

# AWS Advanced

**RDS | EC2 database program**

(Create a MySQL instance using RDS and access it using a custom program from an EC2 instance using an appropriate role)



Learning Outcomes

* Get an EC2 instance using the 7 step workflow & SSH in it
* Create a MySQL database using RDS
* Write the custom application and deploy on EC2
* Able to apply roles to EC2 instances

# Final Goal

AWS (N Virginia)

IAM

Cache

Laptop

RDS

(MySQL)

Browser

SSH

EC2

DB

App



# What is needed?

1. AWS Account Credentials
2. EC2 Instances (Linux)
3. Shell script environment (any text editor of your choice)
4. Full access to - EC2, RDS, IAM, Elasticache

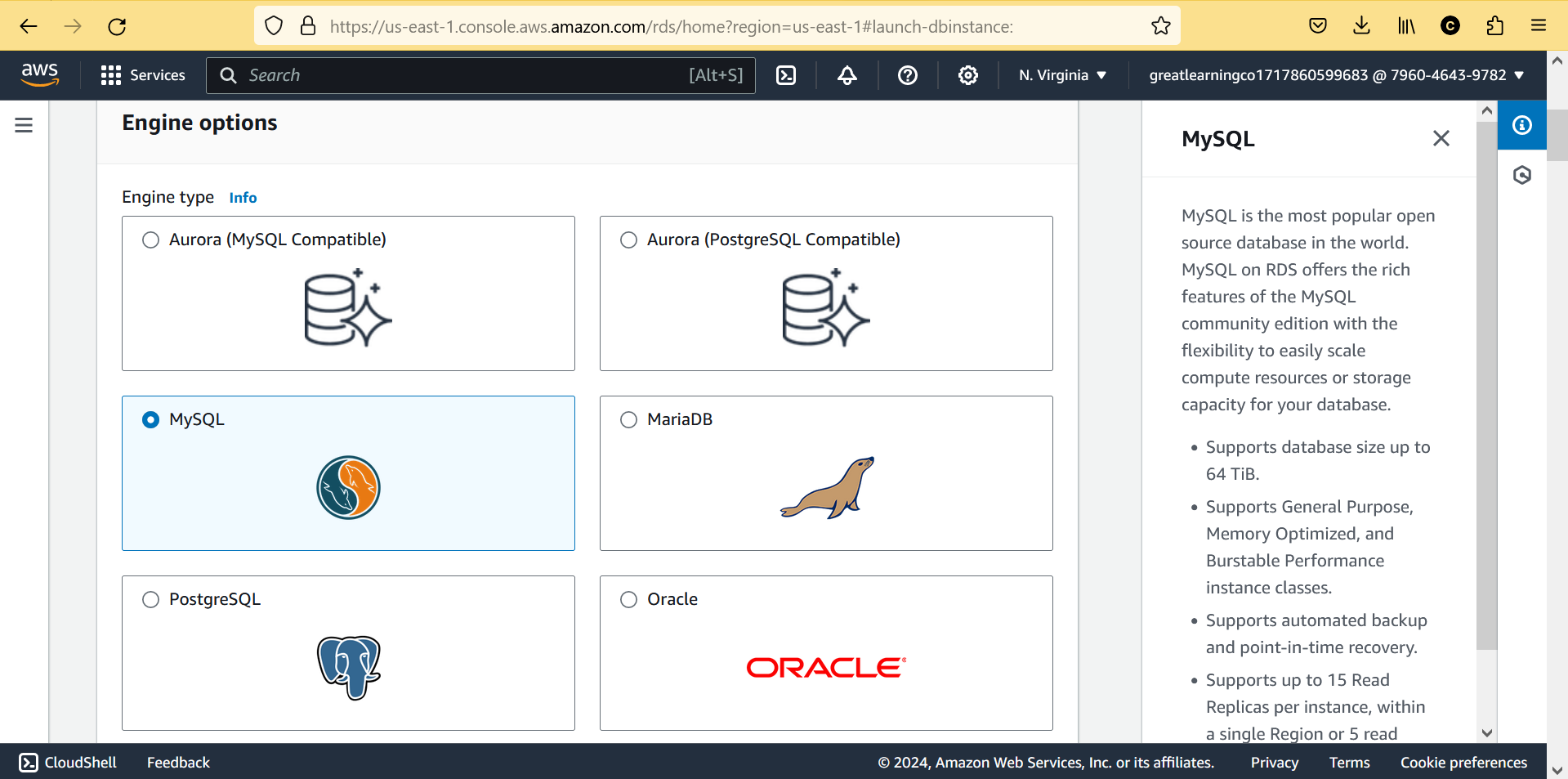


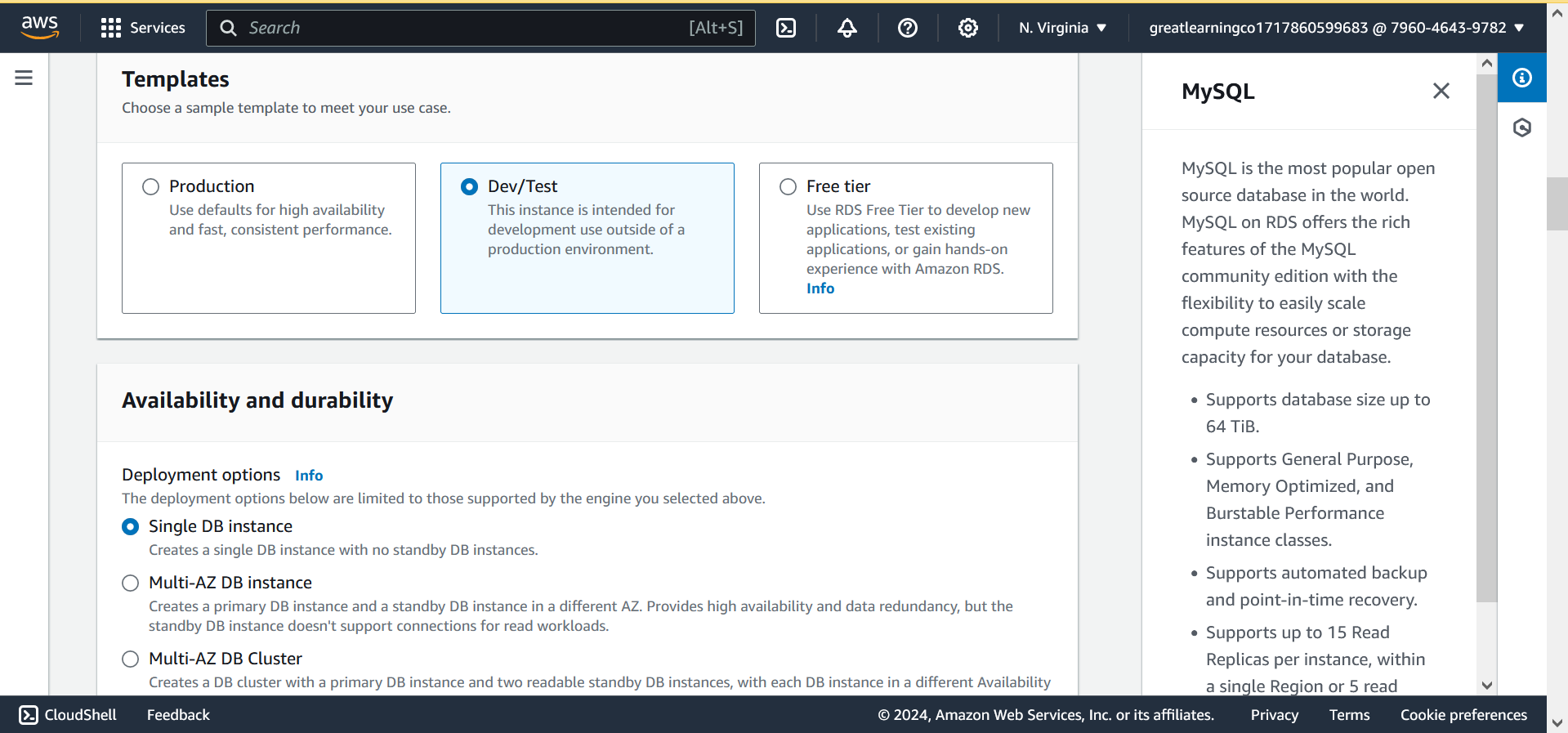
# How to do it? - 1

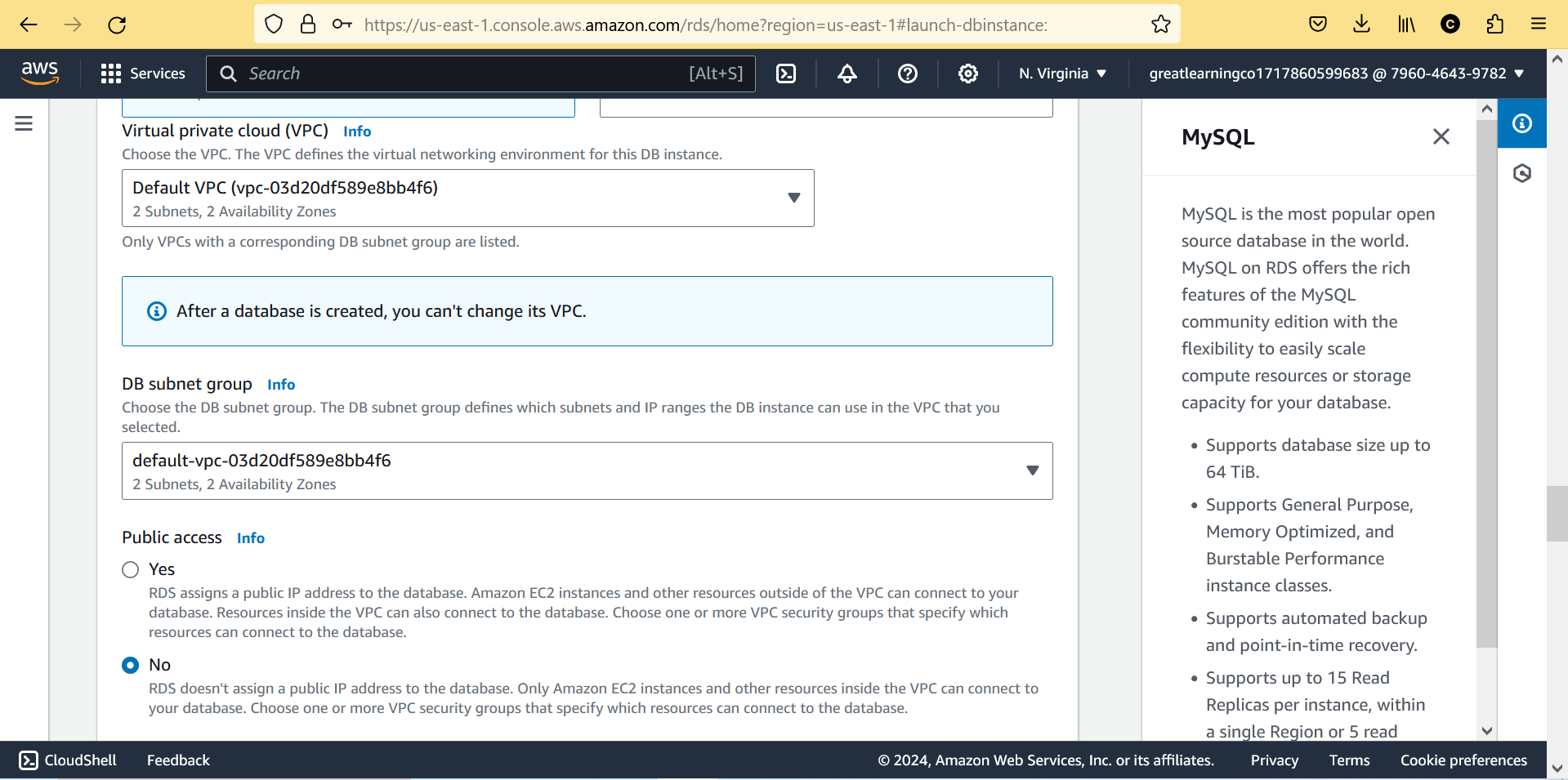
1. Ensure your region is set to "N Virginia"
2. Create a private MySQL RDS
   1. Create a "Dev/Test" MySQL instance
   2. Select "Non publicly" available DB
   3. DB instance = "flipbasket", UID/PWD = root/password
   4. AZ = 2a, create a new SG
   5. IMP When the RDS instance is created check the SG to ensure that the "inbound" rule does not have any specific IP address, instead use the VPC CIDR block of the EC2 instance. Otherwise we will not be able to connect from the EC2 instance
3. Create 1 EC2 instance using the 7 step workflow
   1. Use the usual Ubuntu 20.04 LTS AMI in AZ1
   2. Download a new PEM file and SSH to the instance
4. IAM
   1. Attach the pre-created role “labinstanceprofile” to the instance OR use an IAM role using

the permission “RDSFullAccess”

