RDS Hands-on

Problem Statement:

You work for XYZ Corporation. Your company has multiple branch offices all over the country. It wants to deploy its application on AWS that reads data from users' devices and sends reports to all the branches. The application also requires executing complex relational joins and updates. The company wants high availability for its application.

You are asked to perform the following tasks:

1. Build a highly scalable database; make sure that it is always available and accessible, and manage automated backups for this database with a retention period of 2 days

2. Build a database architecture, which lets your application read data from different regions

3. Establish a database architecture that processes the data collected for near real-time analysis

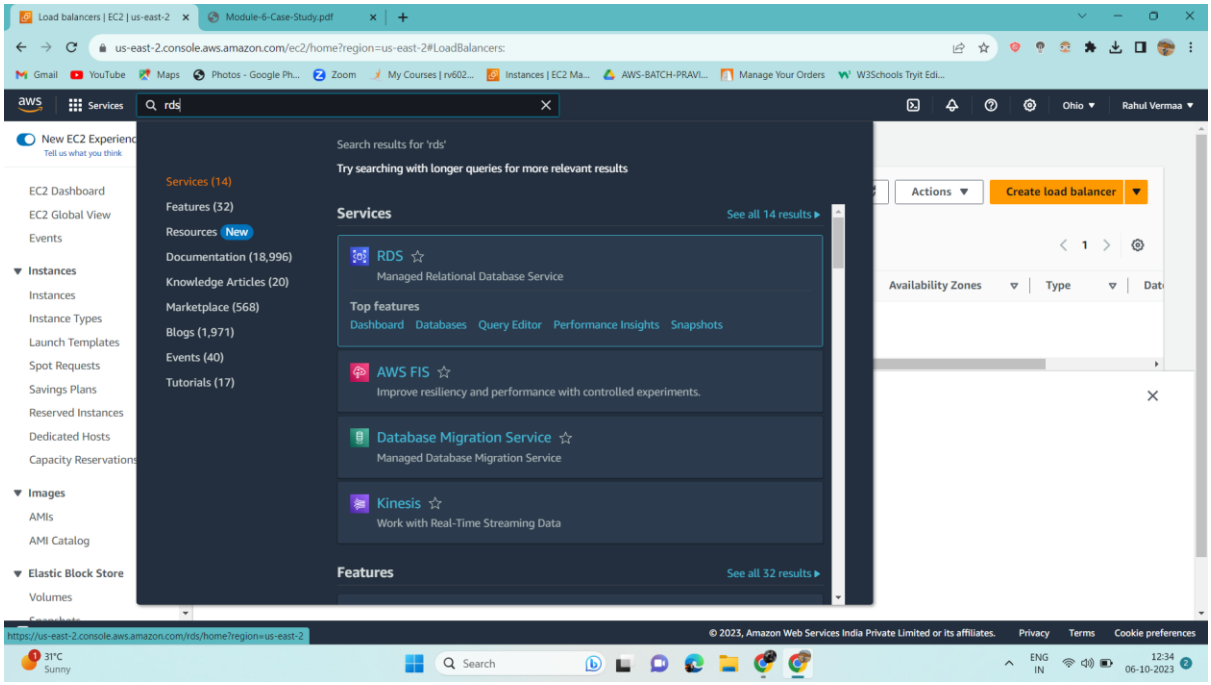
4. Upload data to a DynamoDB table that you have created

5. Take a backup of the table; delete the table, and then restore it

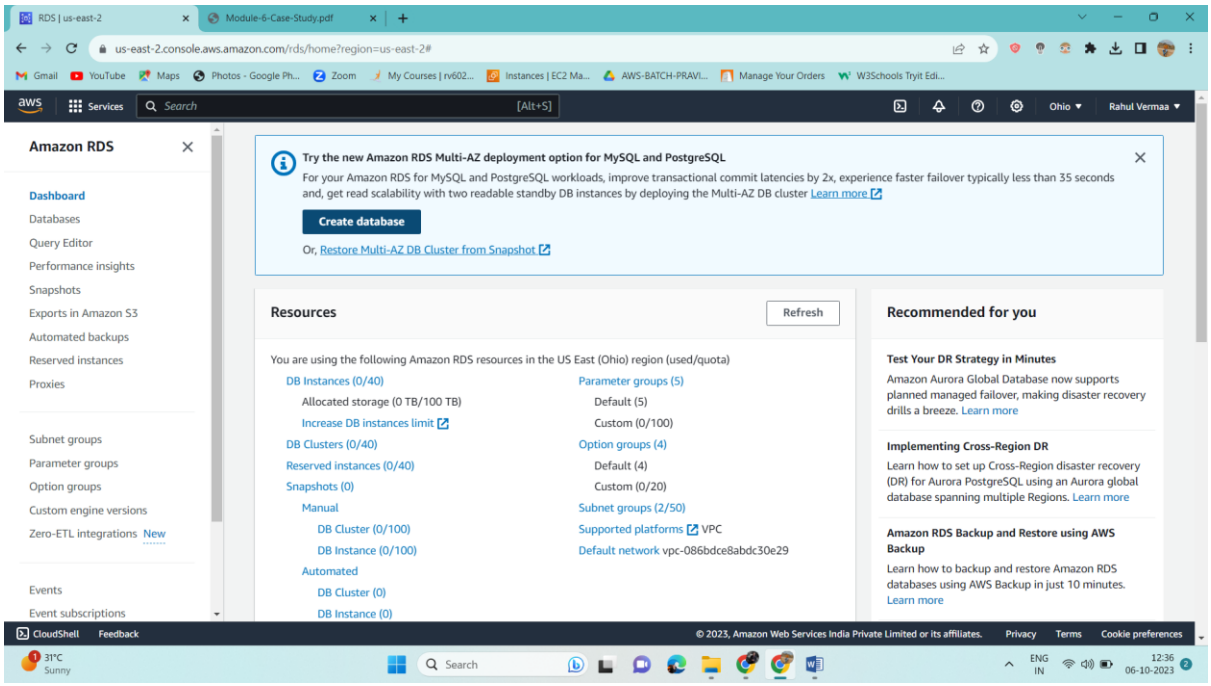
As the user base of the application increases, your company observes a significant latency in the transfer of data to the branches. It realizes that most of the time, the same kind of data is being fetched and sent to the branches. Therefore, you are asked to:

