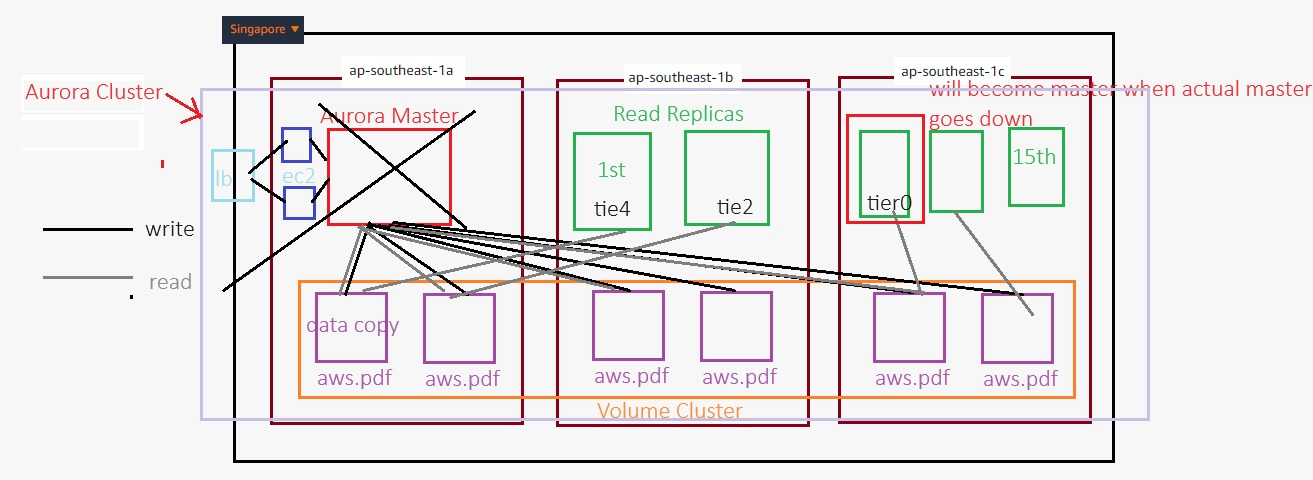
### **AIM : Deploying EC2 with RDS Aurora in Different AZs Using Free Tier**

#### **Objective**

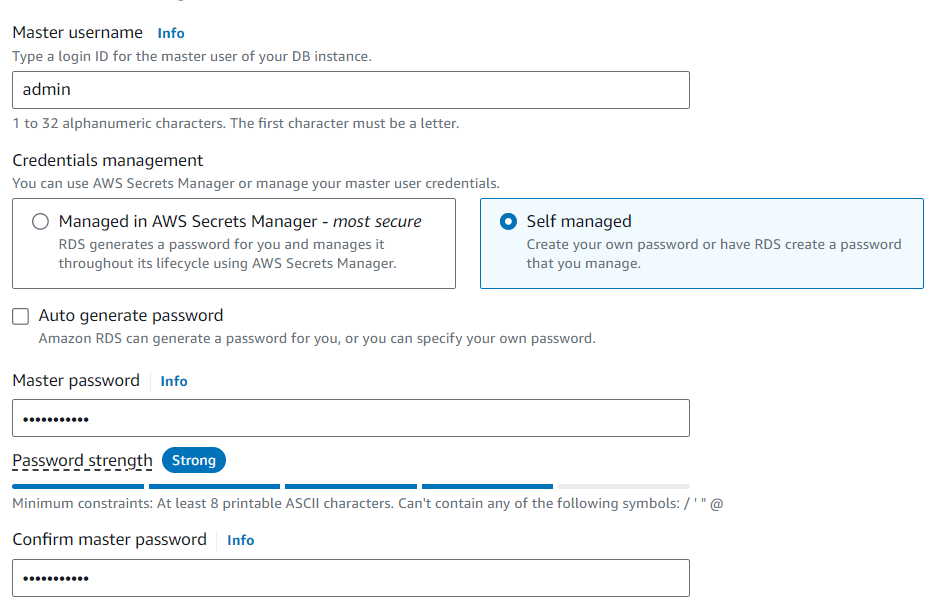
The objective of this documentation is to guide you through deploying an EC2 instance connected to an RDS Aurora database, both located in different Availability Zones (AZs). The setup will include a read replica for the RDS instance and enabling automated backups. The goal is to ensure high availability, data durability, and efficient read performance while staying within the AWS Free Tier.

#### **Architecture Flow Diagram**



1. **VPC Setup**:
   * Create a Virtual Private Cloud (VPC) with multiple subnets in different AZs.
   * Ensure one subnet is designated for the EC2 instance and another for the RDS Aurora instance.
2. **EC2 Instance Configuration**:
   * Launch an EC2 instance in one of the subnets within the VPC.
   * Configure security groups to allow communication between the EC2 instance and the RDS Aurora instance.
3. **RDS Aurora Instance Setup**:
   * Deploy an RDS Aurora instance in a different AZ from the EC2 instance.
   * Create a read replica in another AZ to improve read scalability and availability.
   * Enable automated backups for the RDS Aurora instance.
4. **Data Flow**:
   * The EC2 instance serves as the application server, interacting with the RDS Aurora instance for database operations.
   * The RDS Aurora instance processes write operations, while the read replica handles read queries to offload the main database.
5. **Security and Backup**:
   * Implement IAM roles and policies to manage access.
   * Enable automated backups and Multi-AZ failover for the RDS Aurora instance.