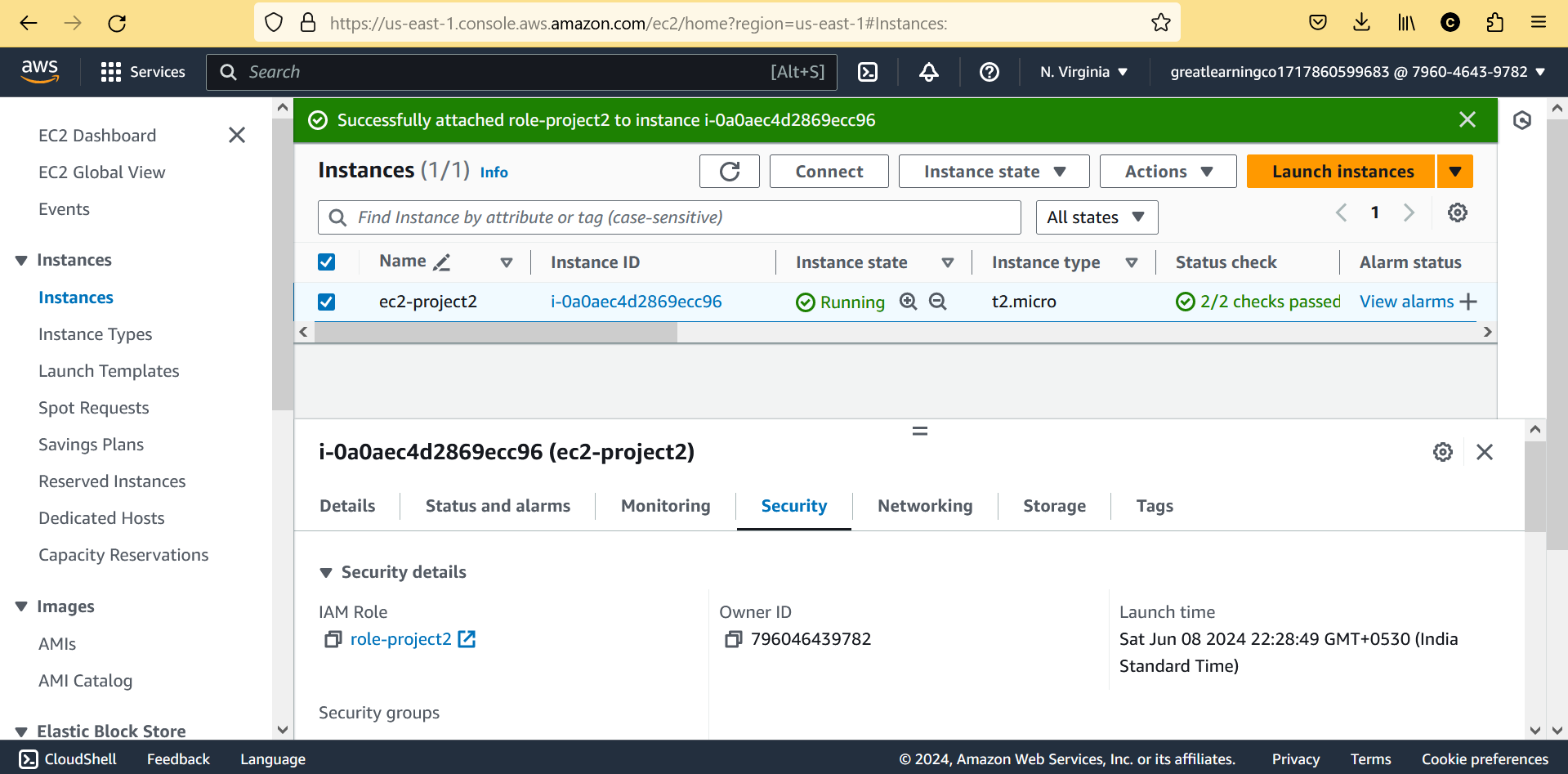
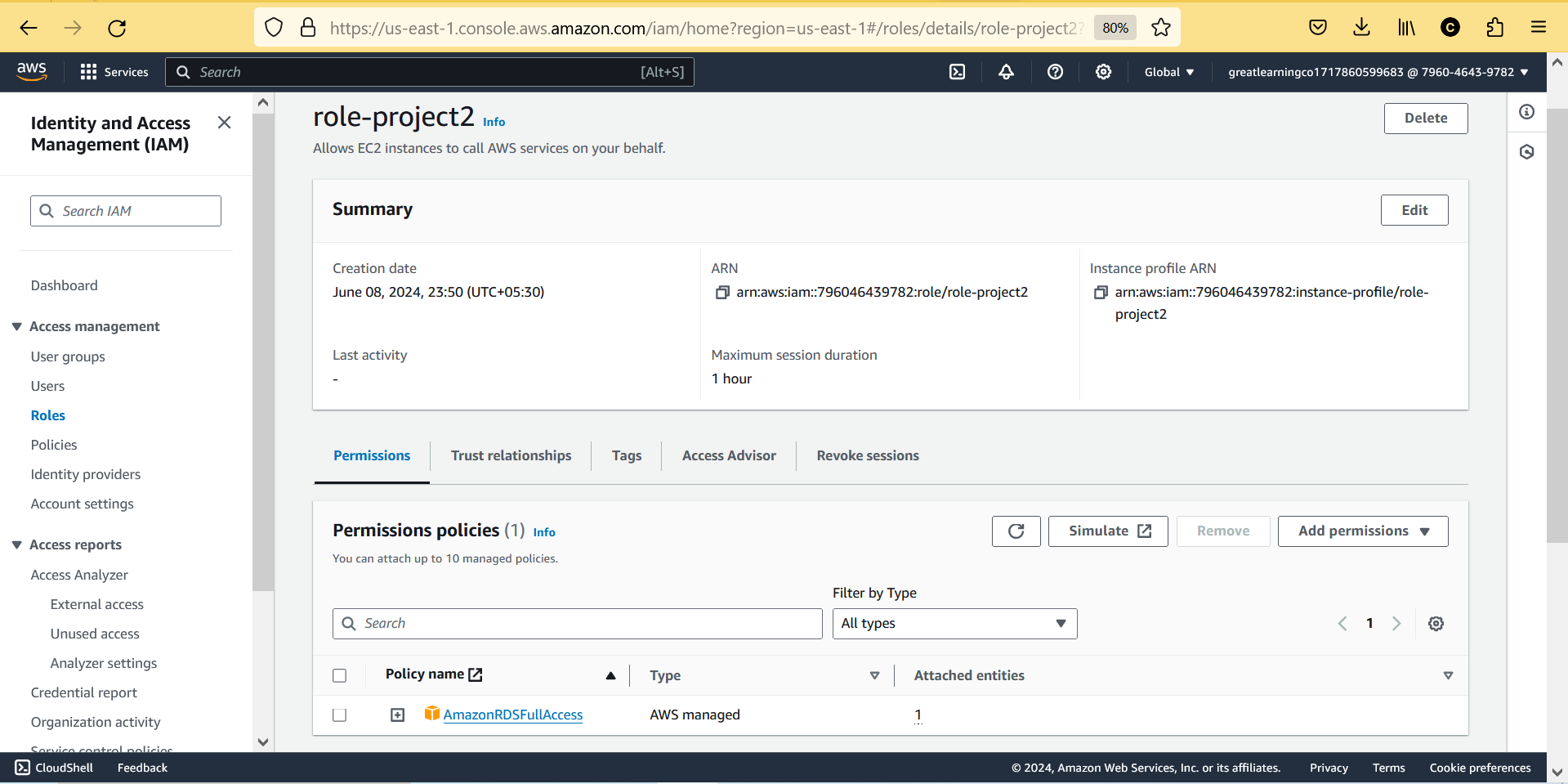
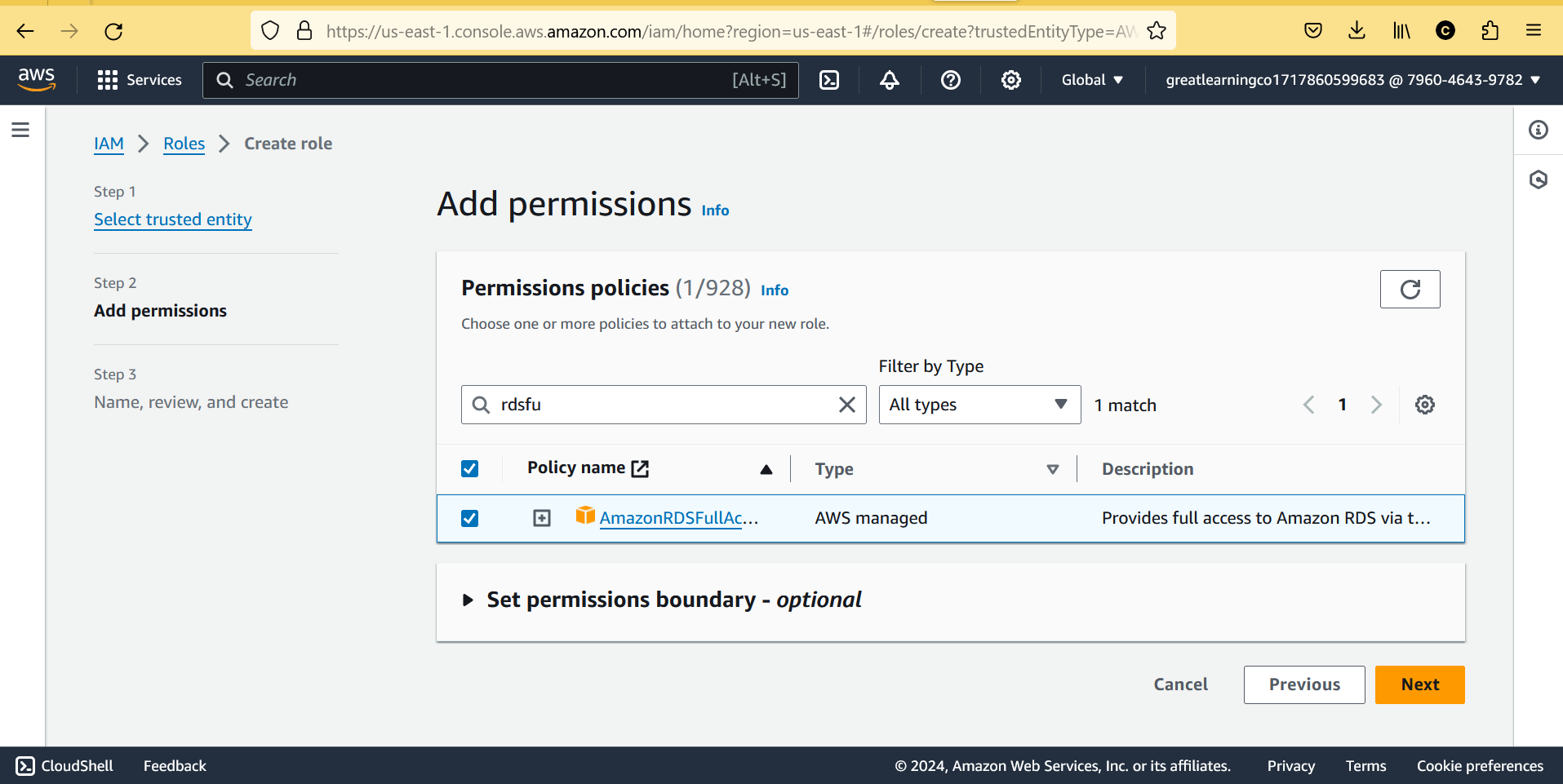
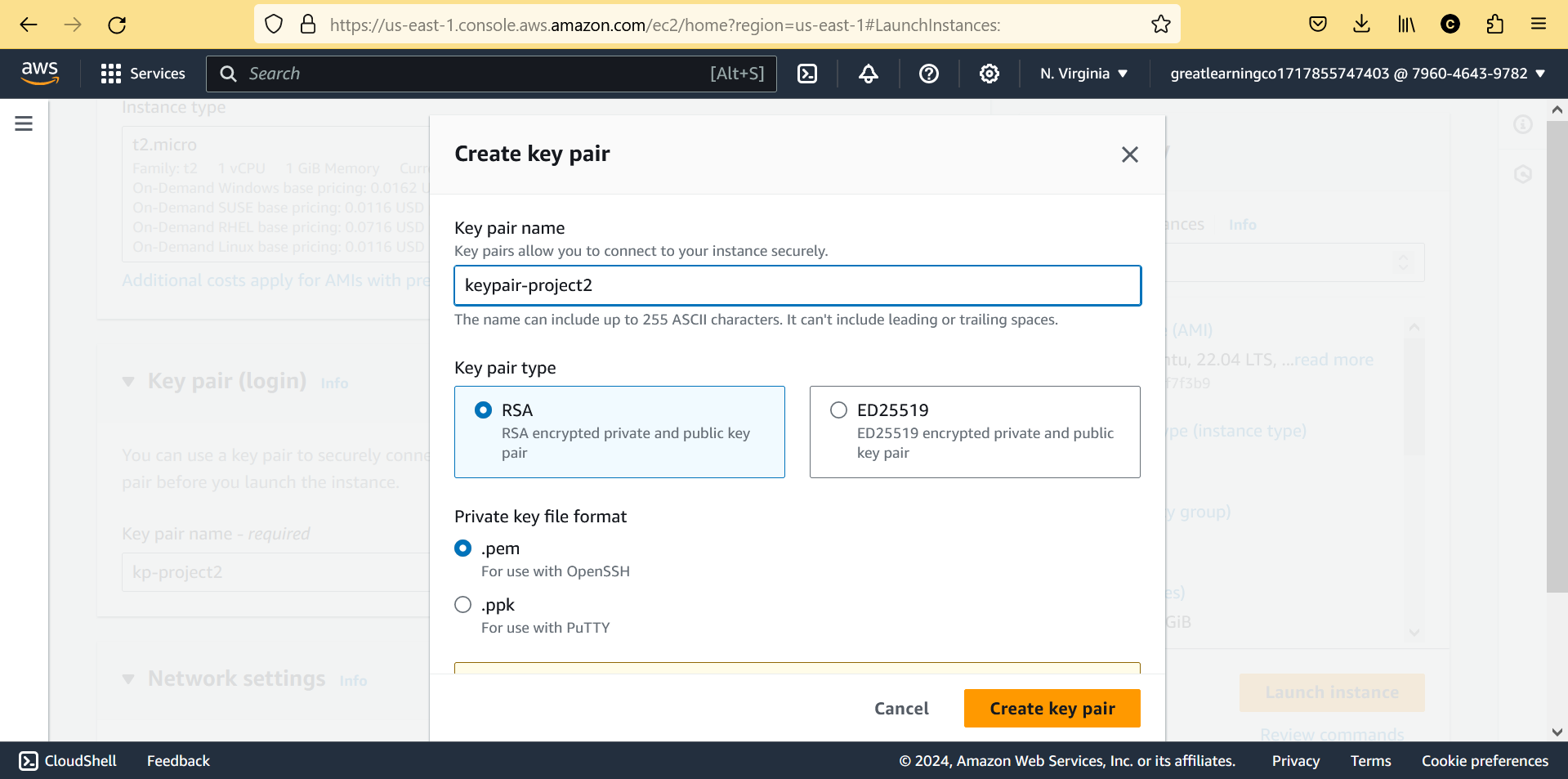
