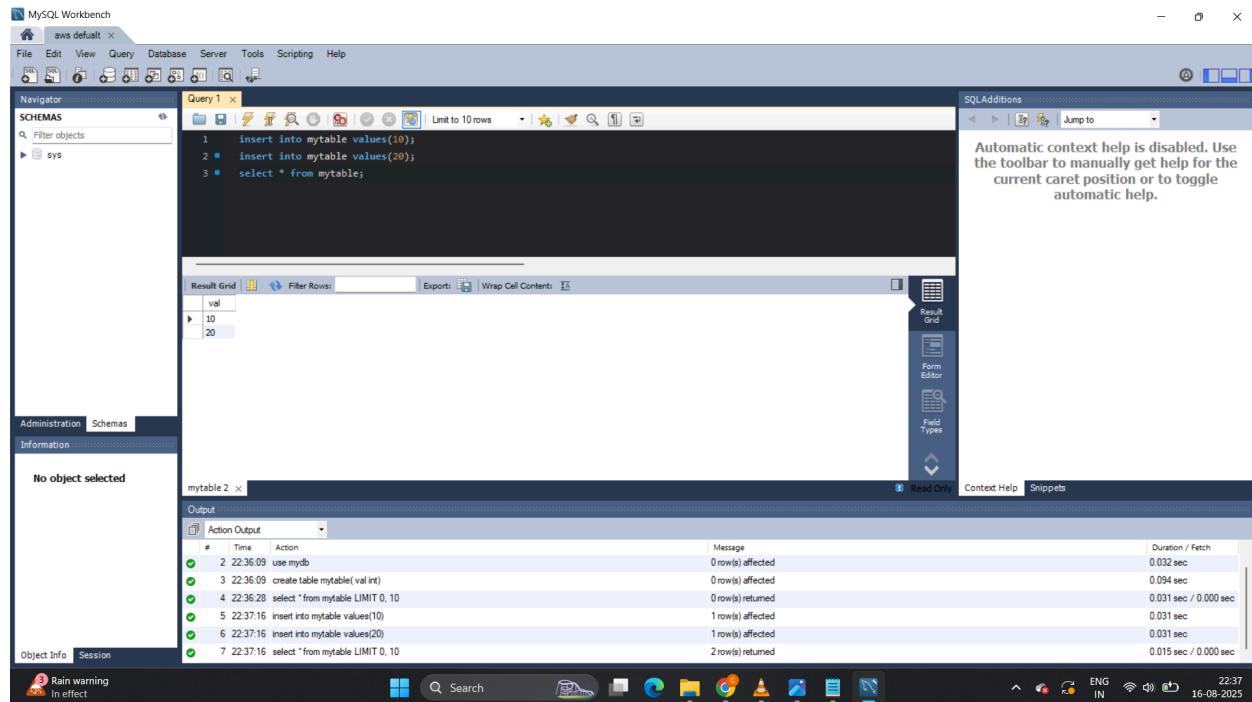
```

1 create database mydb;
2 use mydb;
3 create table mytable( val int);