#### **Step-by-Step Implementation**

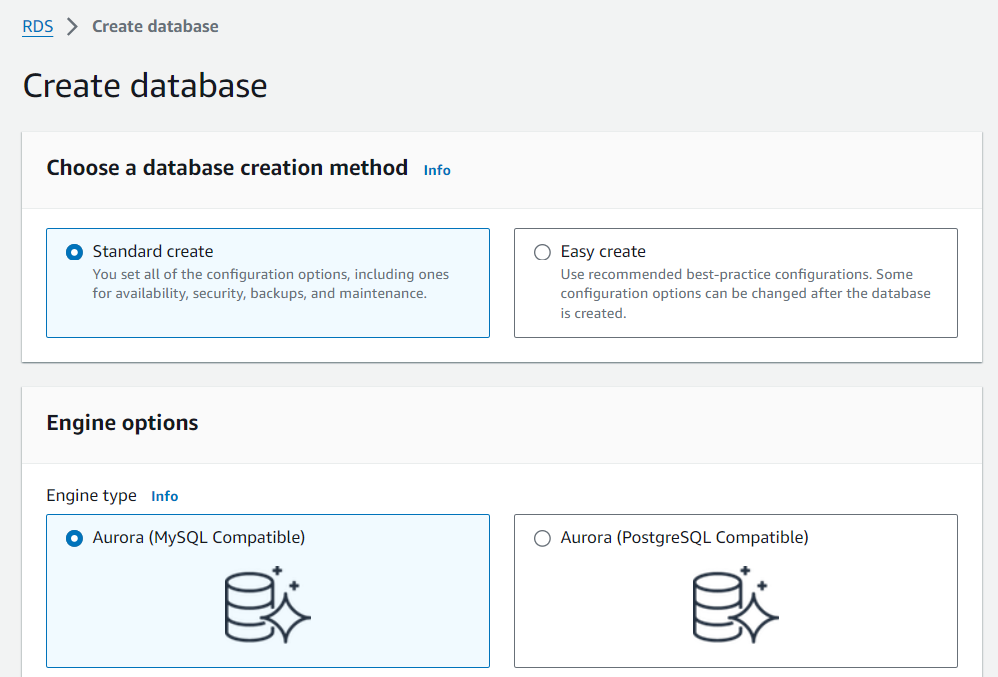
1. **Create a VPC**:
2. 

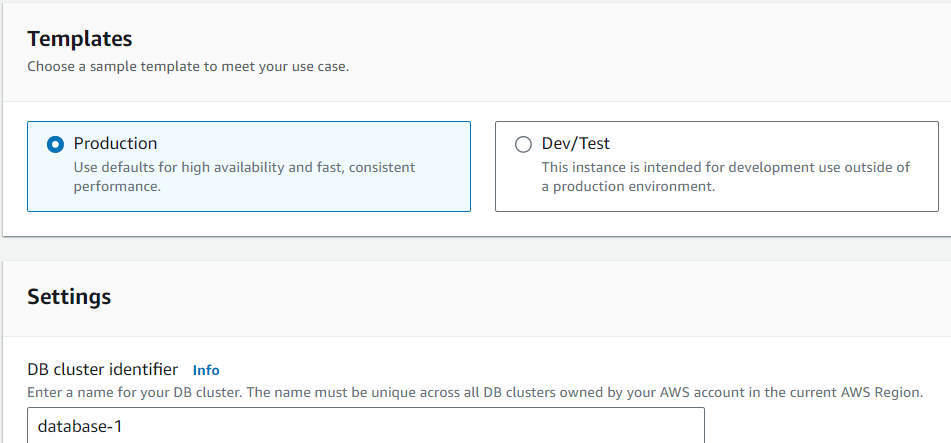
Navigate to the VPC dashboard and create a new VPC.

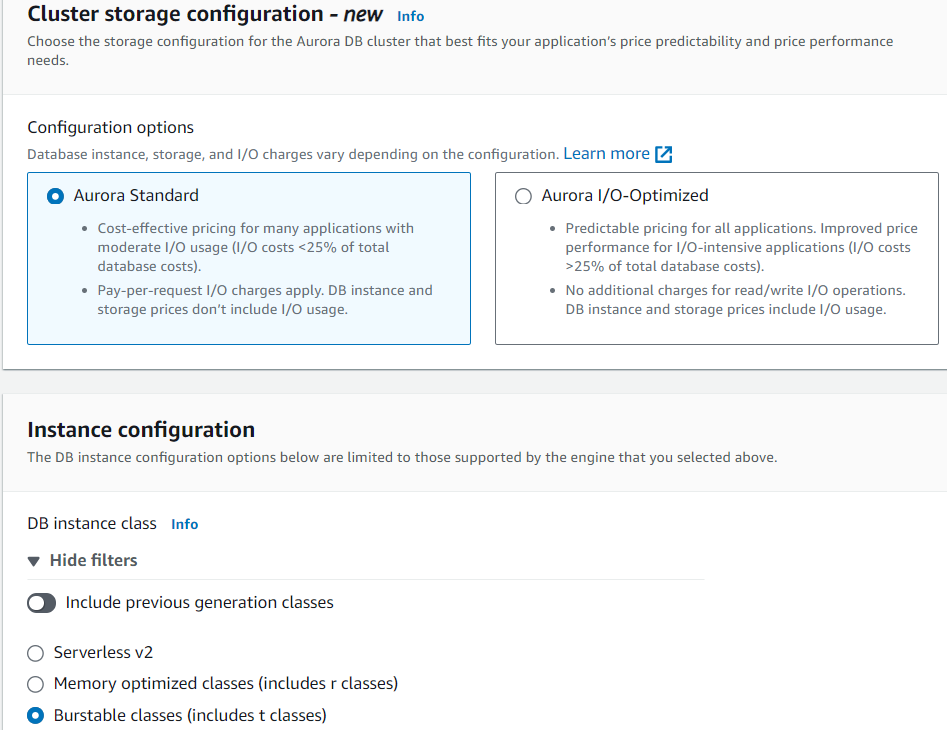
* + Add subnets in different AZs (e.g., us-east-1a, us-east-1b).
  + Create a route table and associate it with the subnets.

1. **Launch EC2 Instance**:
   * Go to the EC2 dashboard and launch a new instance.
   * Choose an AMI that is Free Tier eligible (e.g., Amazon Linux 2).
   * Place the instance in one of the subnets created in your VPC.
   * Configure a security group allowing SSH access and inbound connections from the RDS instance.
2. **Create RDS Aurora Instance**:
   * In the RDS dashboard, create a new Aurora database cluster.
3. Select a different AZ for the RDS instance compared to the EC2 instance.
4. Enable Multi-AZ deployment for high availability.
5. Configure the read replica in a separate AZ to handle read operations.

