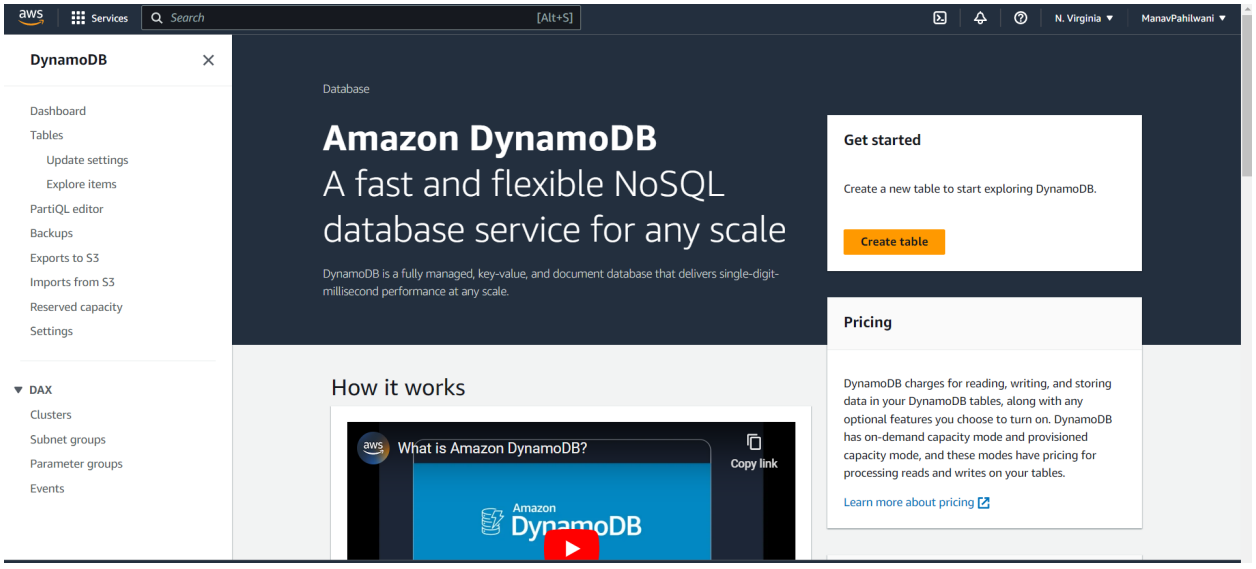
Name - Manav Pahilwani
Roll No - 37
Class - D11AD

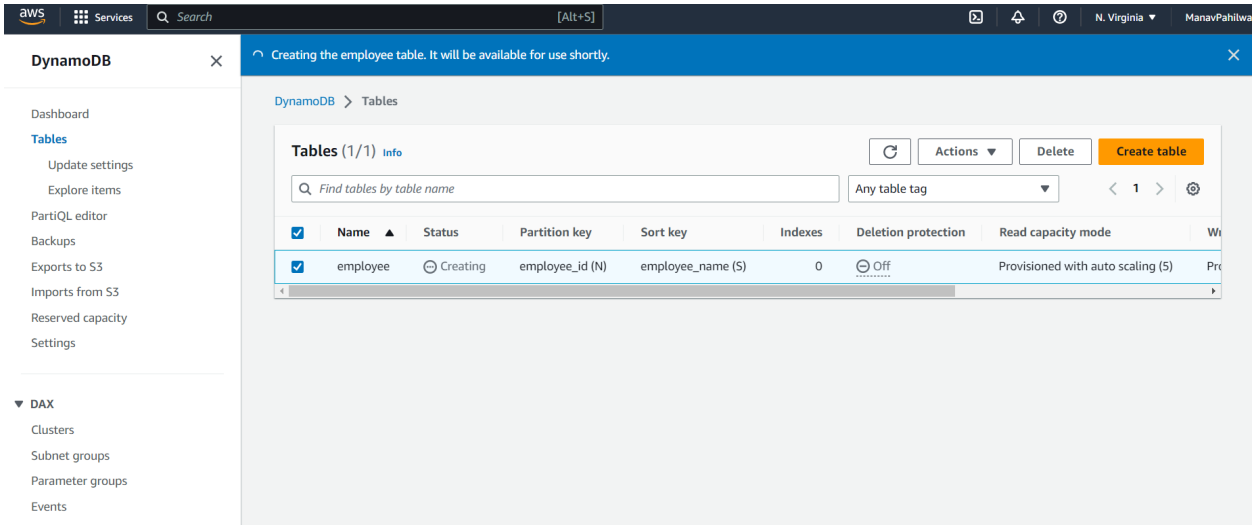
Aim: To study and ImplementDatabase as a Service on SQL/NoSQL databases like AWS RDS, AZURE SQL/ MongoDB Lab/ Firebase.

Theory :- Database As A Service:
Database as a service (DBaaS) is a cloud computing managed service offering that provides access to a database without requiring the setup of physical hardware, the installation of software, or the need to configure the database. Most maintenance and administrative tasks are handled by the service provider, freeing up users to quickly benefit from using the database. DBaaS uses The DBaaS model is ideal for small to medium-sized businesses that do not have well-staffed IT departments. Offloading the service and maintenance of the database to the DBaaS provider enables small to medium-sized businesses to implement applications and systems that they otherwise could not afford to build and support on-premises. Workloads involving data with stringent regulatory requirements may not be suitable for a DBaaS model. Furthermore, mission-critical applications that require optimal performance and 99.999% of uptime may be better suited for on-premises implementation. This is not to say that mission-critical workloads cannot run on cloud services, but much of the DBaaS adoption to date has been for less crucial applications, such as development and pilot programs.

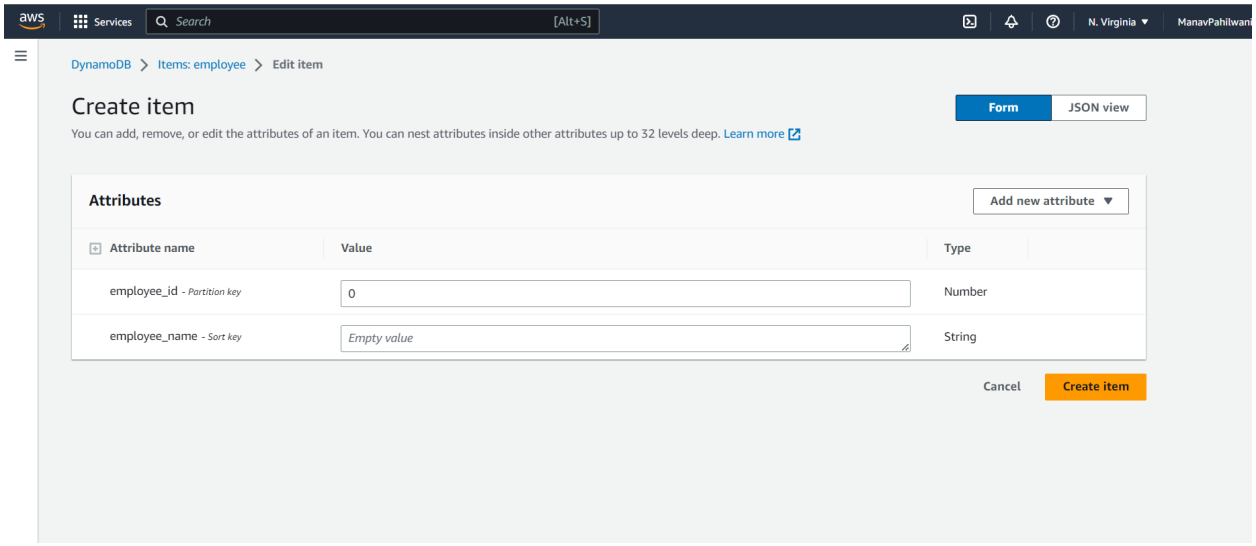
Implementation -
DynamoDB
Step 1: Select Database in Services



