**RDS-Relational Database Service**

Amazon Relational Database Service (Amazon RDS) makes it easy to set up, operate and scale relational databases in the cloud. It provides cost-efficient and resizable capabilities while automating time-consuming administration tasks such as hardware provisioning, database setup, patching, and backup. It frees you up to focus on your applications so that you can provide them with the fast performance, high availability, security and compatibility they need.

Amazon RDS is available on multiple database instance types - optimized for memory, performance or I/O - and gives you six familiar database engines to choose from, including Amazon Aurora, PostgreSQL, MySQL, Maria DB, Oracle Database and SQL Server Huh. You can use the AWS Database Migration Service to migrate easily or replicate your existing databases to Amazon RDS.

**How does Amazon RDS work?**

Databases store large amounts of data that applications can draw upon to help them perform various tasks. A relational database uses tables to store data and is called relational because it organizes data points with defined relationships.

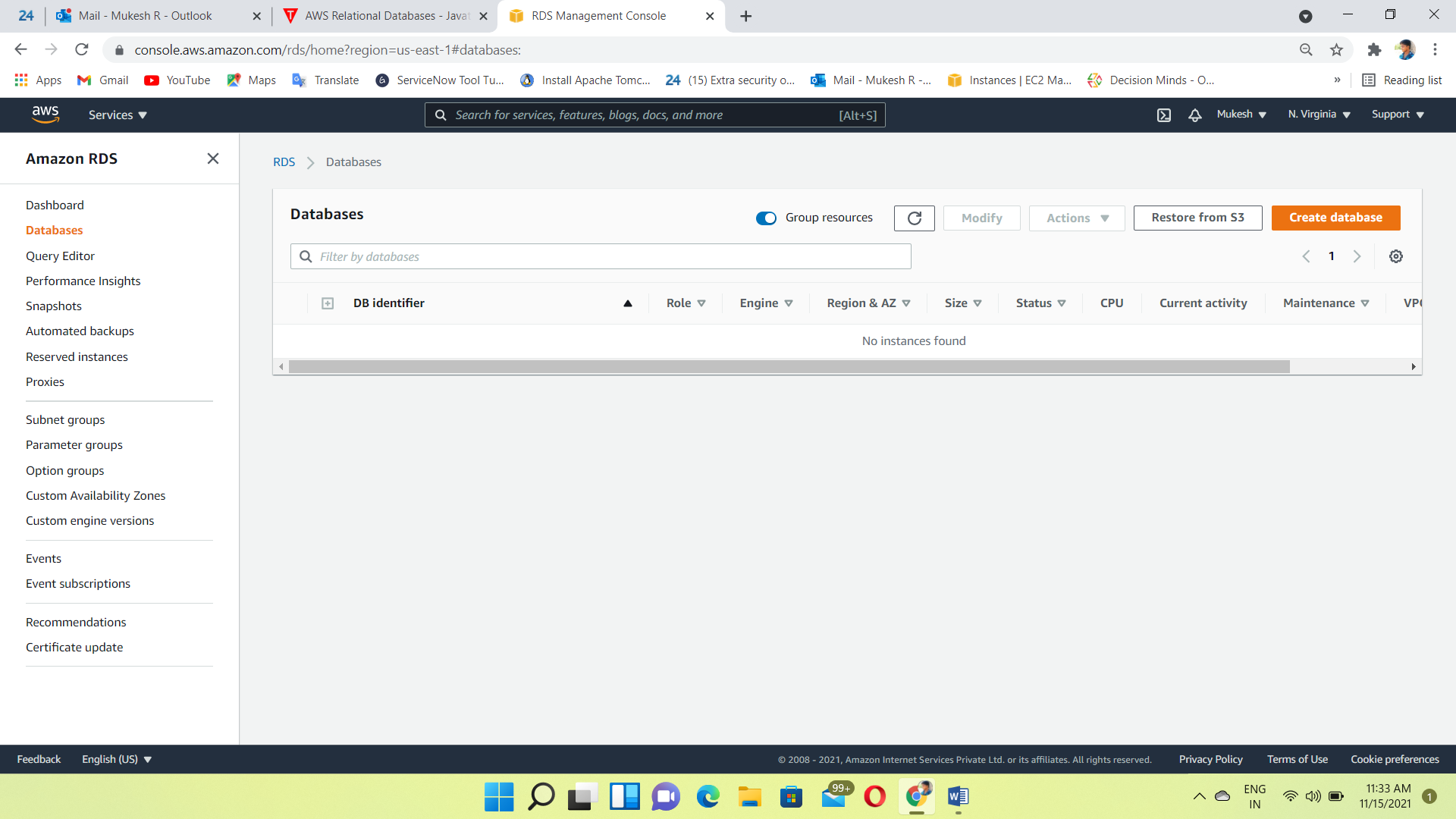
Administrators control Amazon RDS with the AWS Management Console, Amazon RDS API calls, or the AWS command-line interface. They use these interfaces to deploy database instances to which users can apply specific settings.

Amazon provides several instance types with different resources, such as CPU, memory, storage options, and networking capability. Each type comes in a variety of sizes to suit the needs of different workloads.

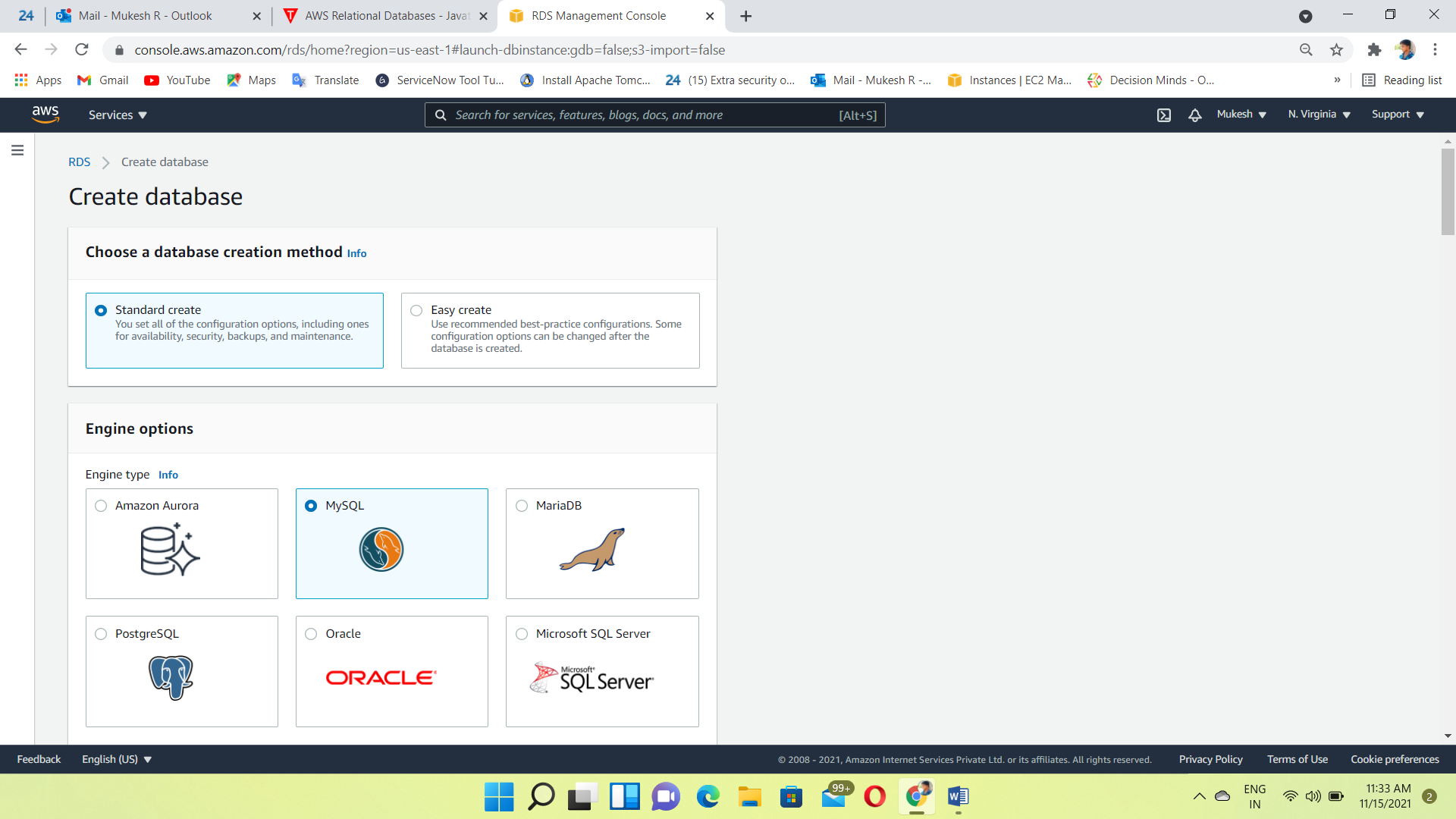
RDS users can use AWS Identity and Access Management to define and set permissions to access RDS databases.

