Read the following instructions before completing the given task.  
  
Step 1: Click on the Online Task Link

Step 2: Download the document

Step 3: Check the online questions

Step 4: Complete the task and take a screenshot for each and every task and upload

Step 5: Save the document

Step 6: Upload the document in Google form

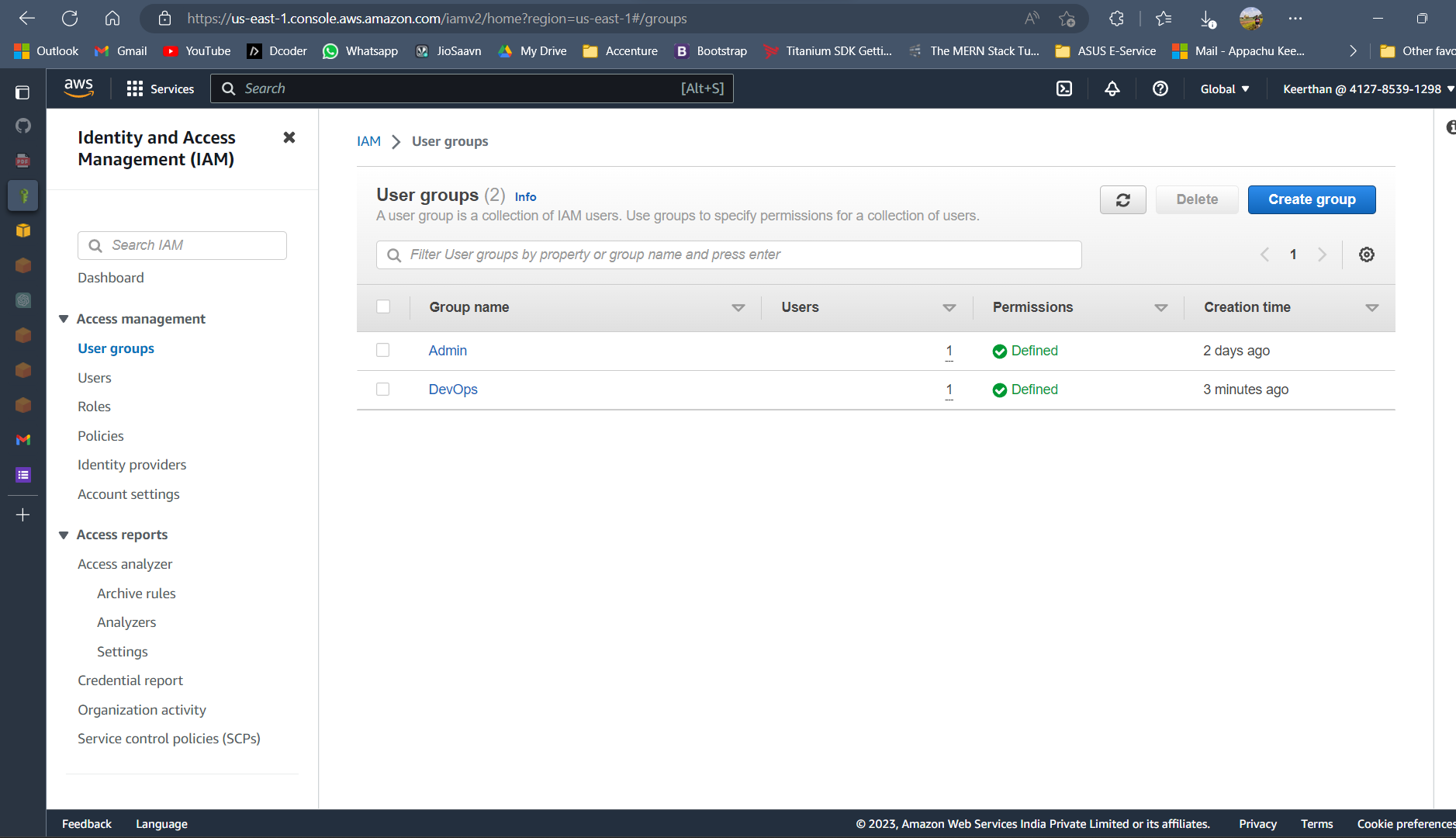
* **AWS Account is mandatory to complete the tasks**.
* **Include AWS account Name in your screenshots**
* **Resume / Copied and similar tasks will be rejected directly.**