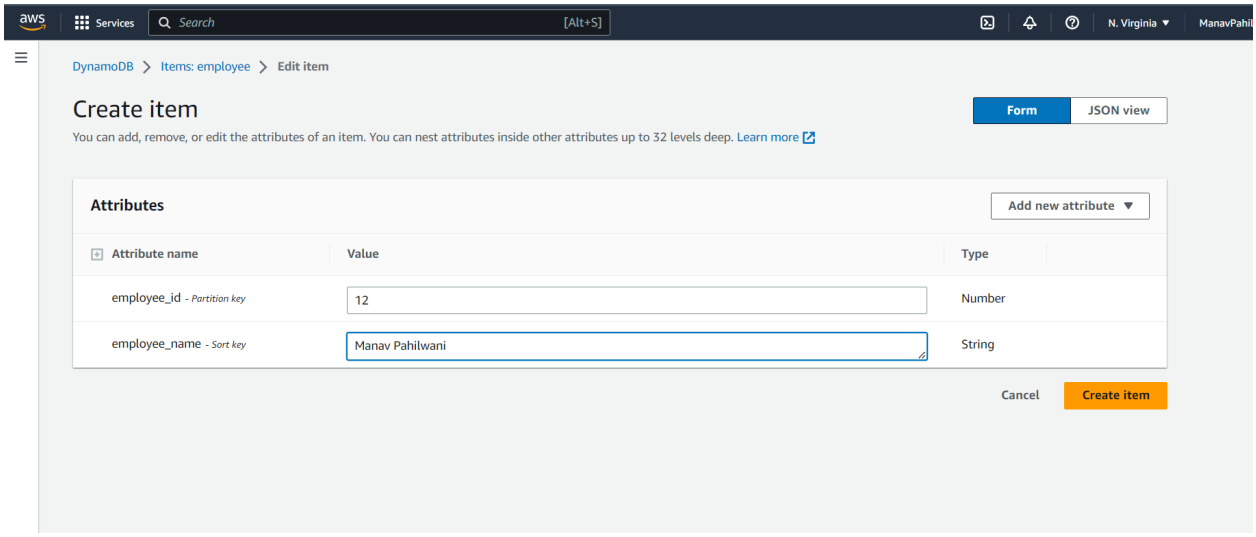
Step 2: Go to DynamoDB



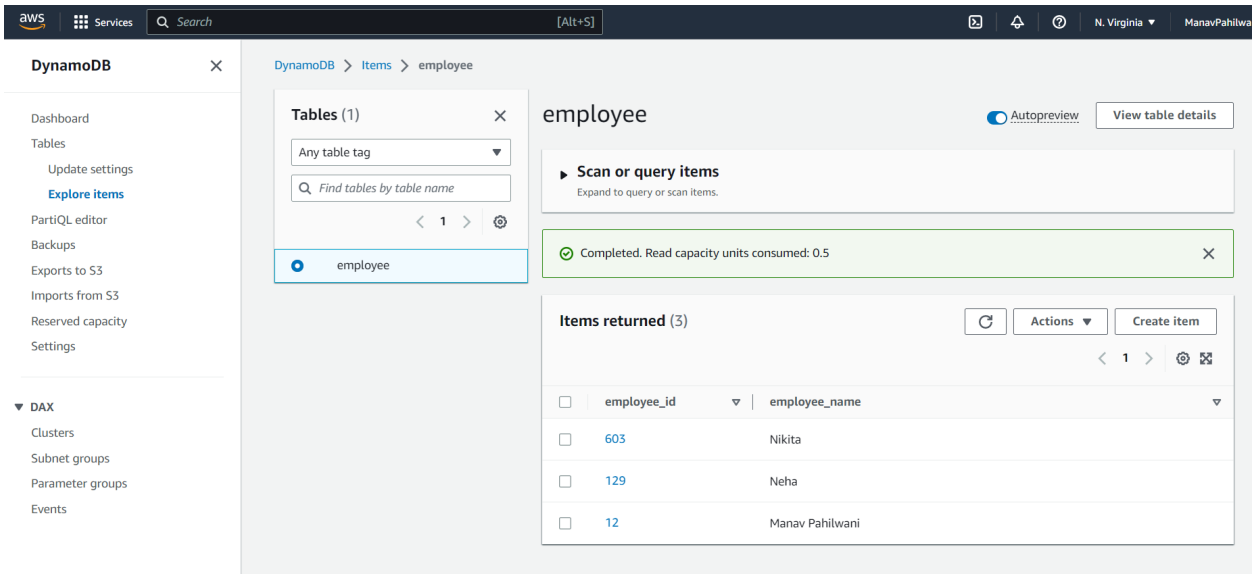
Step 3: Click on create table
Give a name for the table
Select partition key and sort key