```

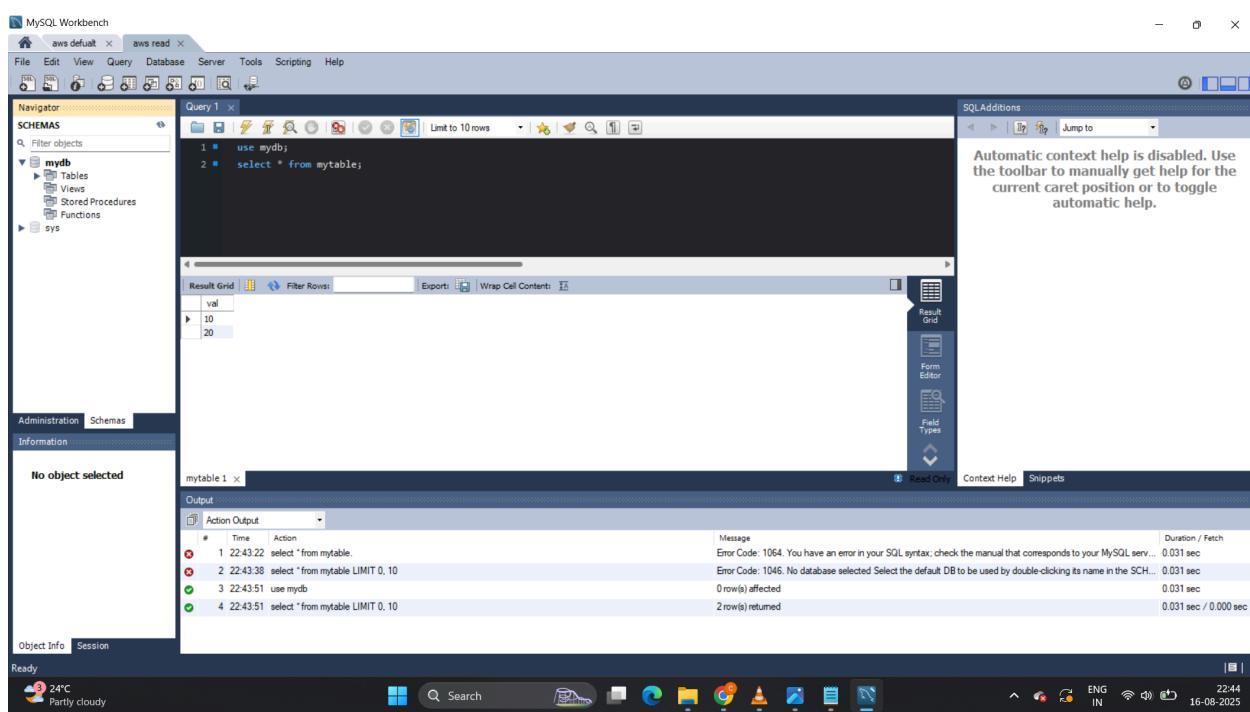
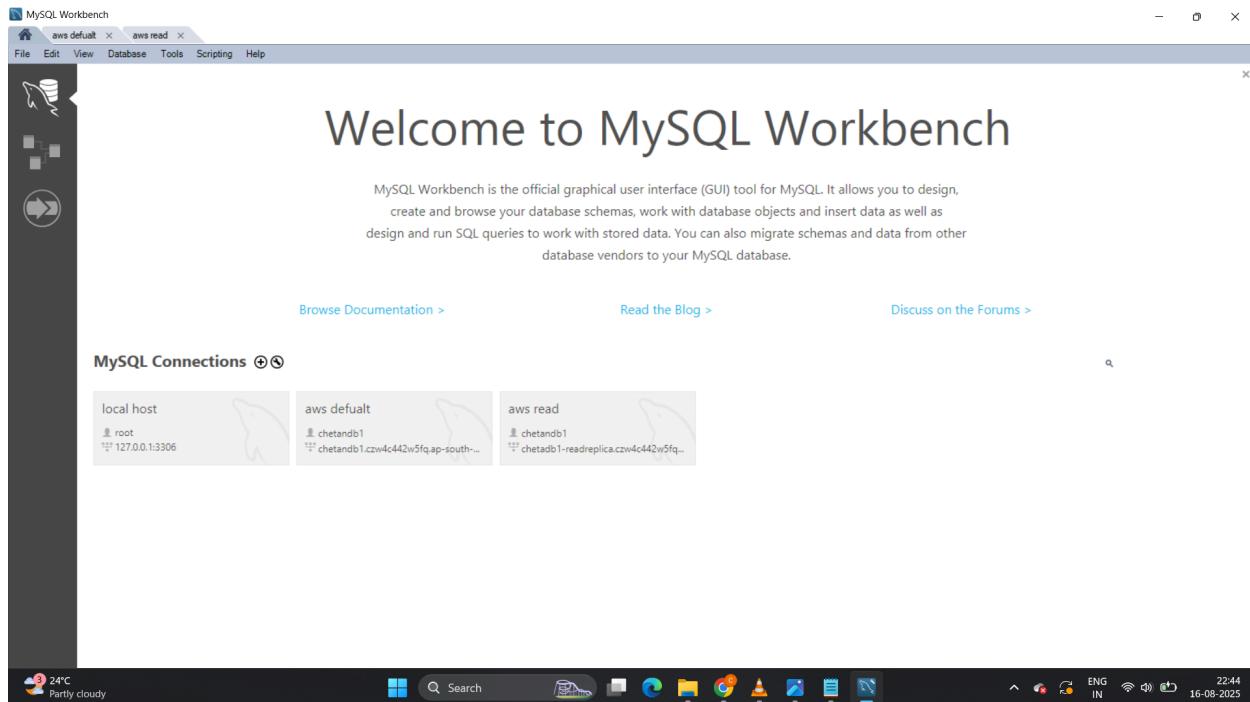
Output