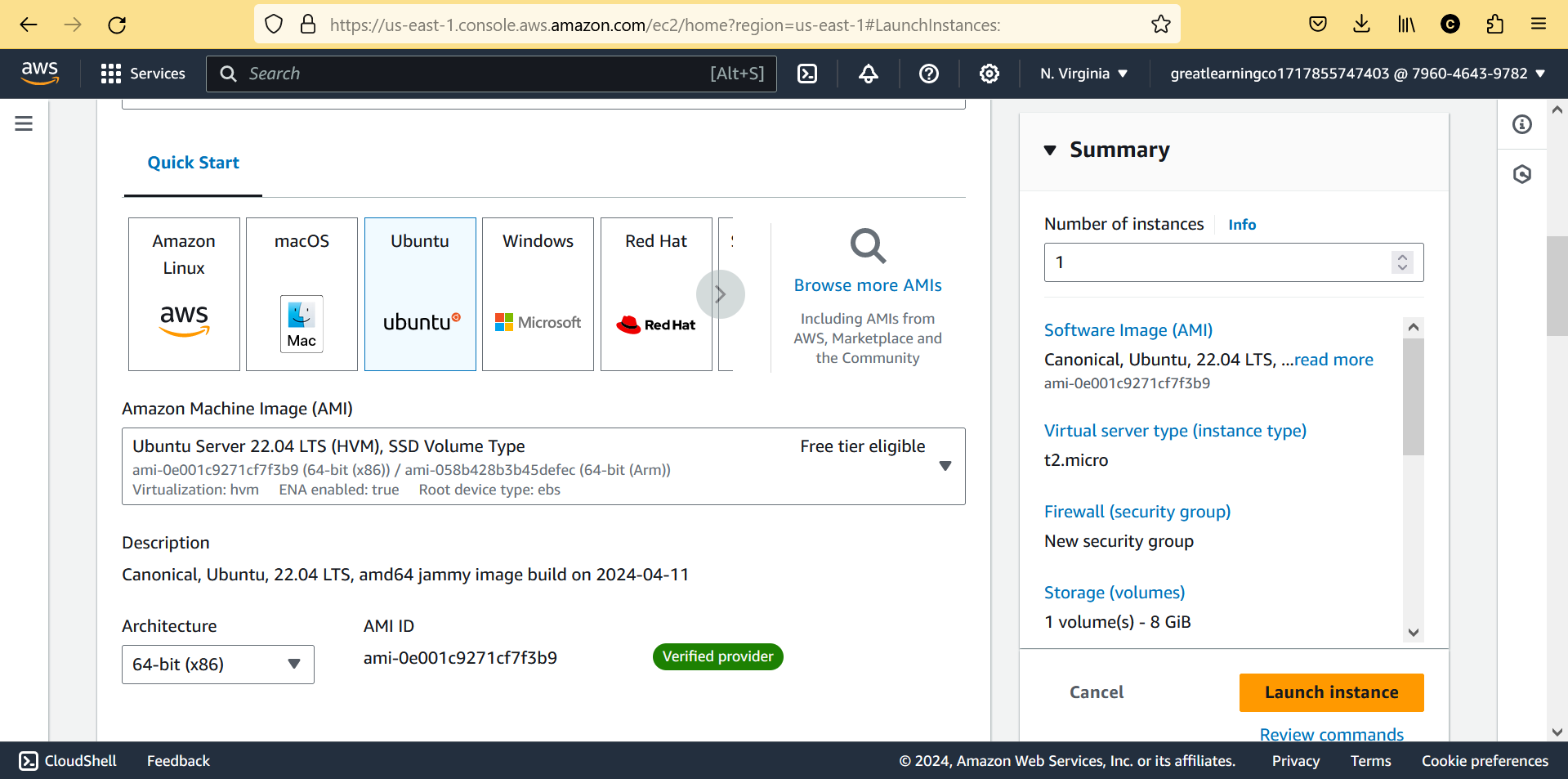
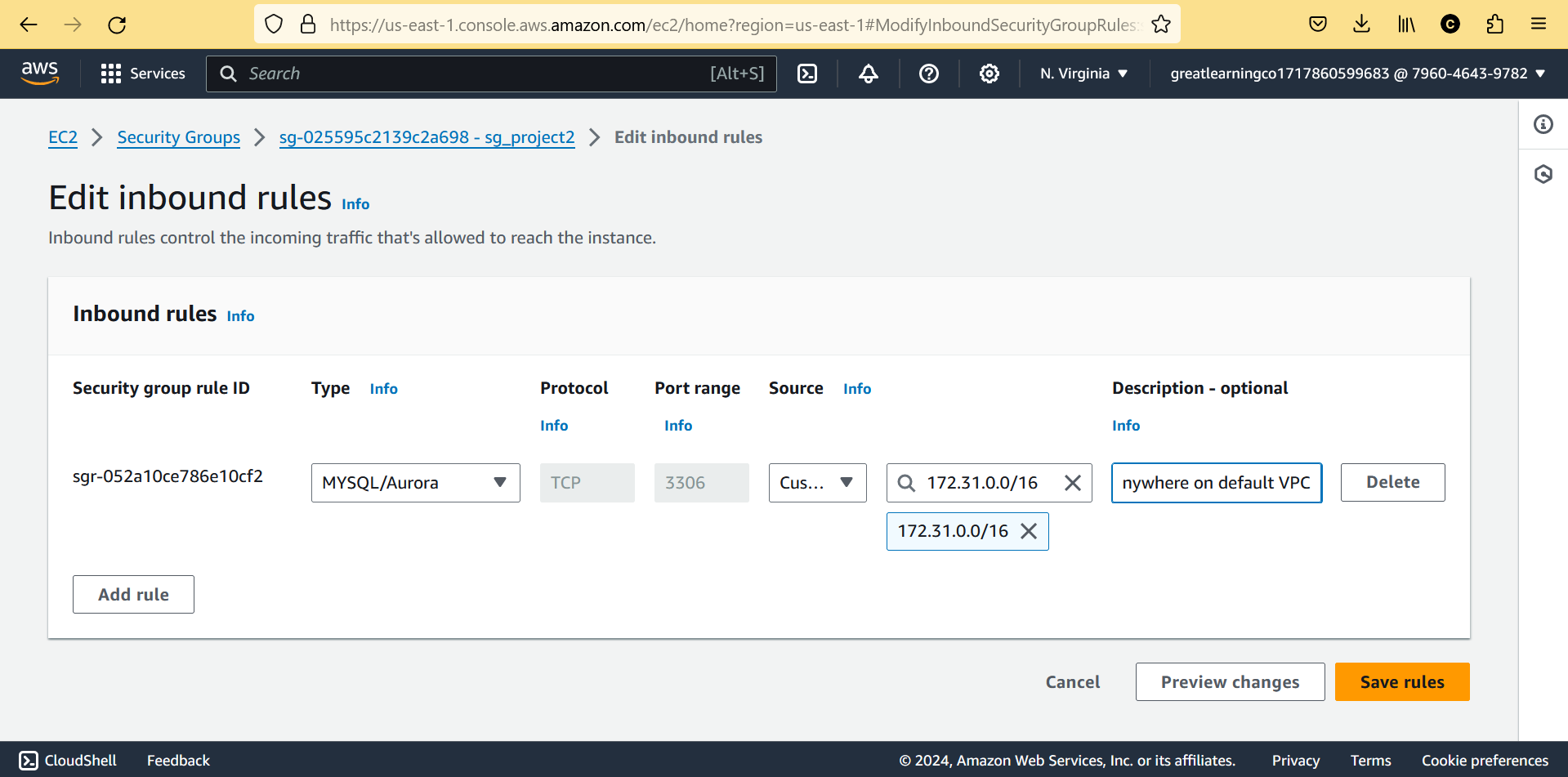
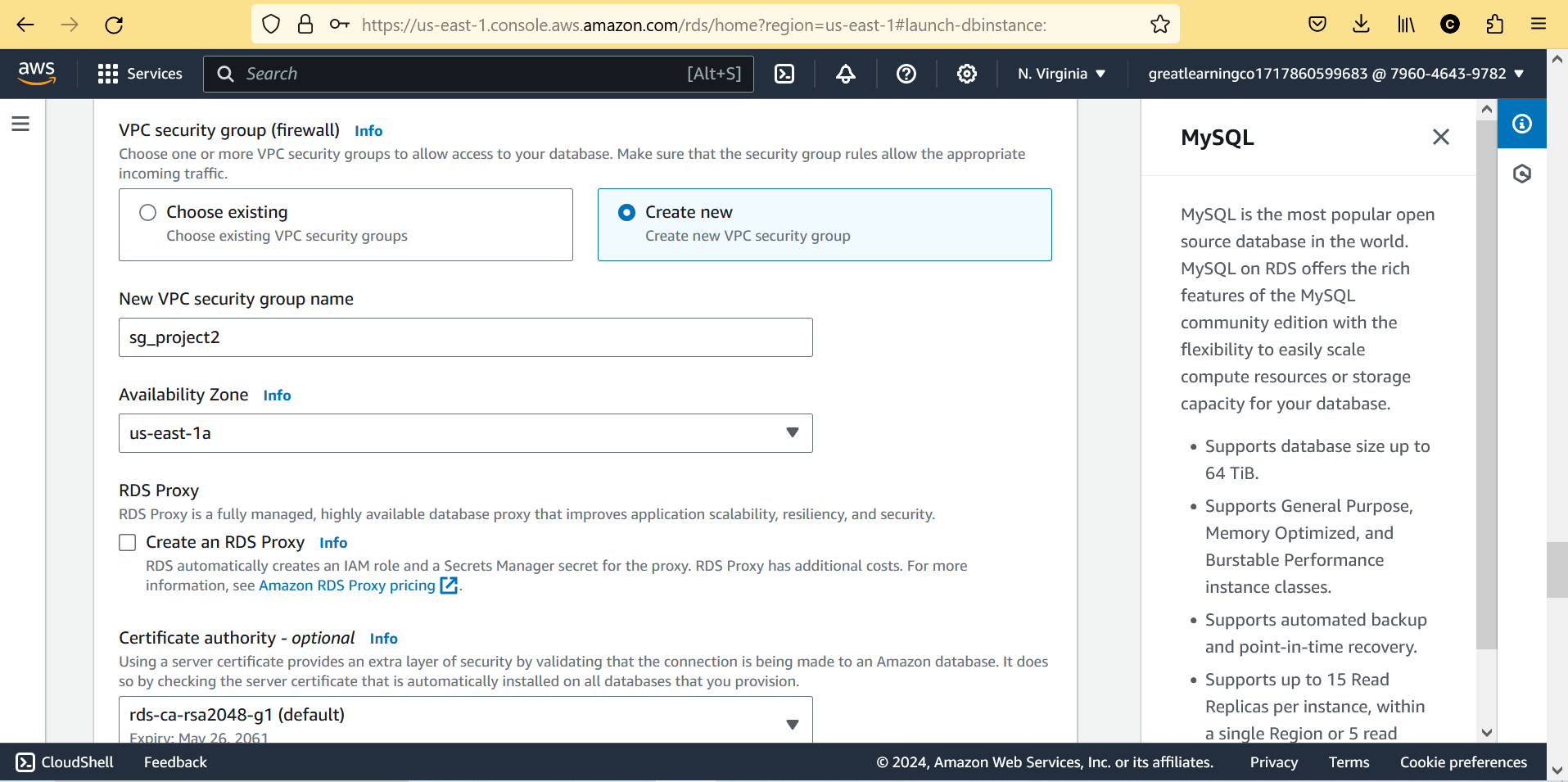
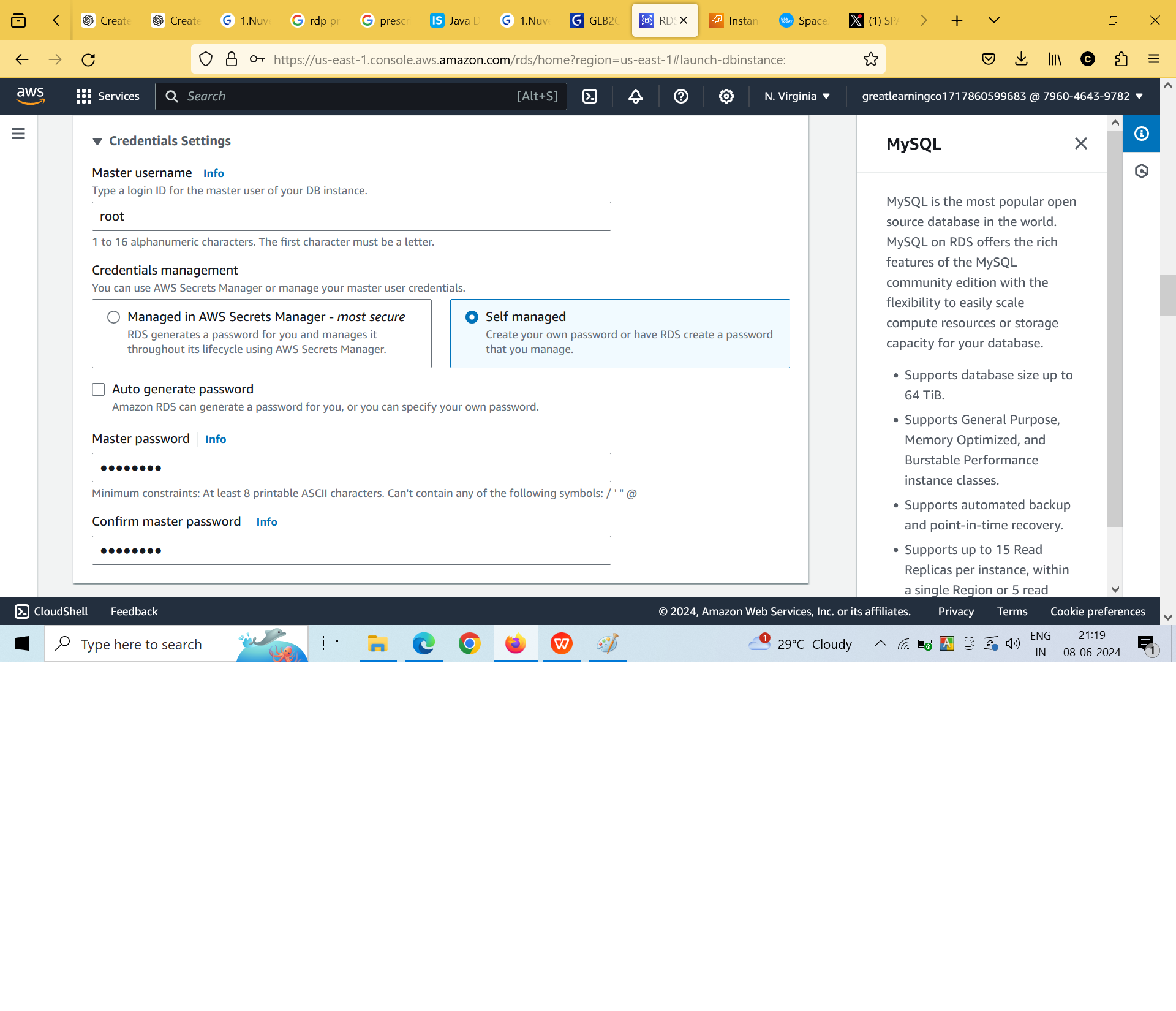
**Screenshots for Step-1:**