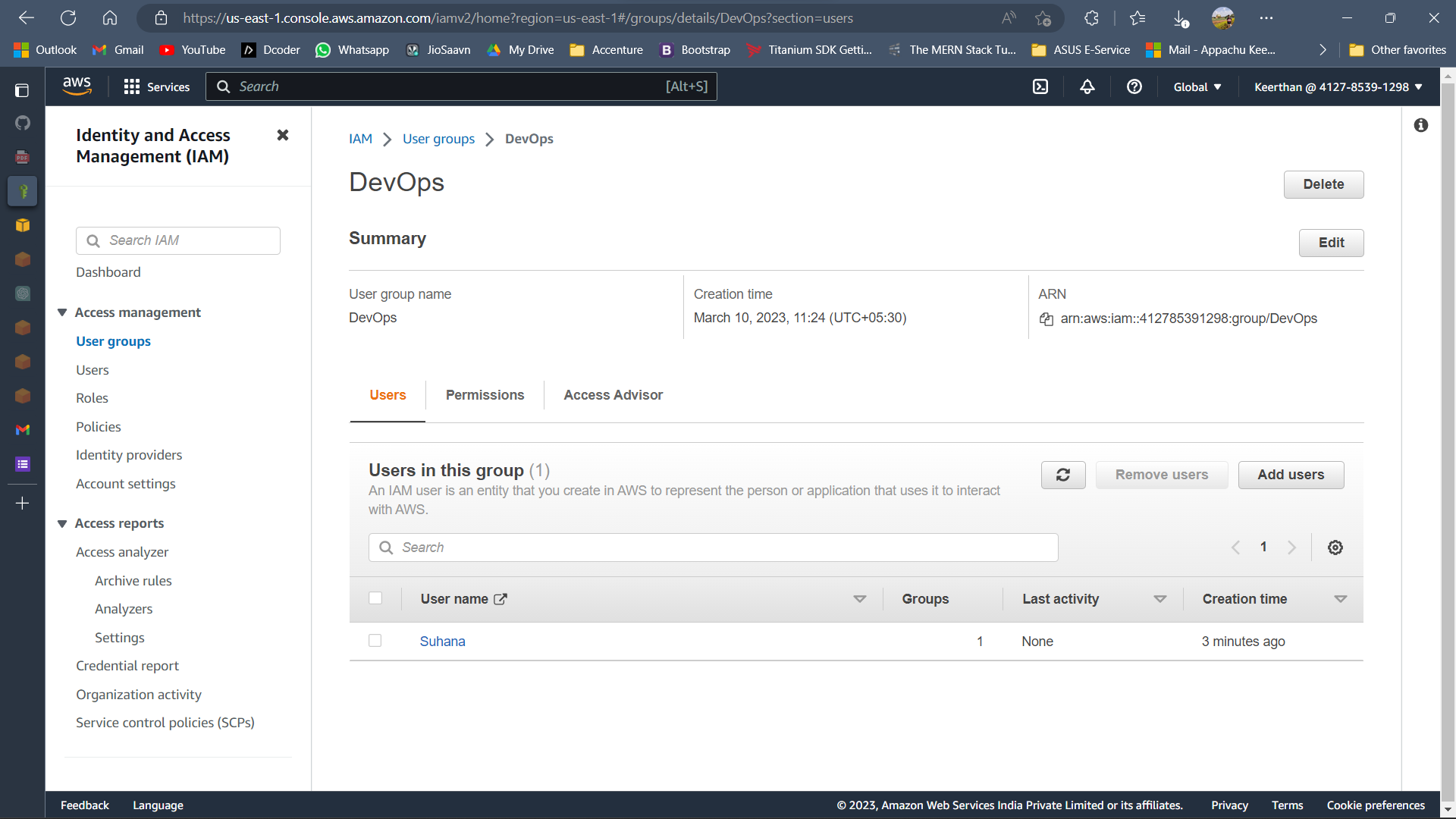
Assignment 1. ->   
==============

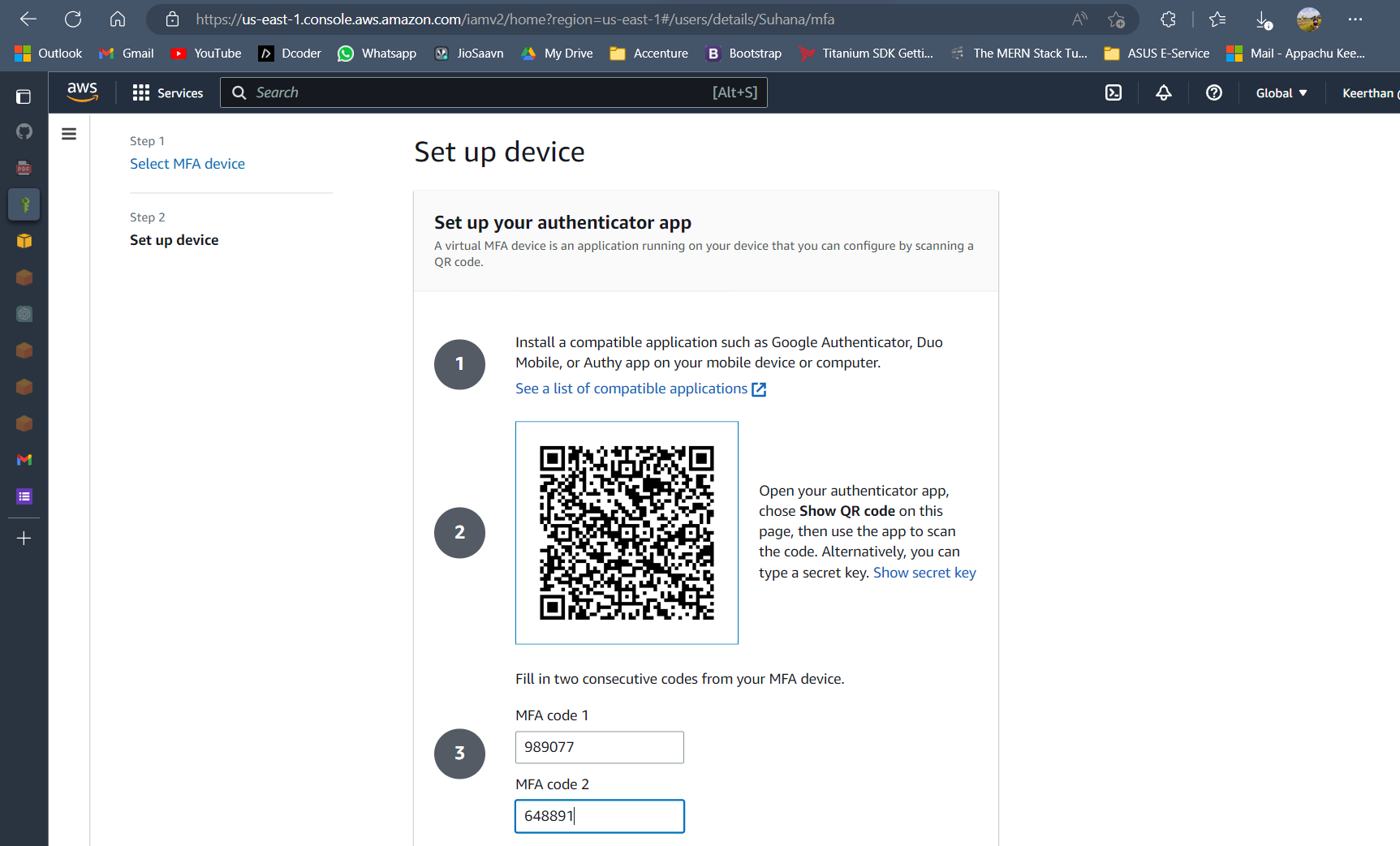
**Create the user "Suhana", enable MFA, create the "Devops" group and add Suhana to it, and perform login as Suhana**

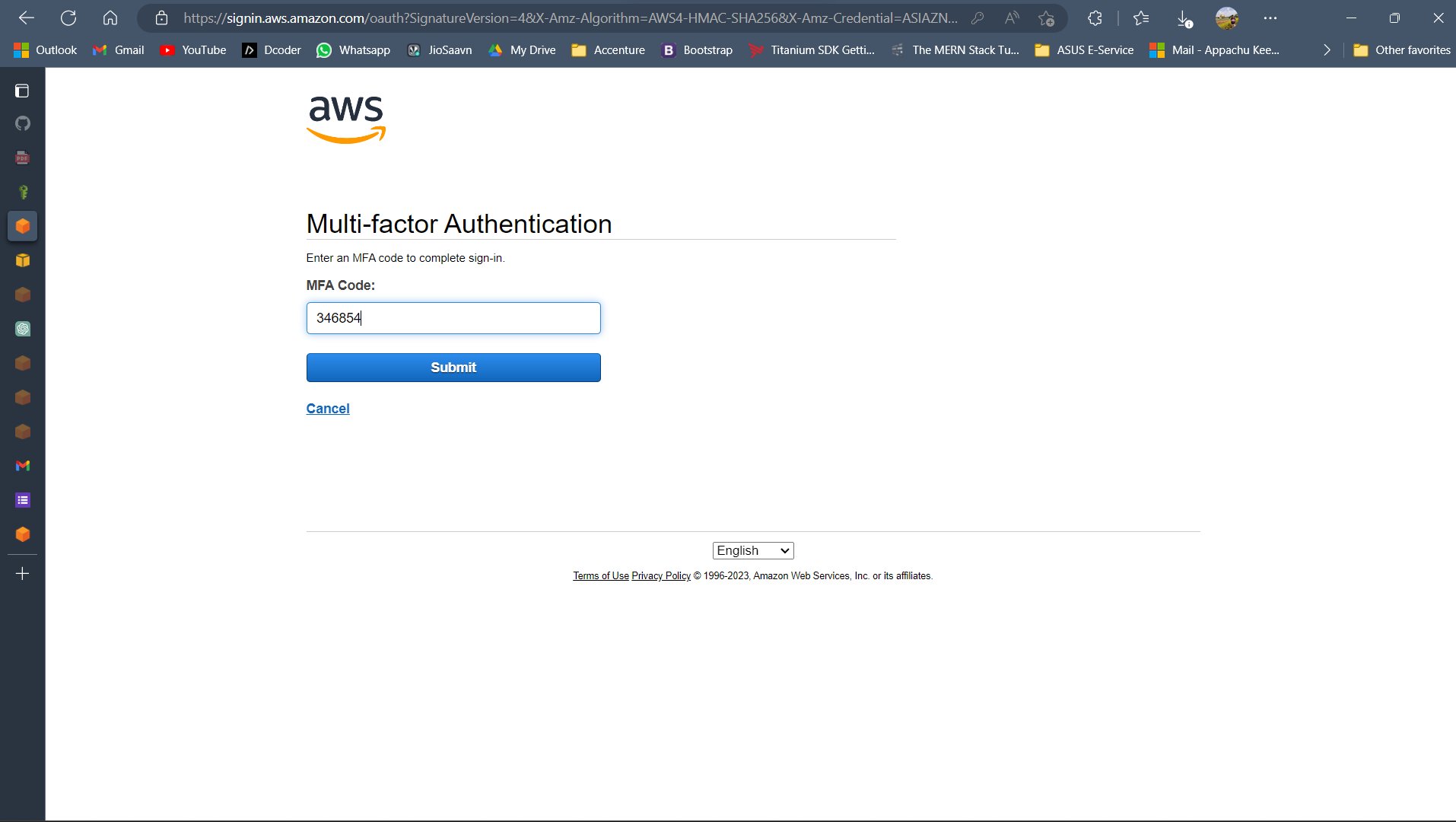
**Enable Multi-Factor Authentication (MFA) for the "Suhana" user**