**LAB:**

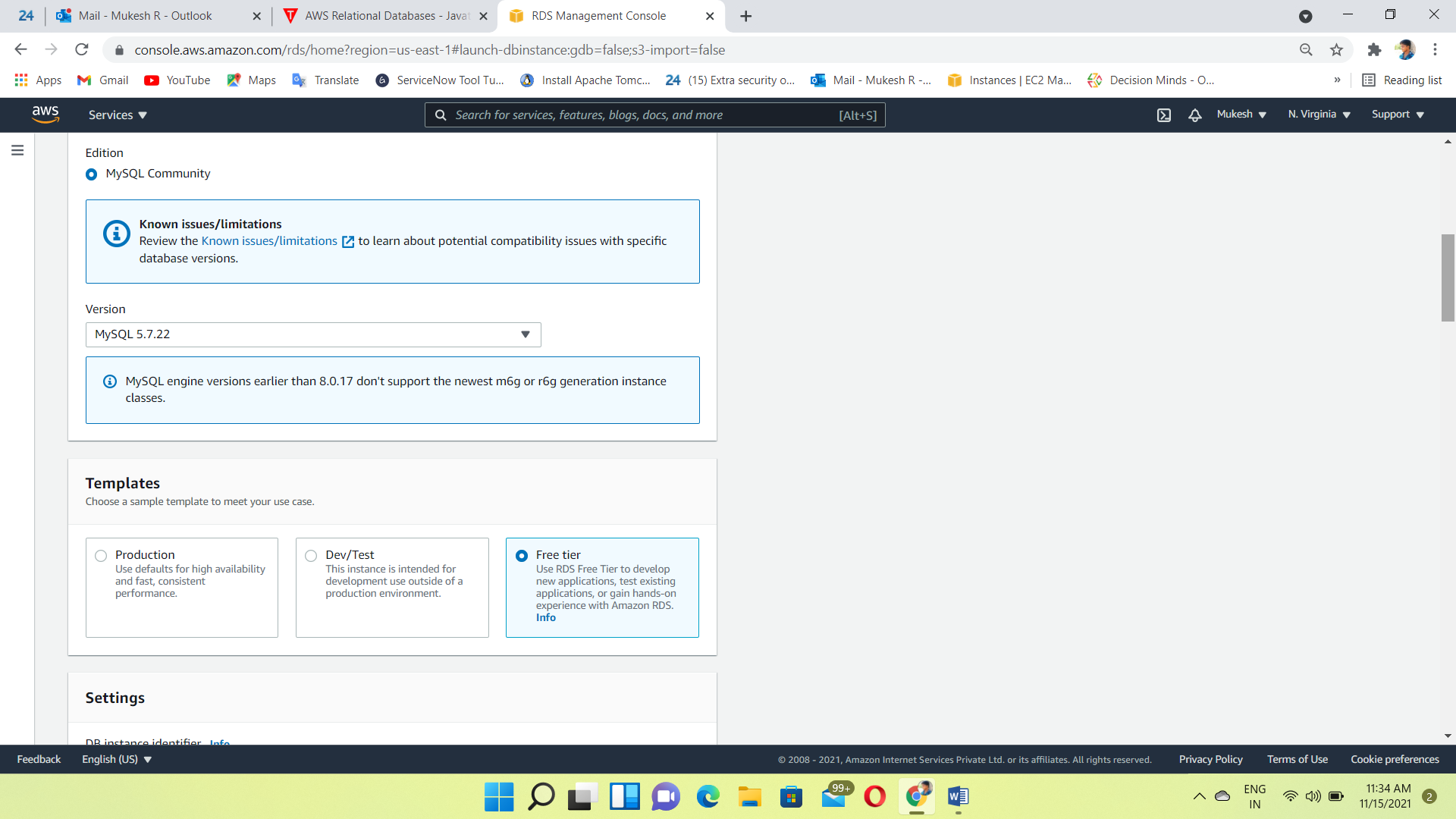
Create a Database



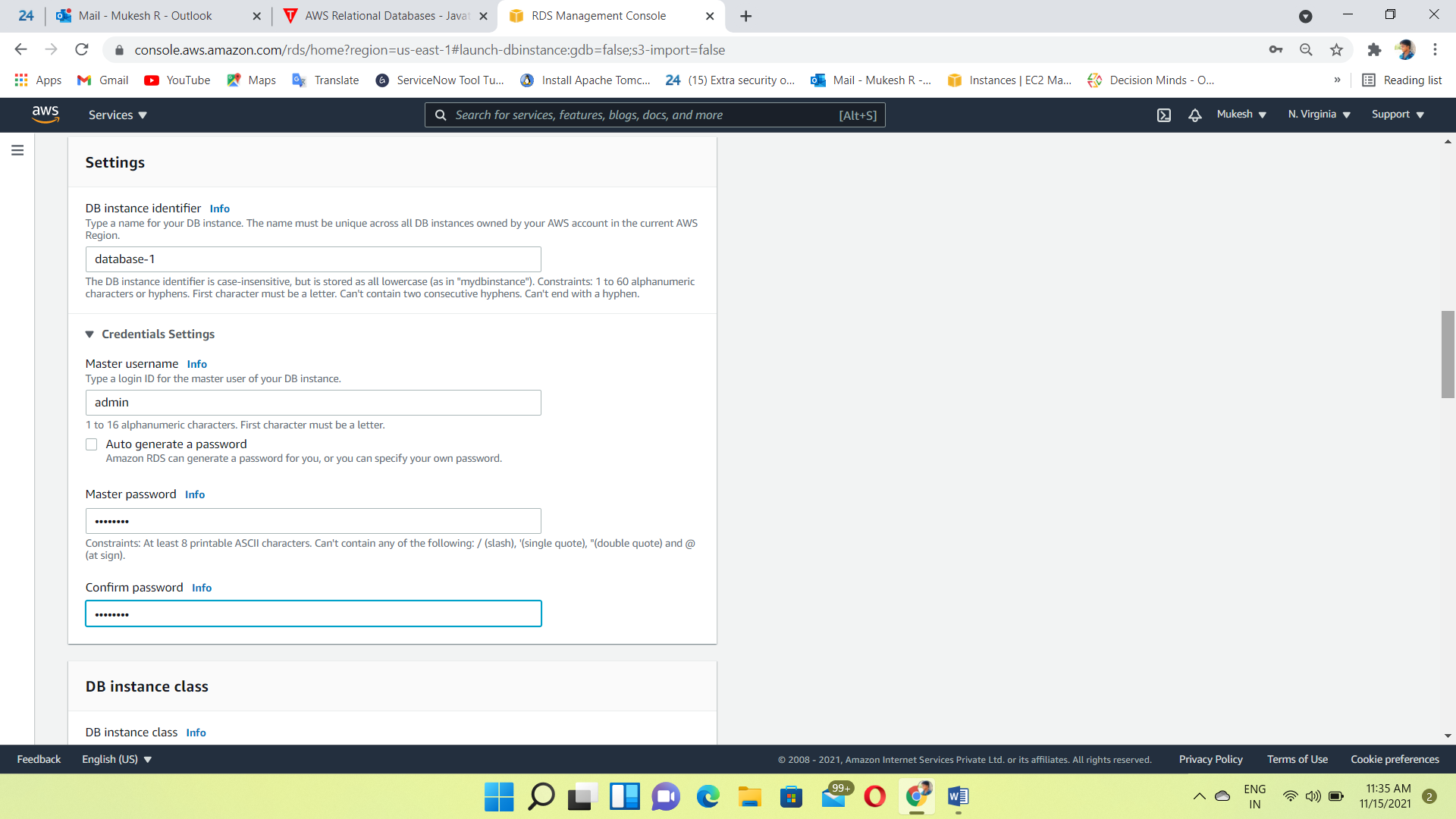
Select MySQL engine type



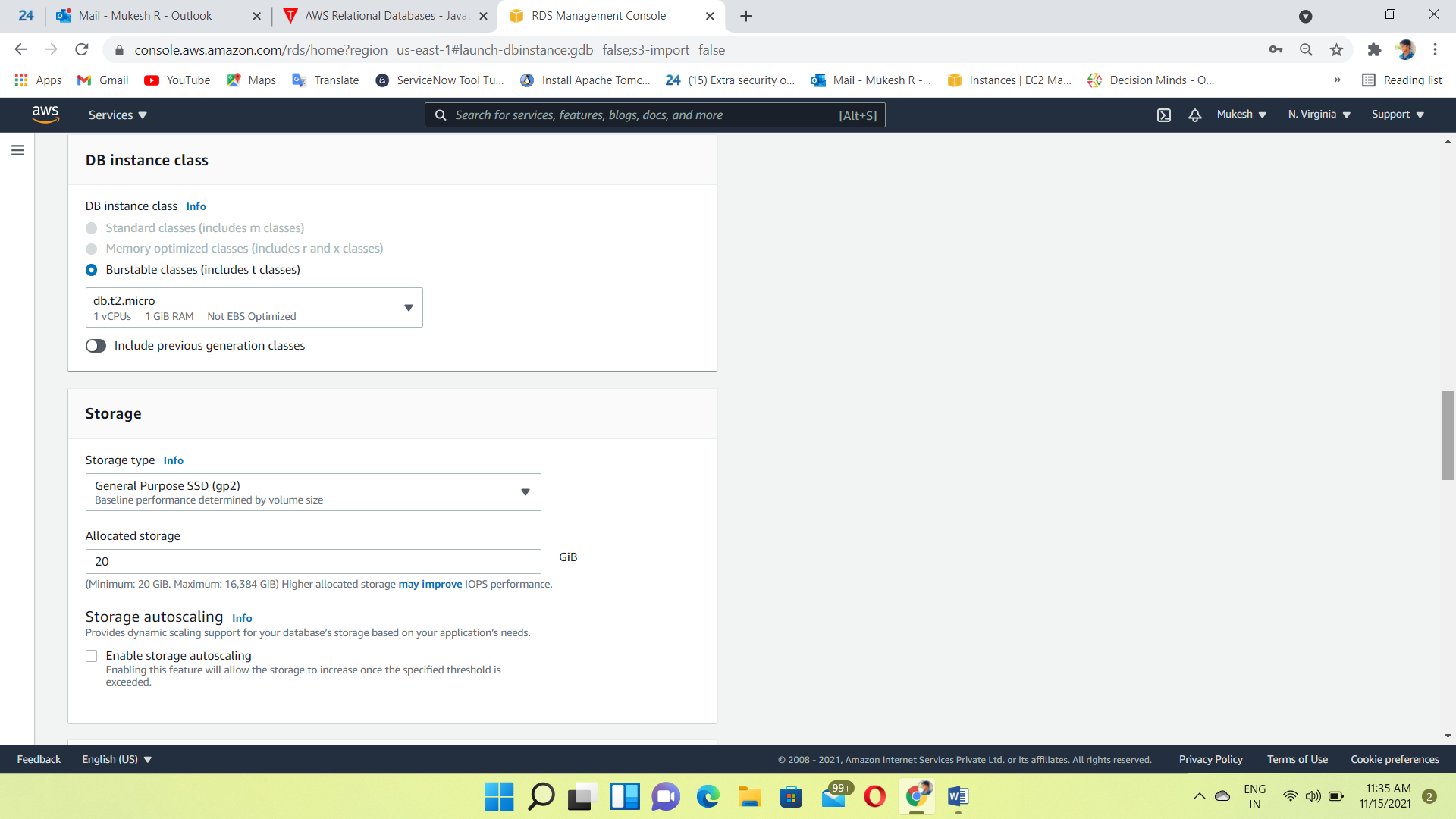
Select Version 5.7.22 and choose free tier