6. Find and implement the solution to resolve the latency issues

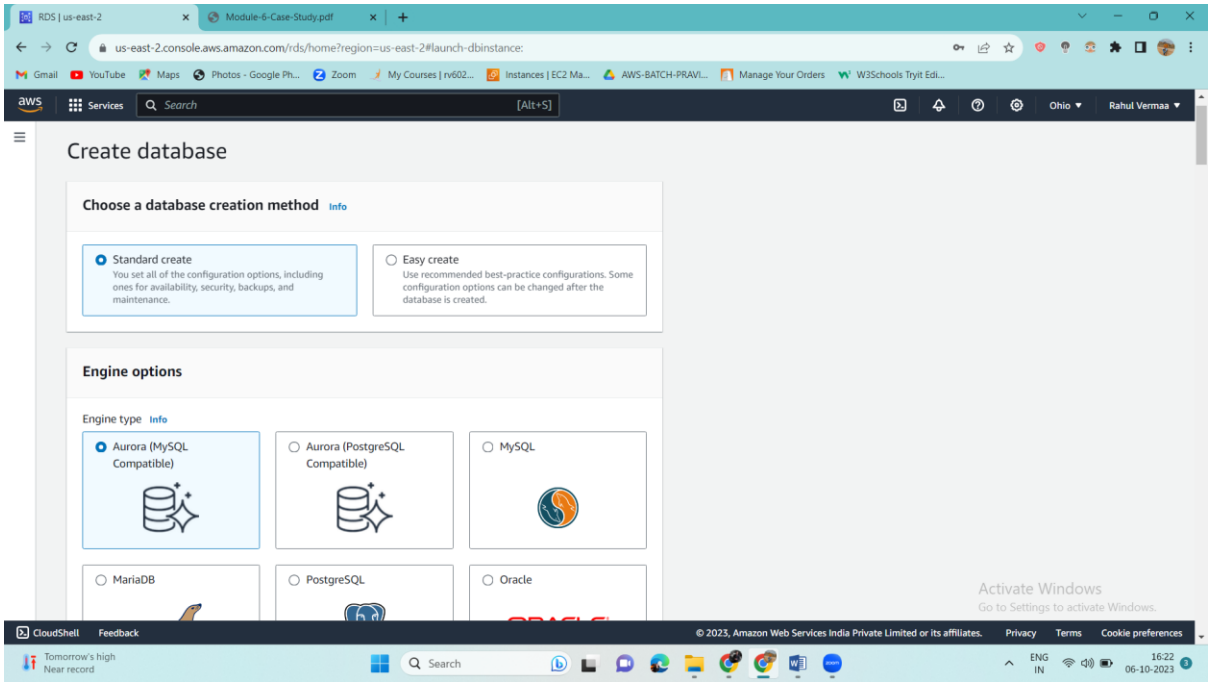
Search RDS in AWS console



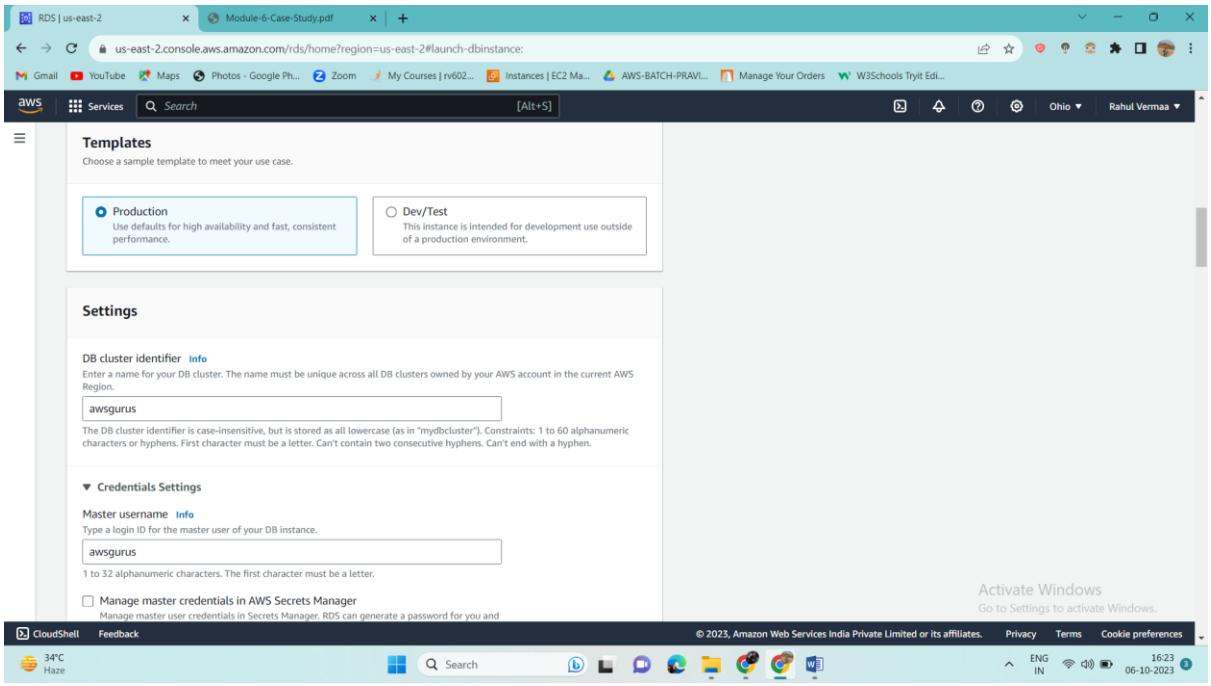
Now click on create database



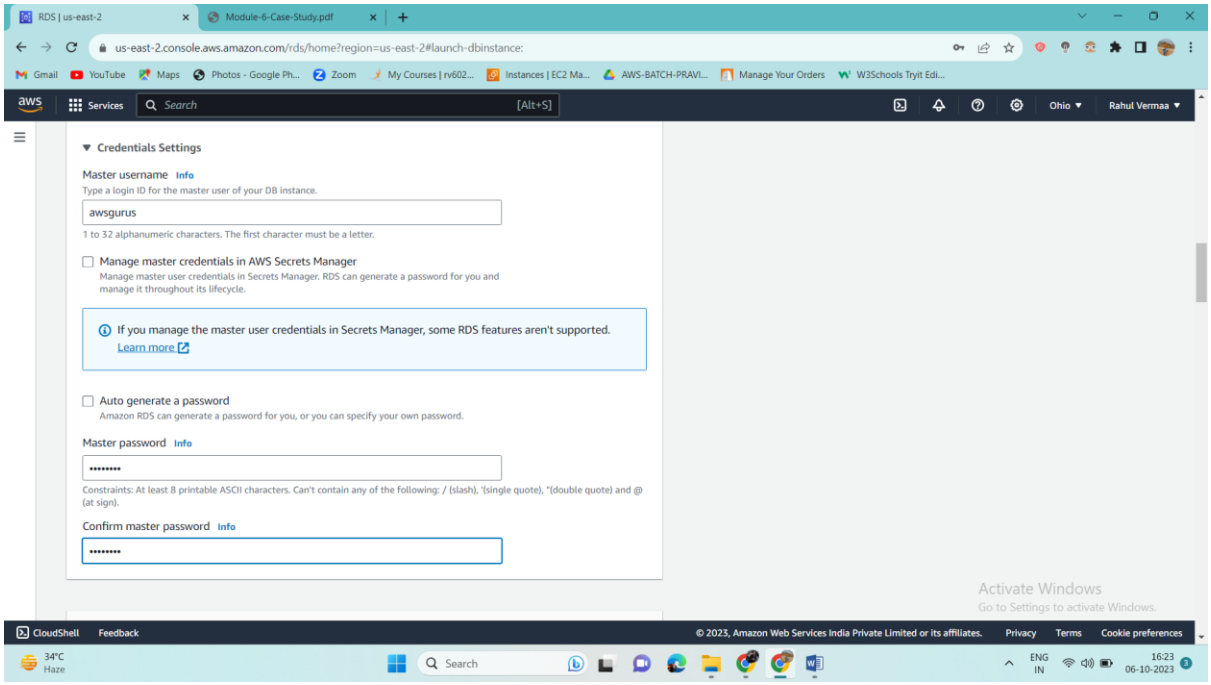
Select standard create and engine type- Aurora



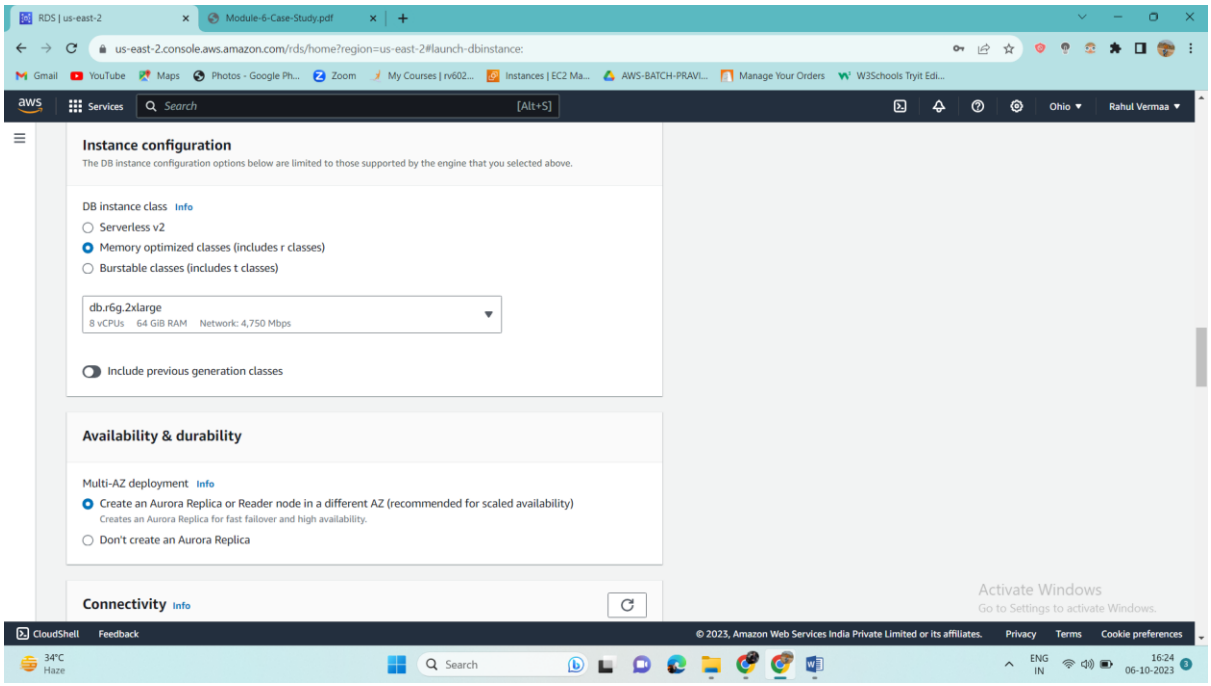
Templates- Production and DB instance Identifier- awsgurus



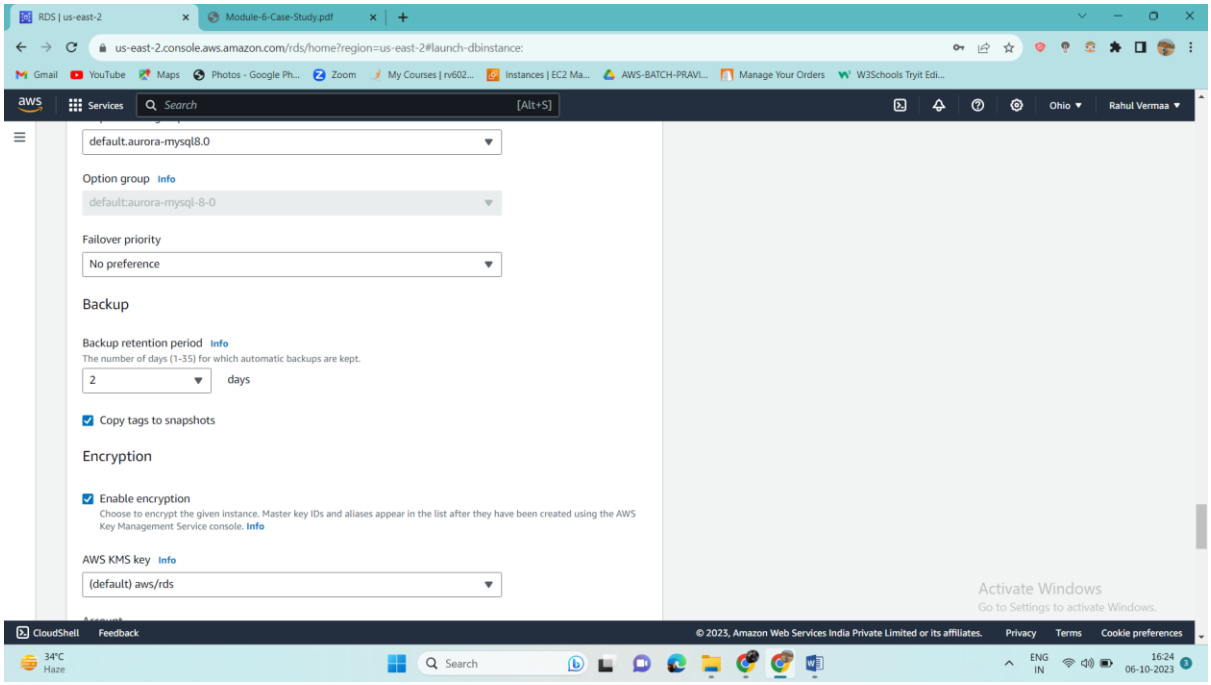
Username- awsgurus and password is also same awsgurus