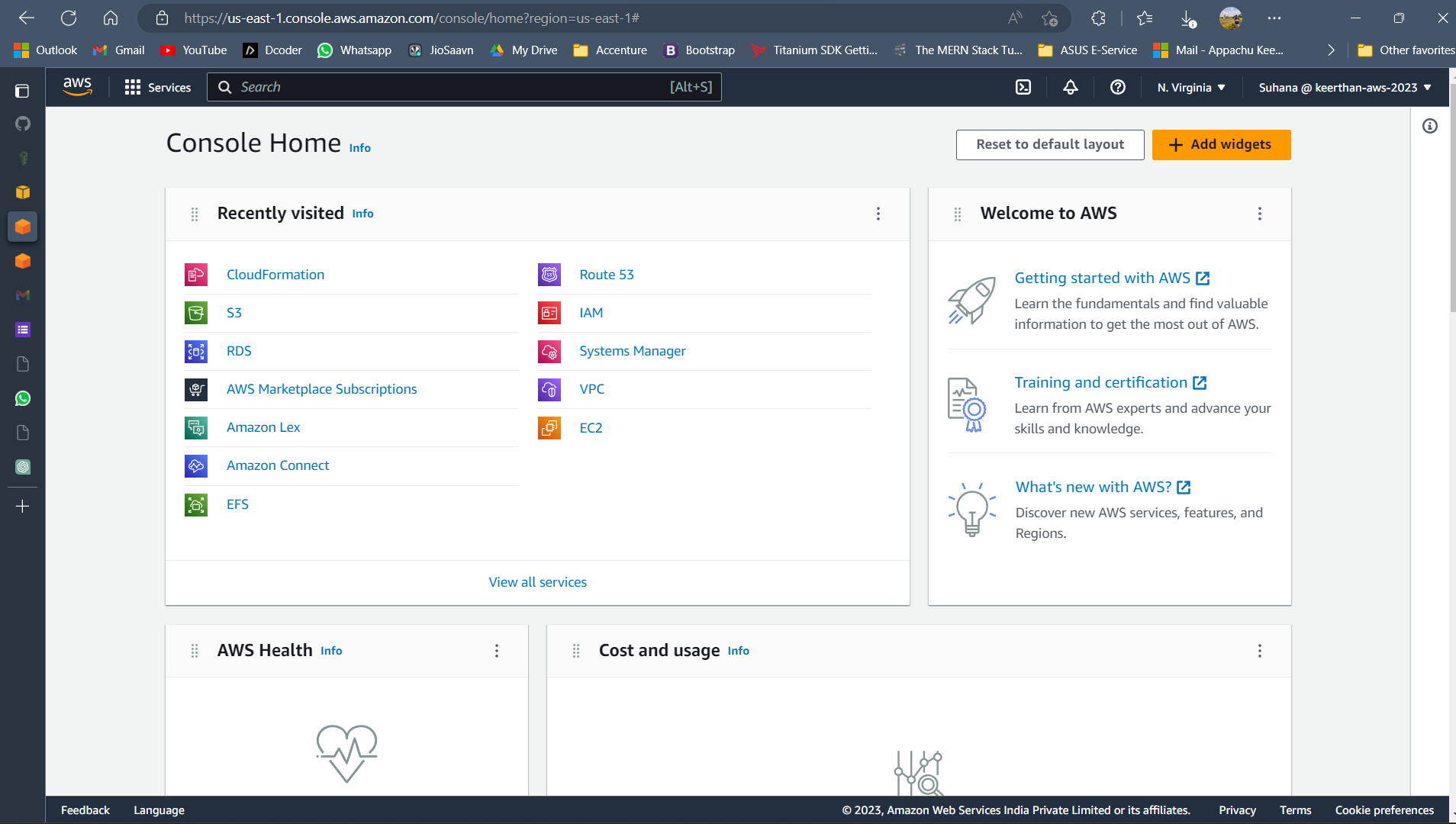
Upload the final output Screenshot









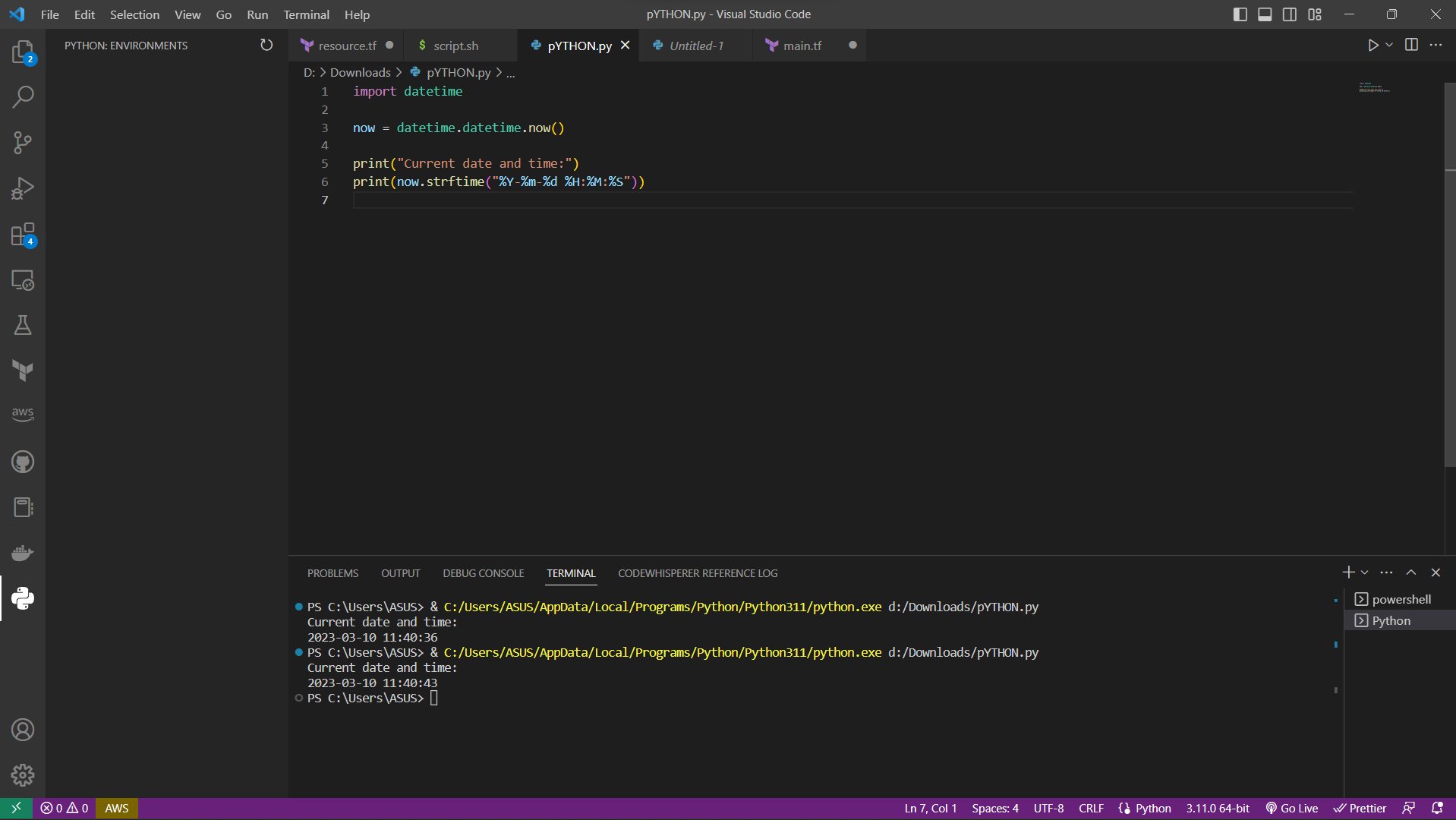


Assignment 2. ->

=============

**Shell:** **Write a Python program to display the current date and time.