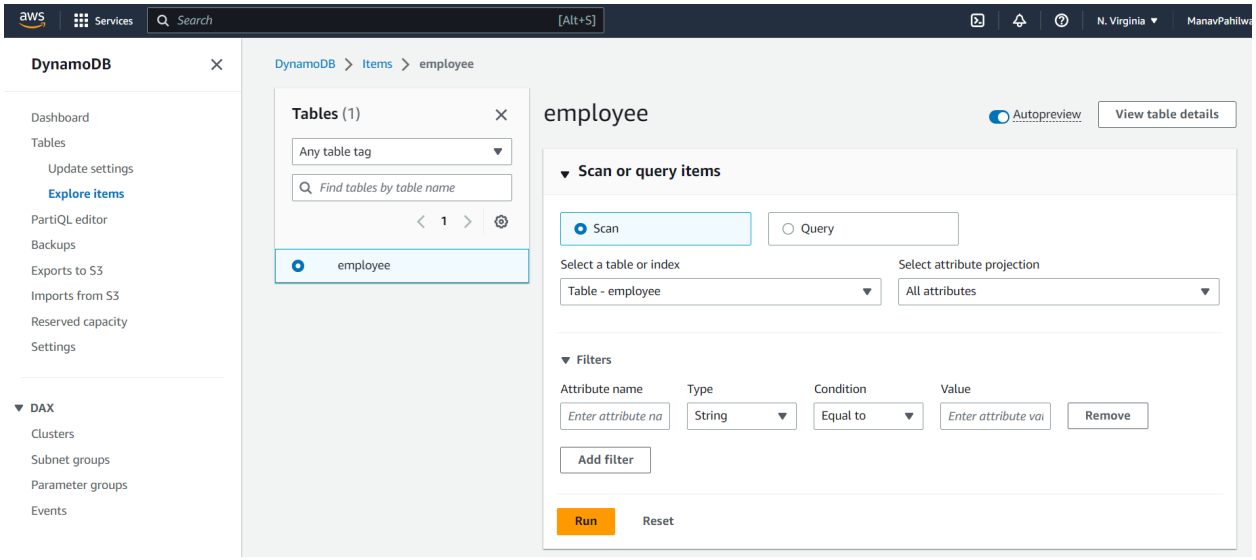
Step 4: Click on create table

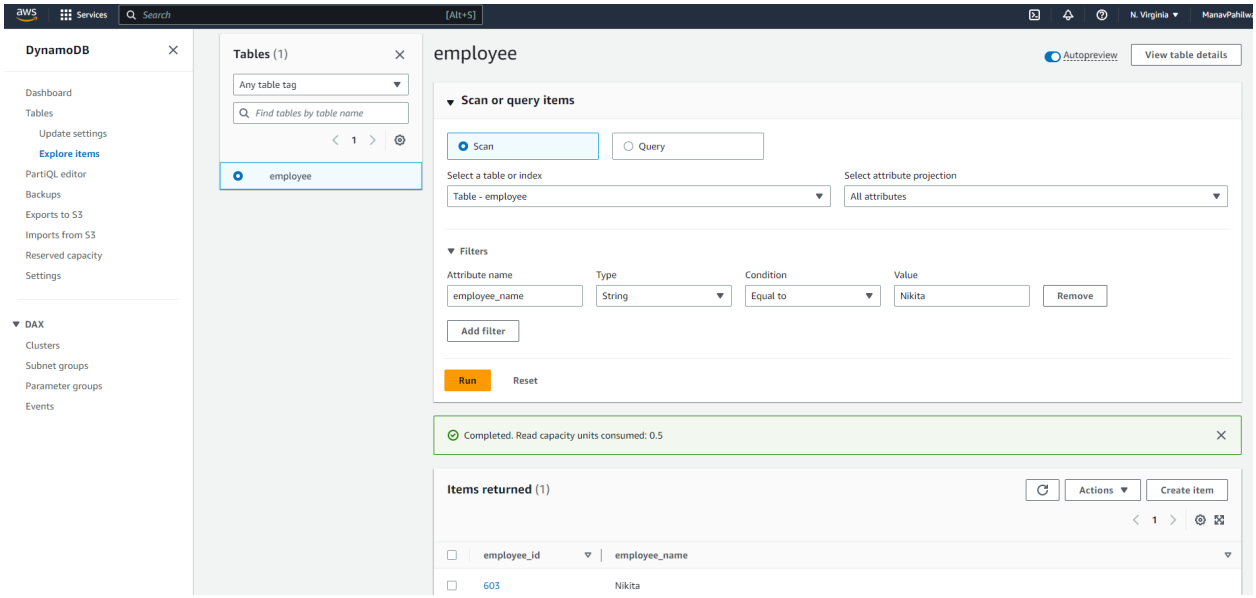


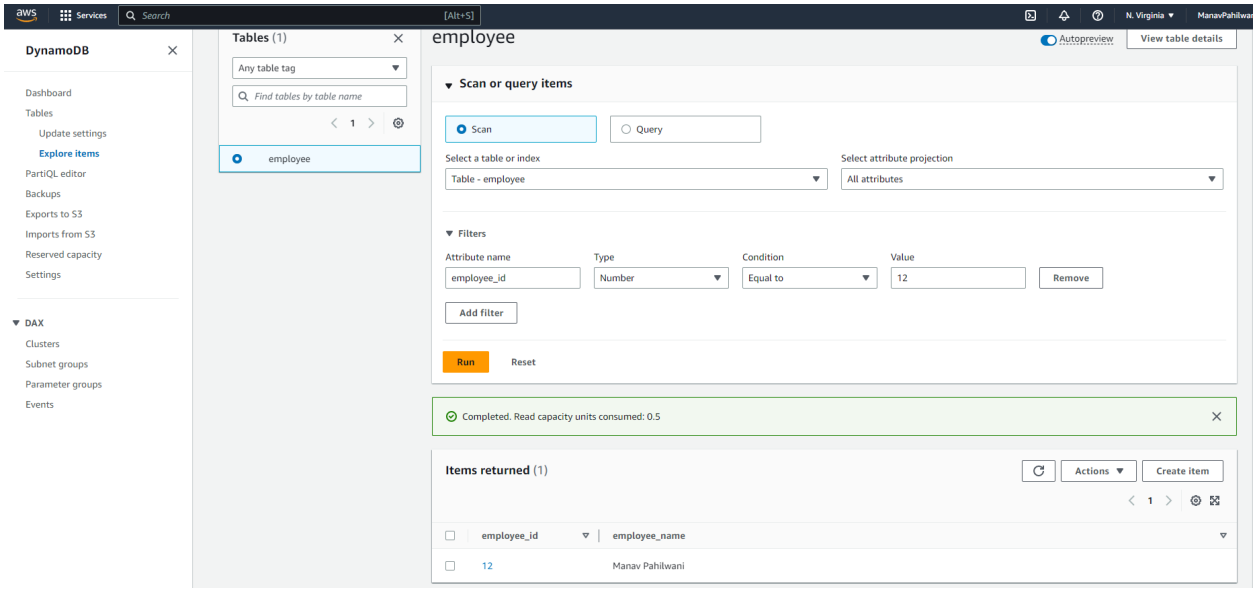
Step 5: Go to actions and click on create an item