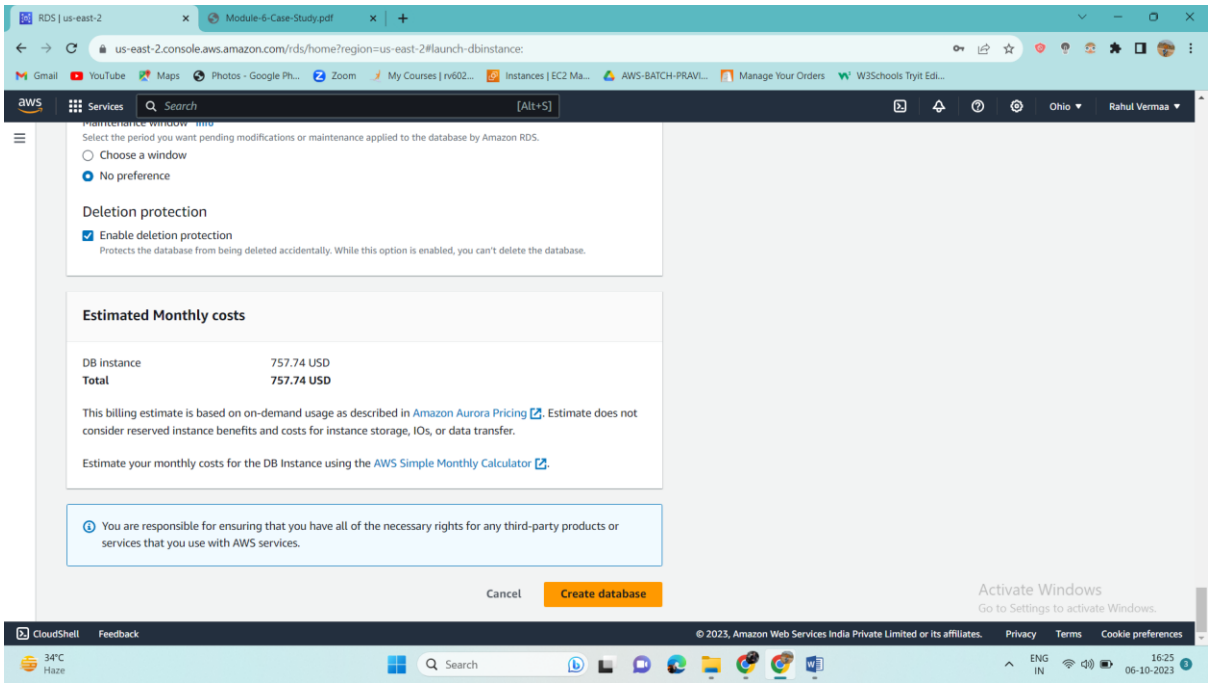
Select multi AZ deployment



Enable autoscaling and backup retention period should be 2 days

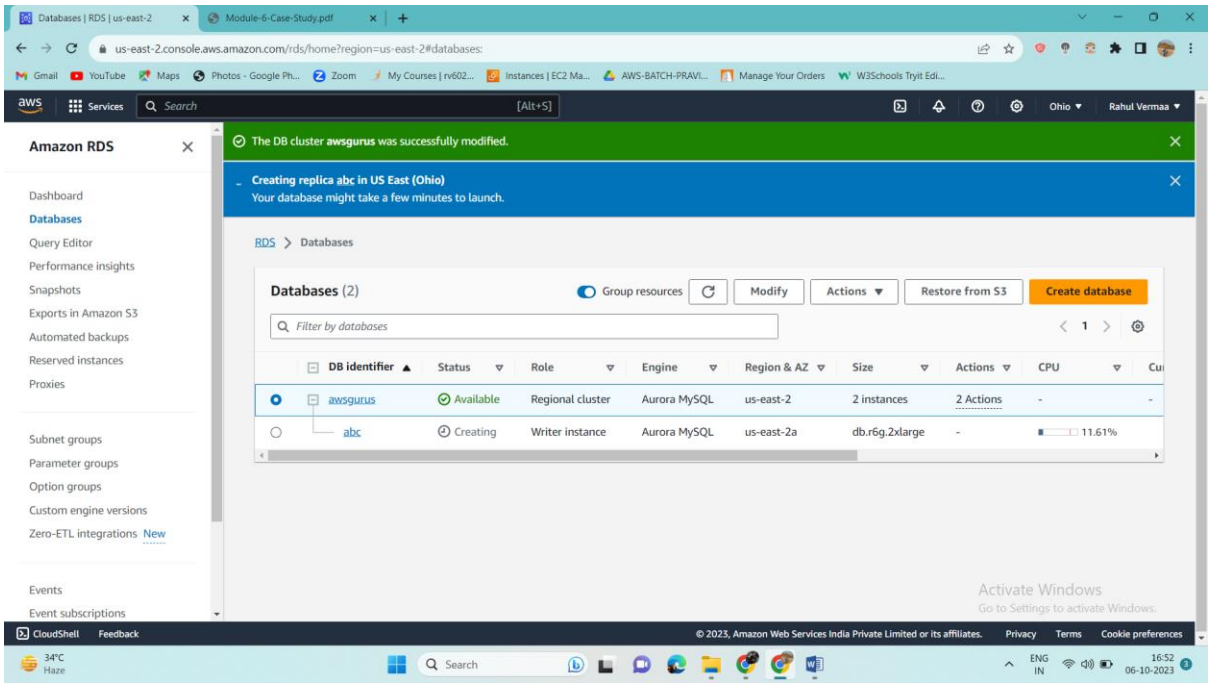


Now click on create database

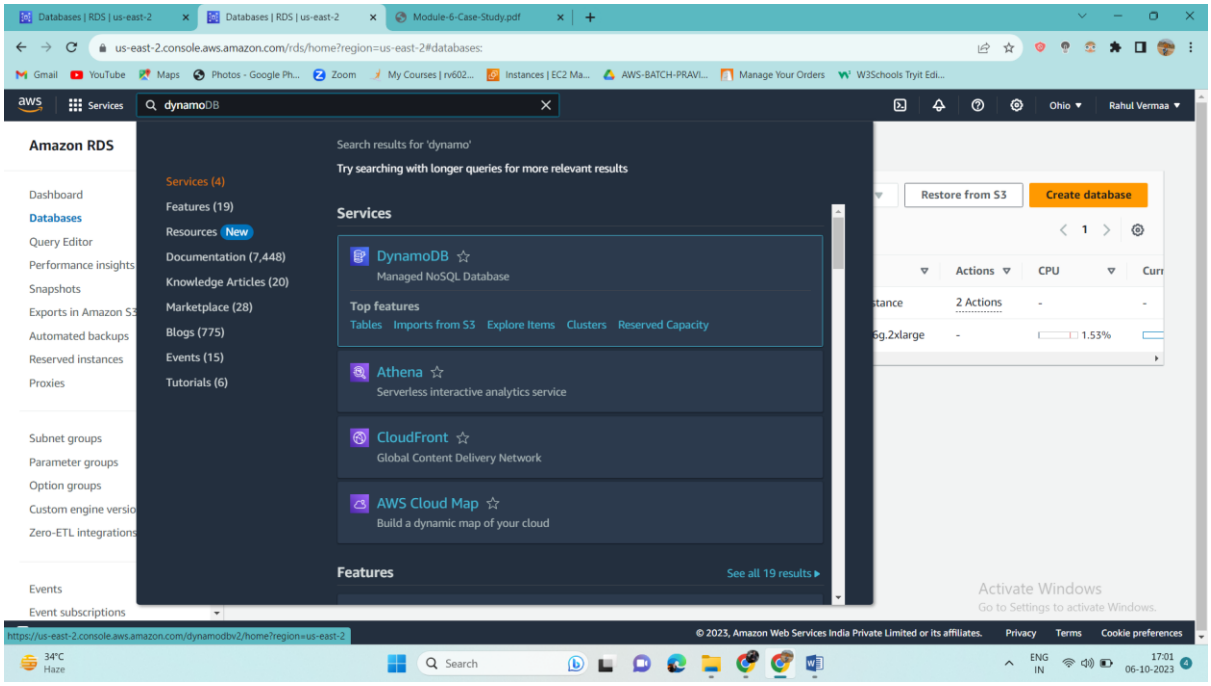


Our database is created successfully

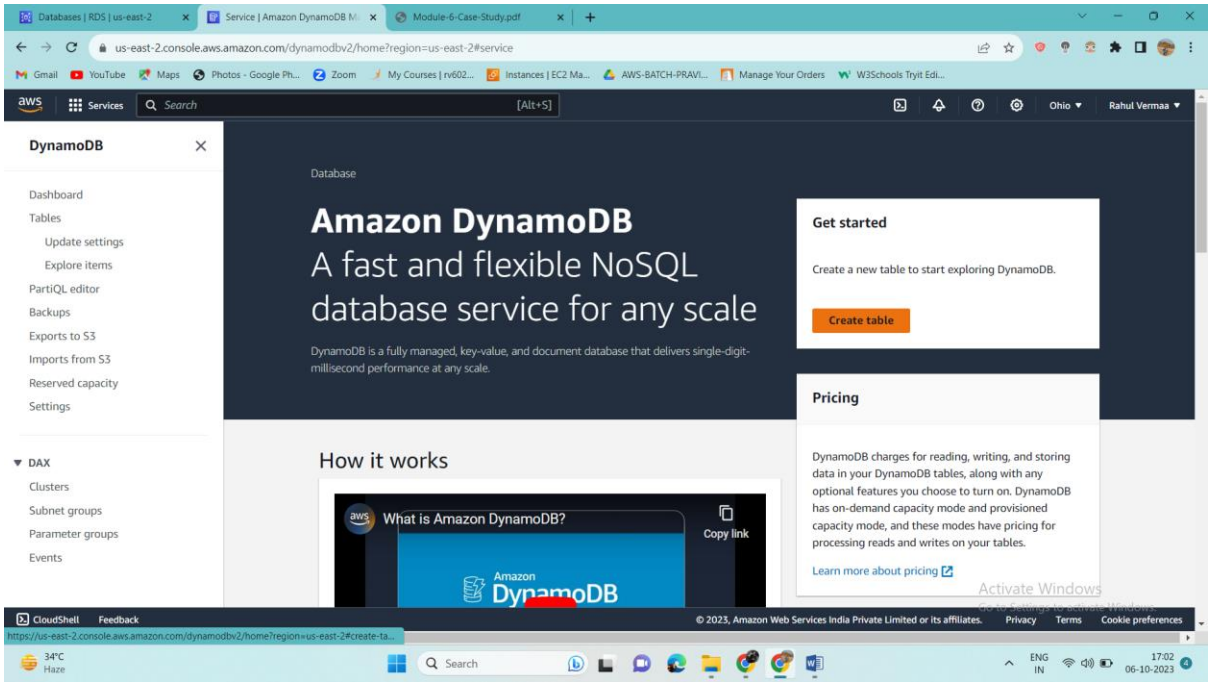
Now will create read replica for that just select your database and go to actions and click add reader



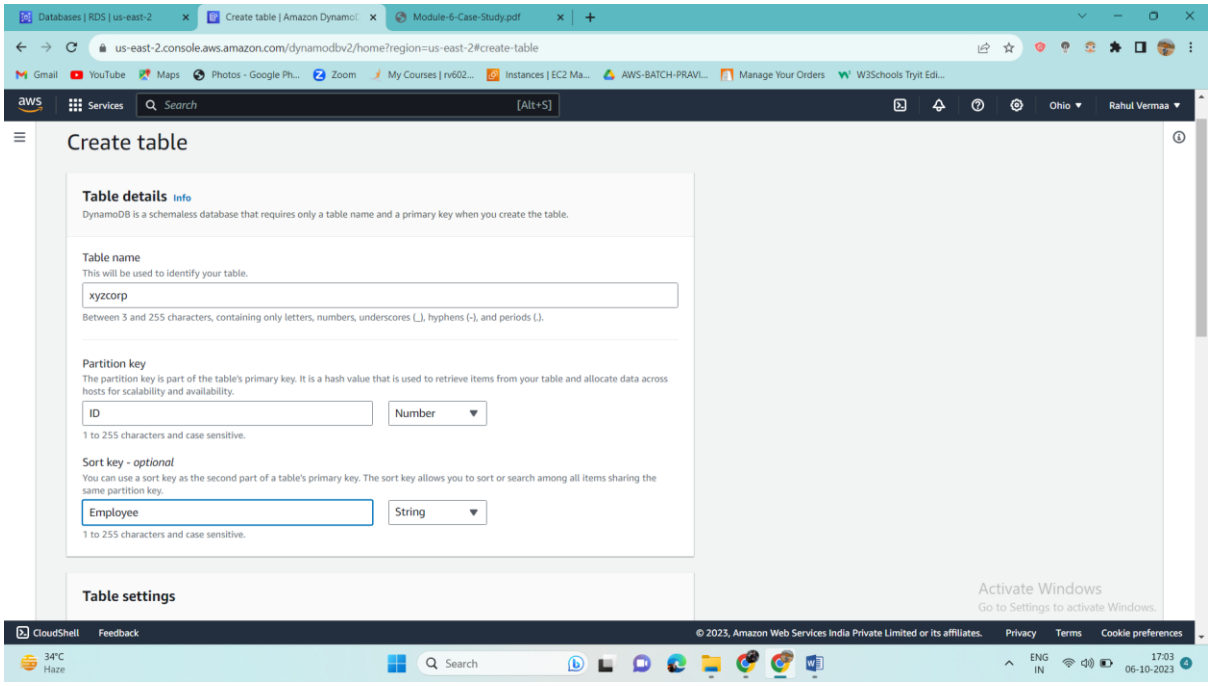
Establish a Database Architecture to process the data collected for near real-time analysis. Now for this we have to create DynamoDB