**

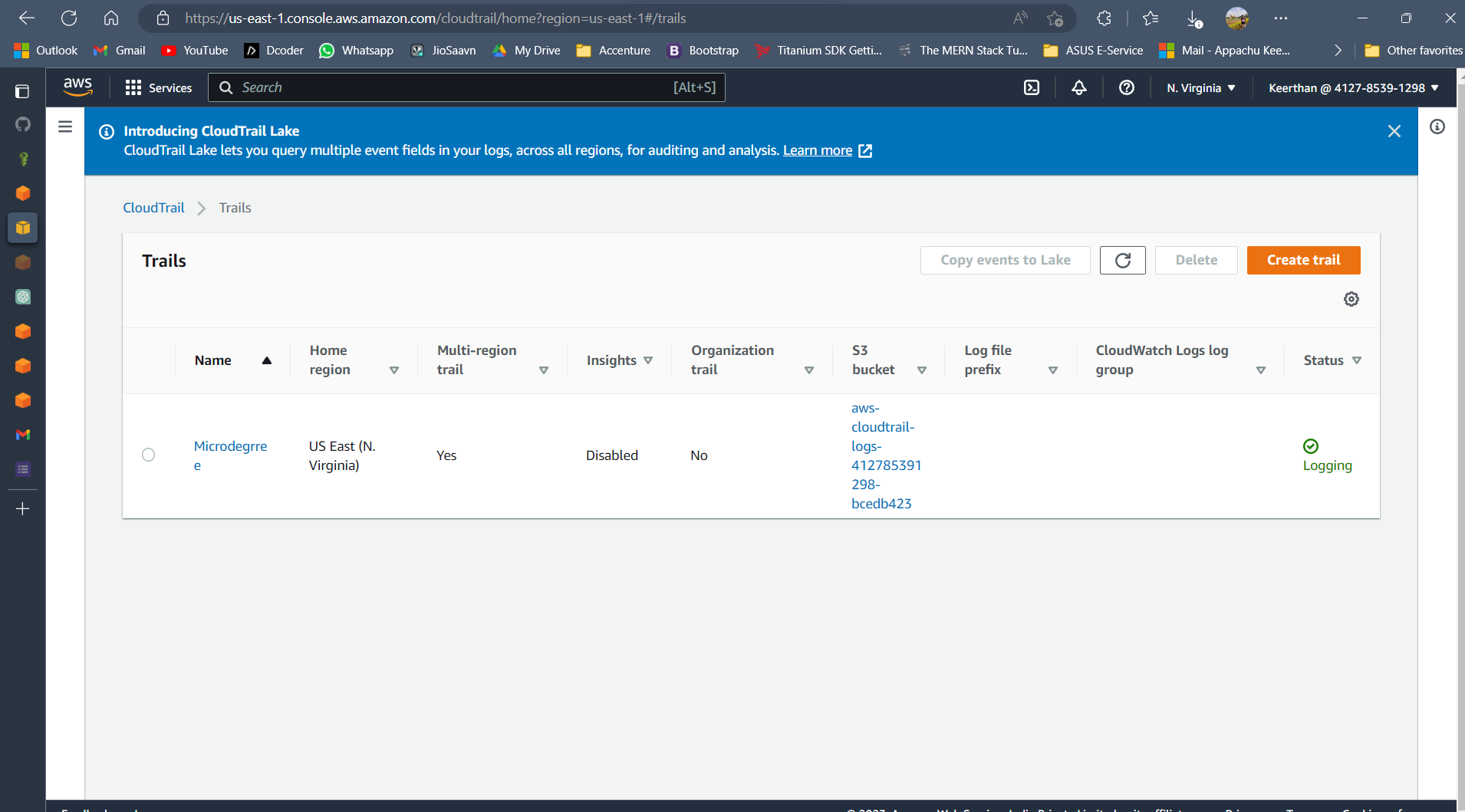
Upload the final output Screenshot



Assignment 3. ->

============

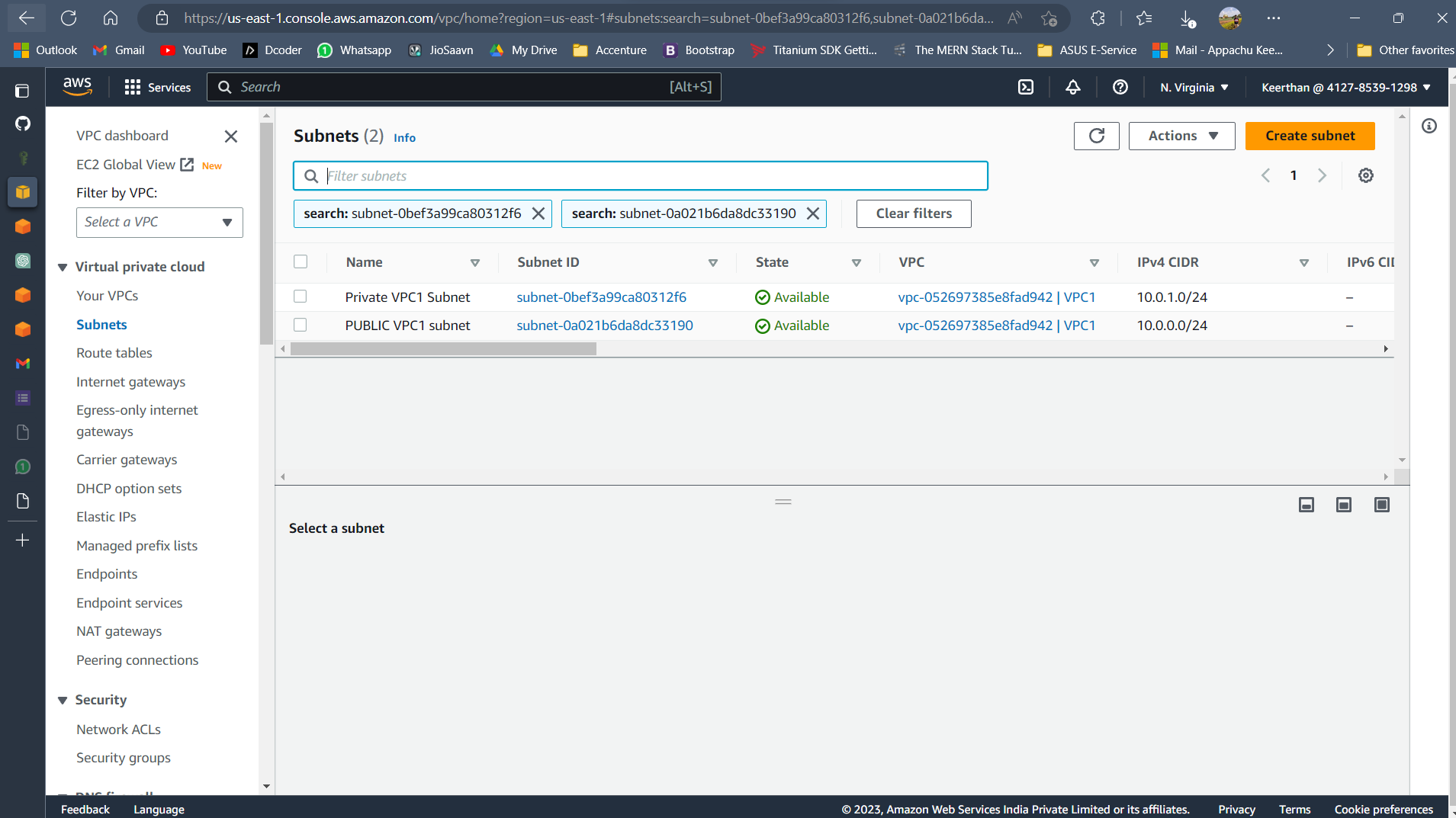
**Create Cloud-trail and store logs in S3 buckets, enable for multi-regions**