**4. Choose the option for "MySQL-compatible" Aurora.**

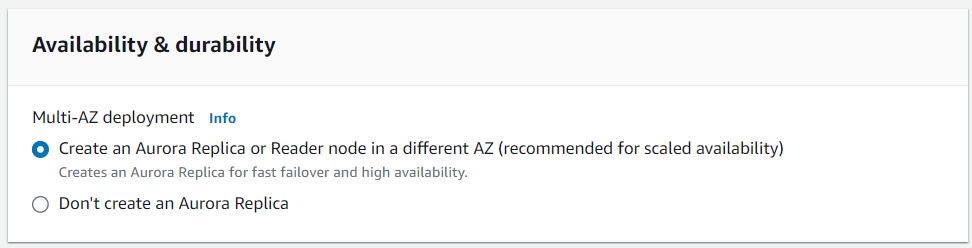


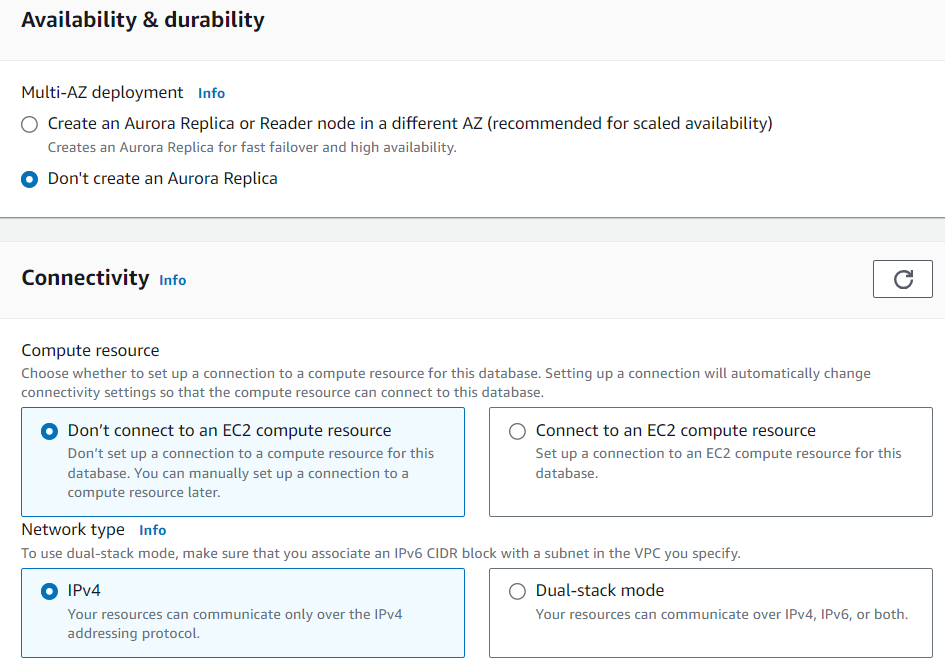


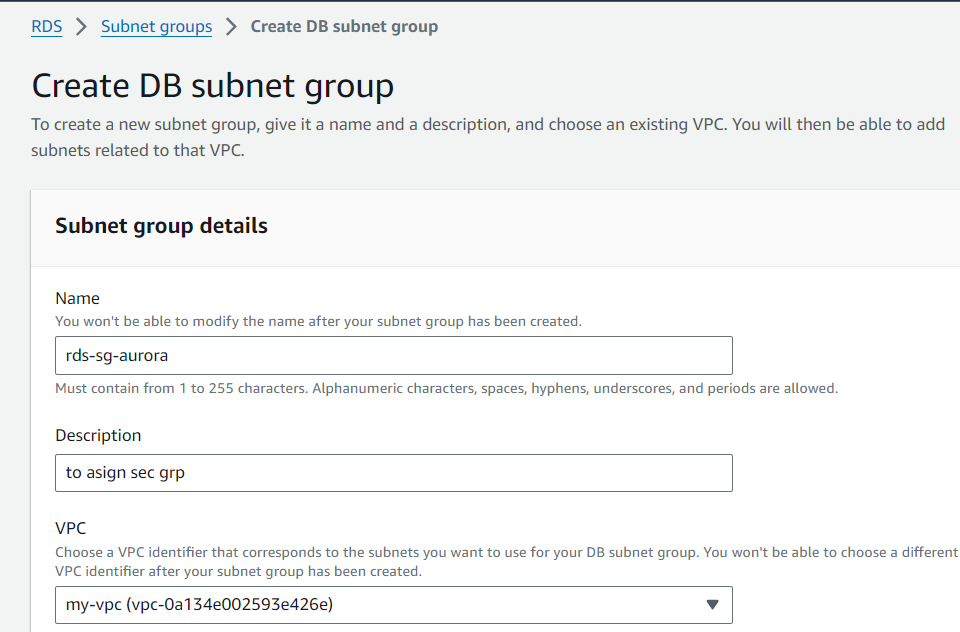


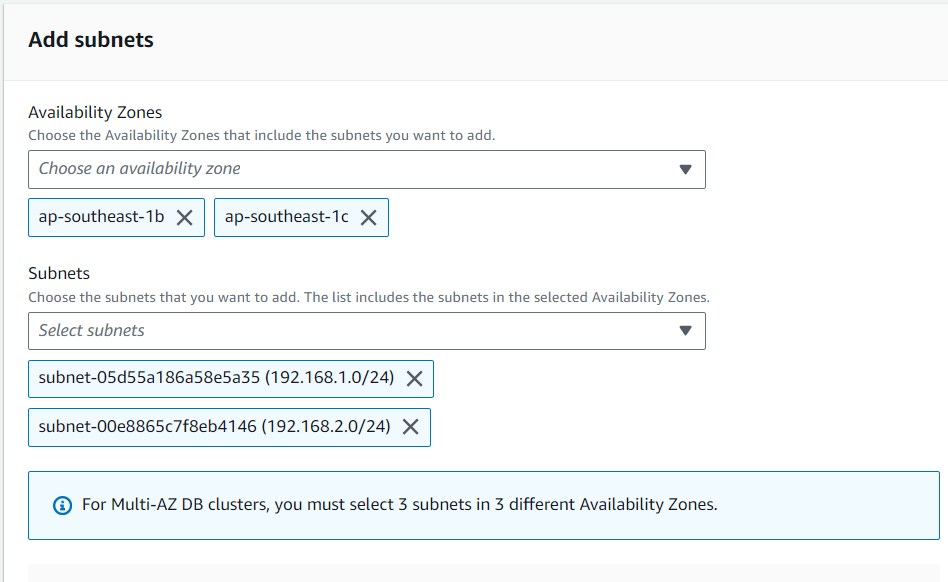
**Configure the read replica in a separate AZ to handle read operations.**