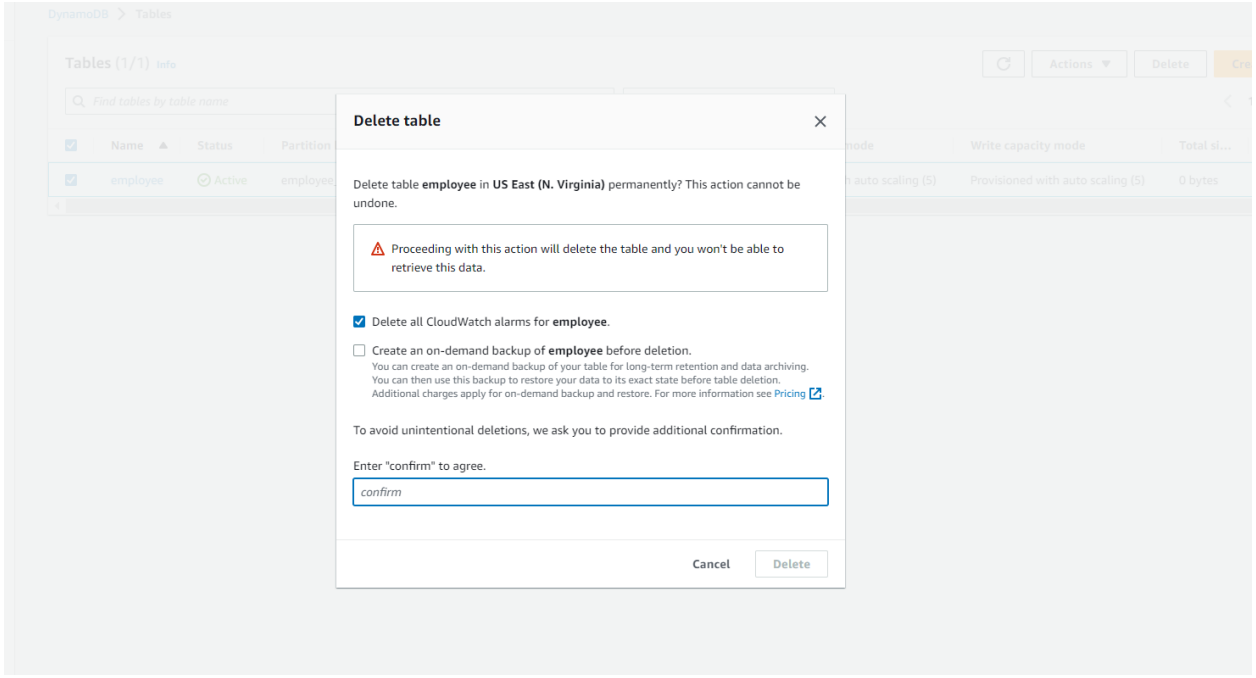
Step 6: Go to scan and then filter

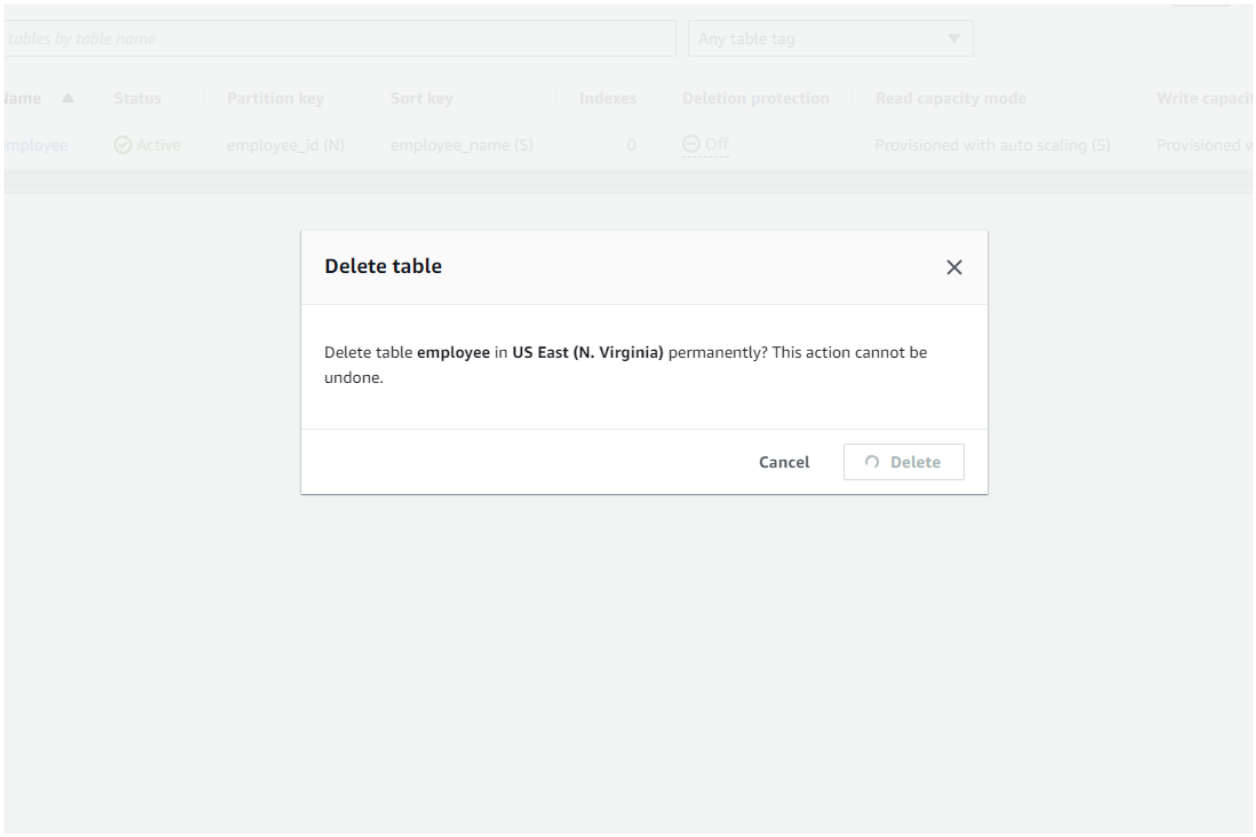


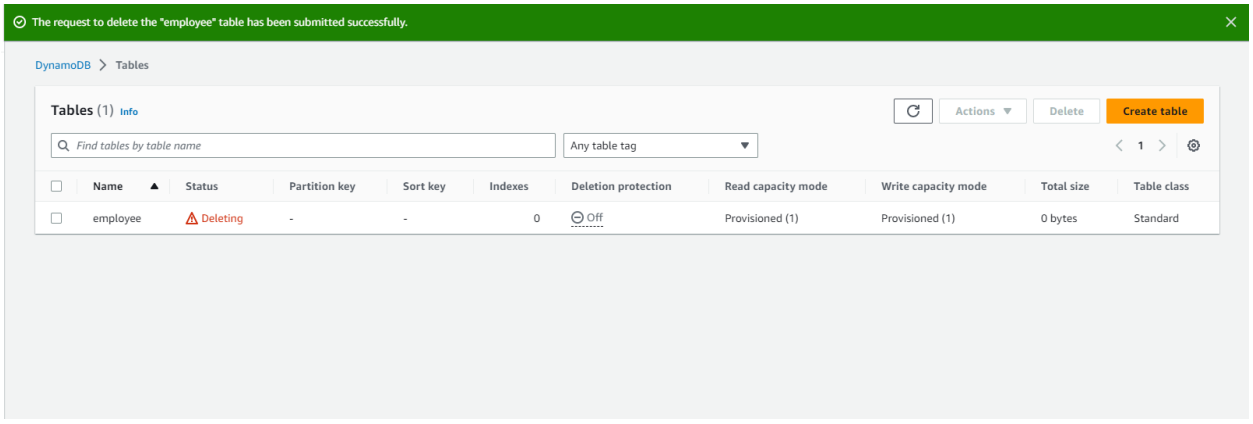




Step 7: Delete the table

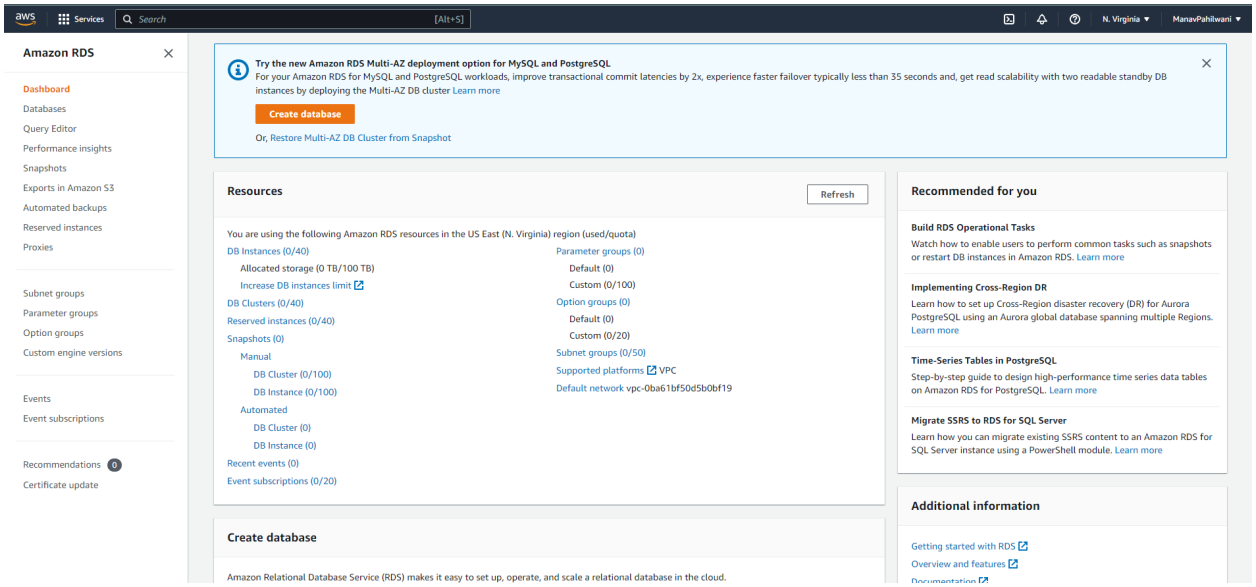


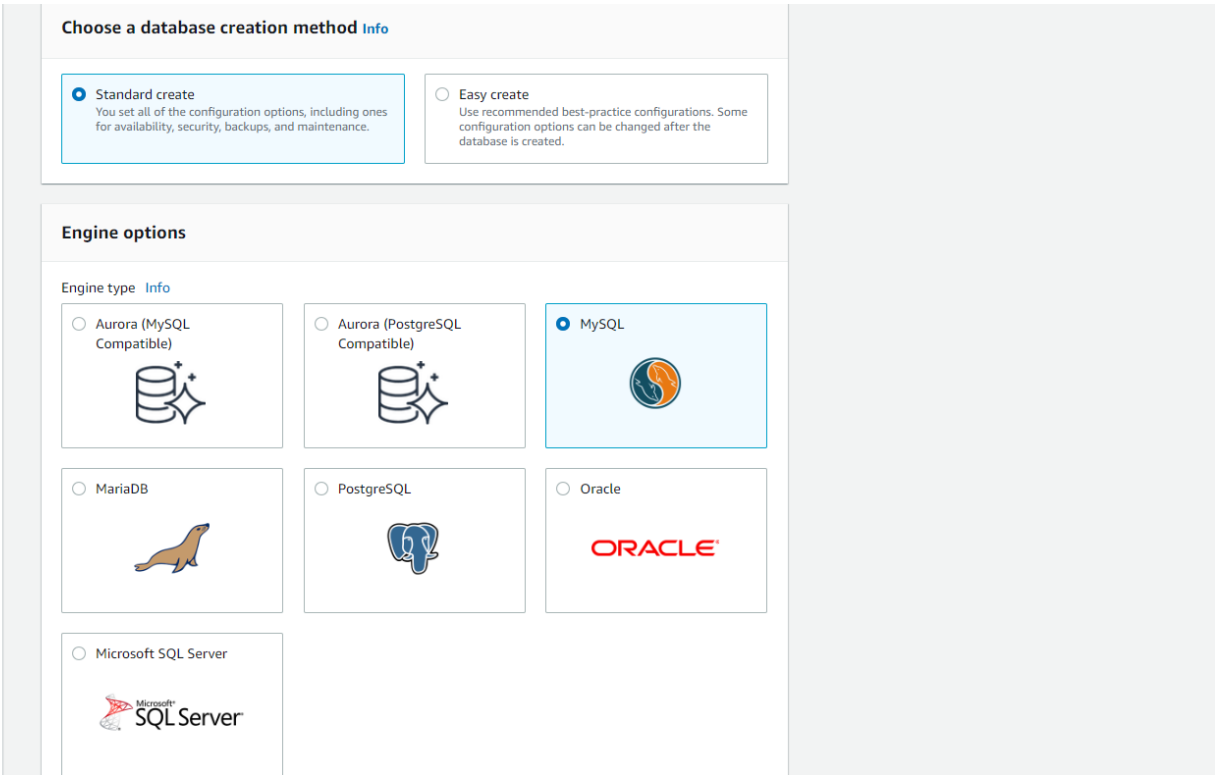

Any table tag
Status
Partition key
Sort key
Indexes
Deletion protection
Read capacity mode
employee
Active
employee_id (N)
employee_name (S)
0
Off
Provisioned with auto scaling (S)
Delete table