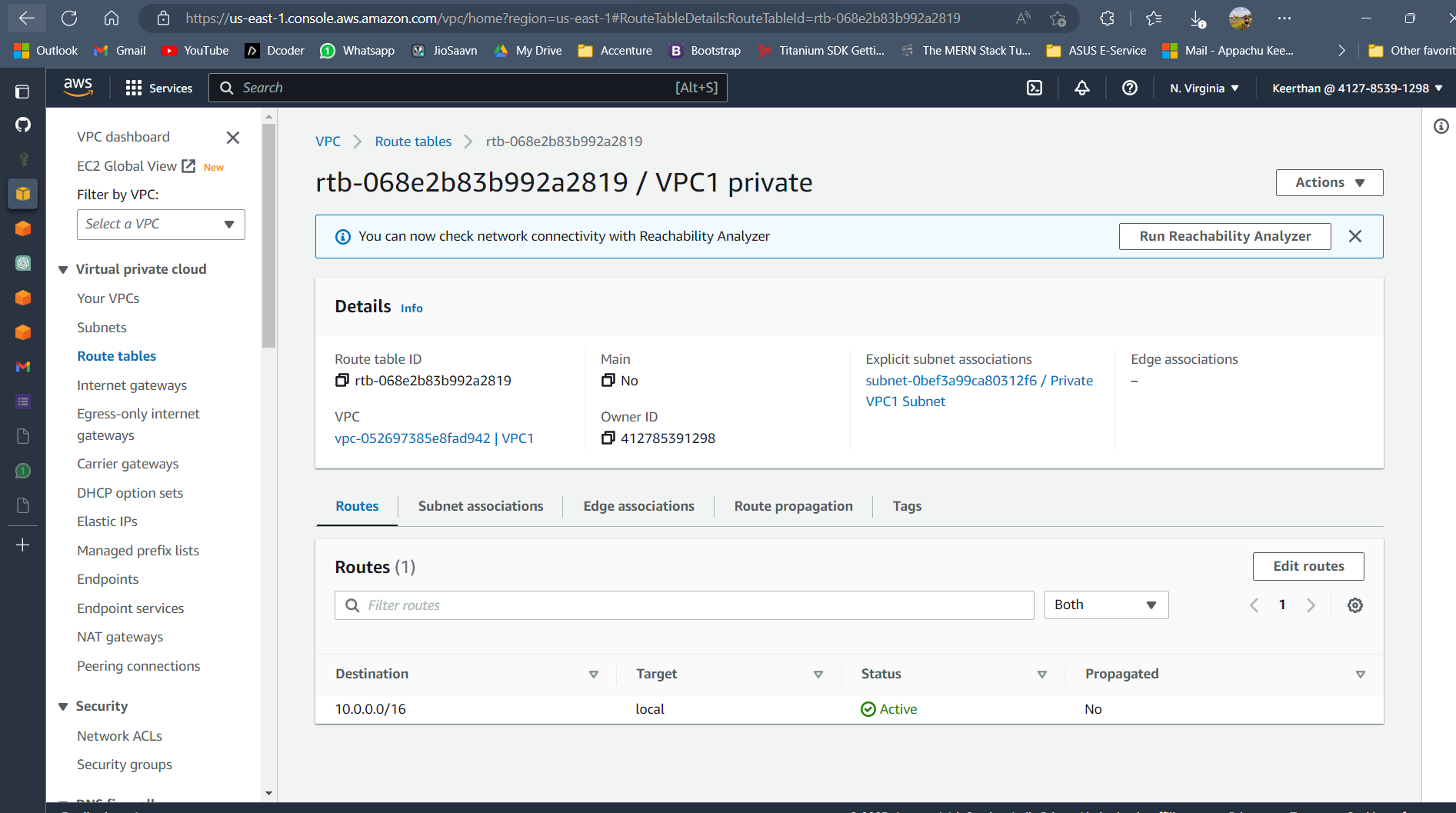


Assignment 4

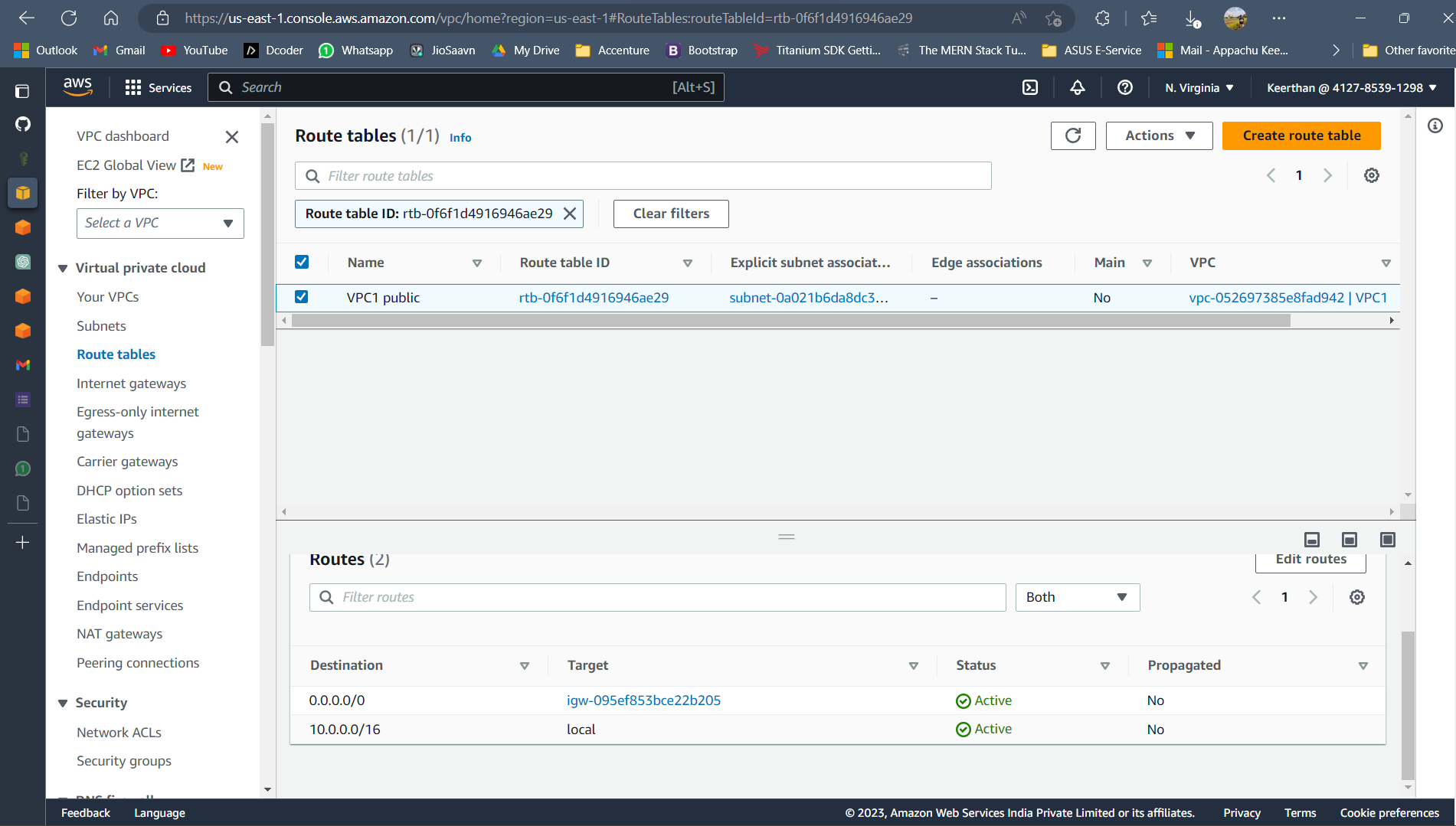
**Create a VPC with 2 subnets (public and private) in different AZs, attach an internet gateway, and launch EC2 instances. Try to access the private instance through the instance in public subnet (Clue:- using SSH)**