****

****

****

****

****



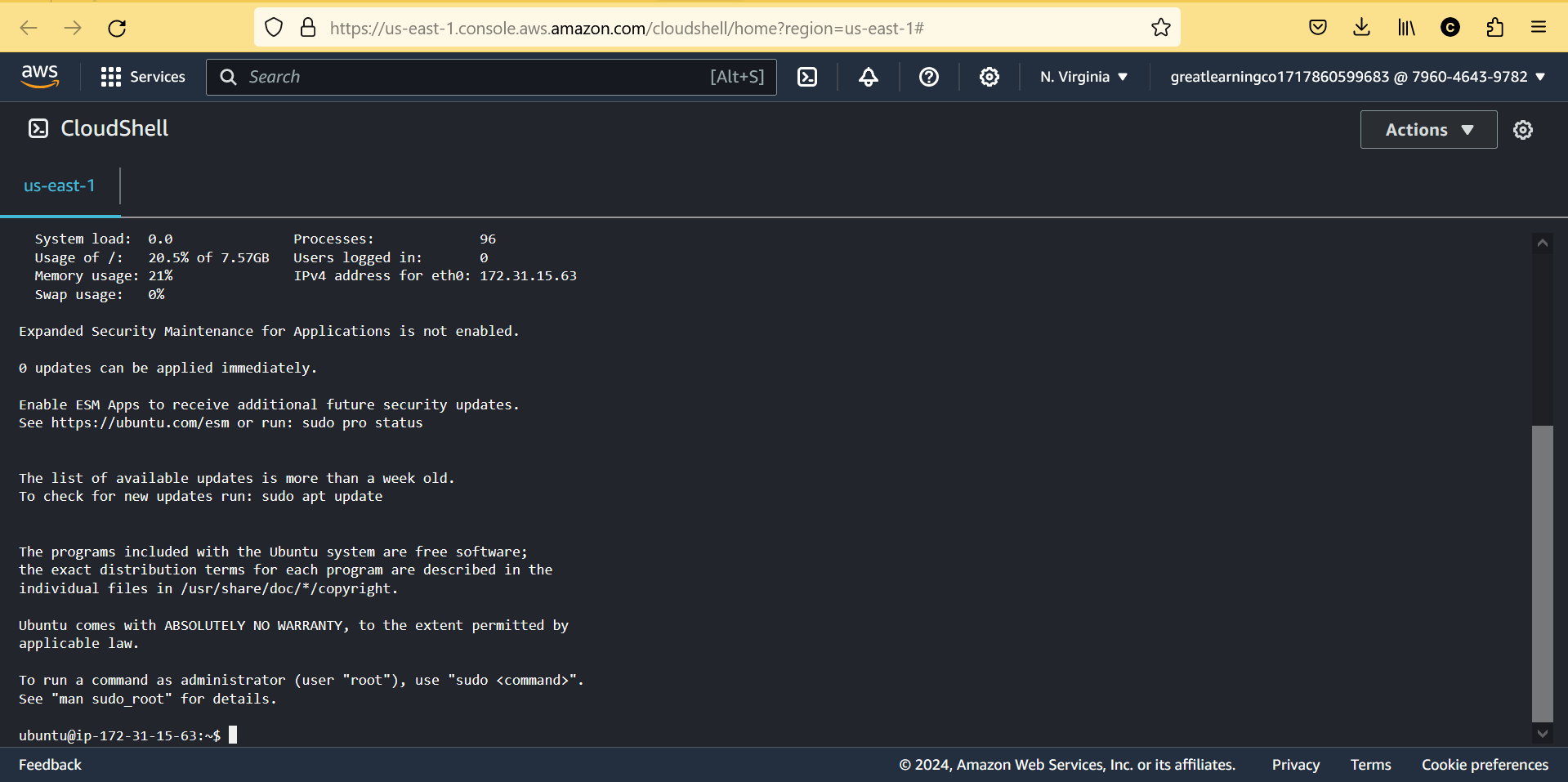
# How to do it? - 2

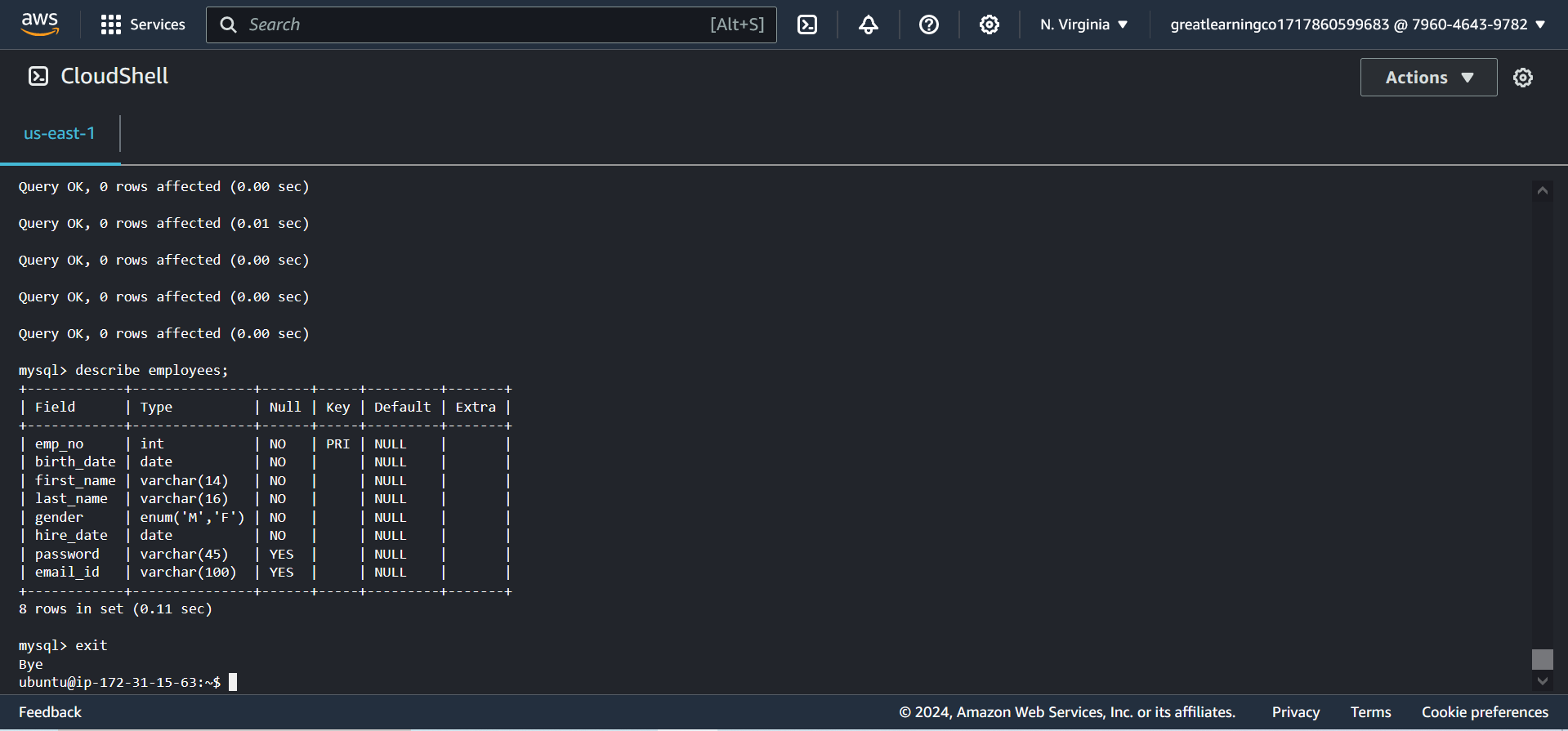
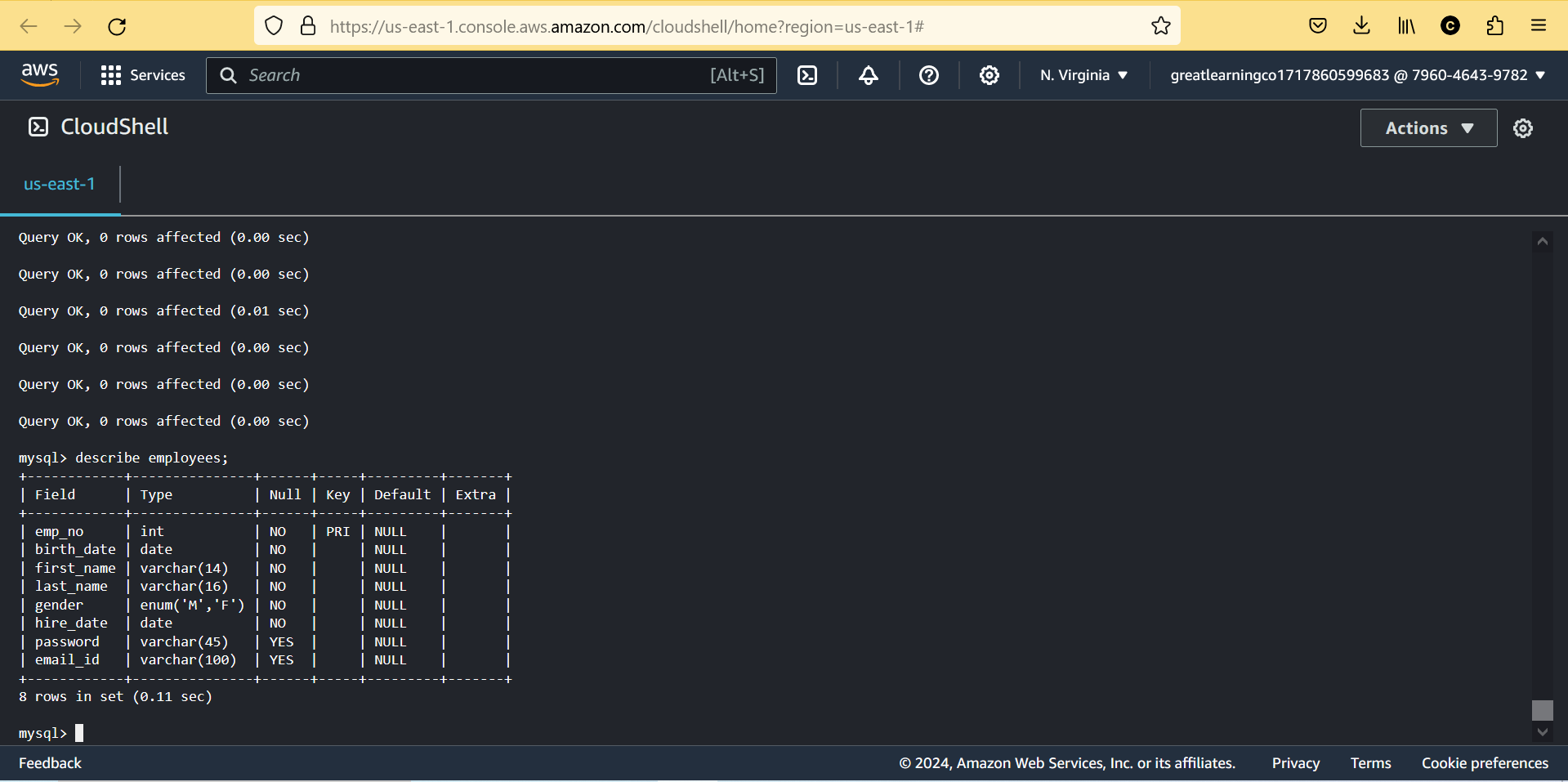
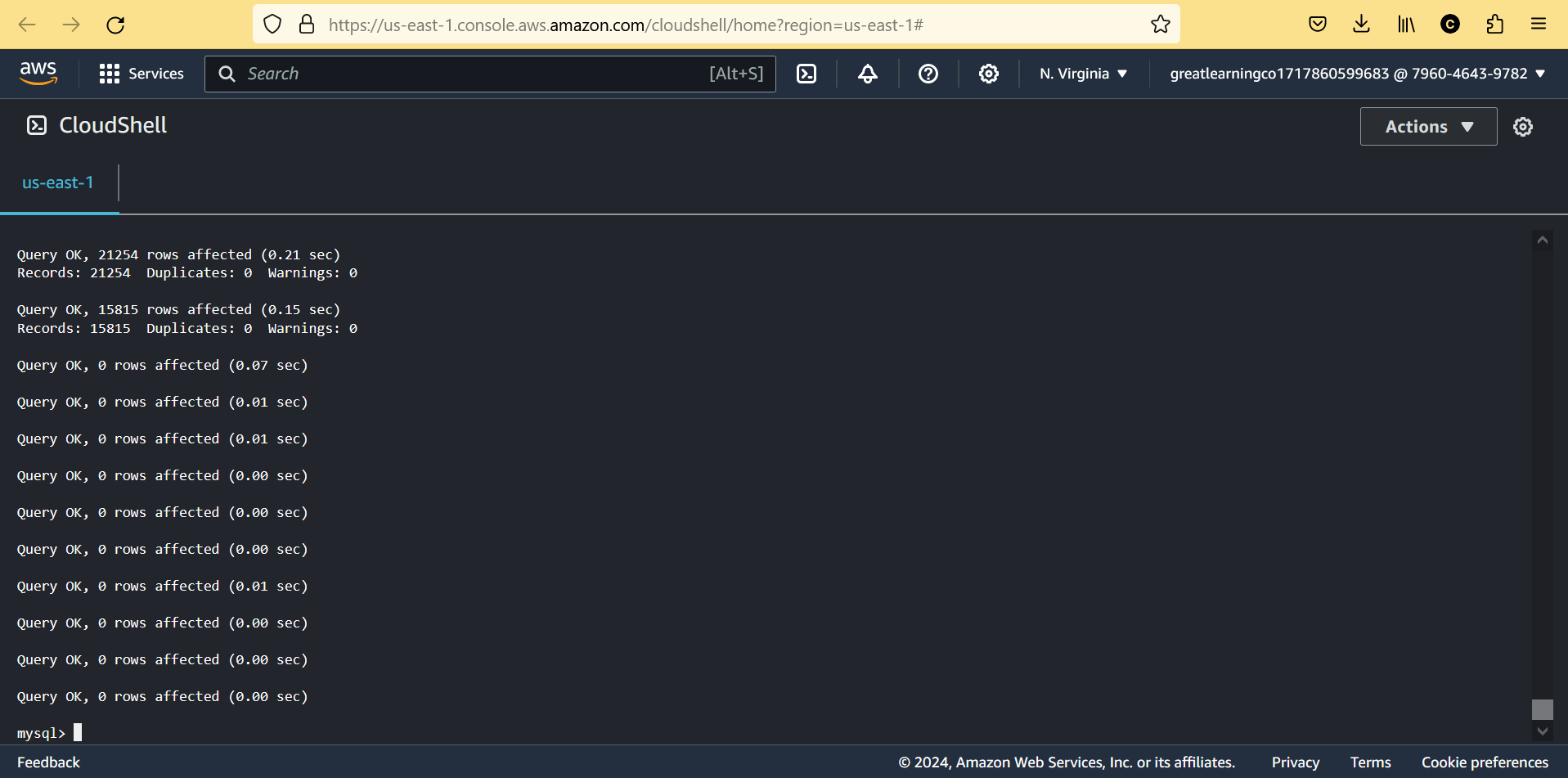
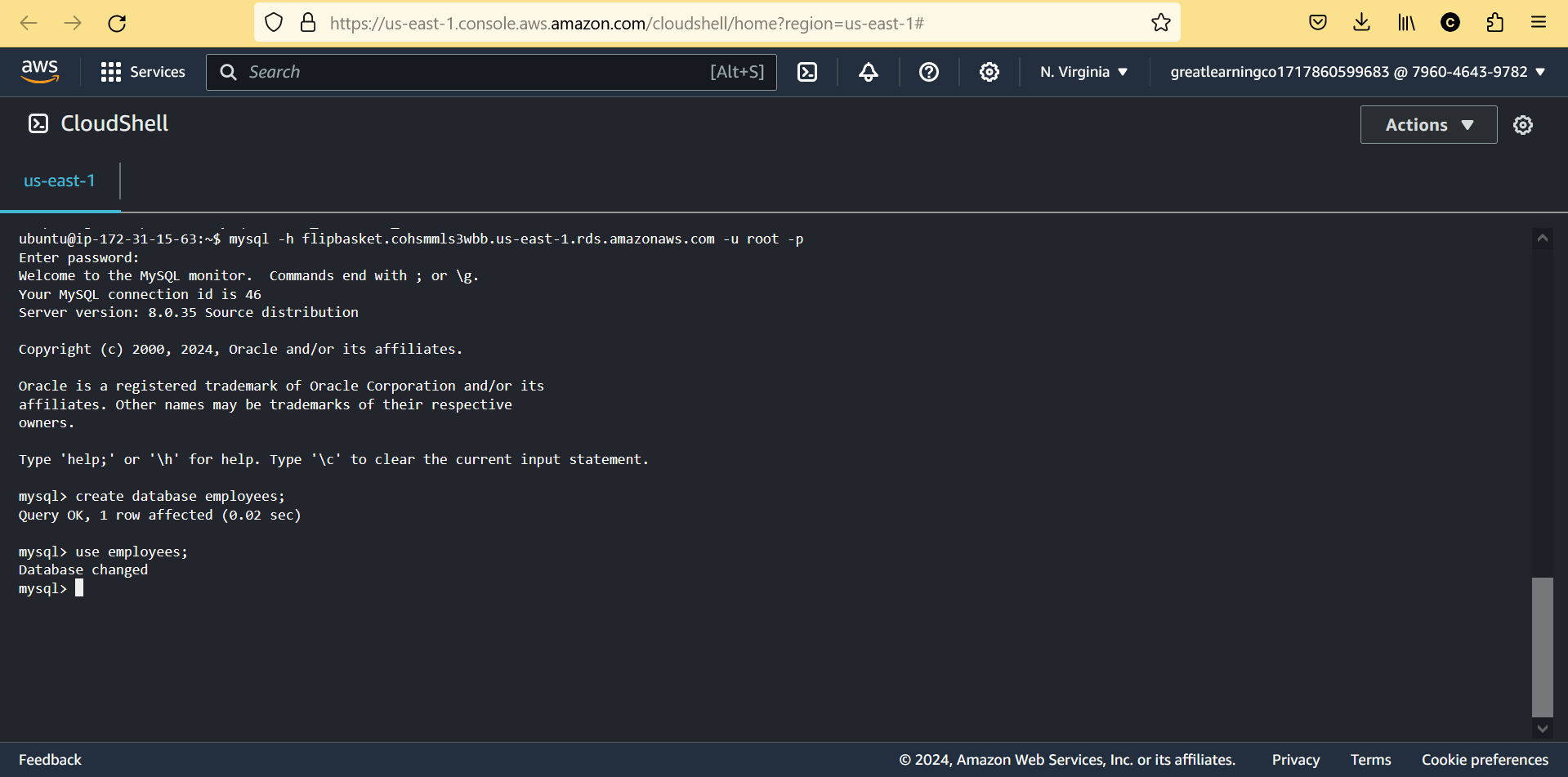
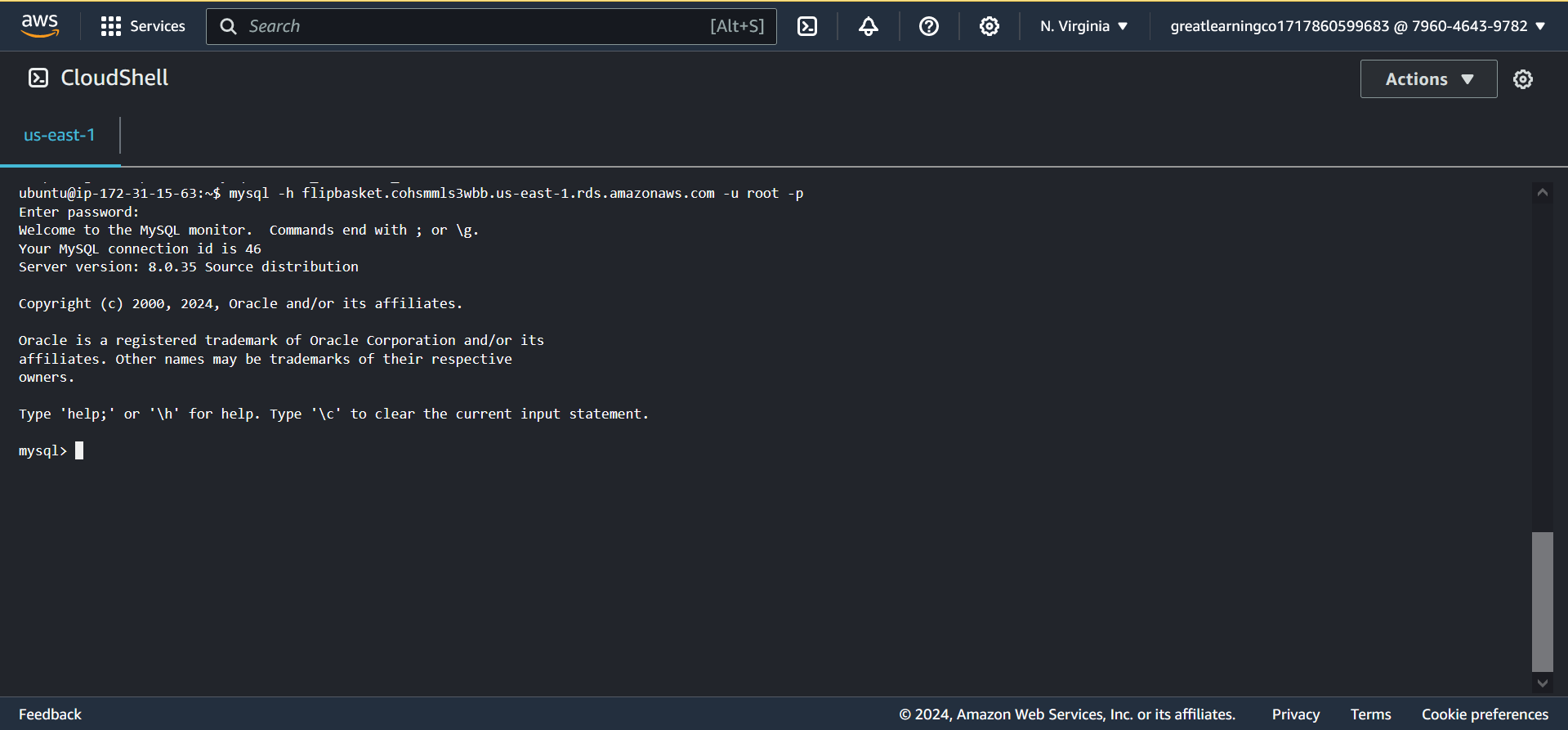
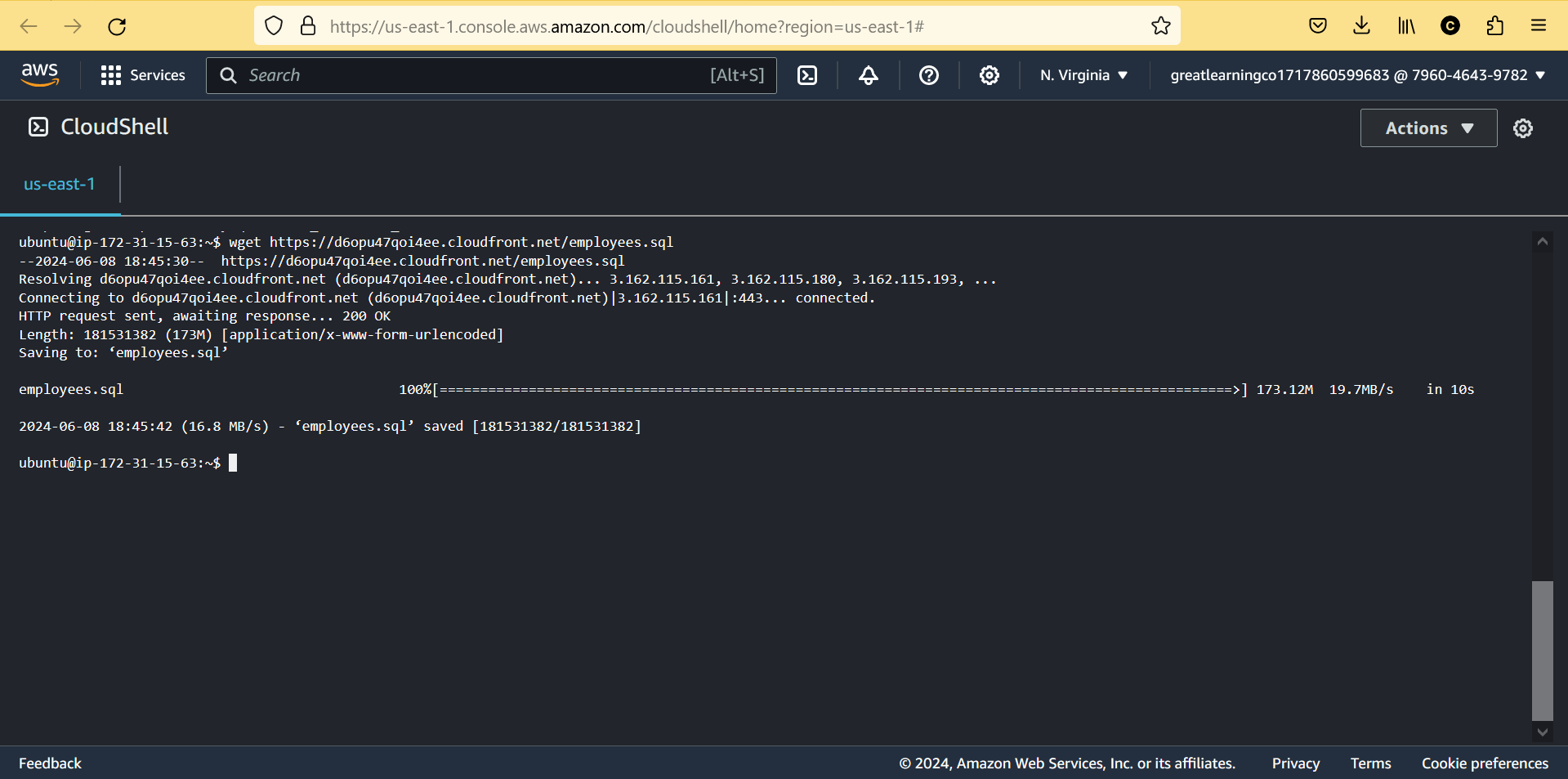