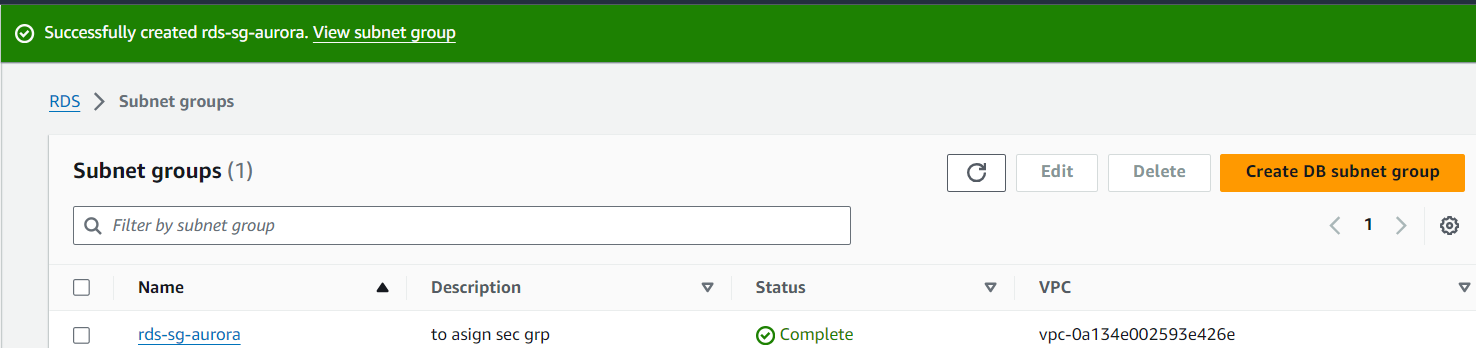
**Enable Multi-AZ deployment for high availability.**