#	Time	Action	Message	Duration / Fetch
1	22:36:09	create database mydb	1 row(s) affected	0.046 sec
2	22:36:09	use mydb	0 row(s) affected	0.032 sec
3	22:36:09	create table mytable(val int)	0 row(s) affected	0.094 sec



The screenshot shows the AWS RDS console for the 'chetandb1-readreplica' database. The left sidebar shows the 'Aurora and RDS' navigation menu. The main area displays the 'Related' section, listing the primary database 'chetandb1' and the replica 'chetandb1-readreplica'. The 'Connectivity & security' tab is selected, showing the following details:

- Endpoint copied**: chetandb1-readreplica.czv4c442w5f.q.ap-south-1.rds.amazonaws.com
- Port**: 3306
- Networking**:
 - Availability Zone: ap-south-1a
 - VPC: vpc-00dcef4d69d6b4f5d
 - Subnet group: default-vpc-00dcef4d69d6b4f5d
- Security**:
 - VPC security groups: default (sg-0eac963e1cef69496), Active
 - Publicly accessible: Yes
 - Certificate authority: rds-ca-rsa2048-g1



5. Multi-Region Read Replica Setup

Objective: Measure replication latency between AWS regions.

Steps:

- Create a cross-region read replica (e.g., source in Mumbai, replica in Singapore).
- After setup, make changes to the source DB.
- Monitor replica lag using CloudWatch metrics or the RDS console.

Finding: Replication introduces delay (from seconds to minutes) due to cross-region network latency. Cross-region replicas improve disaster recovery and support global application deployments.

6. Blue/Green Deployment Exploration

Objective: Investigate blue/green deployment strategies for RDS.

Research:

- RDS supports blue/green deployments for MySQL and PostgreSQL, allowing you to create a staging (green) environment from a production (blue) instance.
- Schema changes (e.g., DDL) are made in green while traffic remains on blue.
- Switchover is nearly instantaneous and with minimal risk.
- Alternatively, AWS Database Migration Service (DMS) can replicate data for migration or upgrades.

Benefit: Enables zero-downtime upgrades and schema changes with rollback capability.

The screenshot shows the AWS RDS console for the 'chetandb1' database. A modal window at the top right indicates 'Successfully deleted DB instance chetandb1-singlezone-readreplica'. The main interface displays two DB instances: 'chetandb1' (Primary, MySQL 8.0.42, ap-south-1a) and 'chetandb1-readreplica' (Replica, MySQL 8.0.42, ap-south-1a). On the right, a context menu for 'chetandb1' is open, listing options like 'Stop temporarily', 'Reboot', 'Delete', and 'Create blue/green deployment'. Below the instances, tabs for 'Connectivity & security', 'Monitoring', 'Logs & events', 'Configuration', and 'Zero-ETL integrations' are visible. The 'Connectivity & security' tab is selected, showing details for the endpoint and networking.

The screenshot shows the 'Create blue/green deployment' wizard. Step 1, 'Create blue/green deployment', is selected. It shows the 'Blue database identifiers' section with 'chetandb1' and 'chetadb1-readreplica' listed. The 'Blue/green deployment name' section shows 'bg-deployment-1' entered. Step 2, 'Review and confirm', is shown as the next step. The bottom of the screen shows the AWS navigation bar and system status.

Green instance storage configuration Green

Storage type Info
General Purpose SSD (gp2)
Baseline performance determined by volume size

Allocated storage Info
100 GiB
Allocated storage value must be 20 GiB to 65,536 GiB

ⓘ Green replicas inherit the allocated storage of the green primary DB instance, which is the same as the blue primary DB instance.

Storage configuration upgrade Info
Storage file system configuration upgrade
RDS recommends a storage file system configuration upgrade for your selected database instance.

ⓘ You are on the latest storage configuration.

Cancel Next

RDS is creating blue/green deployment **bg-deployment-1**.
Set up in progress, click on view details to see all details

chetandb1

DB identifier	Status	Role	Engine	Region	Size	Recommendations	CPU
bg-deployment-1	Provisioning	Blue/green...	-	-	-	-	-
chetandb1	Primary	MySQL Co...	ap-south-1a	db.t3.micro	4.2		
chetandb1-readreplica	Available	MySQL Co...	ap-south-1a	db.t3.micro	3.5		
chetandb1-green-ppt0j	Creating	MySQL Co...	-	db.t3.micro	-		

View details

Modify Actions

Connectivity & security Monitoring Logs & events Configuration Zero-ETL integrations Maintenance & backups Data