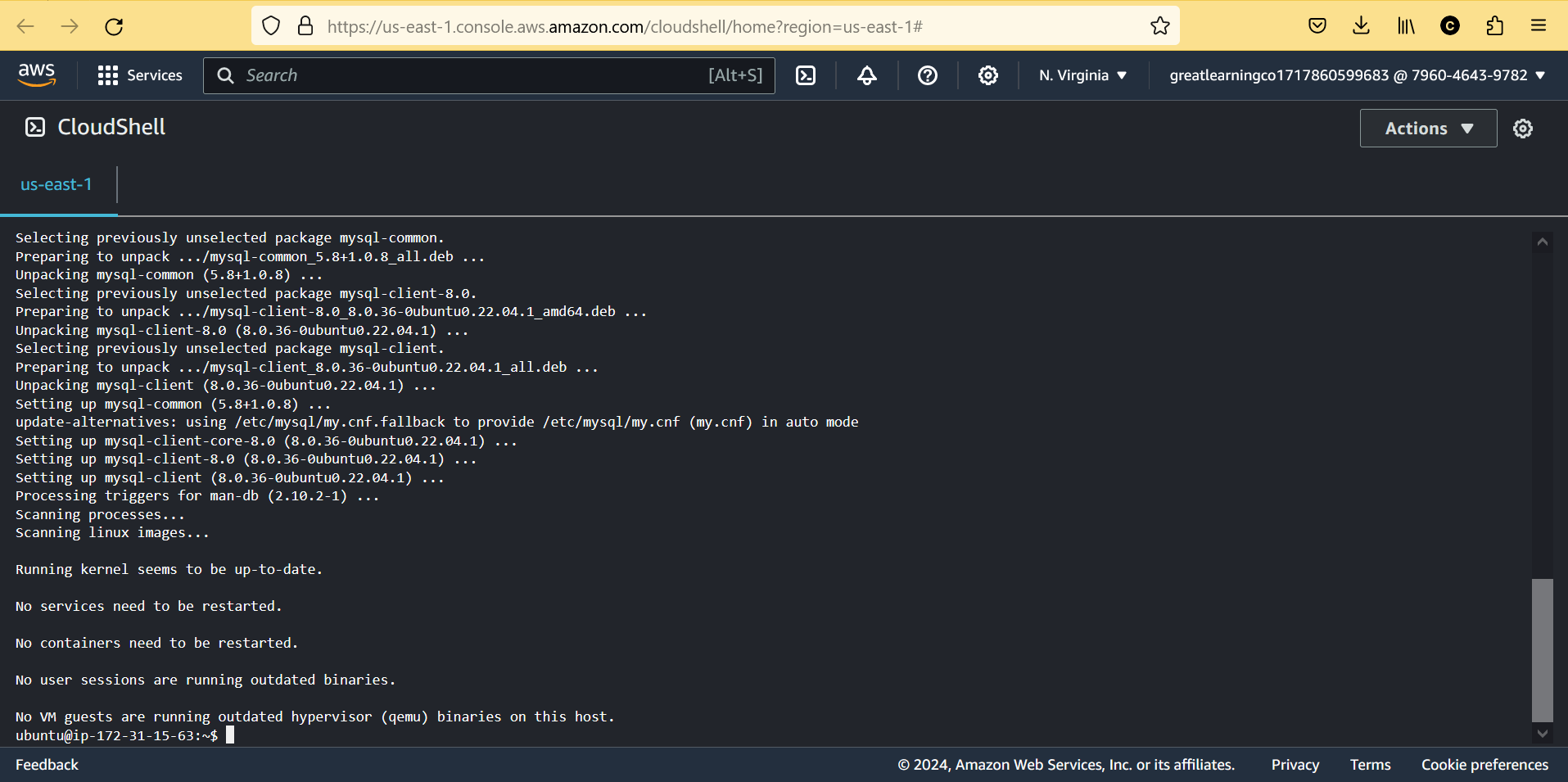
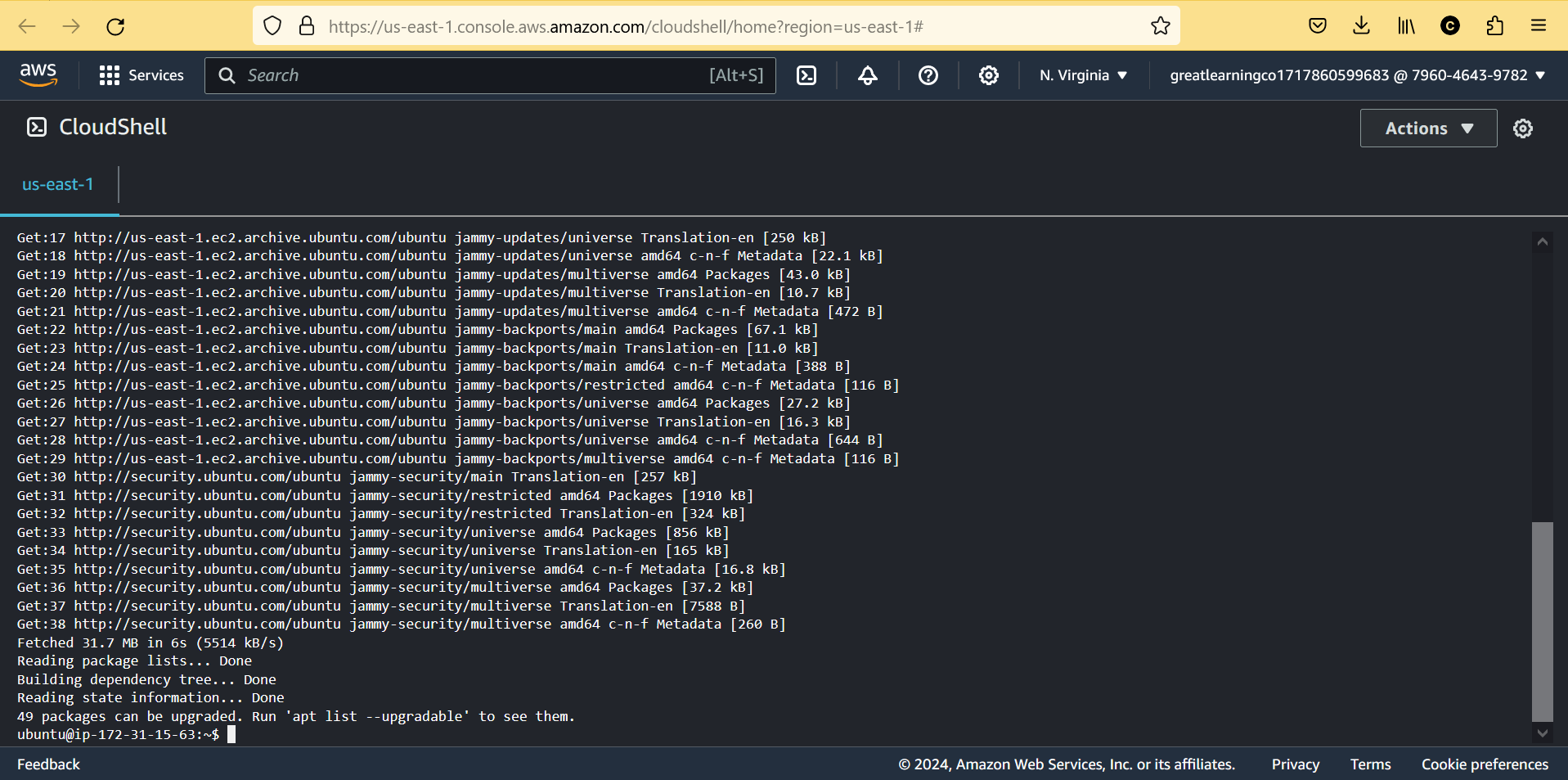
1. SSH to the EC2 instance and fire the following commands
   1. sudo apt update
   2. sudo apt install mysql-client
2. Connect to the RDS instance & create the data
   1. wget https://d6opu47qoi4ee.cloudfront.net/employees.sql
   2. mysql -h [RDS end point] -u admin -p