Delete table employee in US East (N. Virginia) permanently? This action cannot be undone.
Cancel
Delete

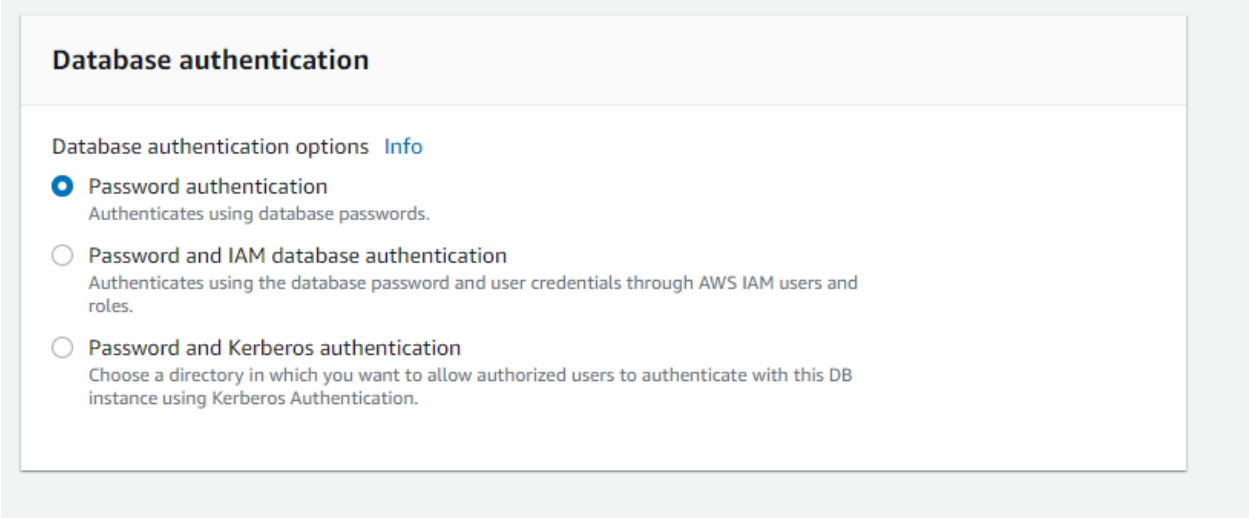

The request to delete the "employee" table has been submitted successfully.
DynamoDB
Tables
Tables (1) Info
Actions
Delete
Create table
Find tables by table name
Any table tag
Name
Status
Partition key
Sort key
Indexes
Deletion protection
Read capacity mode
Write capacity mode
Total size
Table class
employee
Deleting
-
-
0
Off
Provisioned (1)
Provisioned (1)
0 bytes
Standard