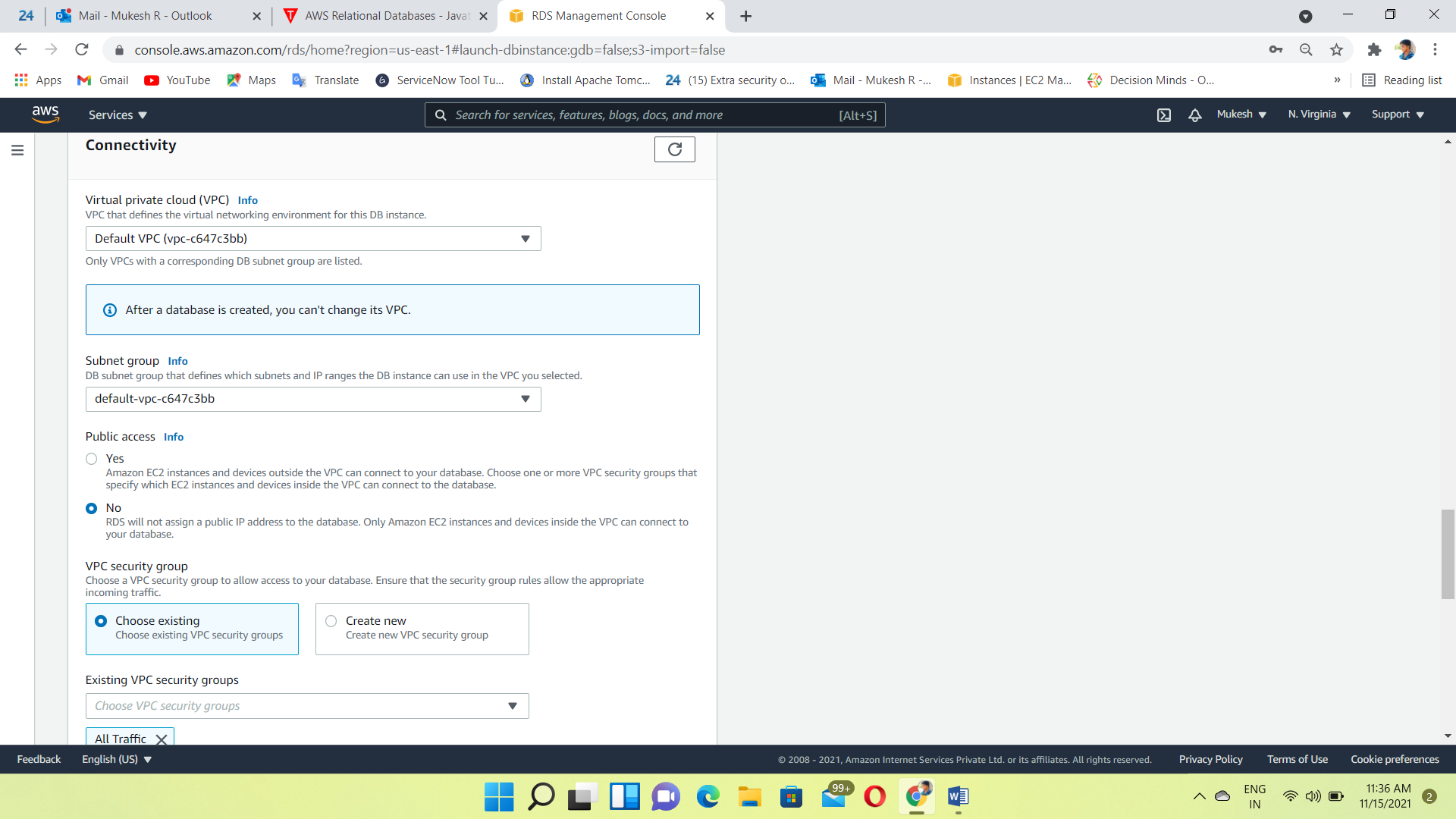
Give to the DB instance name and Username , Password



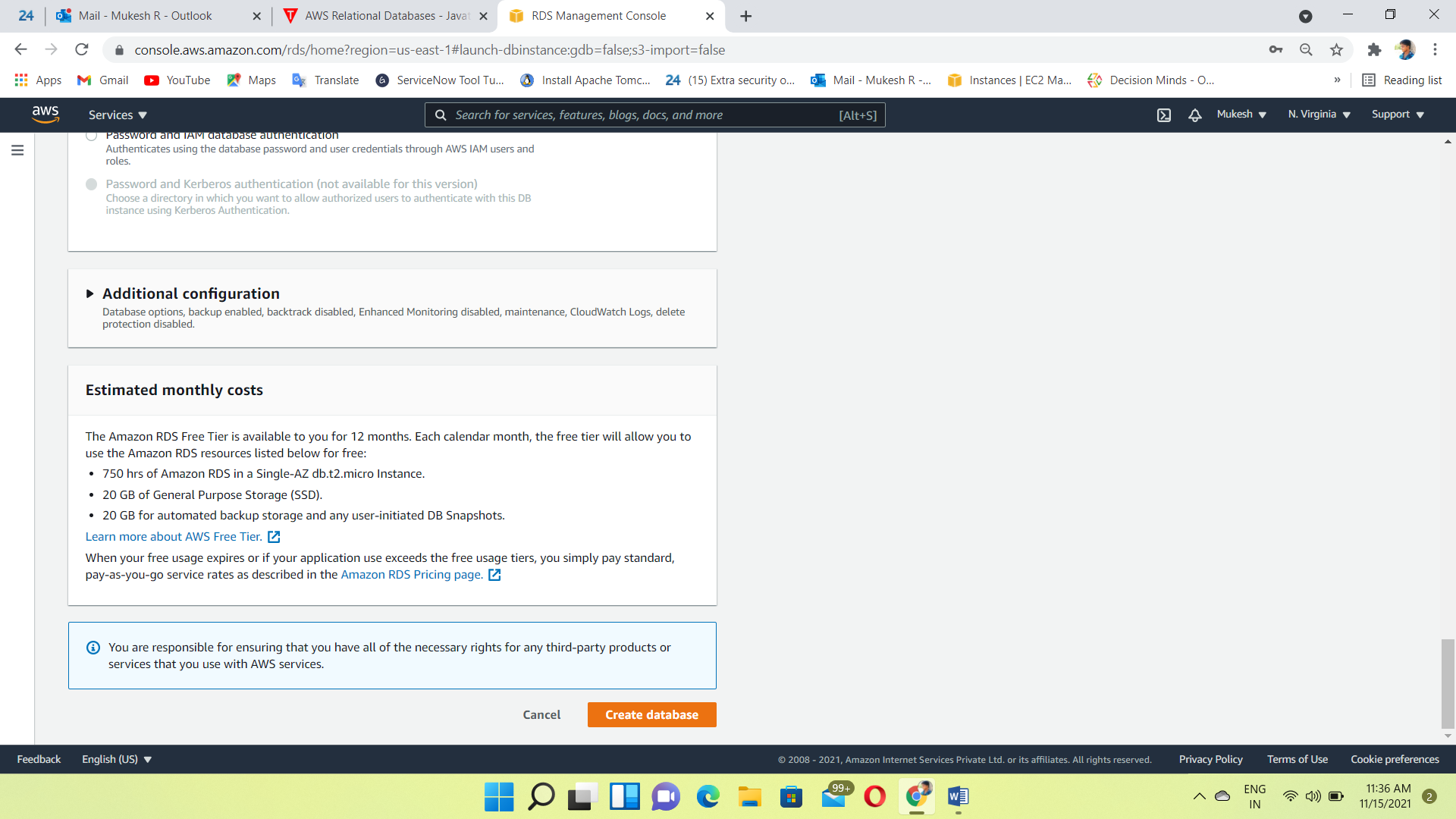
Allocated the storage and Disable the storage auto scaling