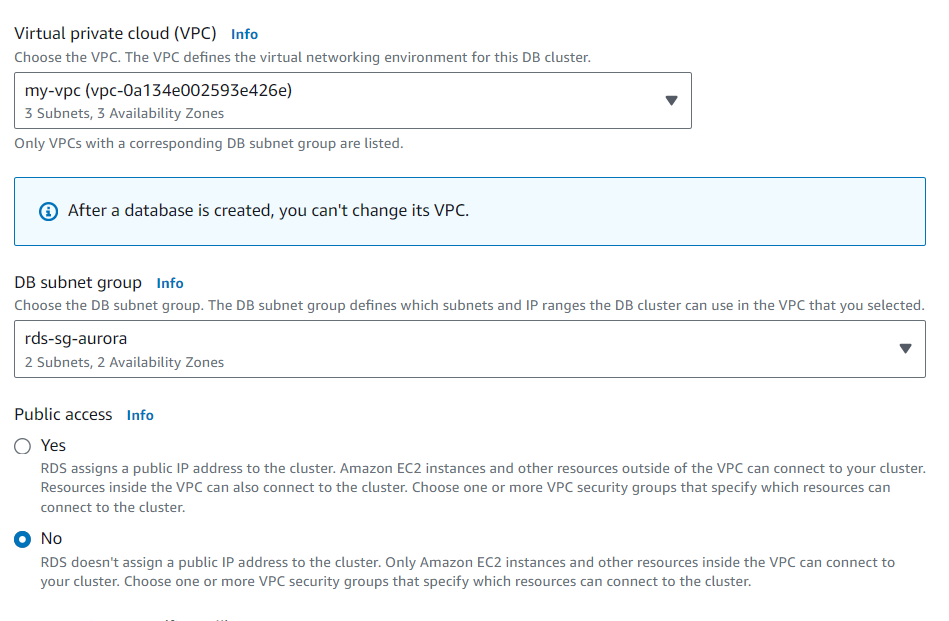


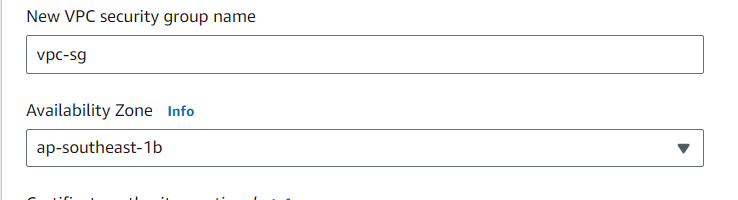
**Create new DB subnet group**

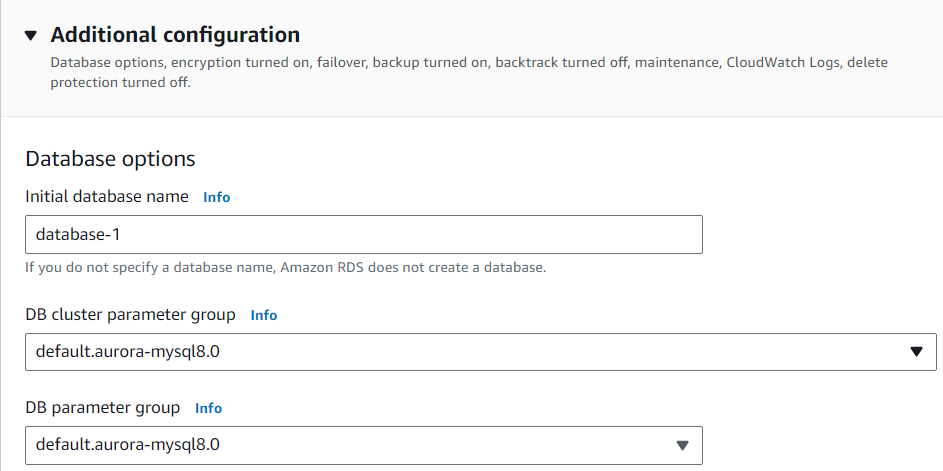
****

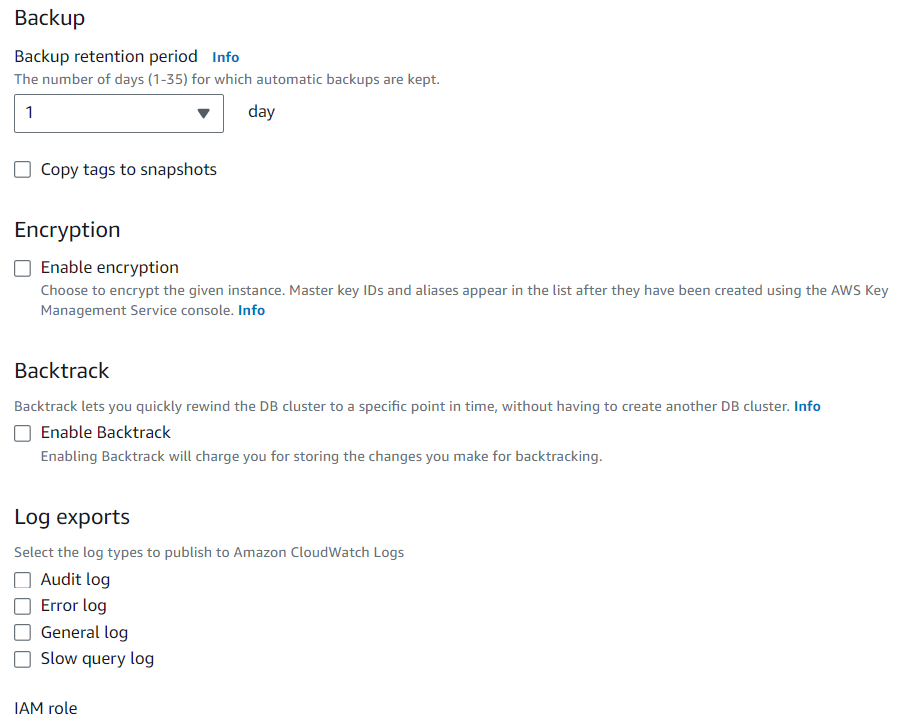
****

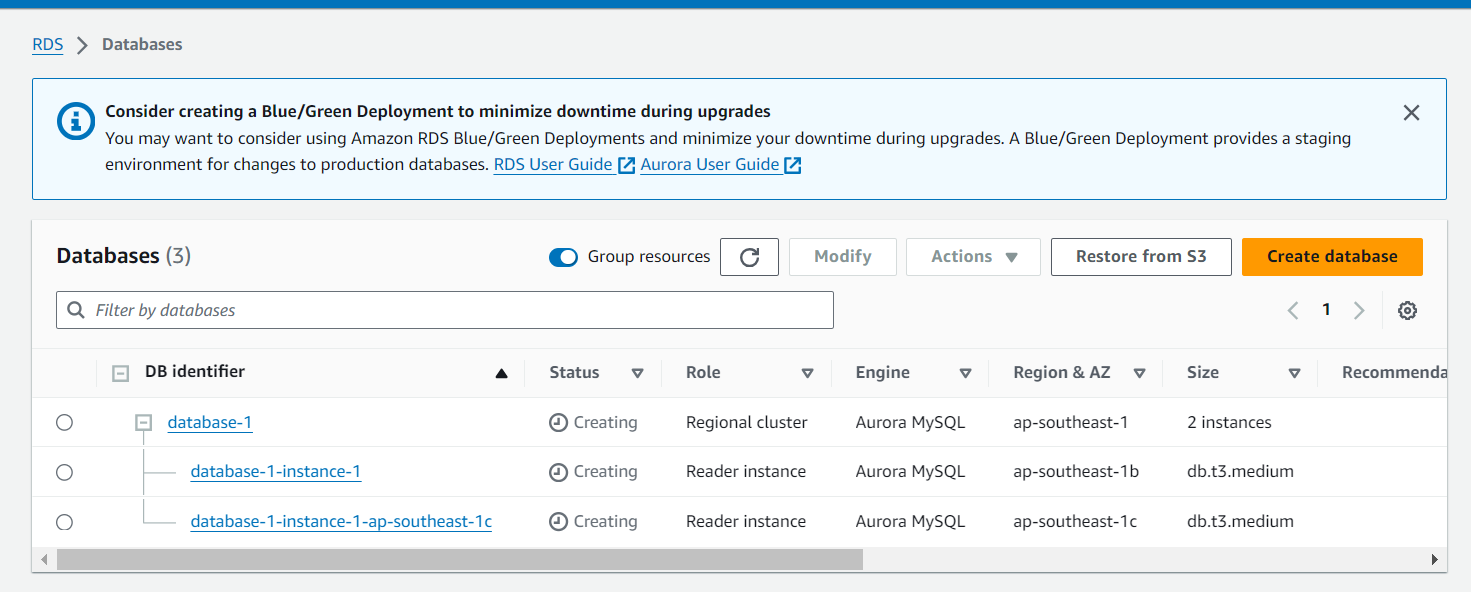
****

****

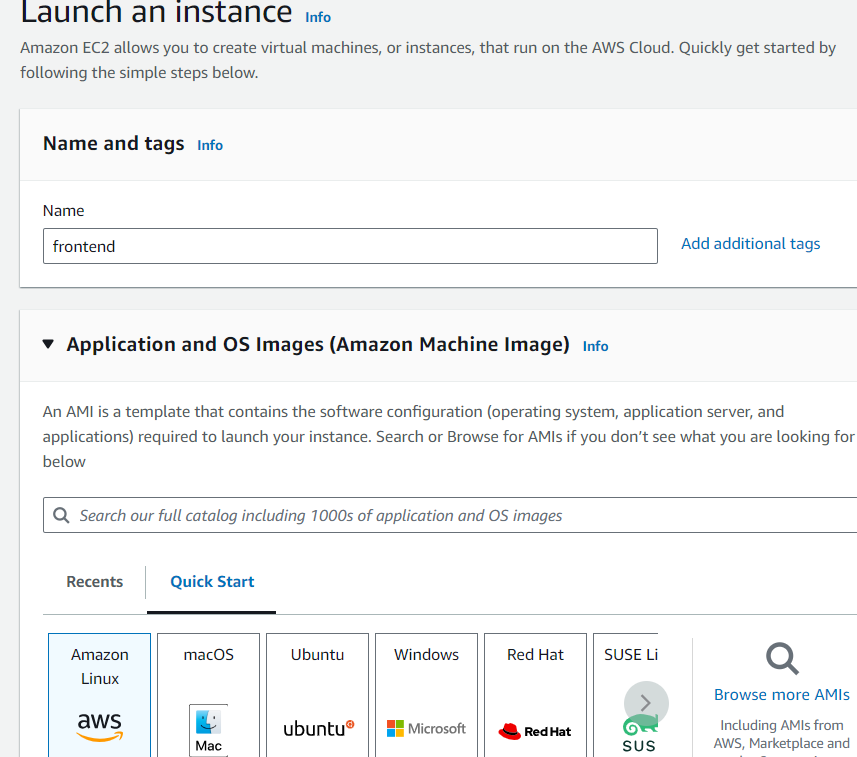
****

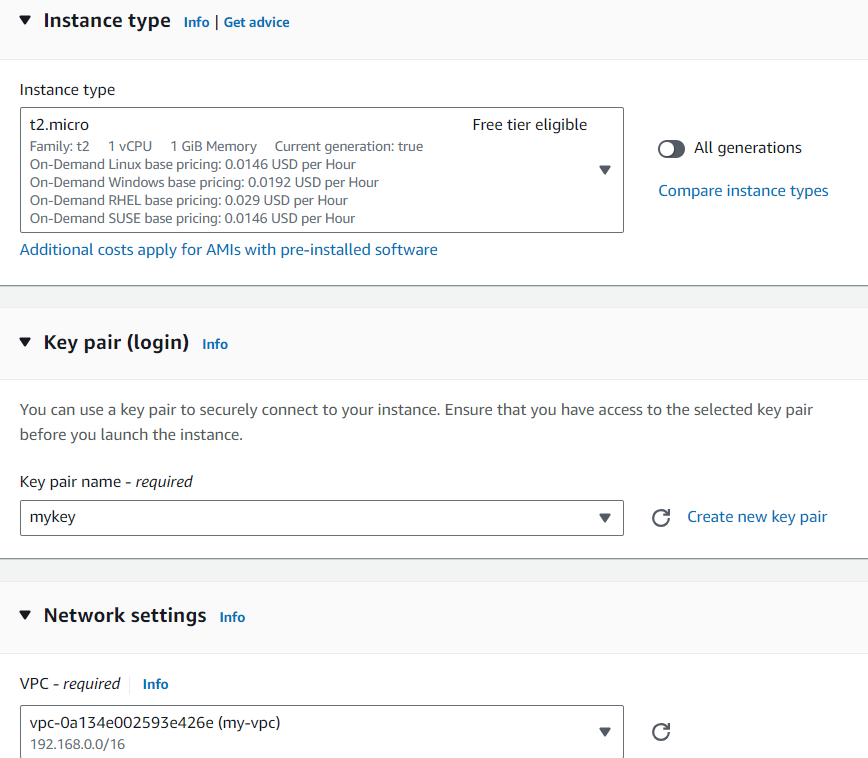
****

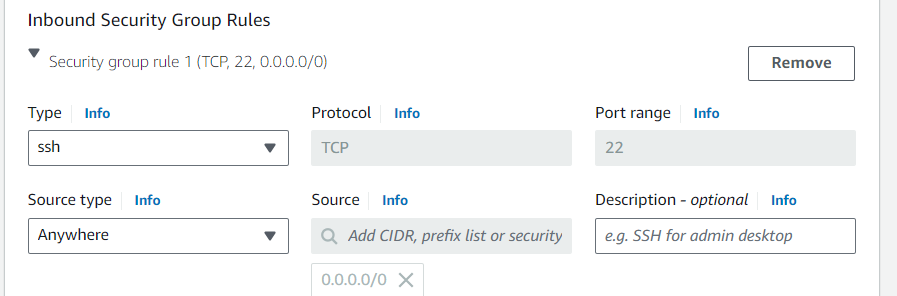
****

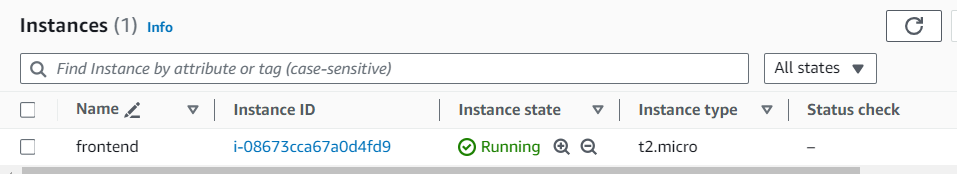