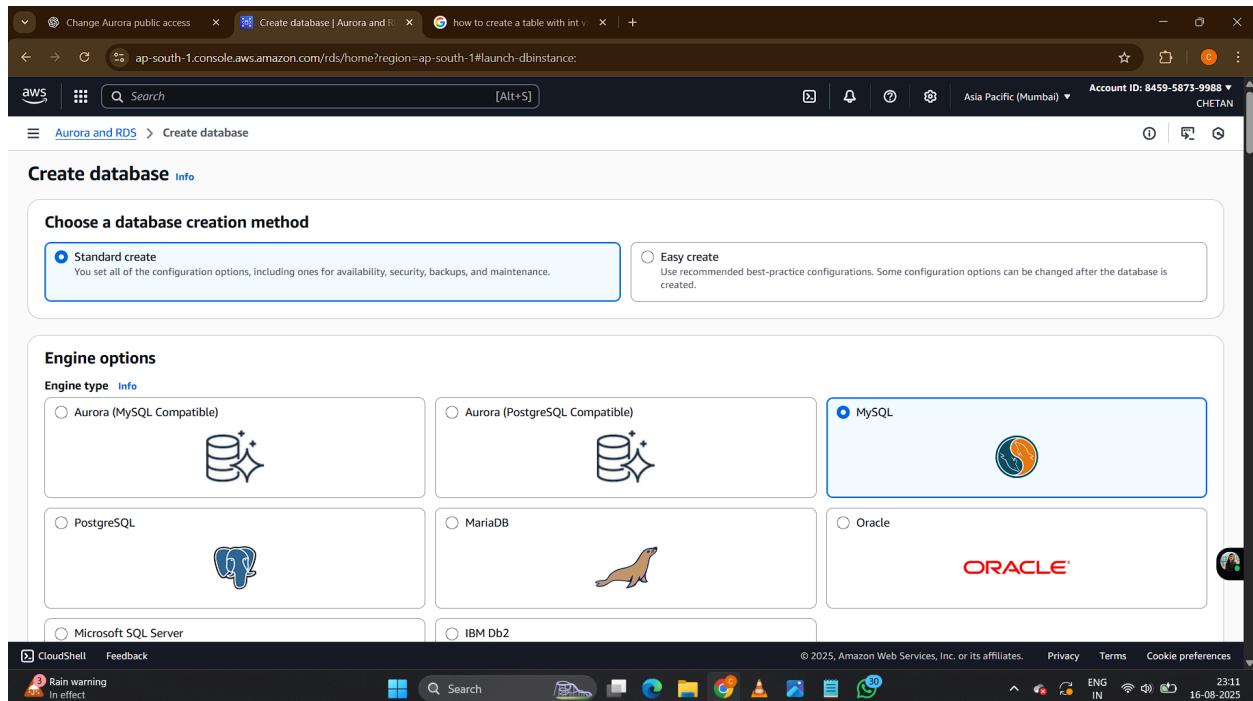
7. Multi-AZ DB Cluster Deployment

Objective: Compare new Multi-AZ DB clusters with traditional Multi-AZ instances.

Steps:

- Create a Multi-AZ DB cluster (one primary, two standby, supports Aurora-like failover).
- Compare performance metrics with standard Multi-AZ instance deployments.
- Clusters allow automatic failover, higher read throughput, and faster recovery.

Finding: DB Clusters offer improved fault tolerance and scalability for critical applications.



Screenshot of the AWS RDS Create Database page showing deployment options:

Availability and durability

Deployment options [Info](#)

Choose the deployment option that provides the availability and durability needed for your use case. AWS is committed to a certain level of uptime depending on the deployment option you choose. Learn more in the [Amazon RDS service level agreement \(SLA\)](#).

- Multi-AZ DB cluster deployment (3 instances)**

Creates a primary DB instance with two readable standbys in separate Availability Zones. This setup provides:

 - 99.95% uptime
 - Redundancy across Availability Zones
 - Increased read capacity
 - Reduced write latency
- Multi-AZ DB instance deployment (2 instances)**

Creates a primary DB instance with a non-readable standby instance in a separate Availability Zone. This setup provides:

 - 99.95% uptime
 - Redundancy across Availability Zones
- Single-AZ DB instance deployment (1 instance)**

Creates a single DB instance without standby instances. This setup provides:

 - 99.5% uptime
 - No data redundancy

Screenshot of the AWS RDS Create Database page showing configuration details:

DB cluster identifier

[Info](#)
Enter a name for your DB cluster. The name must be unique across all DB clusters owned by your AWS account in the current AWS Region.