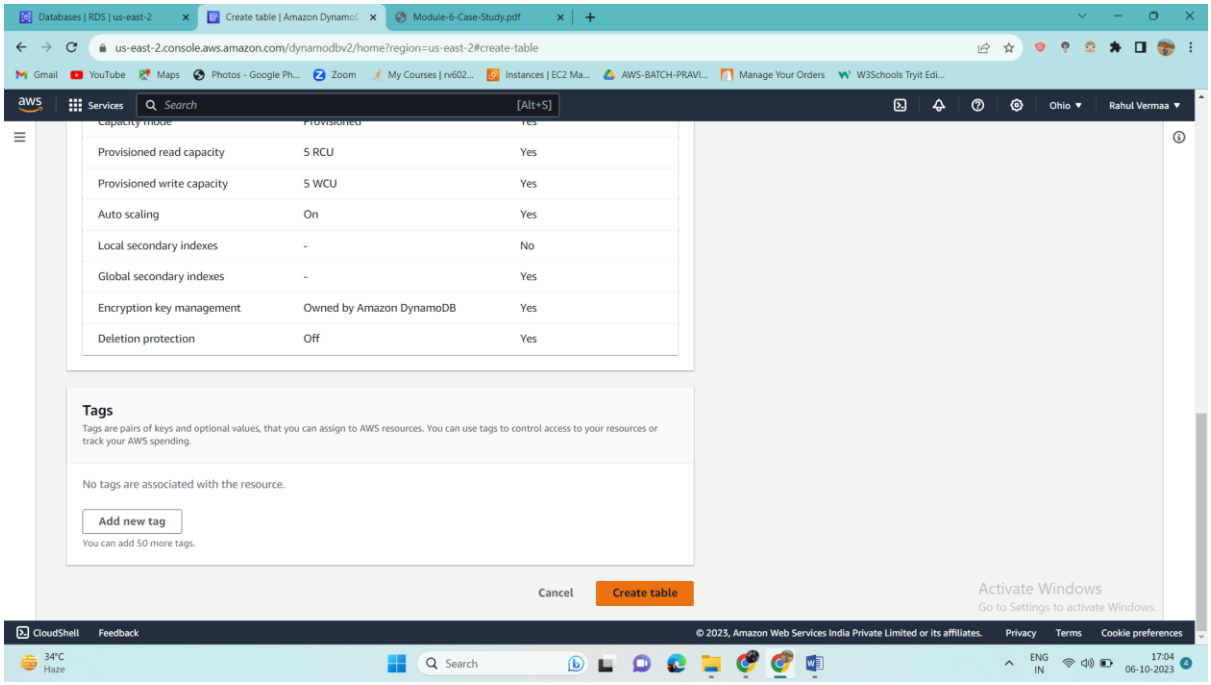
Click on create table



Enter the details and select options

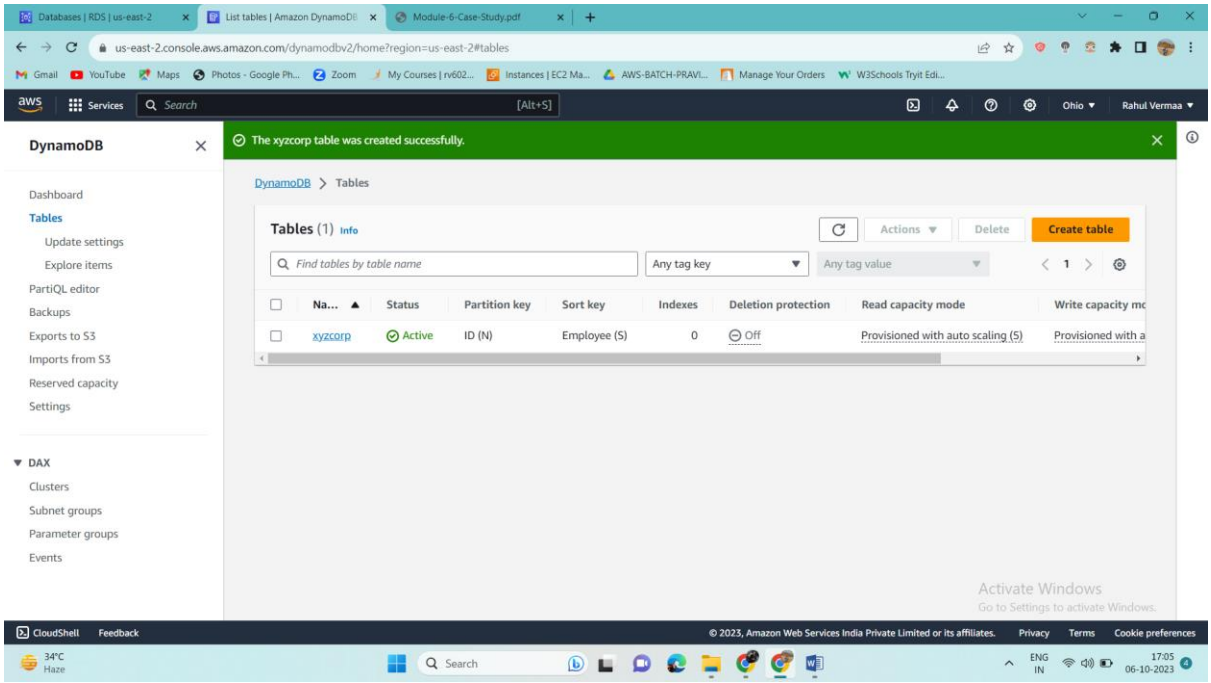


Now click on create table



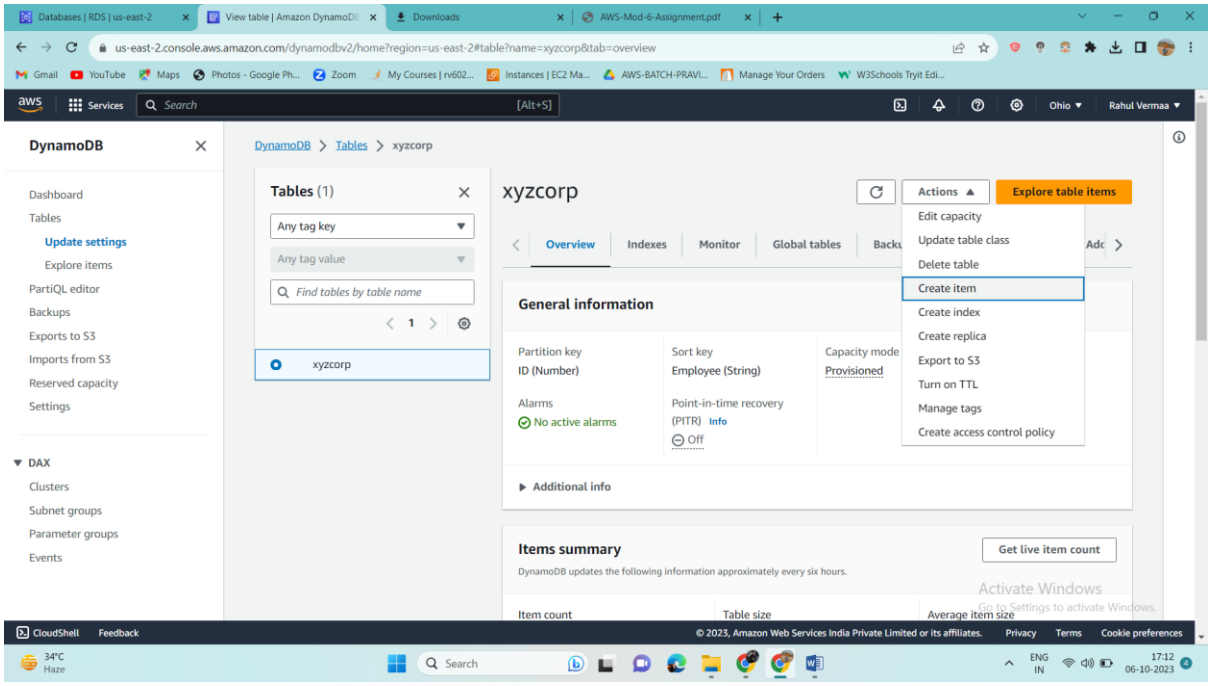
Our table is successfully created.

As it was realized that most of the time, same kind of data is being fetched and sent to the branches, so we can do multi-AZ deployment as shown in point# 2. This can also be achieved by enabling **Auto Scaling option**.