You should get the MySQL prompt after entering the password

* 1. create database employees;
  2. use employees;
  3. source employees.sql
  4. describe employees;
  5. Note the column names in a text editor from the output of the above command
  6. Close the mysql terminal

**Screenshots for Step-2:**

****





# How to do it? - 3

* Follow the steps for the Python program connecting to the RDS instance from EC2
  + sudo apt install python2-minimal
  + sudo apt install python3-pip
  + sudo pip3 install boto3
  + sudo pip3 install mysql-connector-python
  + sudo pip3 install pymysql

In case the pip install fails execute the following 3 lines -

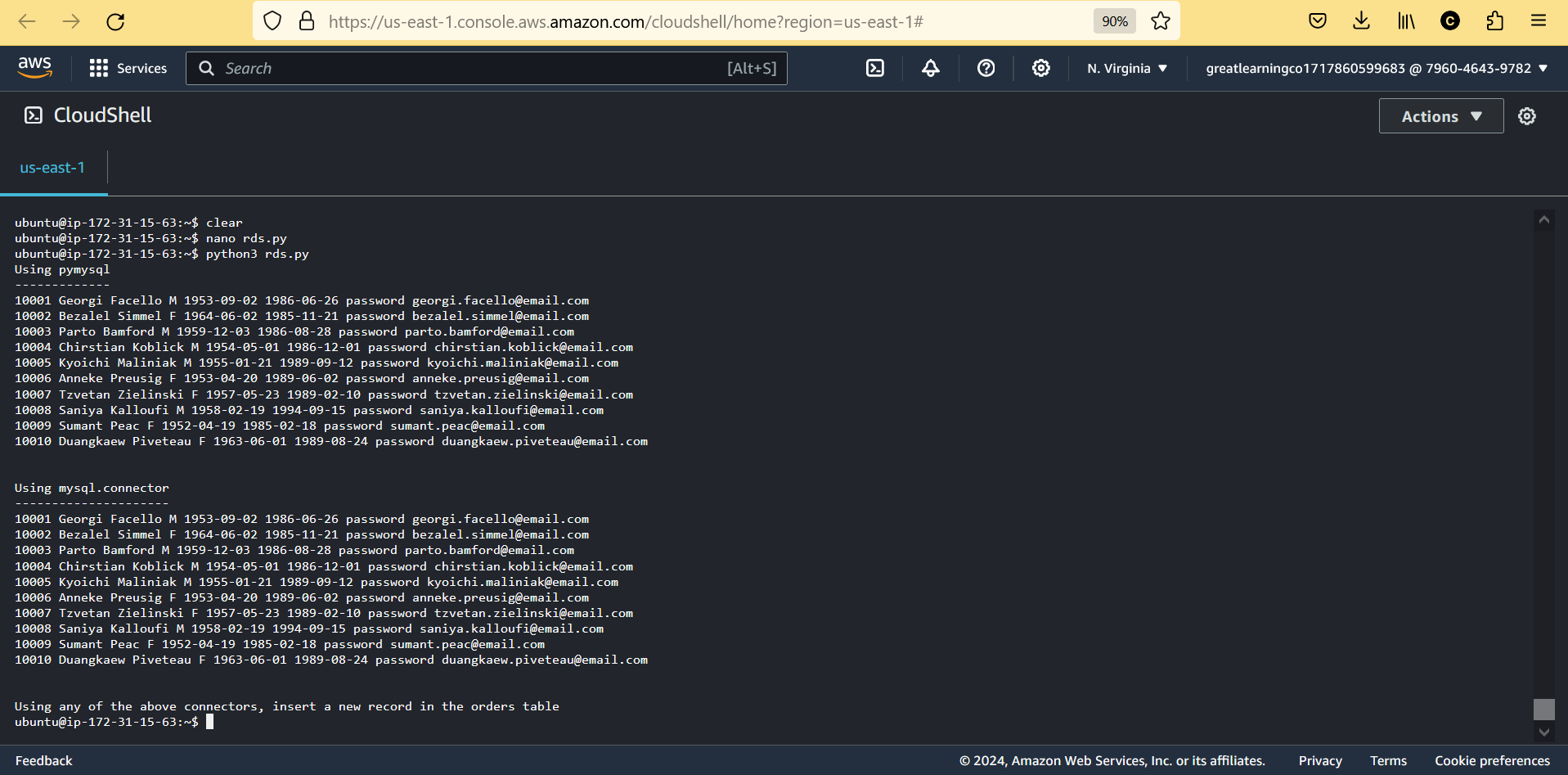
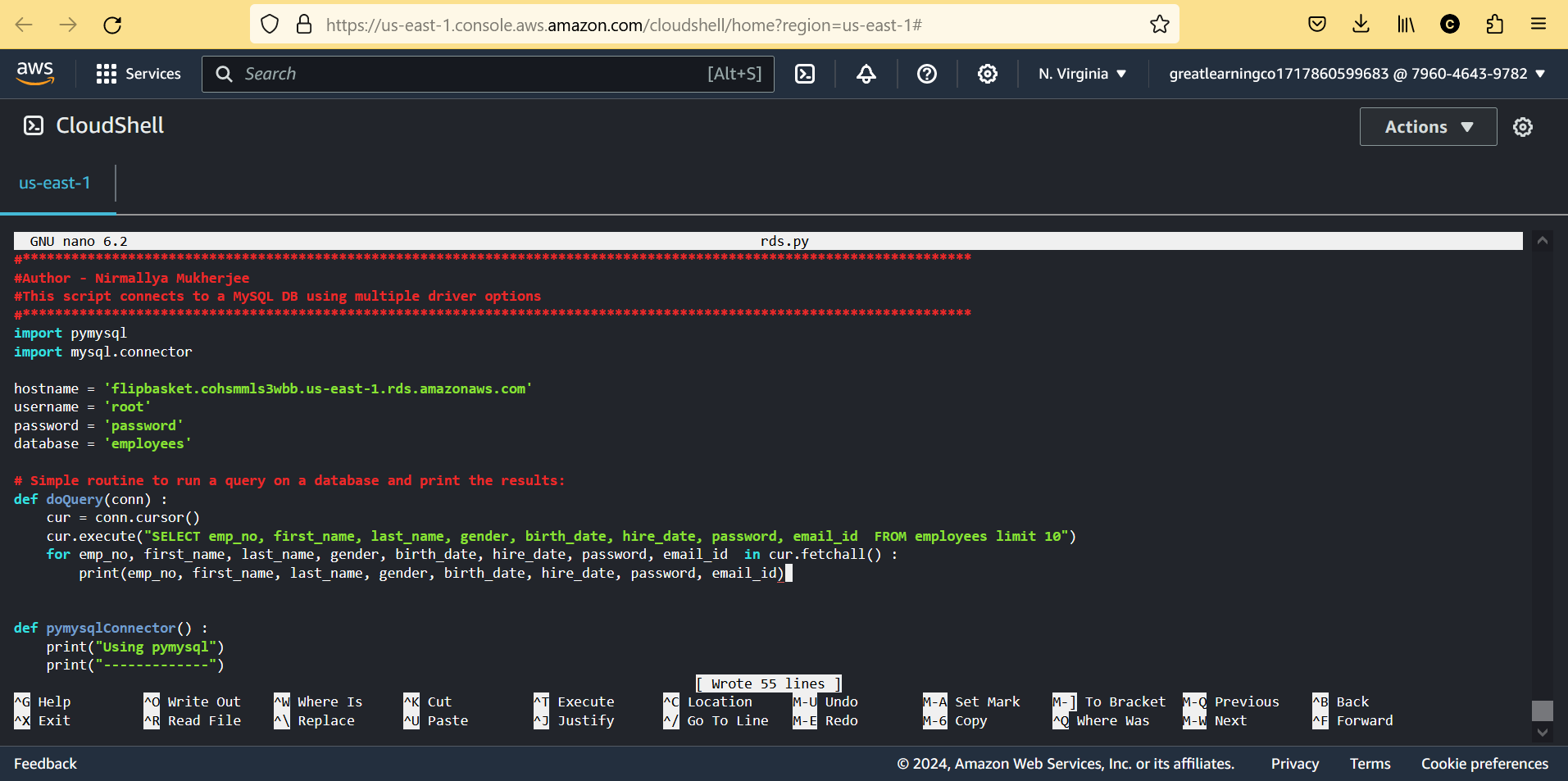
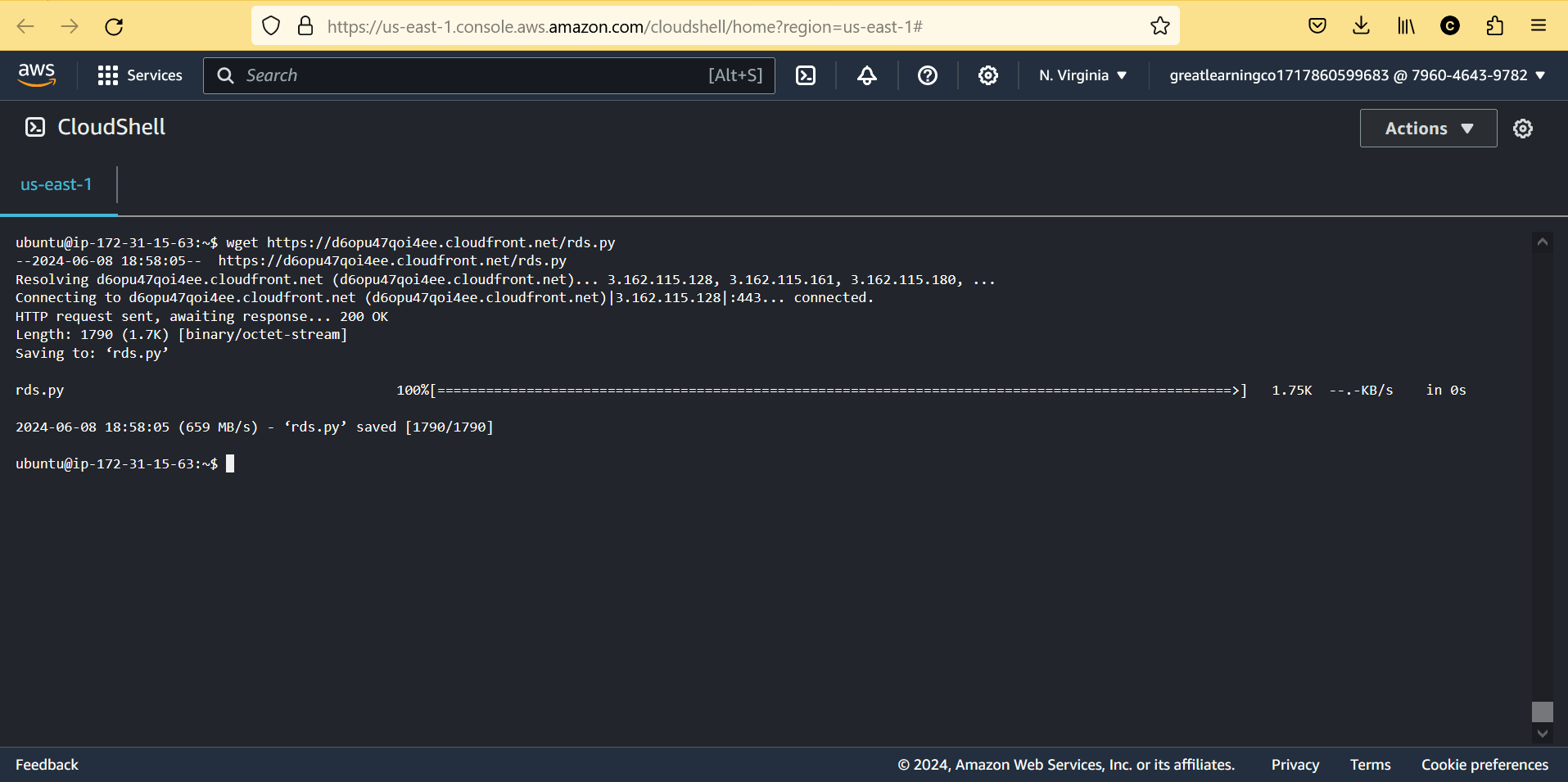
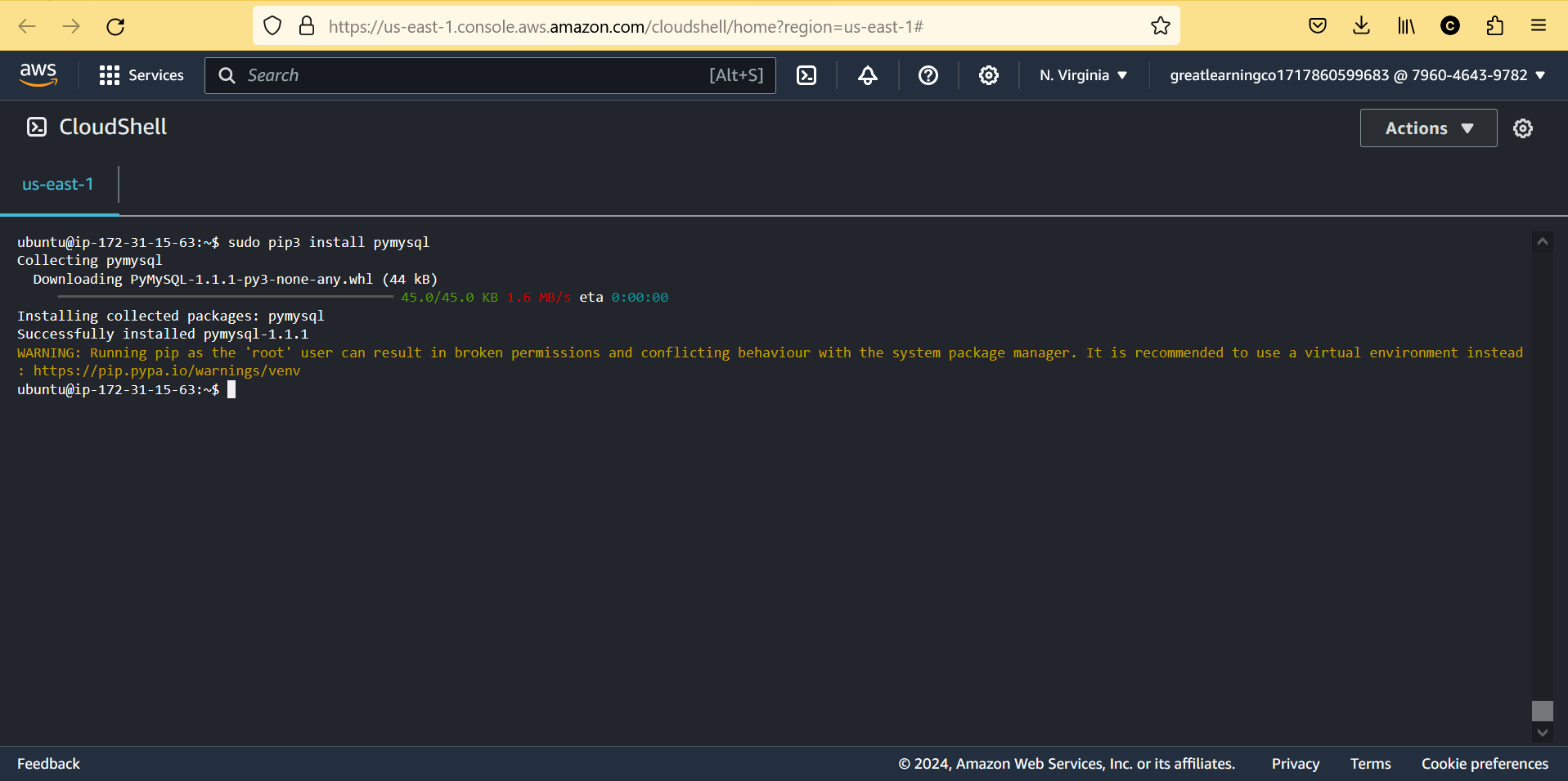
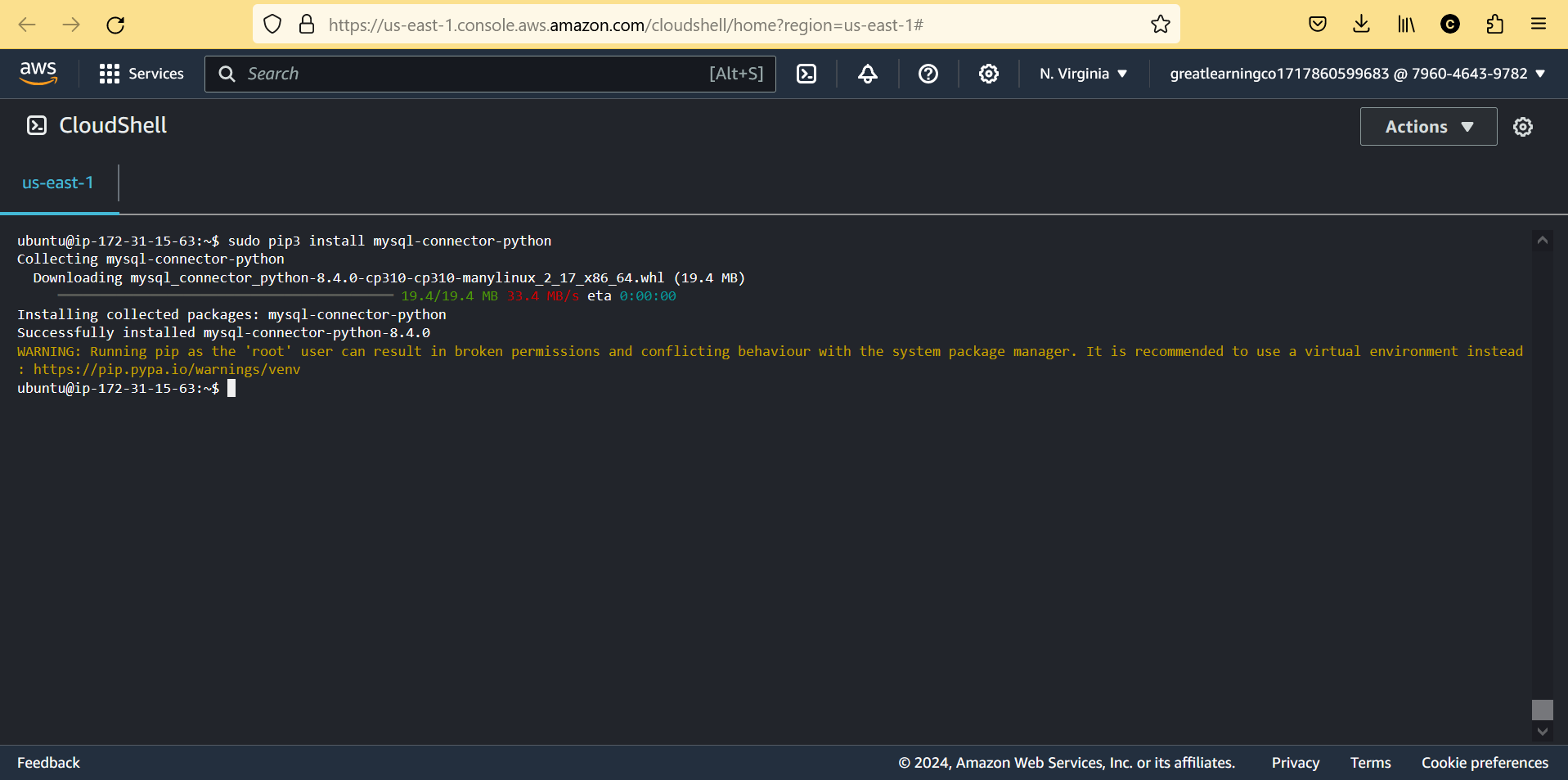
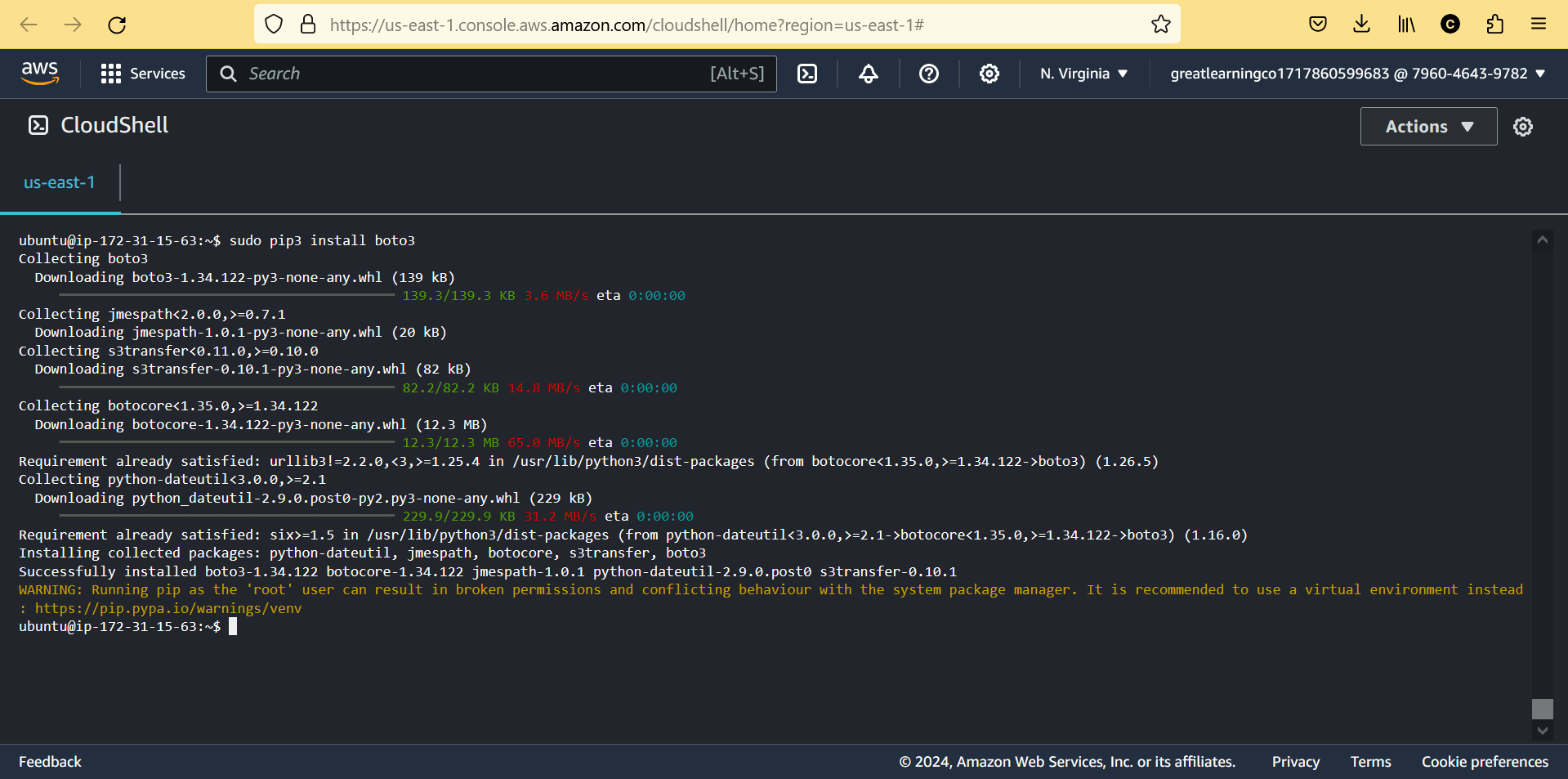
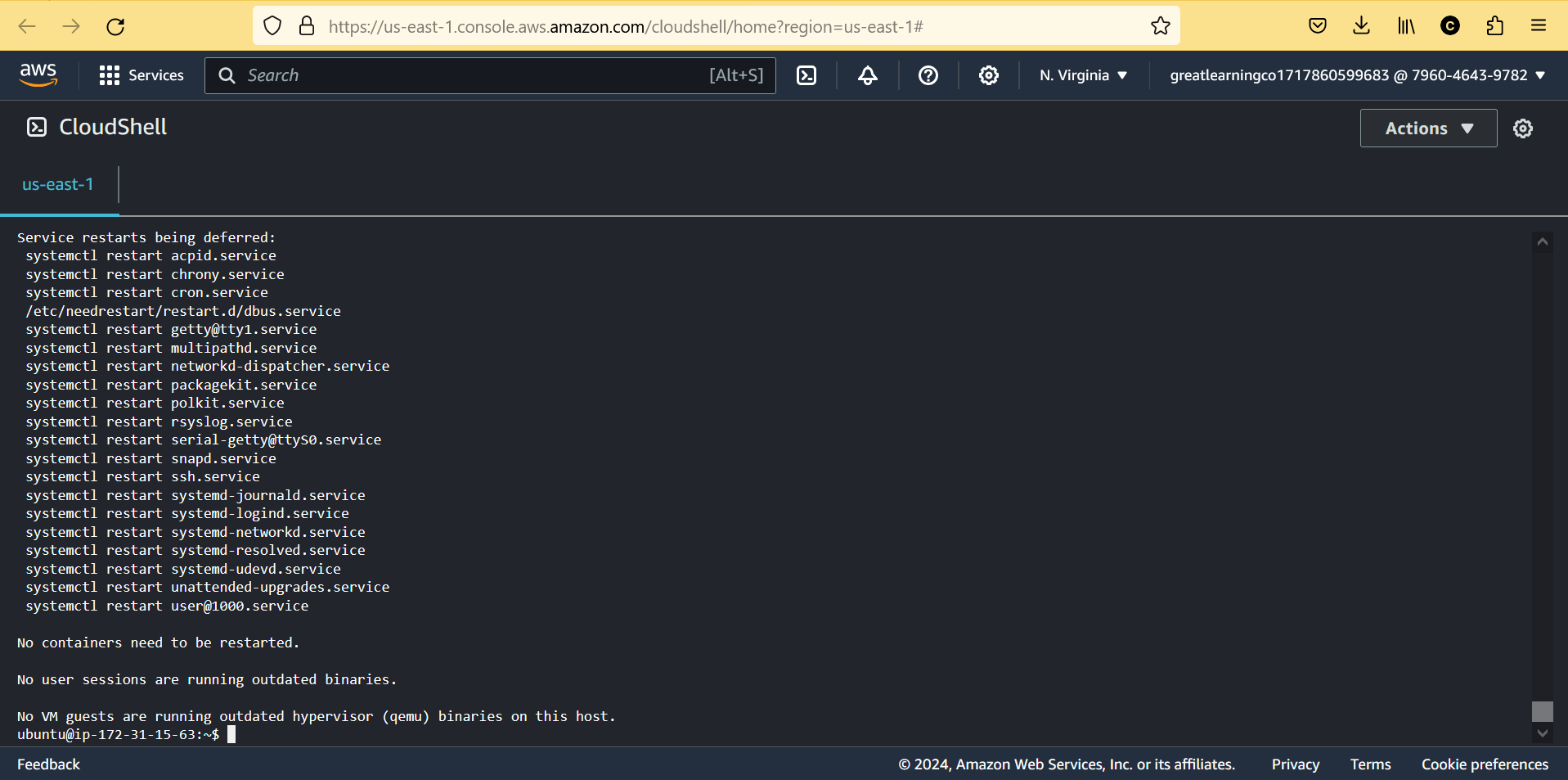
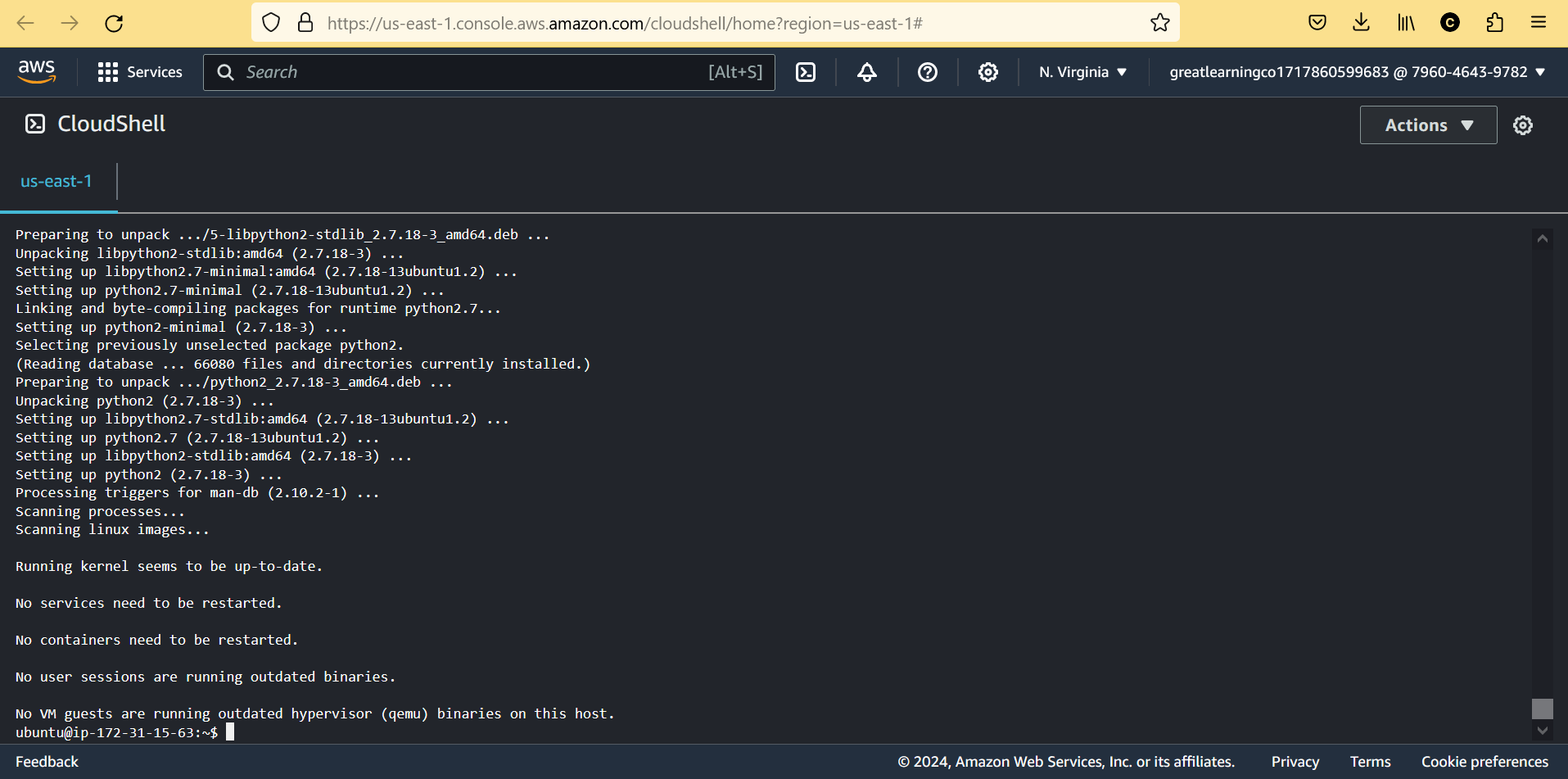
export LC\_ALL="en\_US.UTF-8" export LC\_CTYPE="en\_US.UTF-8" sudo dpkg-reconfigure locales