The DB cluster identifier is case-insensitive, but is stored as all lowercase (as in "mydbcluster"). Constraints: 1 to 63 alphanumeric characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

Credentials Settings

Master username [Info](#)
Type a login ID for the master user of your DB cluster.

1 to 16 alphanumeric characters. The first character must be a letter.

Credentials management
You can use AWS Secrets Manager or manage your master user credentials.

Managed in AWS Secrets Manager - most secure
RDS generates a password for you and manages it throughout its lifecycle using AWS Secrets Manager.

Self managed
Create your own password or have RDS create a password that you manage.

Auto generate password
Amazon RDS can generate a password for you, or you can specify your own password.

Master password

[Info](#)

Password strength Neutral
Minimum constraints: At least 8 printable ASCII characters. Can't contain any of the following symbols: / * @

Confirm master password [Info](#)

The screenshot shows the 'Create database' configuration page for an Aurora DB instance. Under the 'Storage' section, the 'Allocated storage' is set to 100 GiB. The 'Storage type' dropdown is set to 'General Purpose SSD (gp3)'. A note indicates that 'Provisioned IOPS SSD (io2) storage volumes are now available.' Below the storage section, there are tabs for 'CloudShell' and 'Feedback'.

The screenshot continues the 'Create database' configuration. Under 'DB subnet group', it shows 'default-vpc-00decf4d69d6b4f5d' selected, which includes 2 Subnets, 2 Availability Zones. A warning message states: 'The DB subnets must be in 3 Availability Zones (AZs) for the Multi-AZ DB cluster. The current subnets are in 2 AZs (ap-south-1b, ap-south-1a). Add a subnet in a different AZ than the current subnets.' Under 'Public access', 'Yes' is selected, allowing public IP access. Under 'VPC security group (firewall)', 'Choose existing' is selected, with 'default' chosen from the dropdown. The bottom of the screen shows the standard AWS navigation bar with CloudShell, Feedback, and other account details.

The screenshot shows the AWS RDS console under the 'Aurora and RDS' section. A modal window titled 'Creating database chetandb1' is open, stating that the database might take a few minutes to launch. Below it, the 'Databases' list shows four entries:

DB identifier	Status	Role	Engine	Region	Size
chetandb1	Creating	Multi-AZ DB cluster	MySQL Co...	ap-south-1	3 instances
chetandb1-instance-1	Creating	Reader instance	MySQL Co...	ap-south-1c	db.c6gd.medium
chetandb1-instance-2	Creating	Reader instance	MySQL Co...	ap-south-1b	db.c6gd.medium
chetandb1-instance-3	Creating	Reader instance	MySQL Co...	ap-south-1a	db.c6gd.medium

The left sidebar includes links for Dashboard, Databases (selected), Query editor, 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, and Event subscriptions.

Additional Explorations

OLTP vs. OLAP Workload Simulation

Steps:

- Create an "orders" table (OLTP) and "sales_analytics" (OLAP).
- Populate with sample data.
- Set up read replicas to handle OLAP (analytical) queries.
- Measure query speeds: OLAP queries run faster on replicas, reducing primary load.

AWS Endpoint Testing

Steps:

- Connect to RDS Writer Endpoint: Perform read and write operations.
- Connect to Reader Endpoint: Attempt to write; expected error indicates it is read-only.

Cost Estimation Exercise

Steps:

- Use AWS Pricing Calculator to compare:
 - Single-AZ vs. Multi-AZ deployment costs.
 - db.t3.micro vs. db.m5.large monthly costs.
- Multi-AZ and larger instances incur higher costs but offer higher performance and resilience.

Automated Backup & Snapshot Management

Steps:

- Enable automatic backups, set retention to 7 days.
- Create a manual snapshot.
- Restore a new DB instance from the snapshot to test data recovery.

Failover Simulation in Multi-AZ

Steps:

- In the RDS Console, trigger a manual failover.
- Measure time for recovery and database availability post-failover.

Blue/Green Deployment Hands-On

Steps:

- Use AWS DMS or RDS Blue/Green Deployment to roll out a schema change (e.g., new table or column).
- Switchover with no downtime for client applications.

Key Learnings & Exam Preparation

- **Multi-AZ provides high availability** by deploying a standby in a separate AZ.
- **Read replicas** boost scalability but are read-only.

- **Cost optimization** involves selecting appropriate instance types and storage options.
- **Cross-region replication** is vital for global applications and disaster recovery.
- **Blue/green deployments and DMS** enable zero-downtime migrations and upgrades.
- **AWS Pricing Calculator** helps to estimate and control cloud spending.