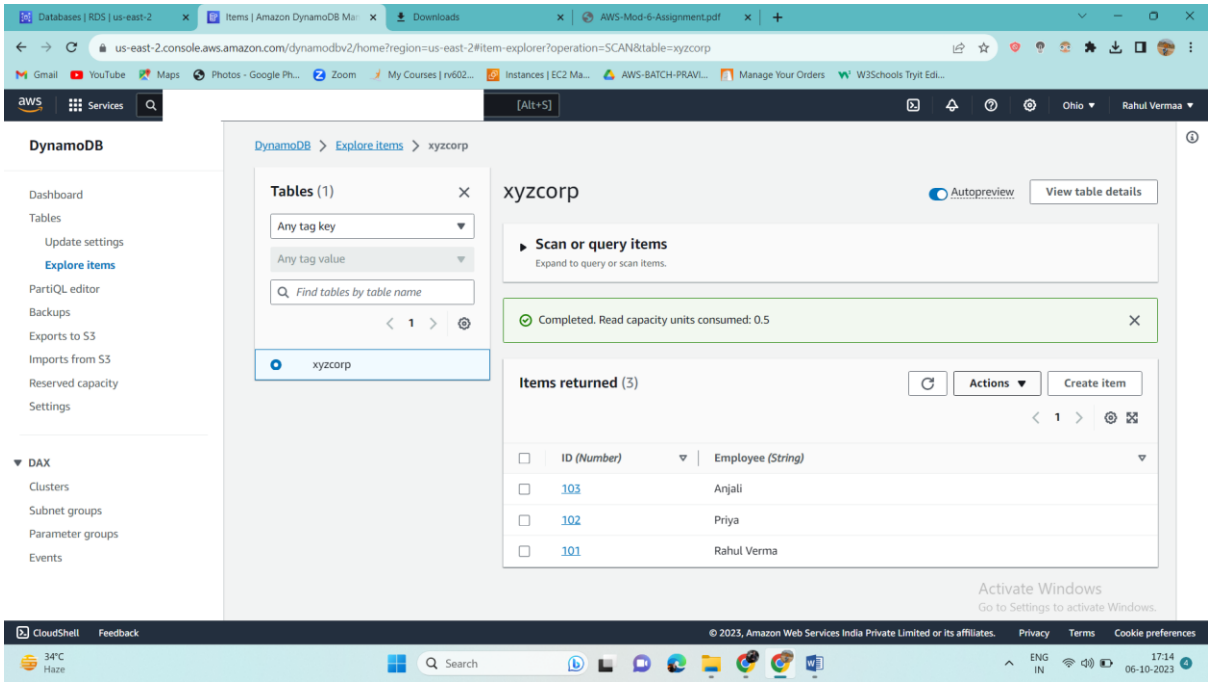


Now we will upload data to a DynamoDB table

For that select your table go to actions and click on create items

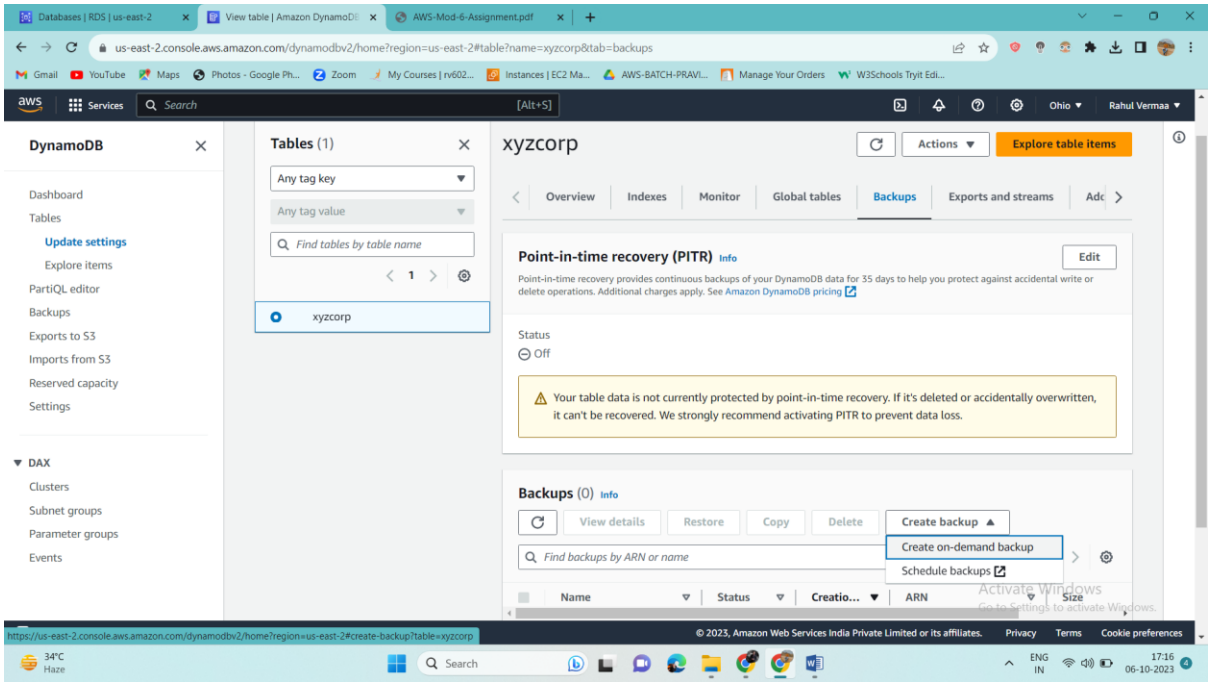


We have added data

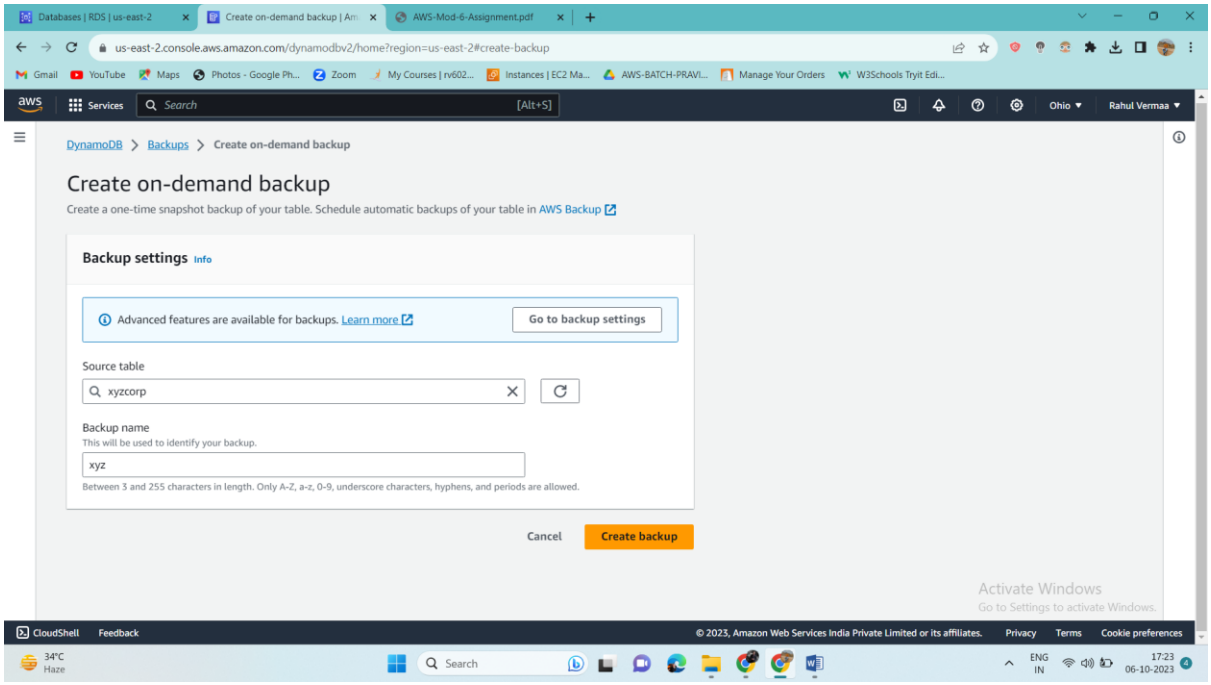


Now will take a backup of the table

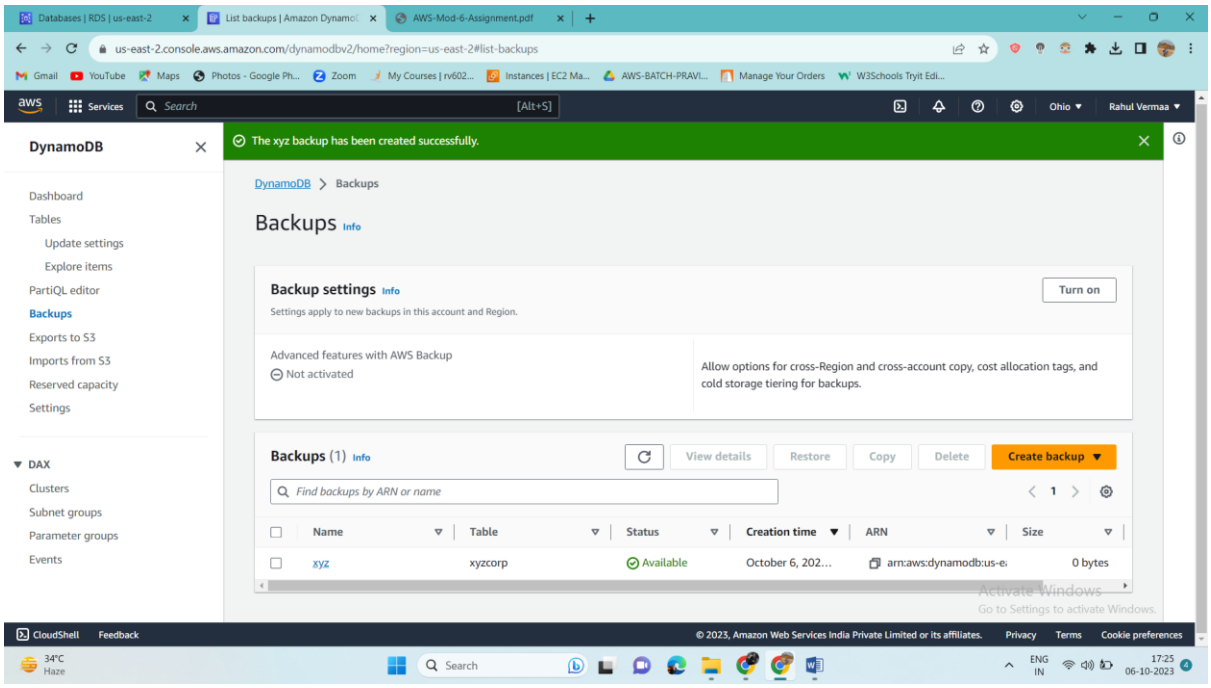
Select your table and to backup option and click on- create backup



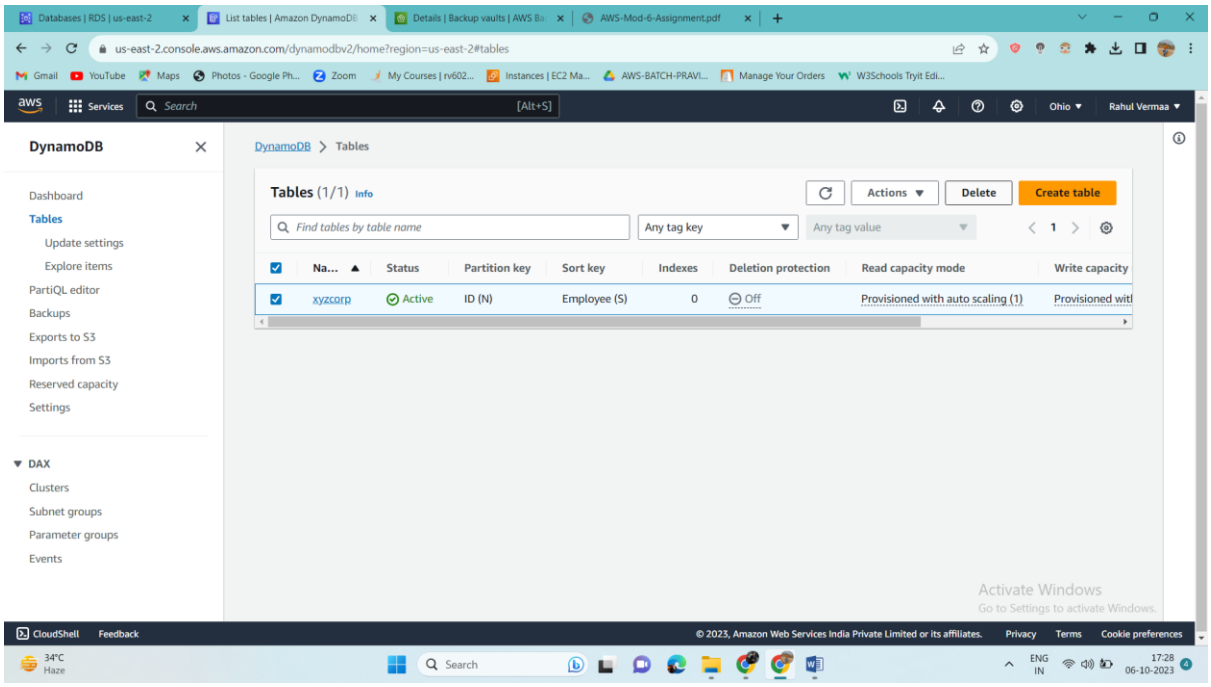
Select options and click on create backup