* Use the rds.py as the template and update the hostname to the endpoint of your RDS instance

and run the program

* + wget https://d6opu47qoi4ee.cloudfront.net/rds.py
  + Modify the python program in all the places marked with "TBD"
  + python3 rds.py
  + In case you get an error for the mysql connector just re-execute the sudo pip install command
  + **Q: state your observations when you remove the role and rerun the program (Hint - RDS SG)**

**Screenshots for Step-3:**

****

**Q: state your observations when you remove the role and rerun the program (Hint - RDS SG)**

**Ans:**

When We detached the role from the EC2 instance, deleted it and re-run the program, it still worked normally and the Python program installed on the EC2 instance could fetch data from the Database created on the RDS instance. It indicates that this infrastructure doesn’t require a dedicated IAM role for the EC2 instance to fetch data from RDS instance..

The possible reasons are:

1. We modified the security group of the RDS instance to include the entire CIDR block of the VPC instead of using a specific IP address. This configuration allows all resources within the VPC, including the EC2 instance, to attempt to connect to the RDS instance at the network level.

2) RDS uses database-specific credentials (username and password) for access control. When our application on the EC2 instance tries to connect to the RDS database, it must provide valid database credentials. If these credentials are correct, the connection will be established- allowing the application to interact with the database.

IAM roles with RDS permissions (like AmazonRDSFullAccess) are primarily used for managing RDS resources via the AWS Management Console, AWS CLI, or AWS SDK. These roles enable users or applications to perform administrative tasks on RDS instances (e.g., creating instances, modifying configurations) but do not govern database-level access control directly.



# How to do it? – 4 (Advanced \*)

* Once you have updated the program to connect with the database, do the following
  + Create a Redis cluster (self managed / Elasticache)
  + Modify the program to include another method "getOrder(...)"
  + Add a class that represents the order
  + This method needs to check if the order exists in the cache, if found then get from cache and return else go to the DB, put in cache and return
    - Use the python modules - redis & pickle

\* This step of the assignment is NOT mandatory, and will not have any impact on the grades. You can choose

to do / skip it only from the learning perspective.



# Mandatory step: Resource Clean-Up

* Cloud is always **pay per use model** and all resources/services that we consume are chargeable. Cleaning up when you’ve completed your lab or project is always necessary. This is true whether you’re doing a lab or implementing a project at your workplace.
* **After completing with the lab, make sure to delete each resource created in the reverse chronological order.**
* Check resources in each cloud region that you have worked on before logging off.
* Since the dashboard doesn’t show cross-region resources, it is up to you to find and delete them.



# Marks Break Down –

Please find the marks breakdown below (Screenshots must

be provided for every step)

1. Creation of RDS instance : 15 marks
2. Data Population in RDS : 15 marks
3. Installation of Python libraries : 5 marks
4. Execution of Python program : 15 marks