RDS:

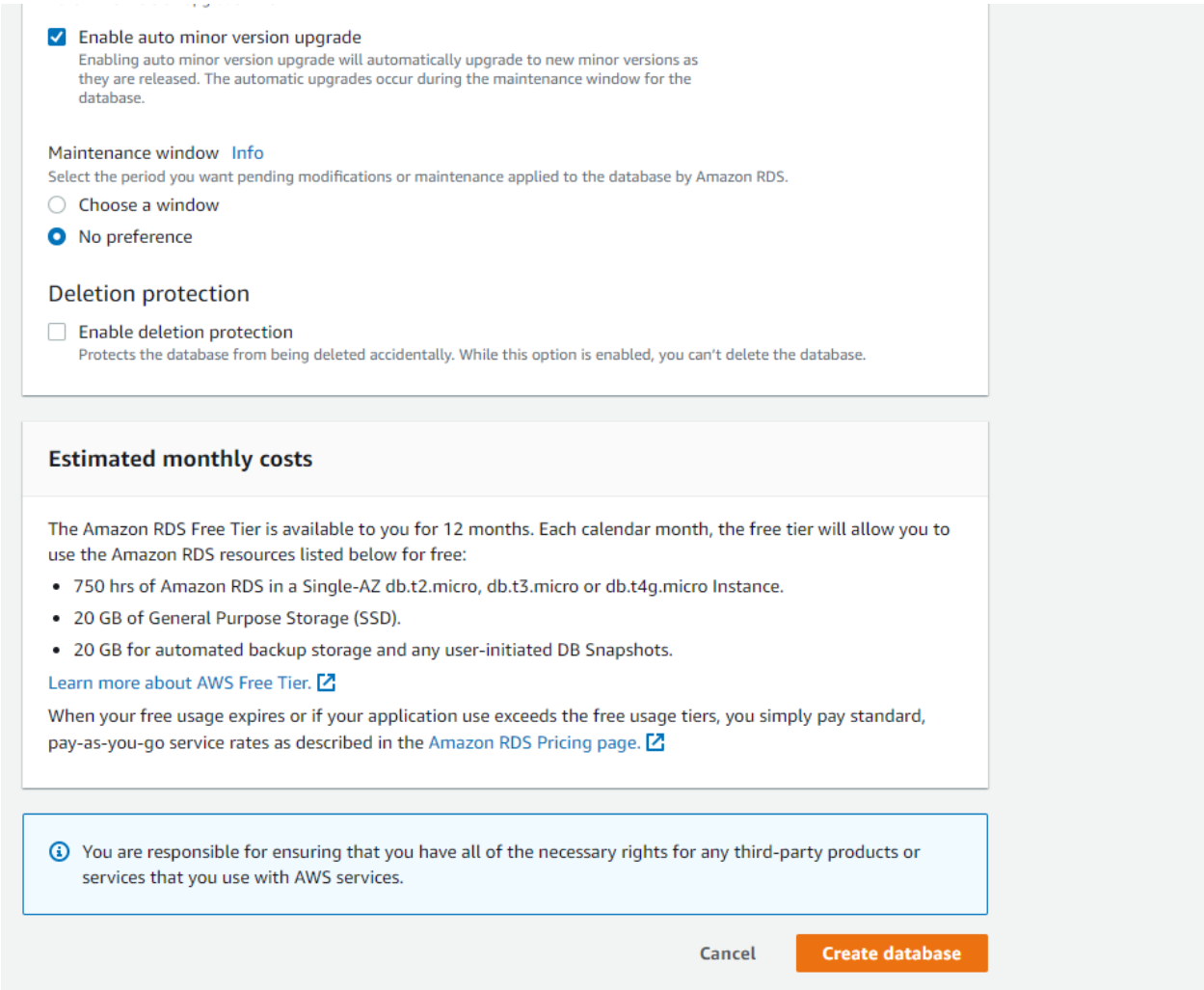
Step 8: Now again go to services. Click on database and go to RDS