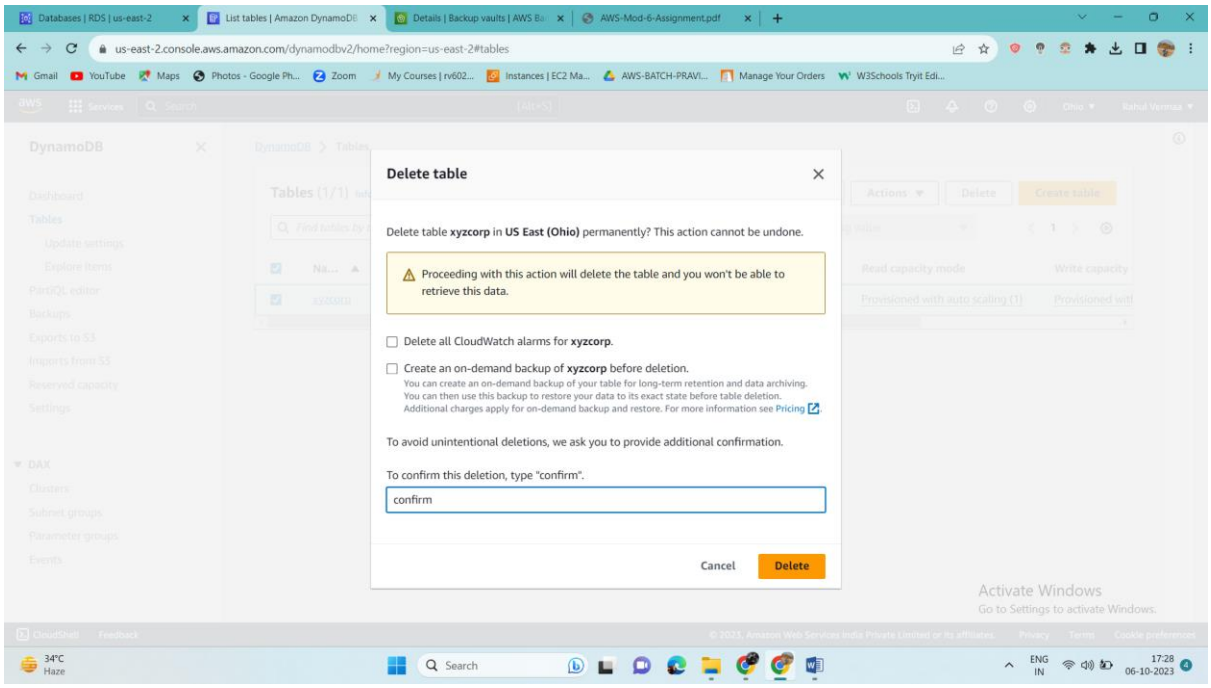
Backup is successfully created



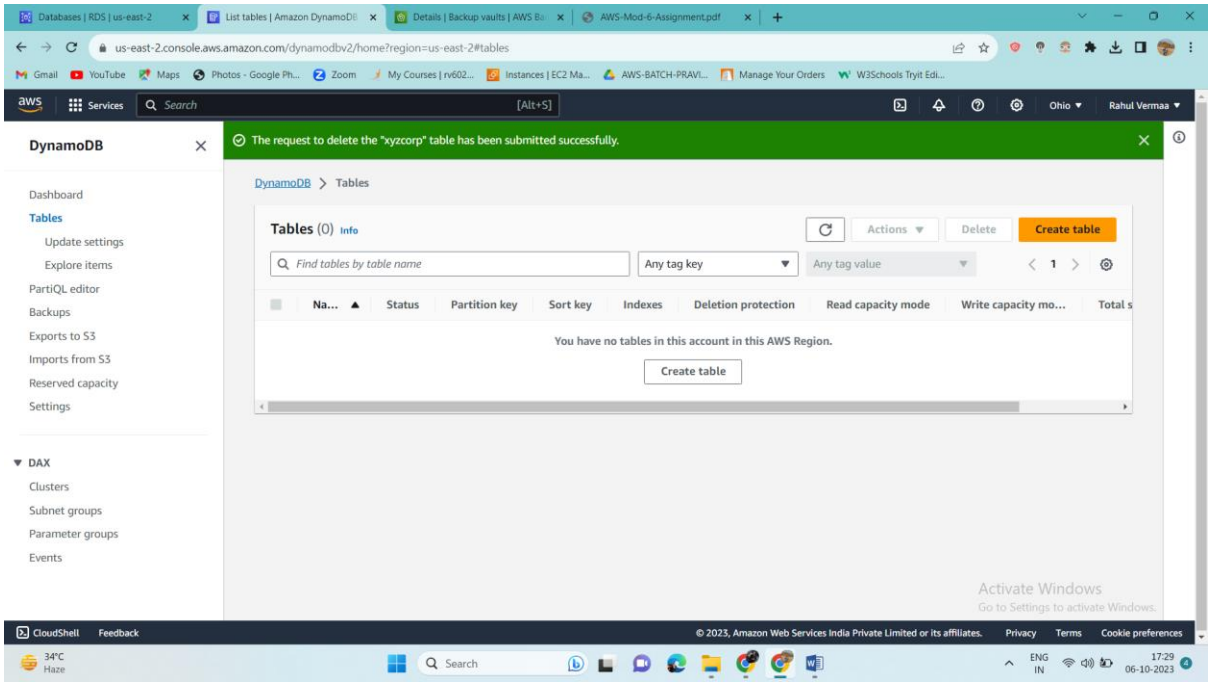
Now we will delete our table



While deleting also we select backup option

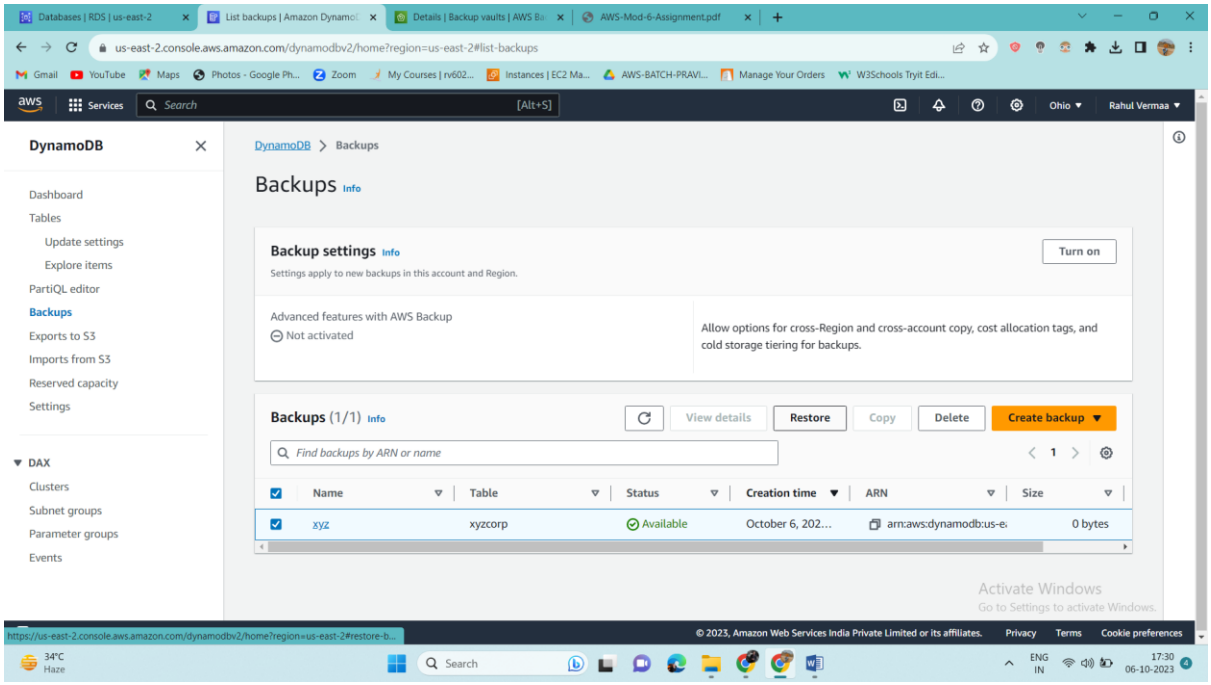


Our table is deleted

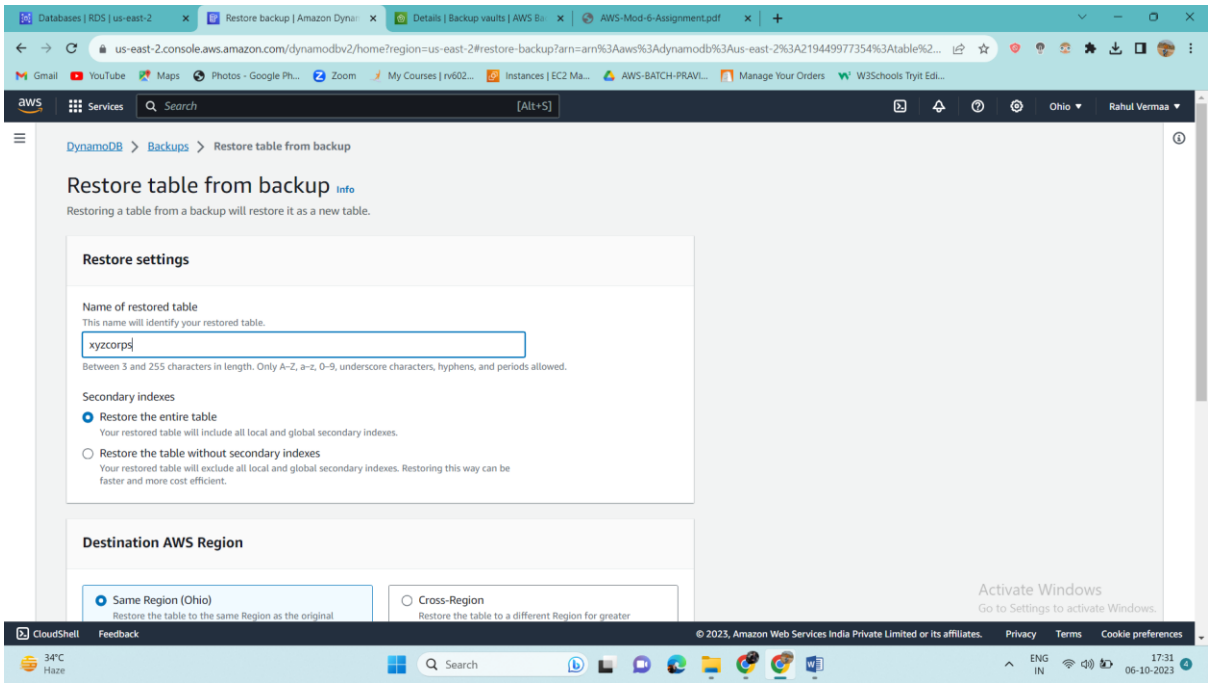


Now we will try to restore it using our backup

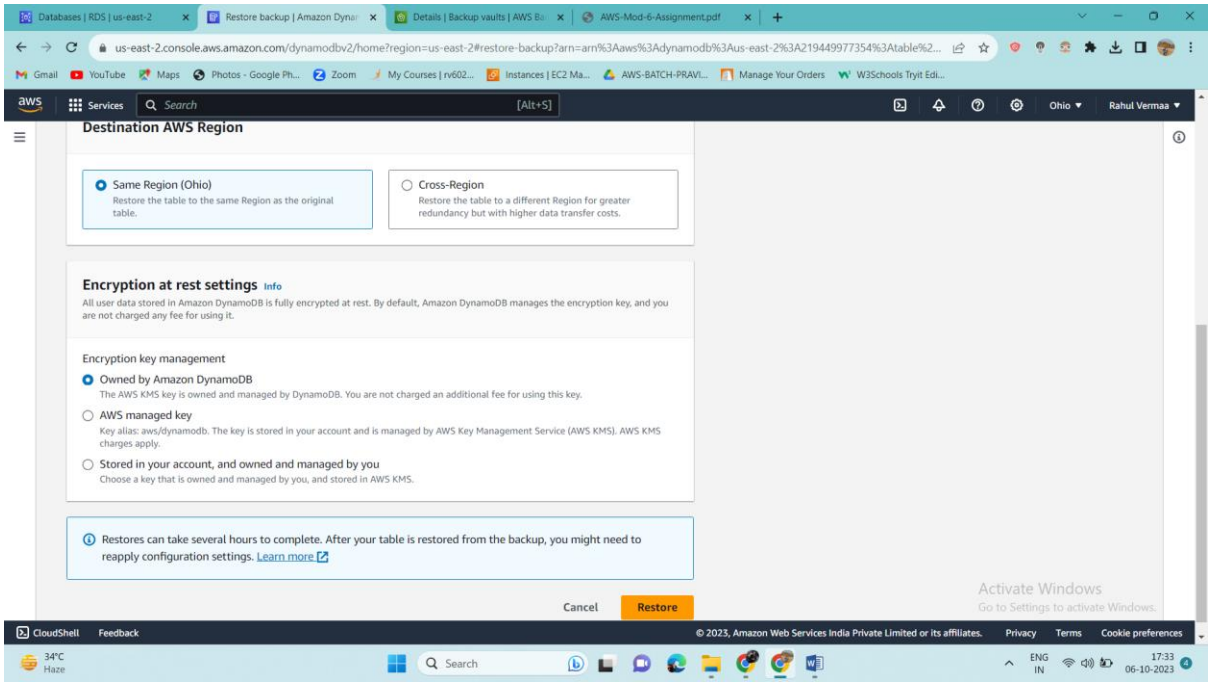
Will select our backup and now just click on Restore