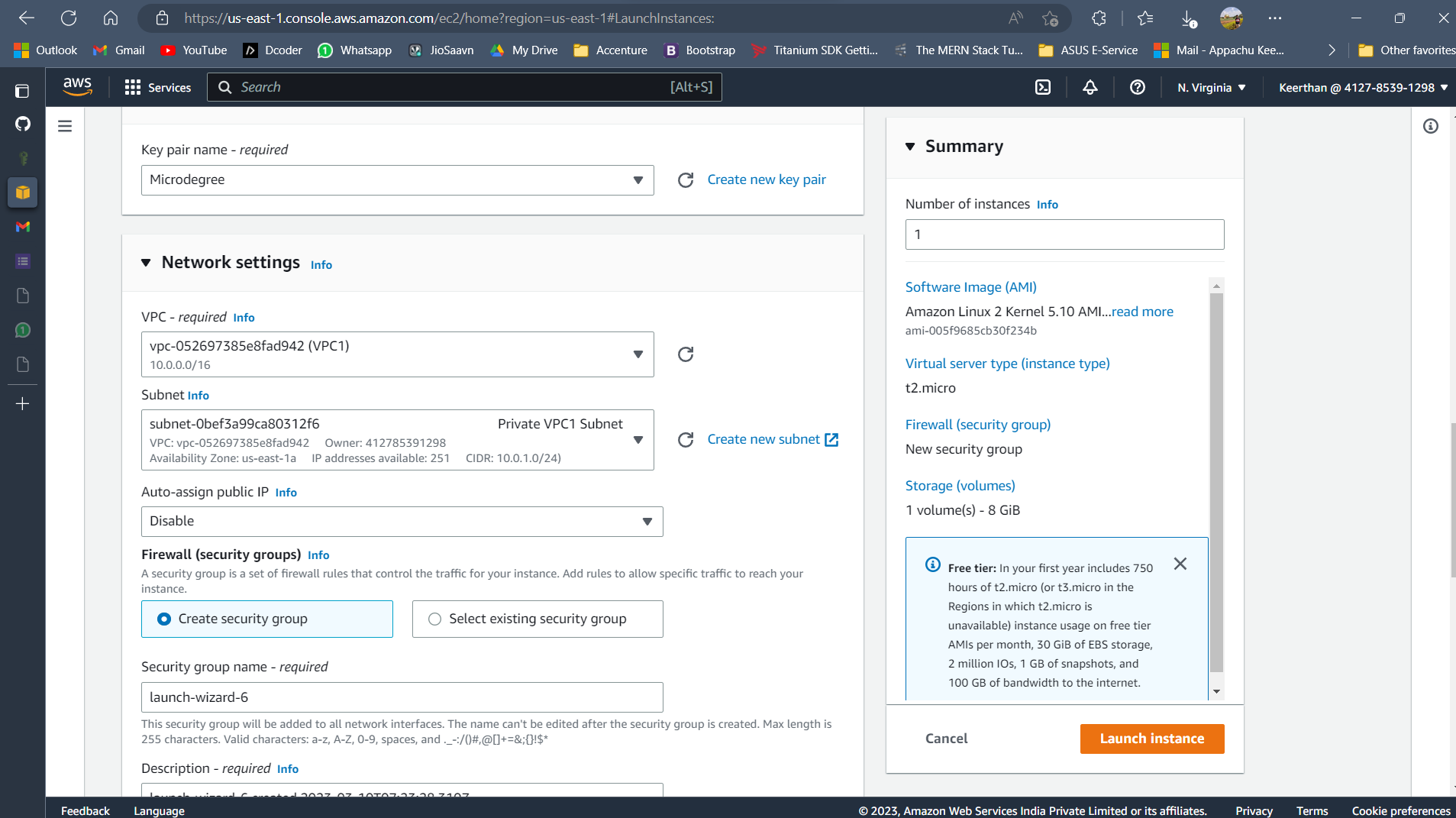


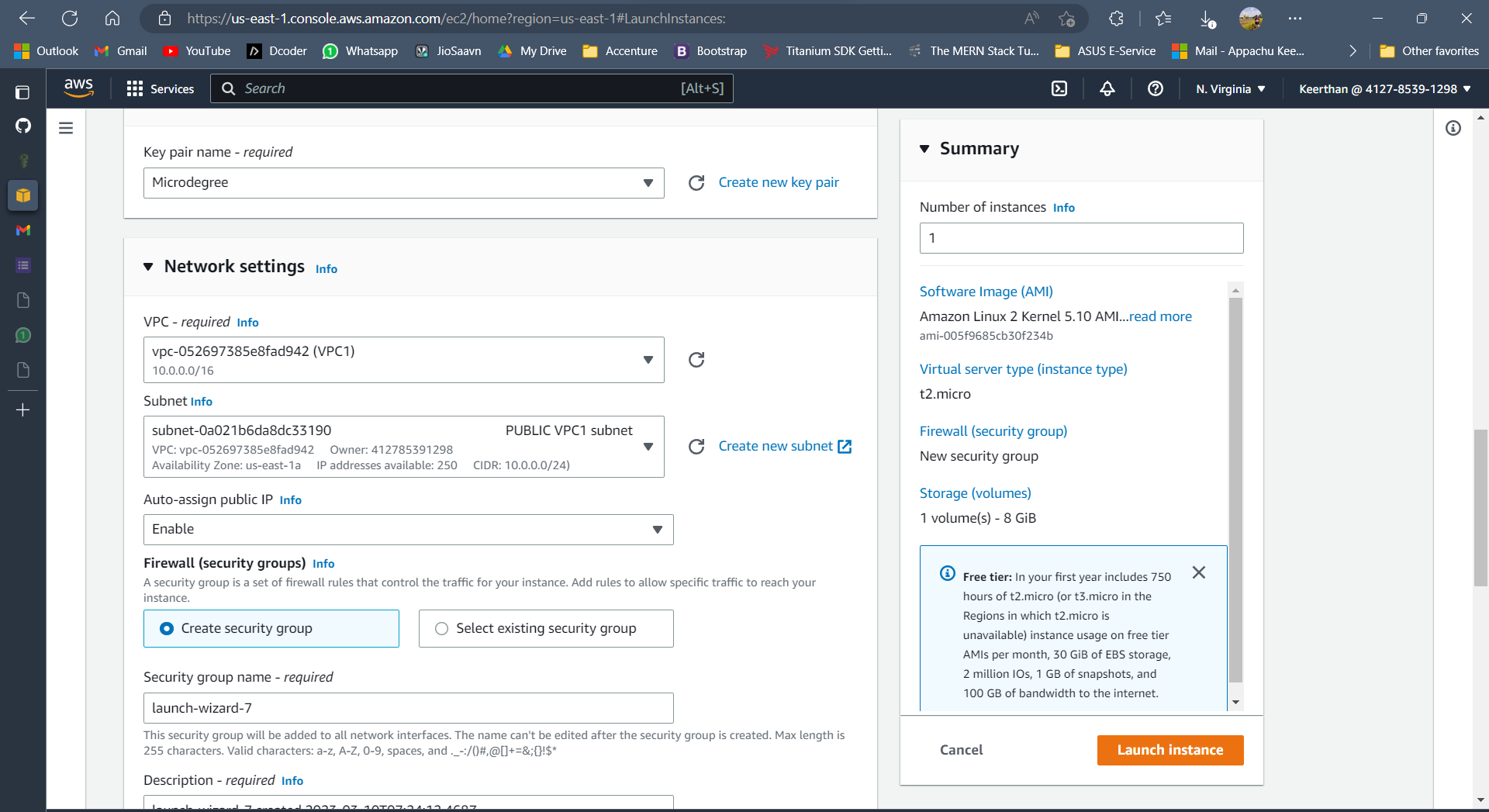
Private



Public