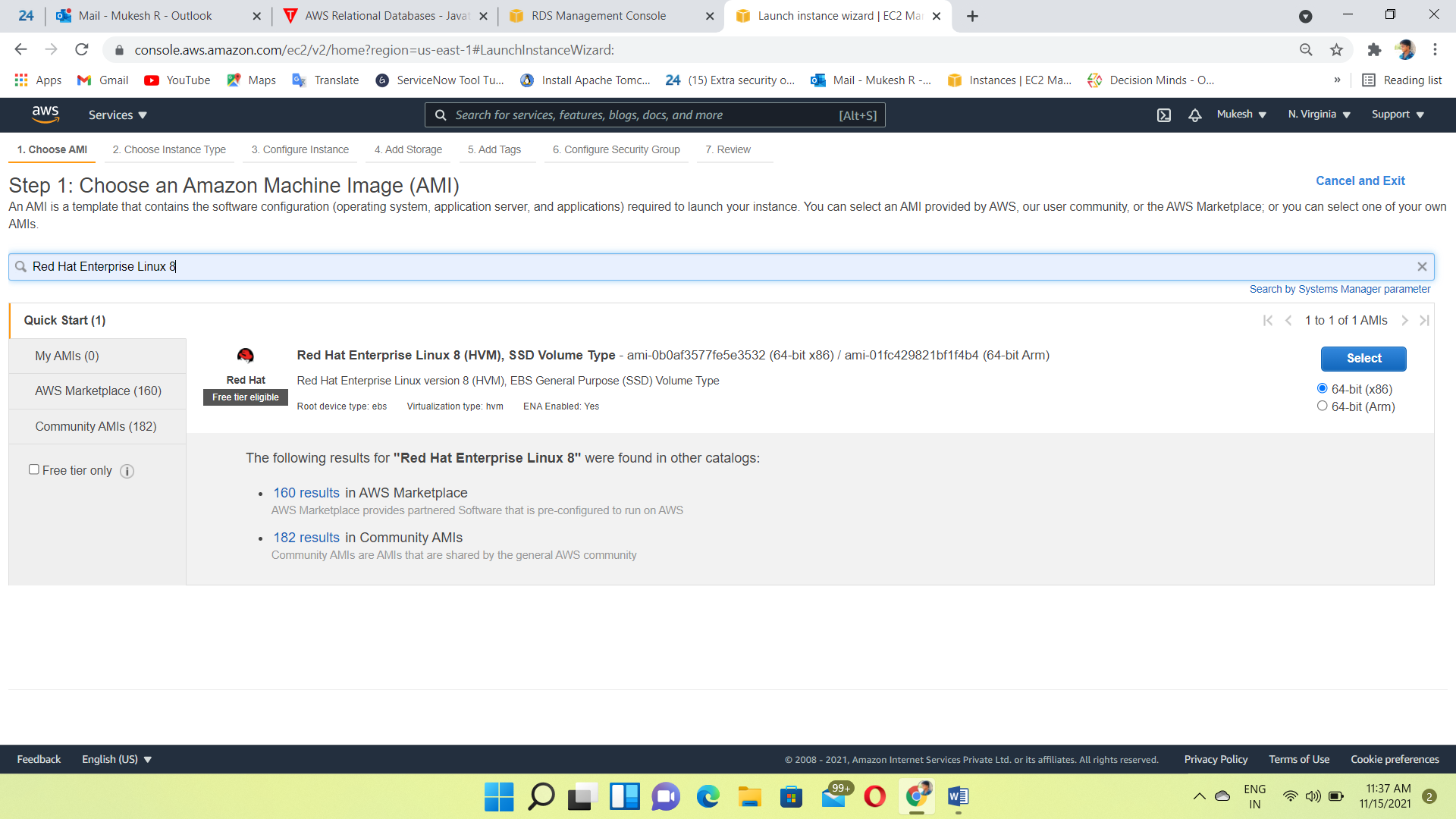
Select the DB instance details🡪VPC, Subnet, Security Group