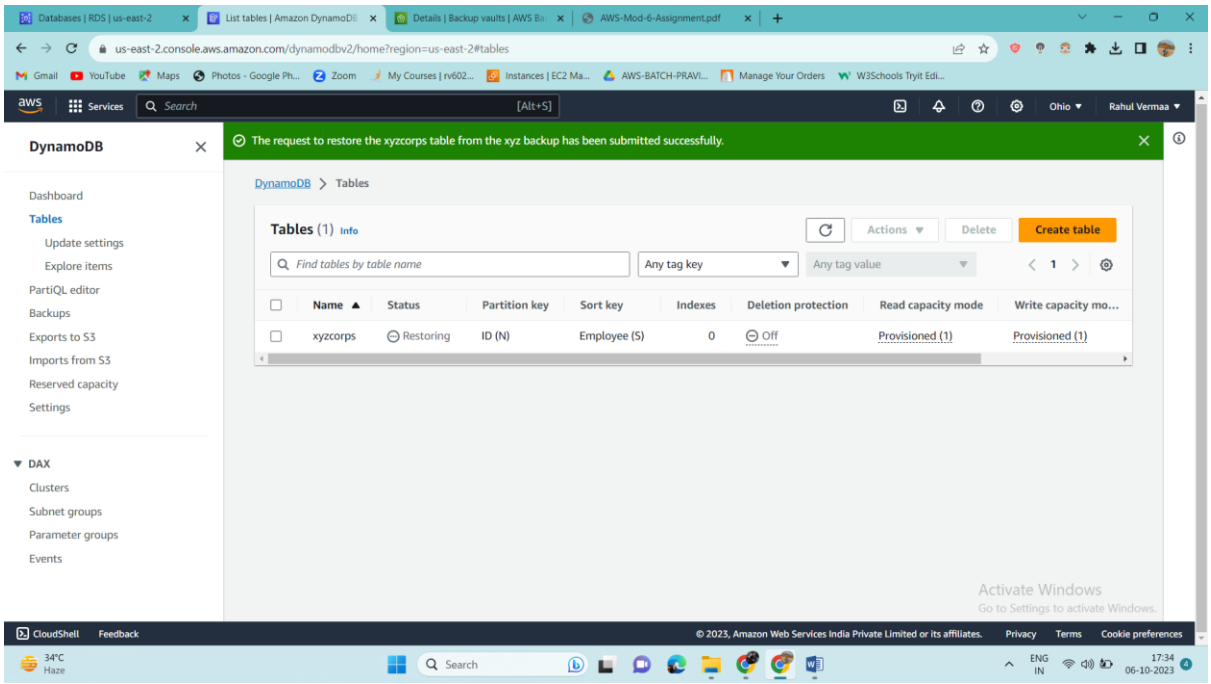
Will define our table name and we want our entire table to be restored so will select first option



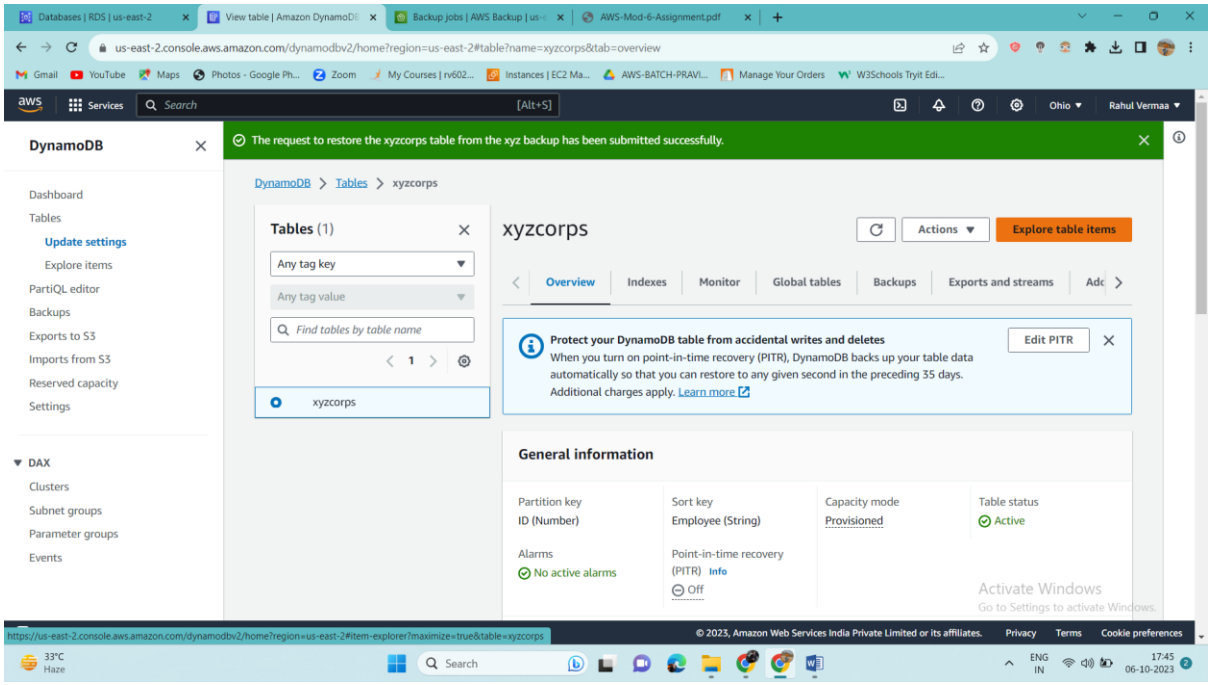
Now we will select same region rest default option and just click on restore



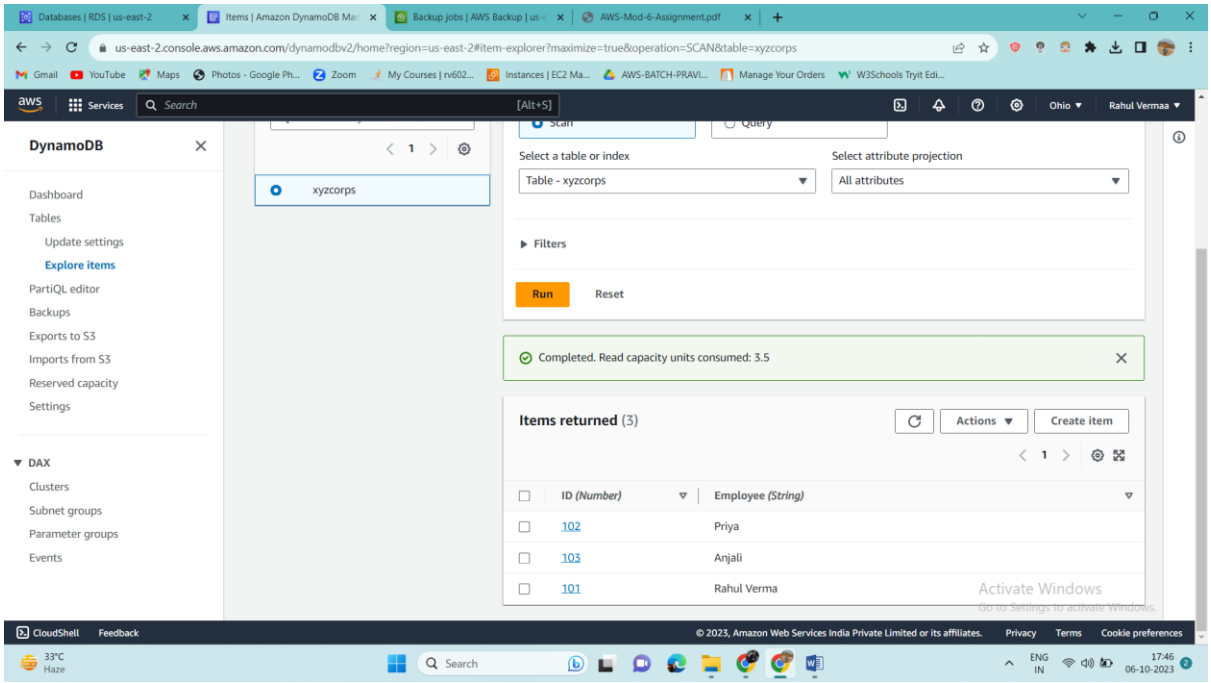
As you can see our table is created and it’s status is showing restoring



Now to check items select your table when it’s status turned to active state and click on expand items