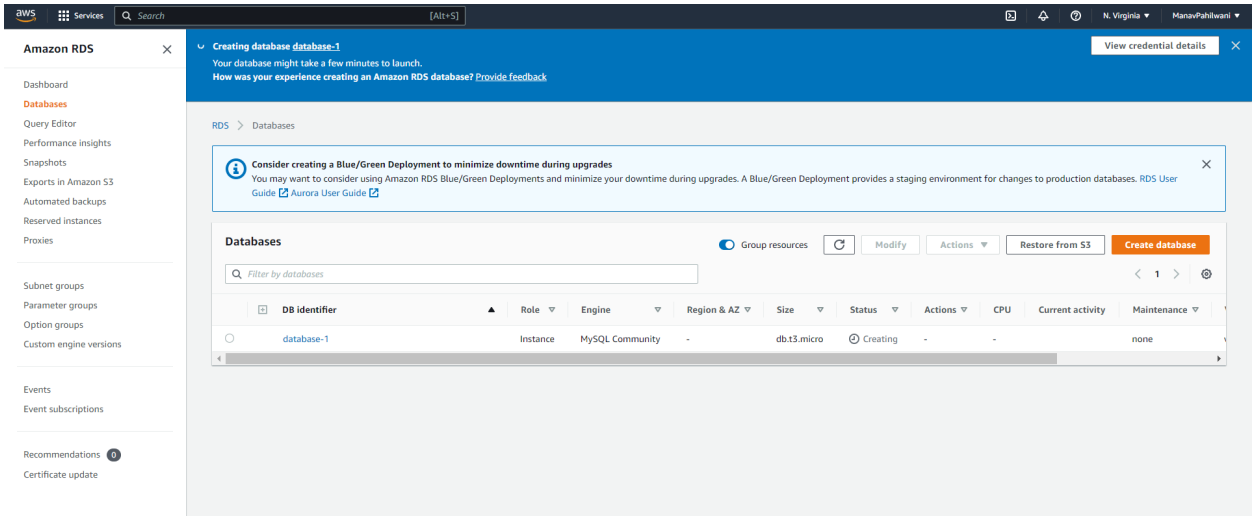




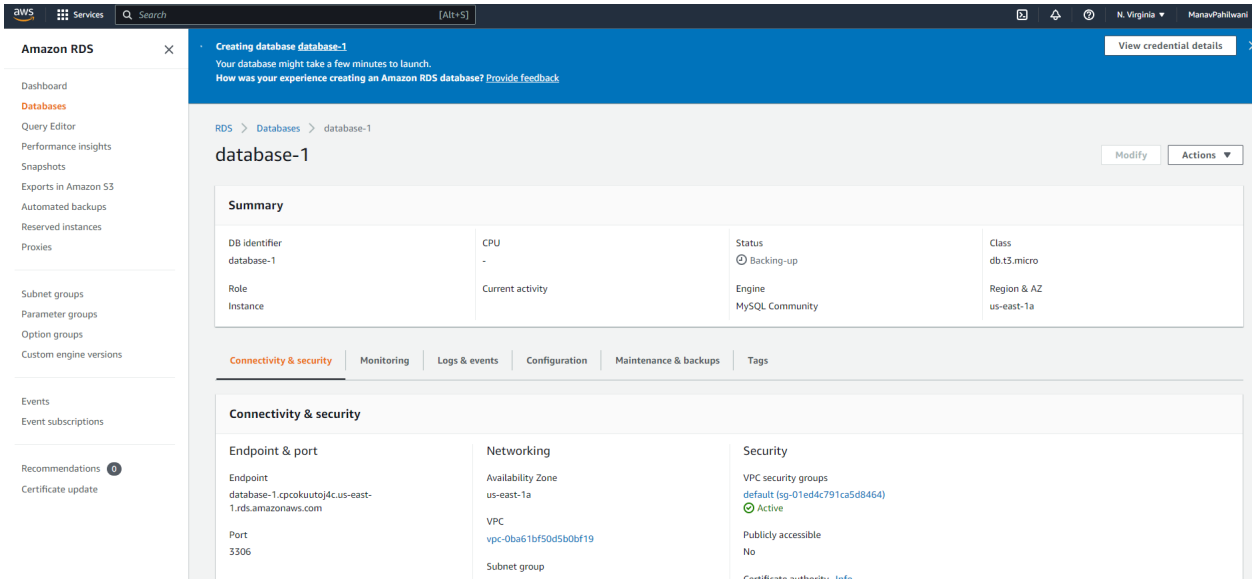


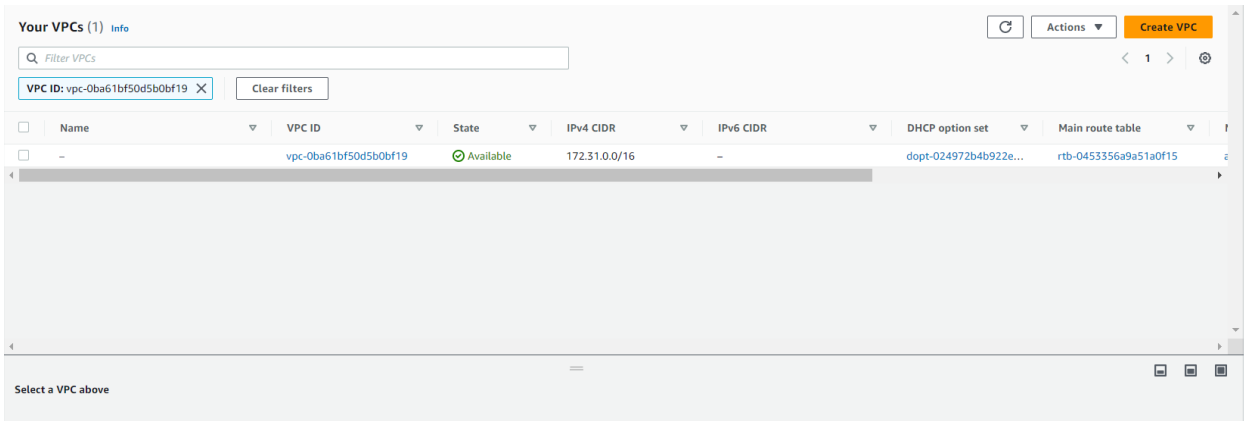
Step 9: Click on create a database. It may take some time to create a database. After it gets created click on it to see details.