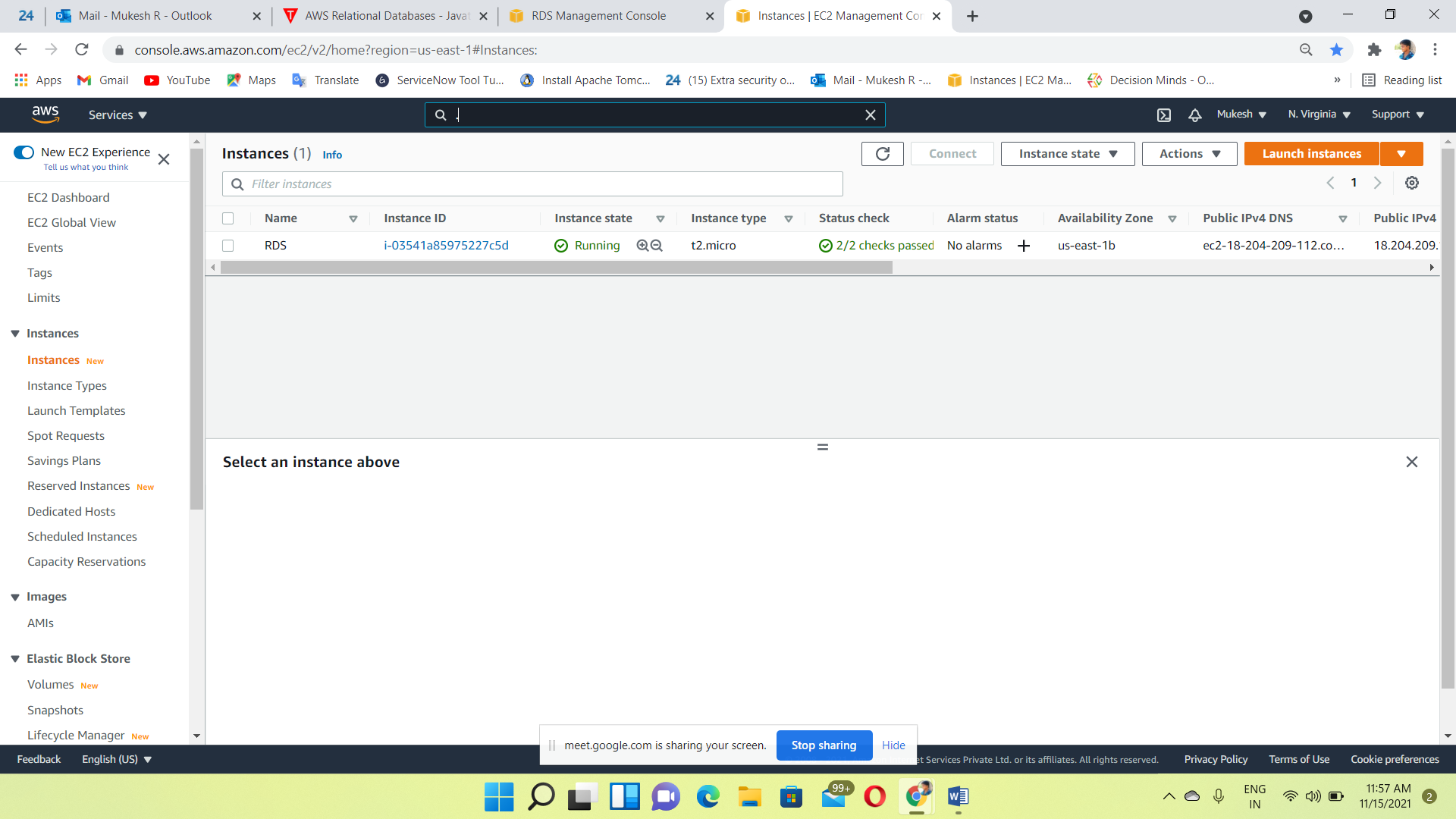
Click on Create Database



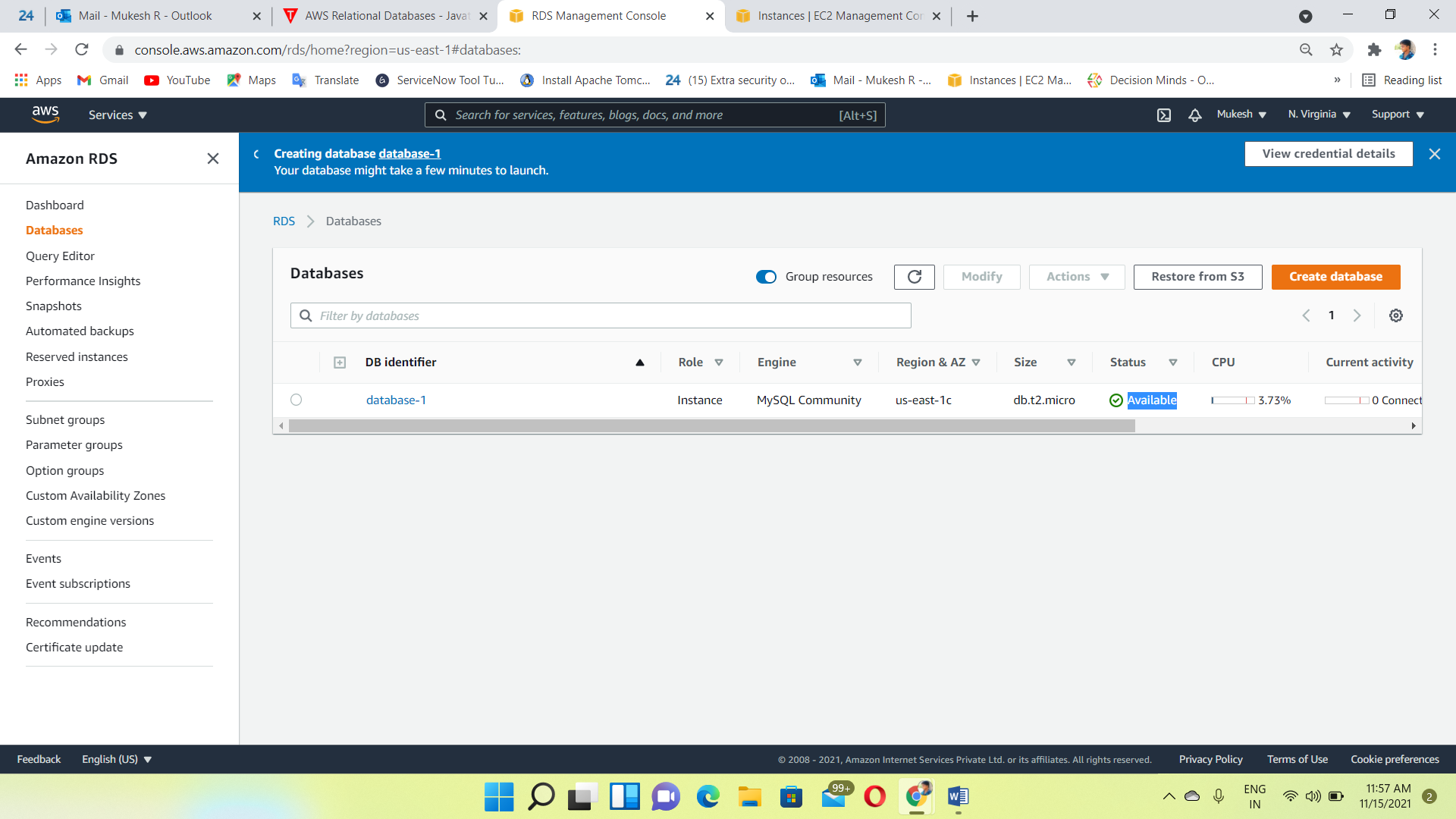
Create a RHEL8 instance for frontend server