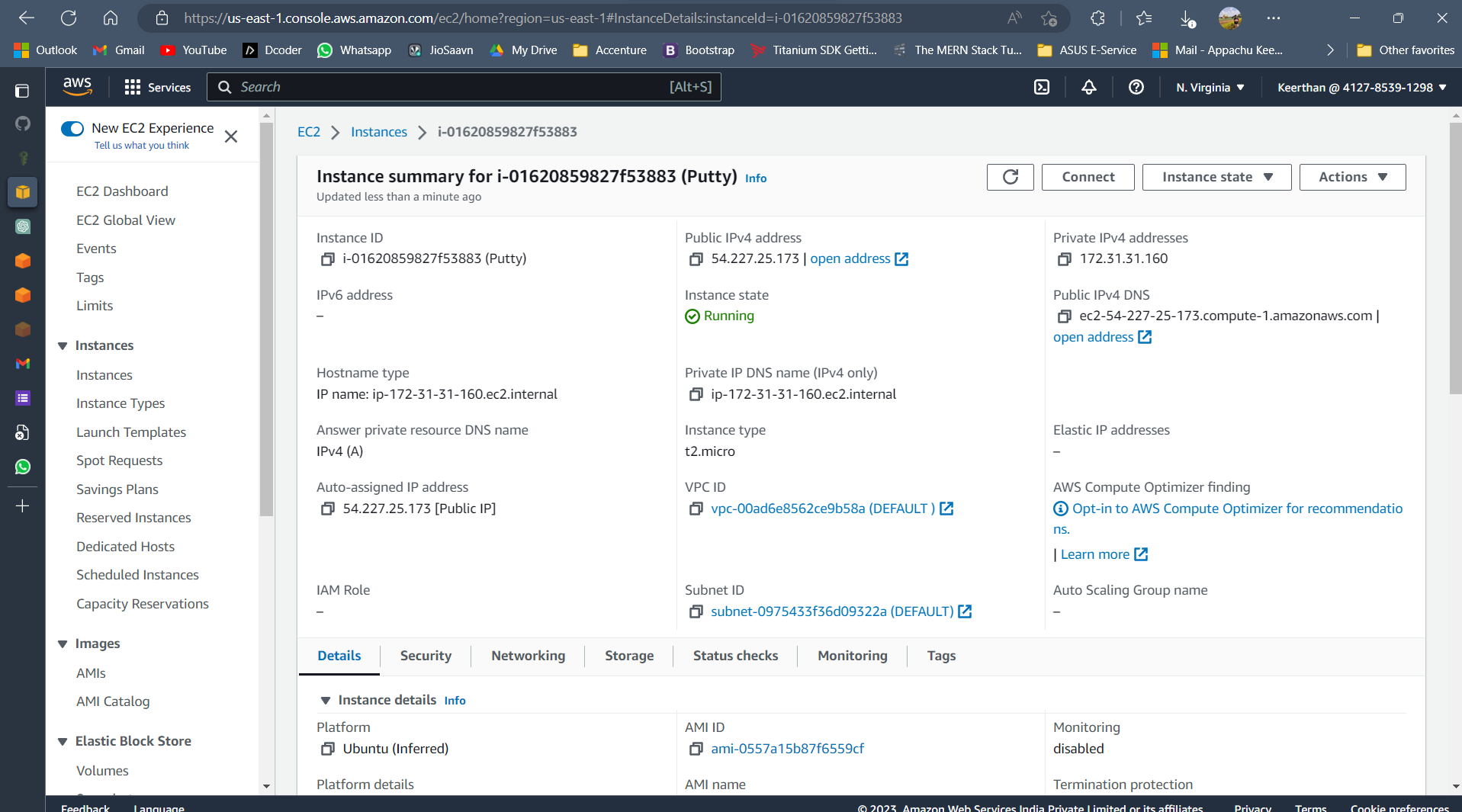


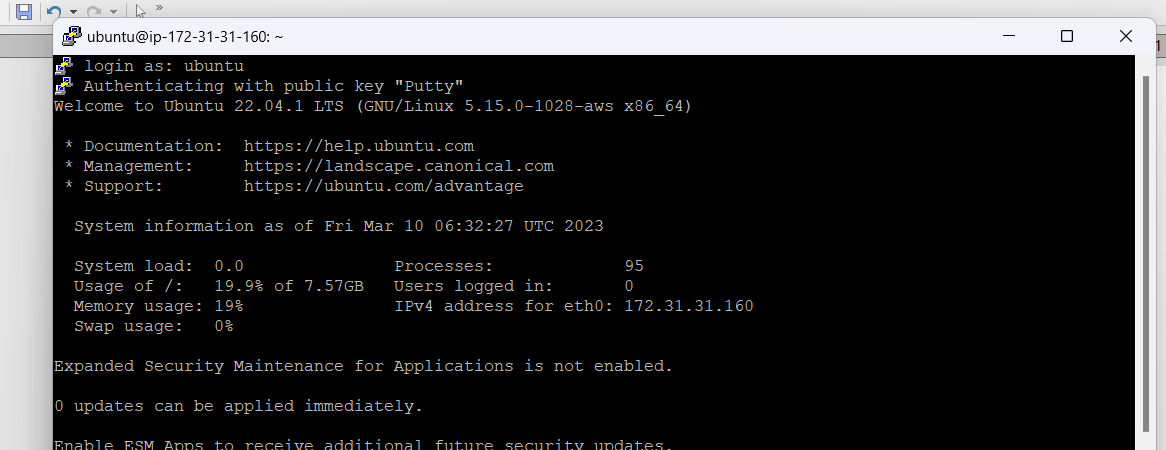


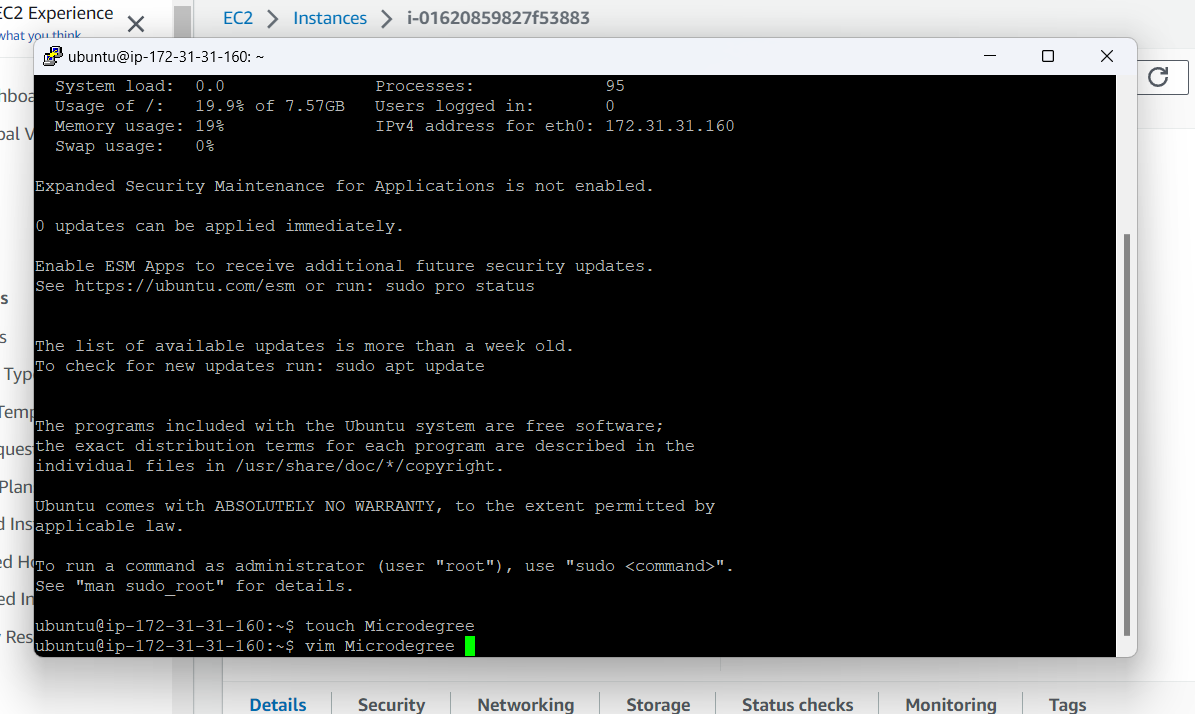