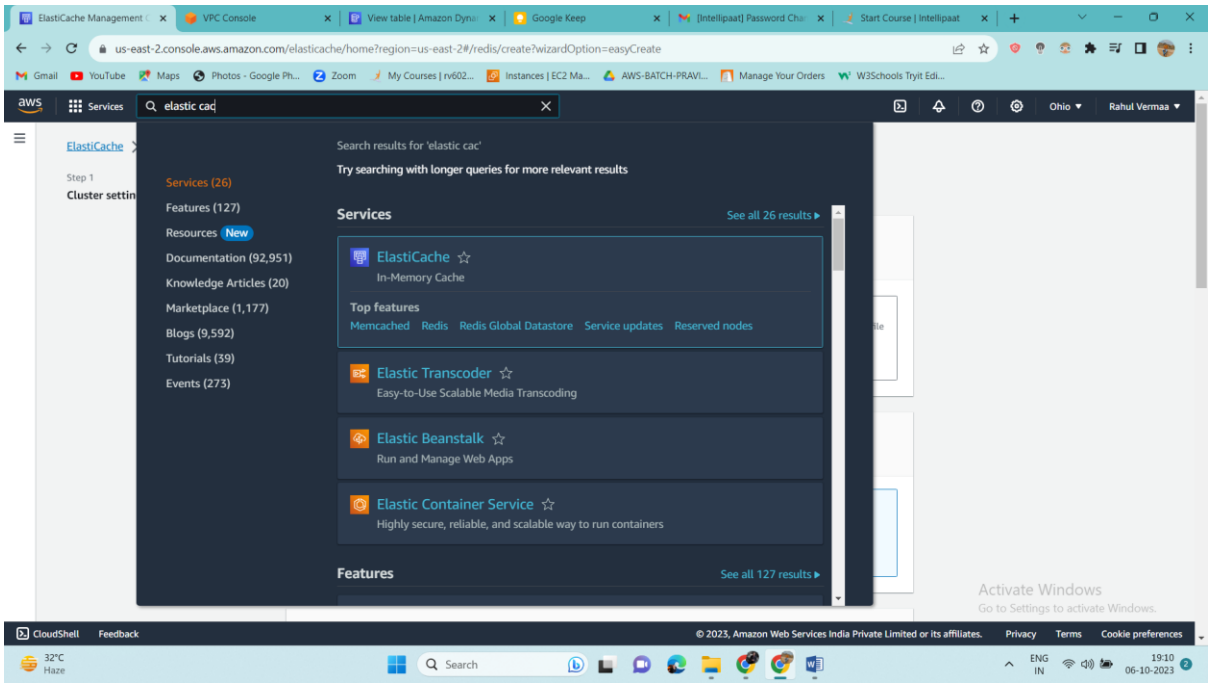
And our item is there



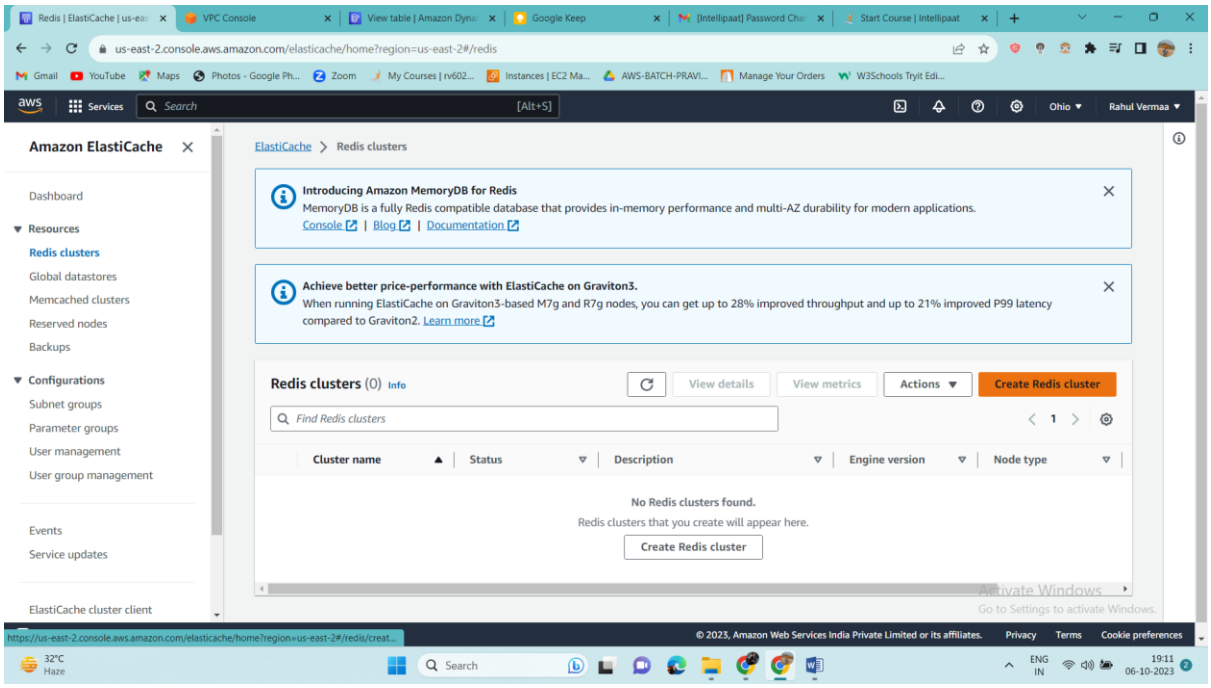
As the user base of the application increases, your company observes a significant latency in the transfer of data to the branches. It realizes that most of the time, the same kind of data is being fetched and sent to the branches. Therefore, you are asked to:

6. Find and implement the solution to resolve the latency issues

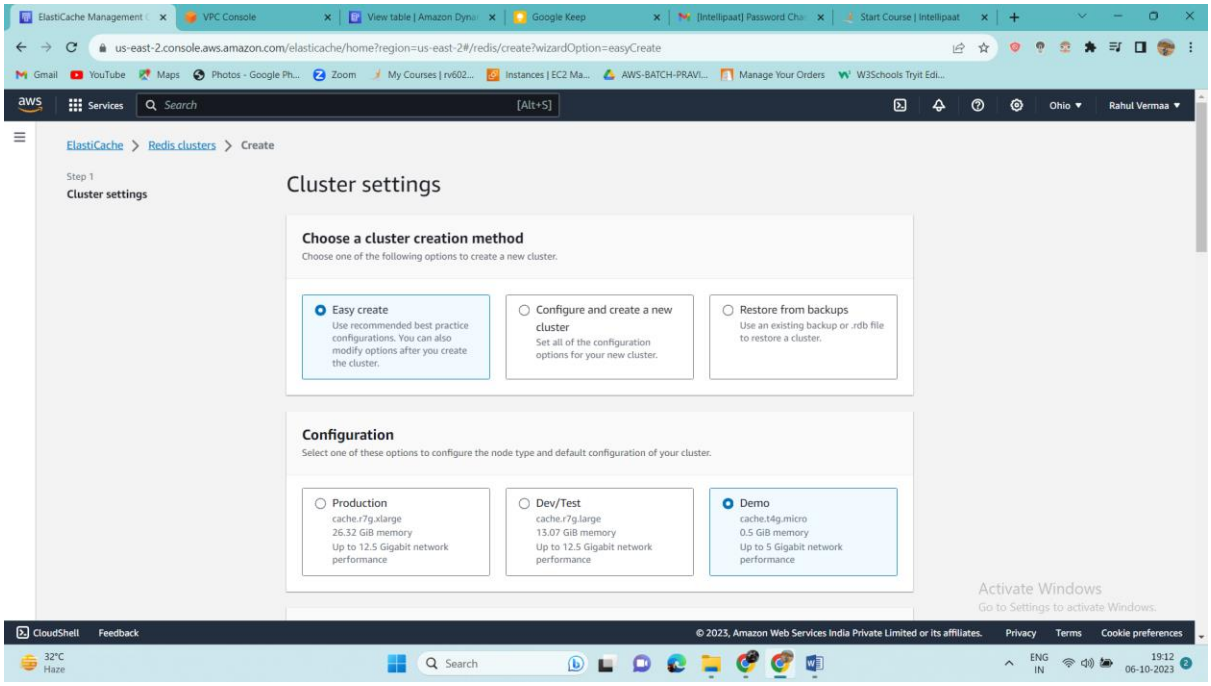
We can use elastic cache cluster in RDS



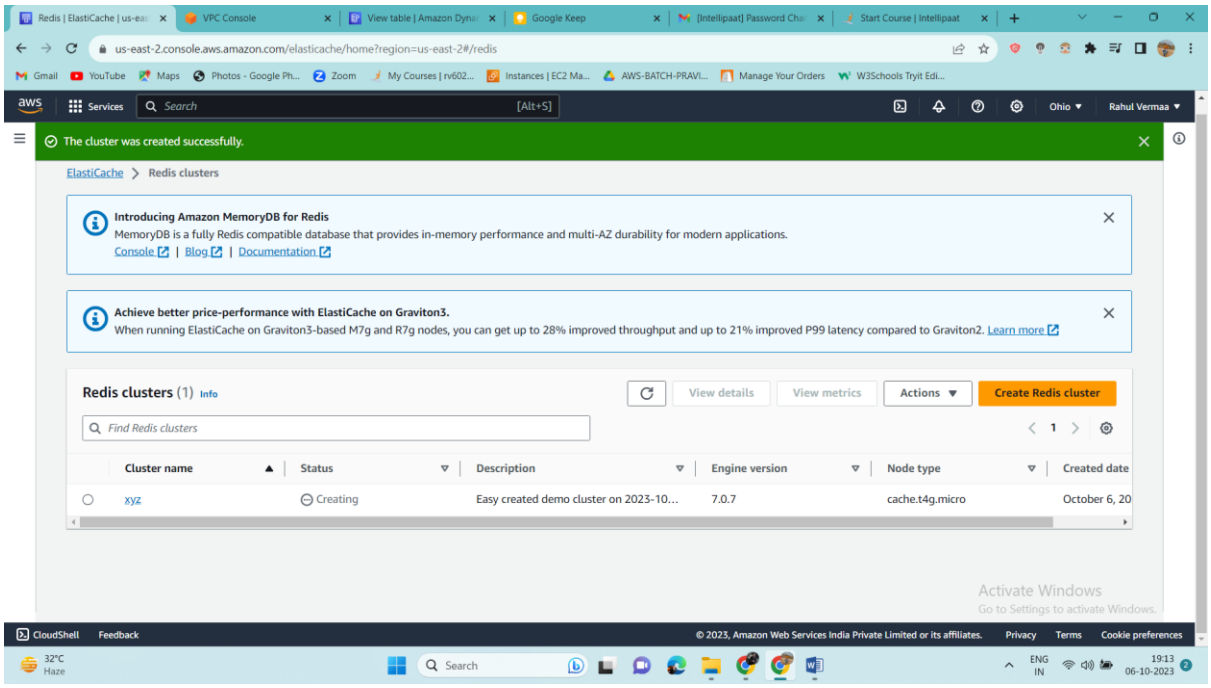
Now click on redis cluster



Select appropriate options and click on create button



And it’s created successfully



And it’s done