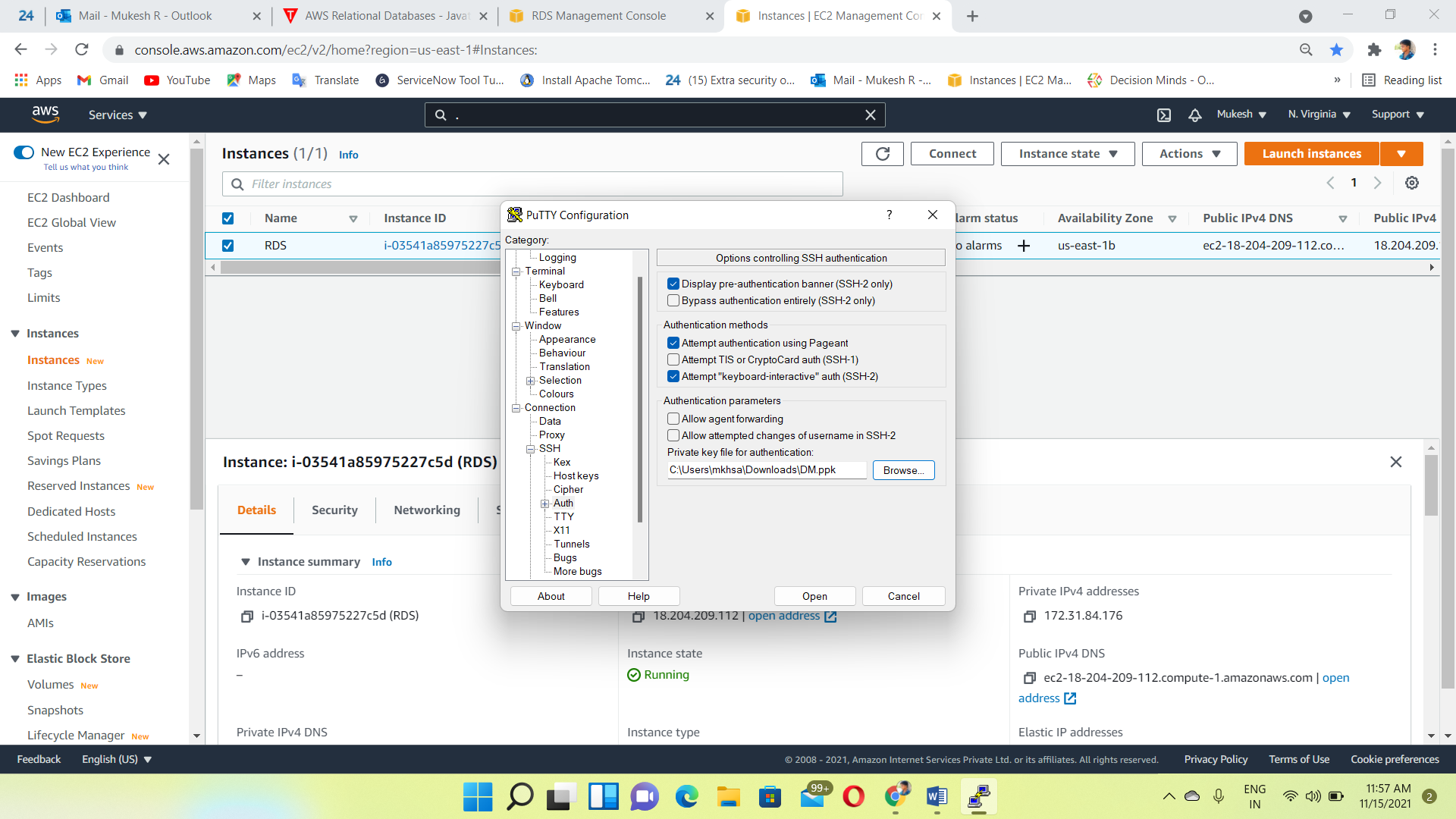
Instance created



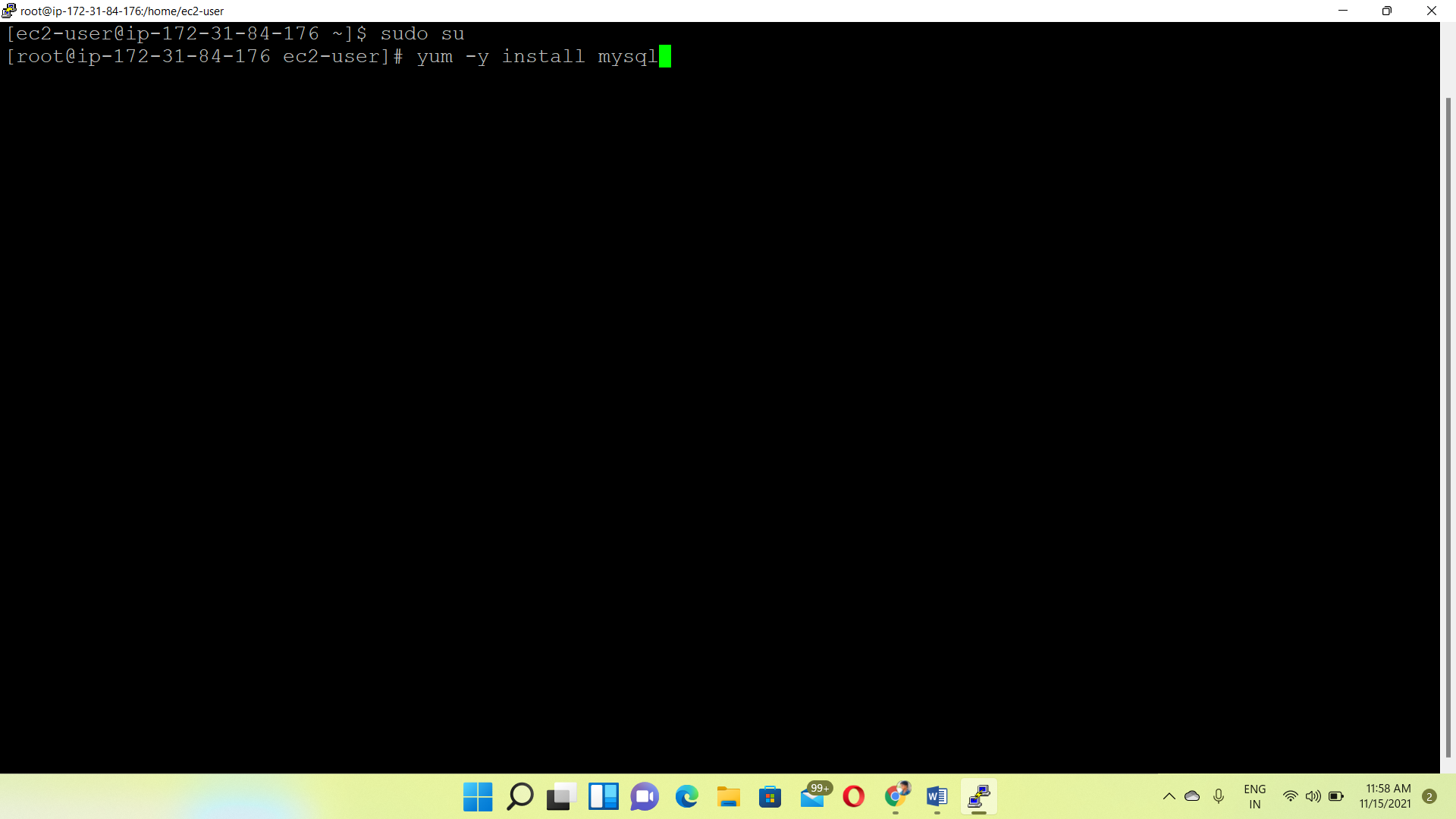
DB instance Also created



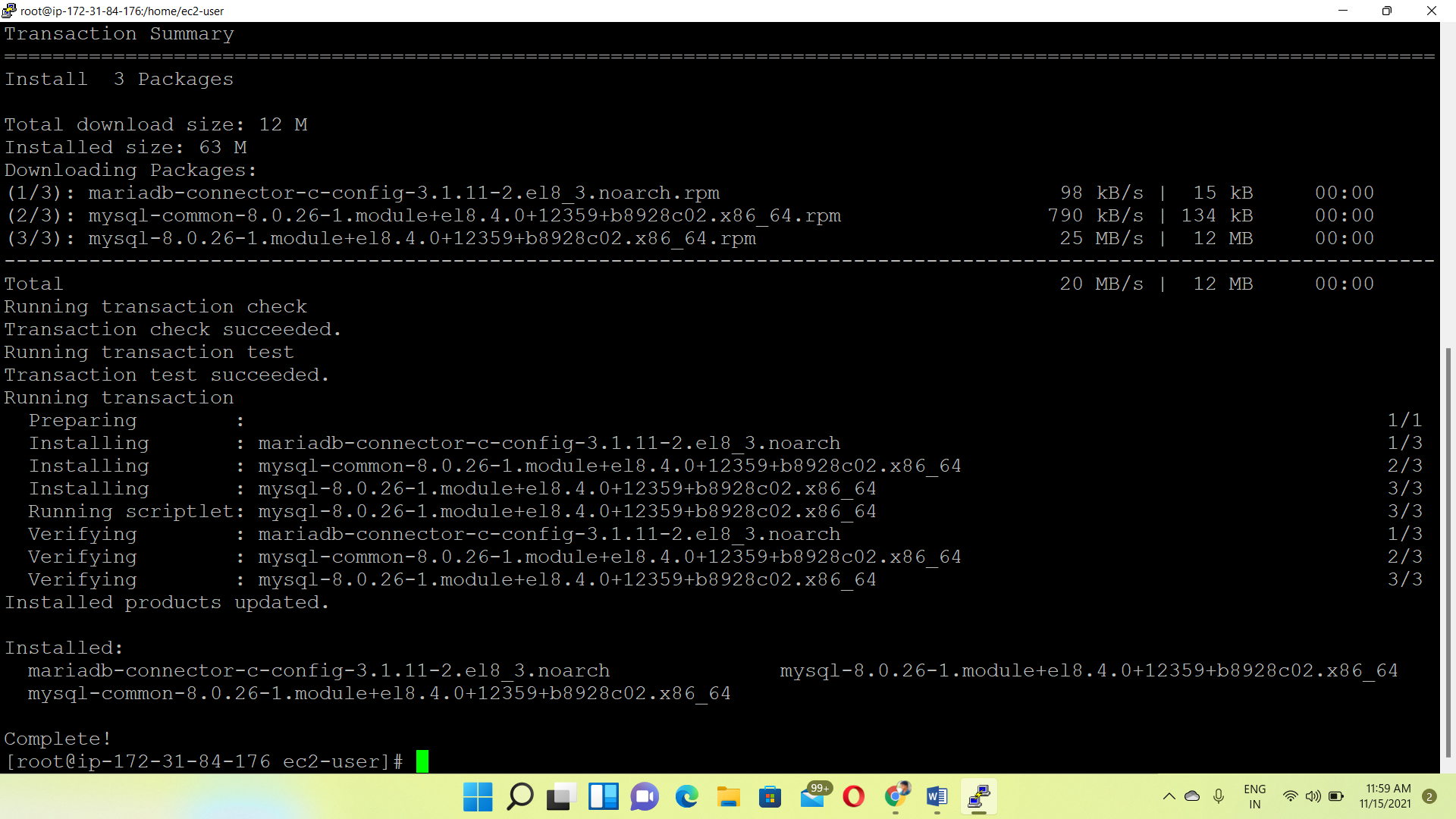