****

**Launch An Instance**

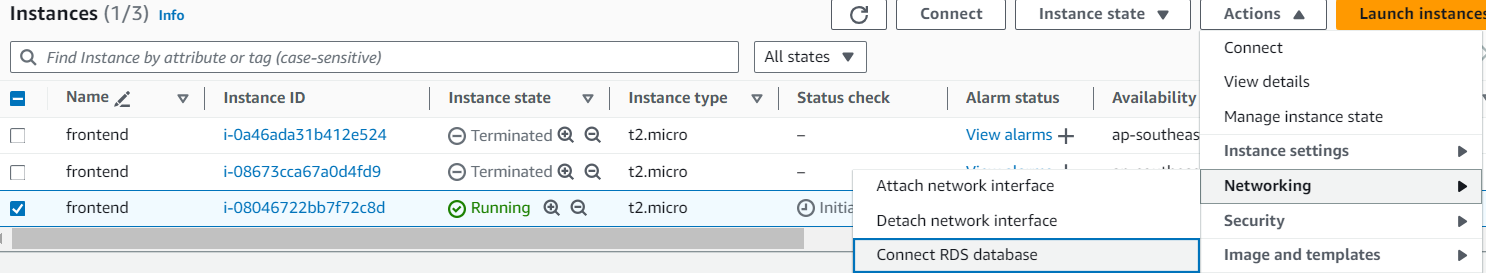
****

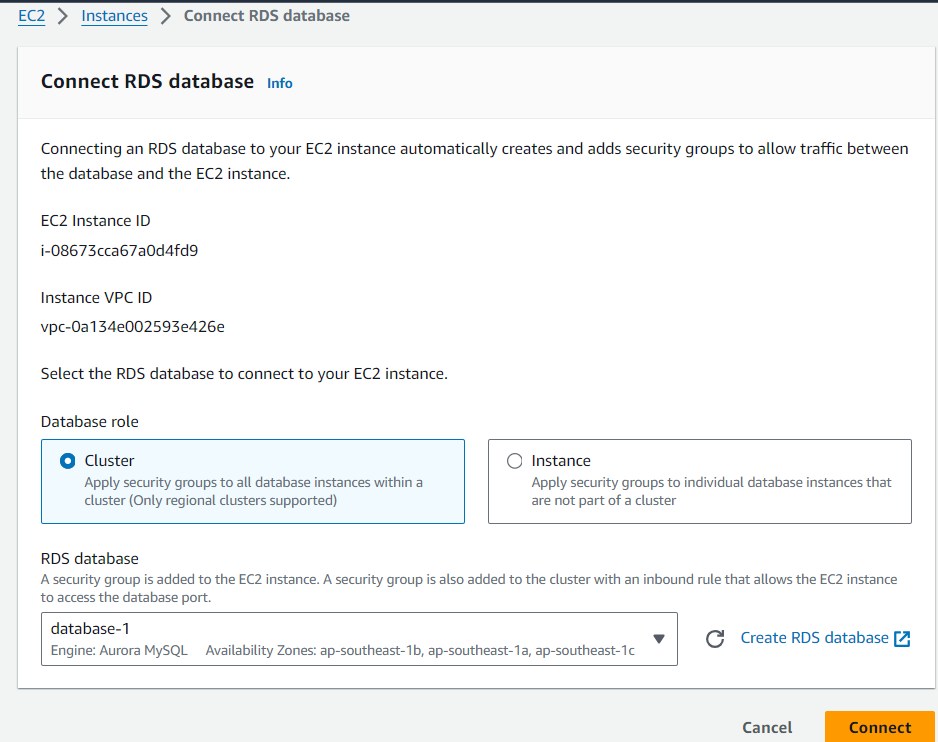
****

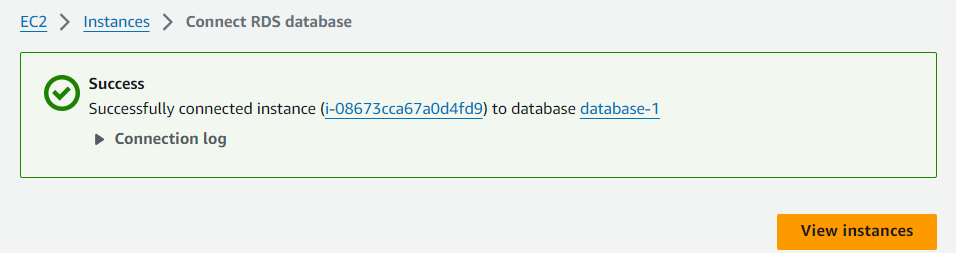
****

****

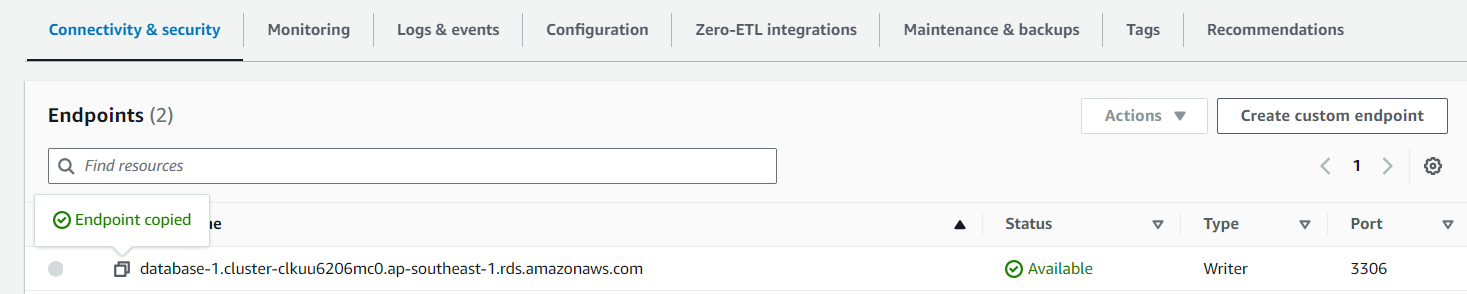
**Connect EC2 with RDS**

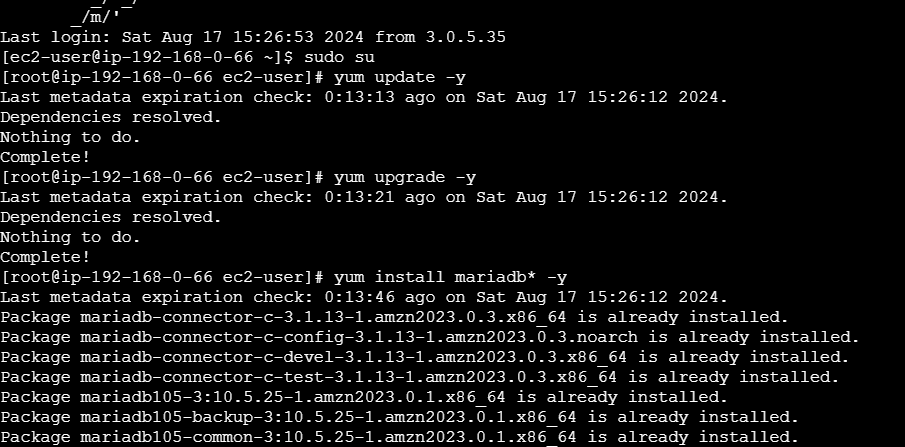
****

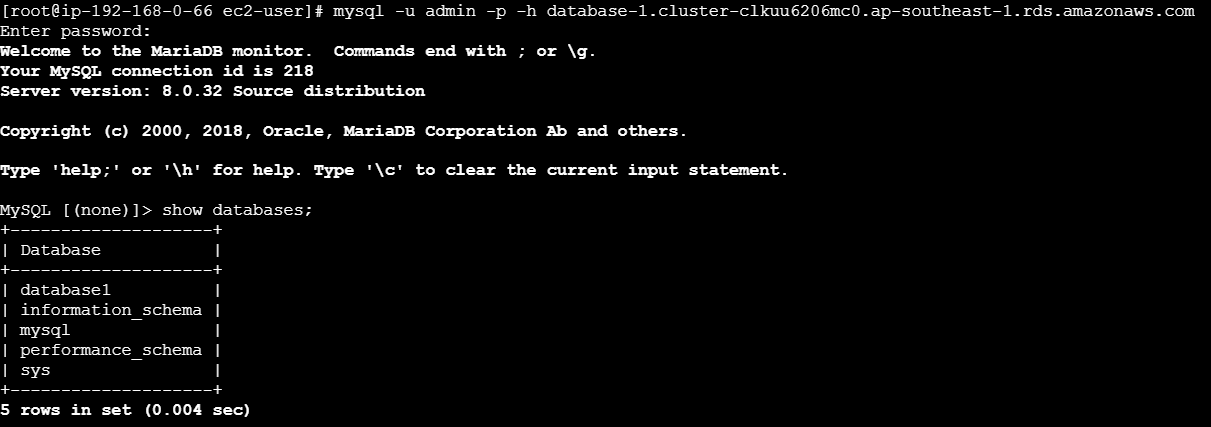
****

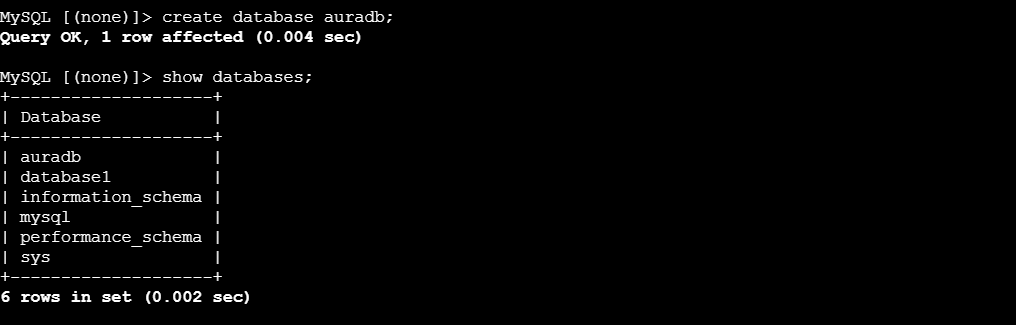
****

**Update the application’s configuration to point to the RDS Aurora endpoint.**

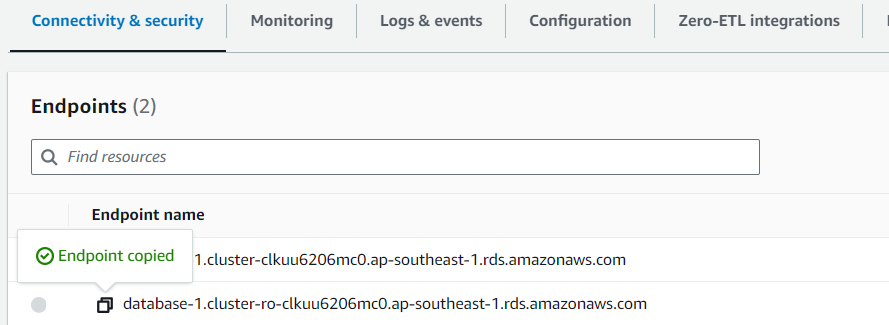