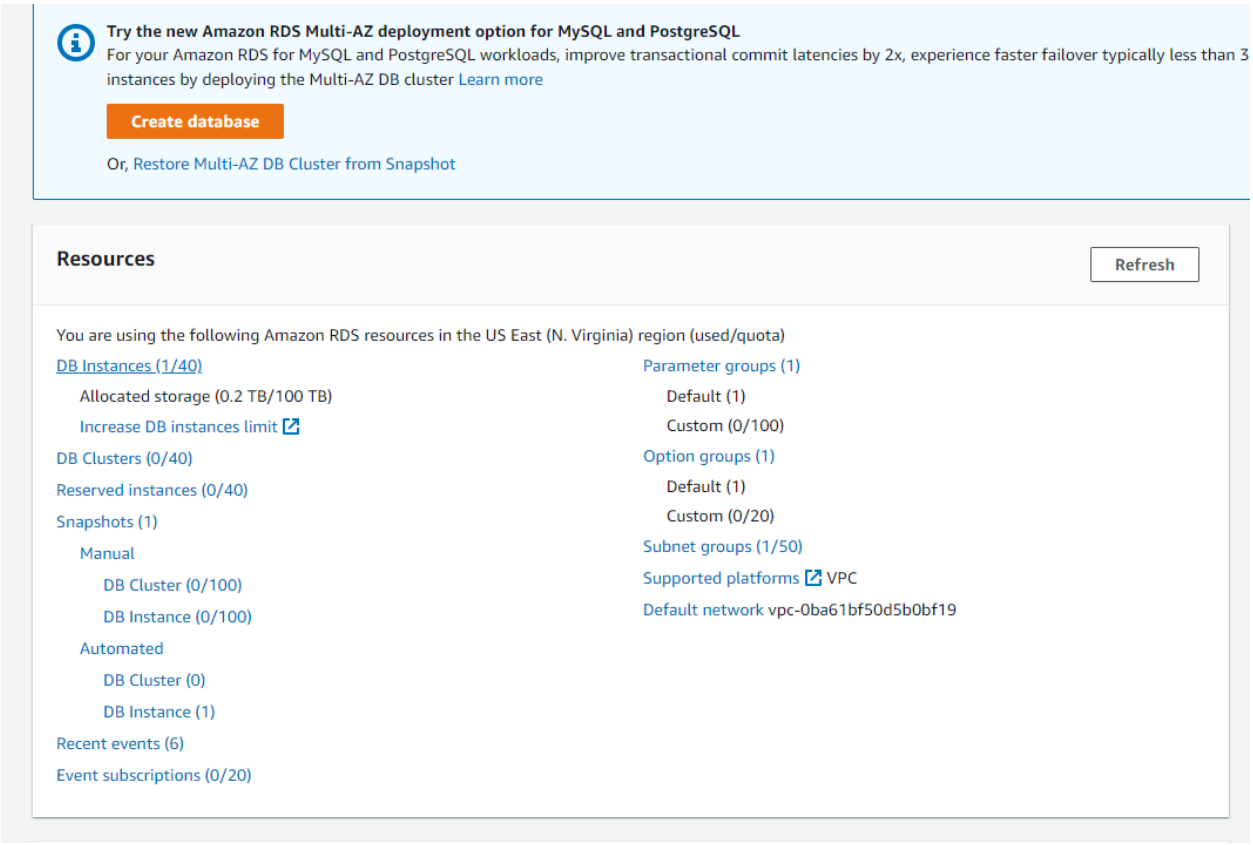


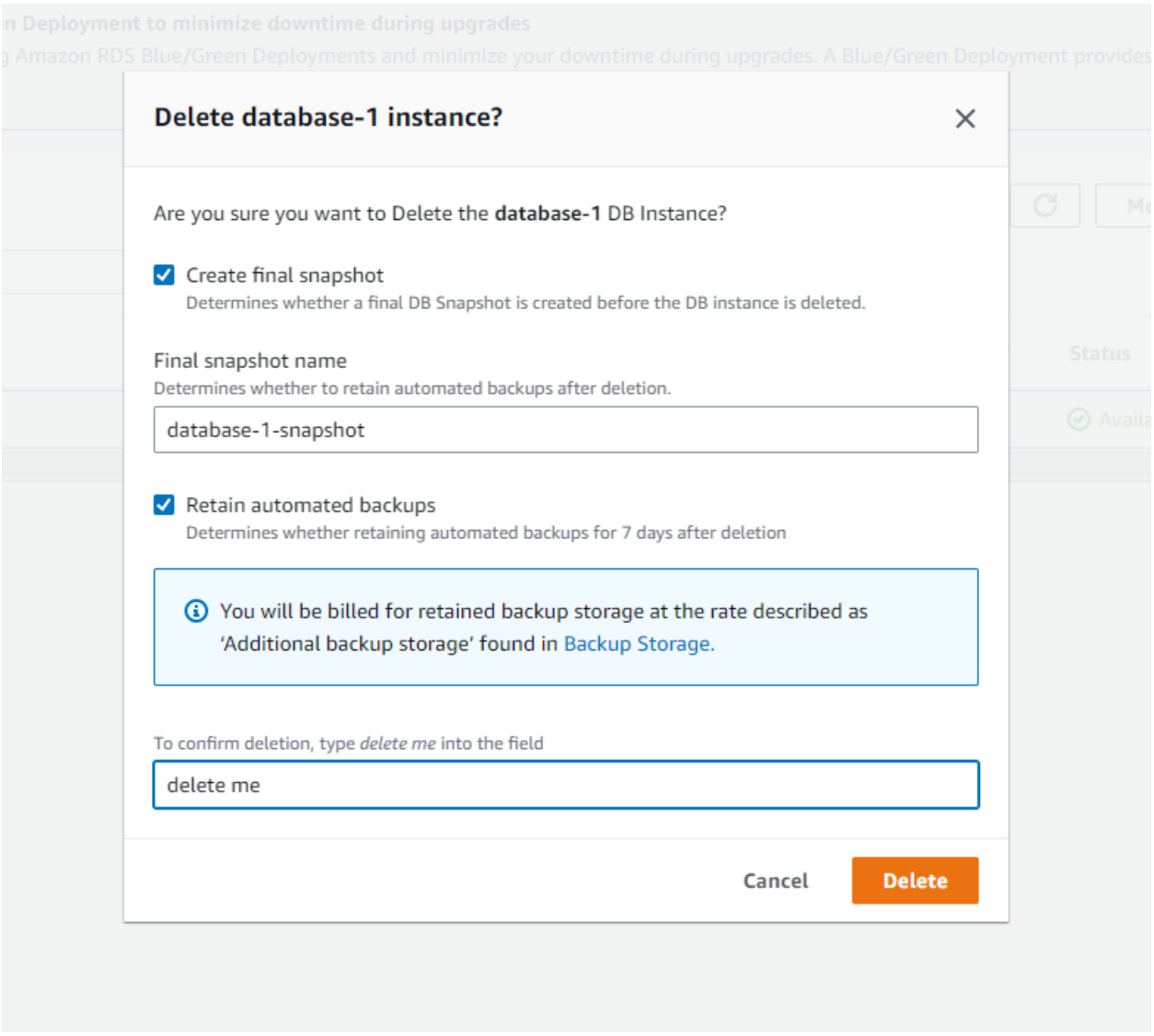
Step 10: Click on vpc

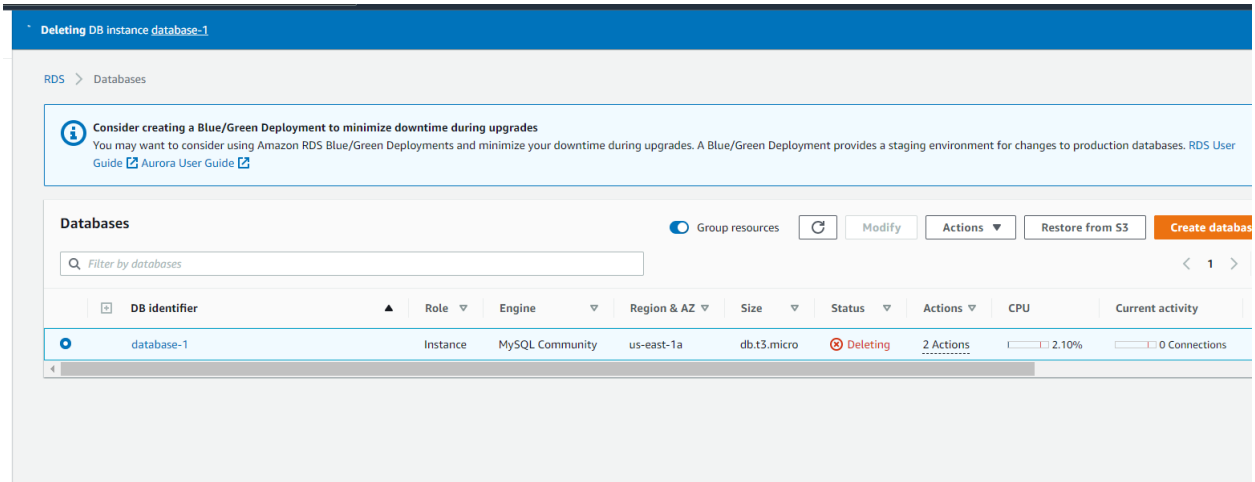




Step 11: Delete the instance. It may take some time to delete the instance (database).




Deployment to minimize downtime during upgrades
Amazon RDS Blue/Green Deployments and minimize your downtime during upgrades. A Blue/Green Deployment provides
Delete database-1 instance?
Are you sure you want to Delete the database-1 DB Instance?
Create final snapshot
Determines whether a final DB Snapshot is created before the DB instance is deleted.
Final snapshot name
Determines whether to retain automated backups after deletion.
database-1-snapshot
Retain automated backups
Determines whether retaining automated backups for 7 days after deletion
You will be billed for retained backup storage at the rate described as 'Additional backup storage' found in Backup Storage.
To confirm deletion, type delete me into the field
delete me
Cancel
Delete
Status


Deleting DB instance database-1
RDS
Databases
Consider creating a Blue/Green Deployment to minimize downtime during upgrades
You may want to consider using Amazon RDS Blue/Green Deployments and minimize your downtime during upgrades. A Blue/Green Deployment provides a staging environment for changes to production databases. RDS User Guide Aurora User Guide
Databases
Group resources
Modify
Actions
Restore from S3
Create databas
Filter by databases
DB identifier
Role
Engine
Region & AZ
Size
Status
Actions
CPU
Current activity
database-1
Instance
MySQL Community
us-east-1a
db.t3.micro
Deleting
2 Actions
2.10%
0 Connections

Conclusion - Amazon Web Services (AWS) provides a wide range of Database as a Service (DBaaS) options to help organizations manage their database needs. AWS's managed database services include Amazon RDS (Relational Database Service) for relational databases such as MySQL, Oracle, and PostgreSQL; Amazon DynamoDB for NoSQL databases; and Amazon Redshift for data warehousing. These services are designed to be highly scalable, reliable, and cost-effective, and they offer a range of features for backup and recovery, monitoring, security, and performance optimization.