Login to the RHEL8 instance using putty



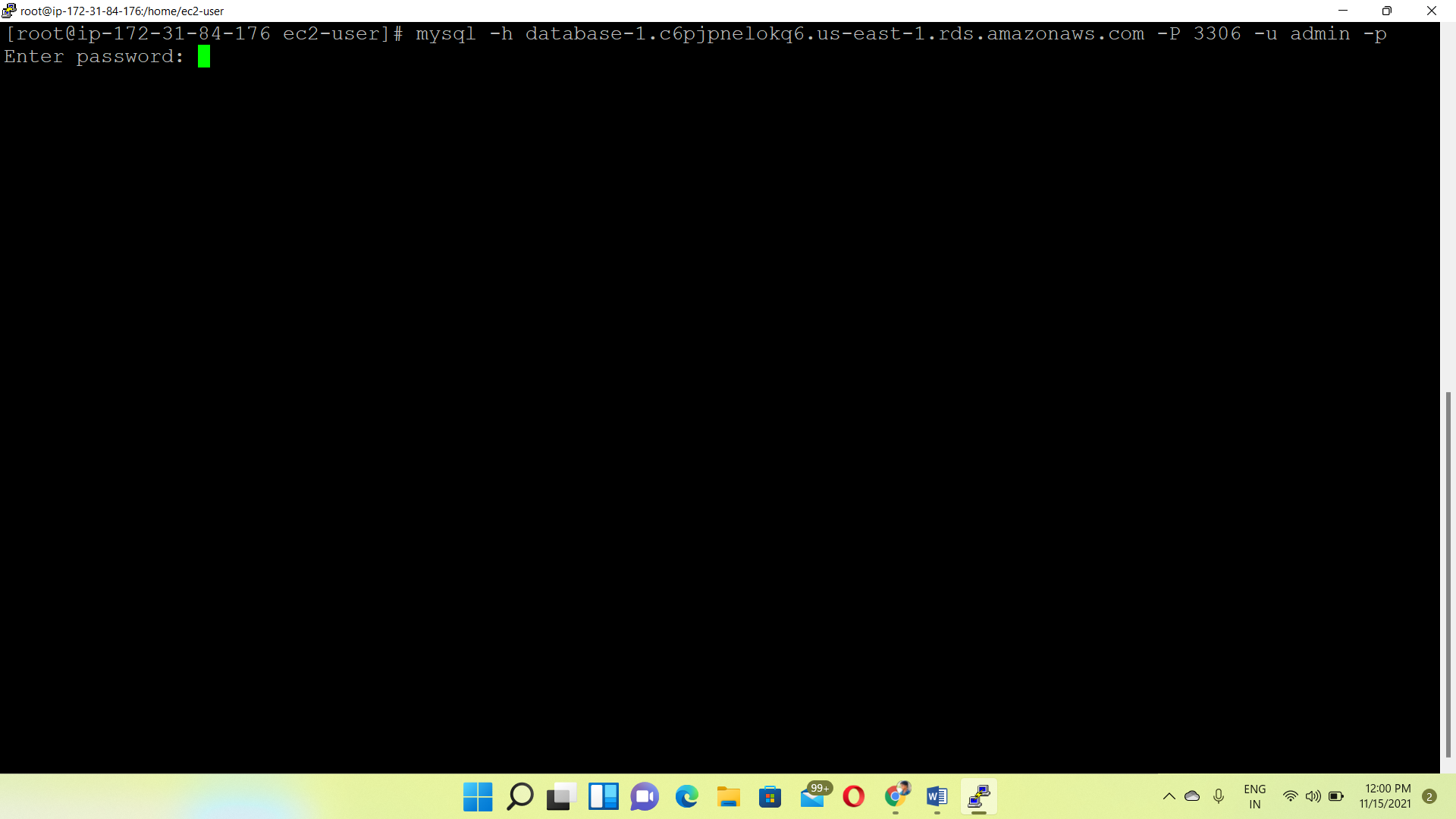
Install MySQL engine🡪yum –y install MySQL



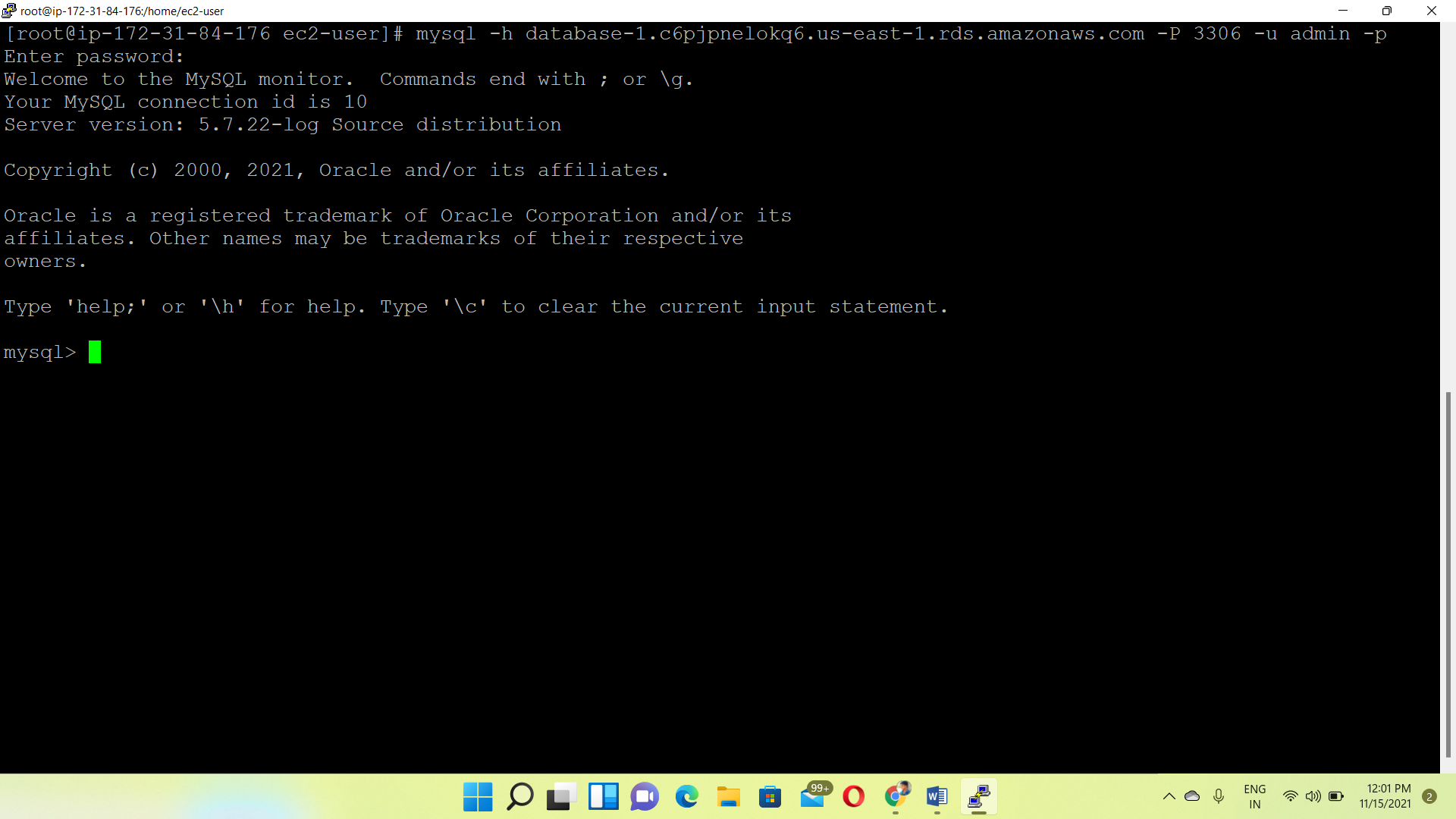
Installation completed



Enter the MySQL endpoint, Port number, Username, and Password



Successfully connected to the MySQL engine from RHEL8 instance



**Output:**

Check the Database working or not🡪show databases