****

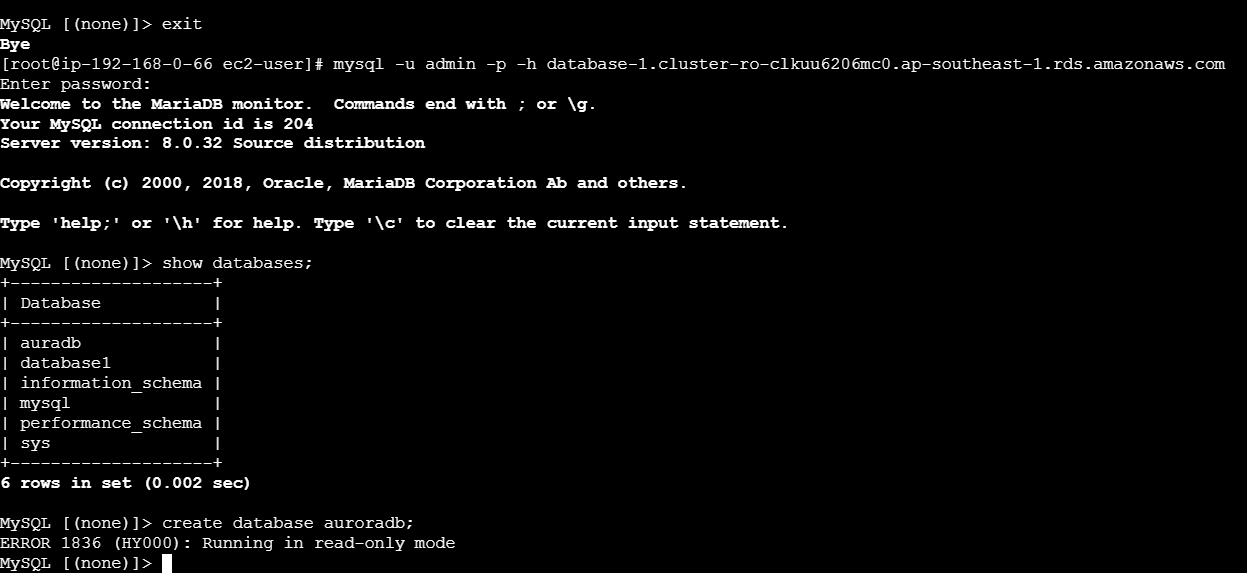
****

****

****

**For read operations, configure the application to use the read replica endpoint.**





**Testing**:

* + Perform some read and write operations to ensure everything is set up correctly.
  + Test the failover by temporarily stopping the primary RDS instance and checking if the application switches to the replica or the secondary AZ.

#### **Conclusion**

By following this guide, you have successfully deployed an EC2 instance and an RDS Aurora database in different AZs, complete with a read replica and automated backups. This architecture ensures high availability, better performance, and cost-effective use of AWS resources under the Free Tier. The use of different AZs for the EC2 instance and the RDS database helps enhance the fault tolerance of the application, while the read replica improves the read performance.