Upload the final output Screenshot

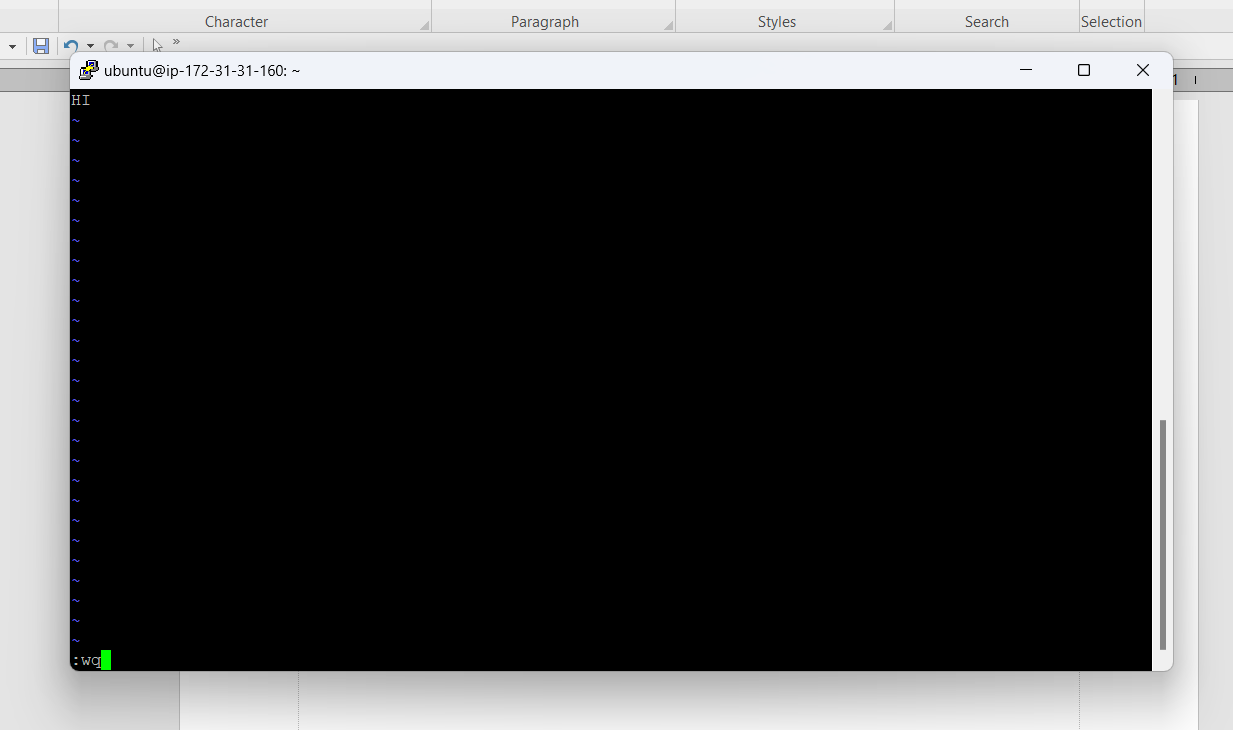
Assignment 5

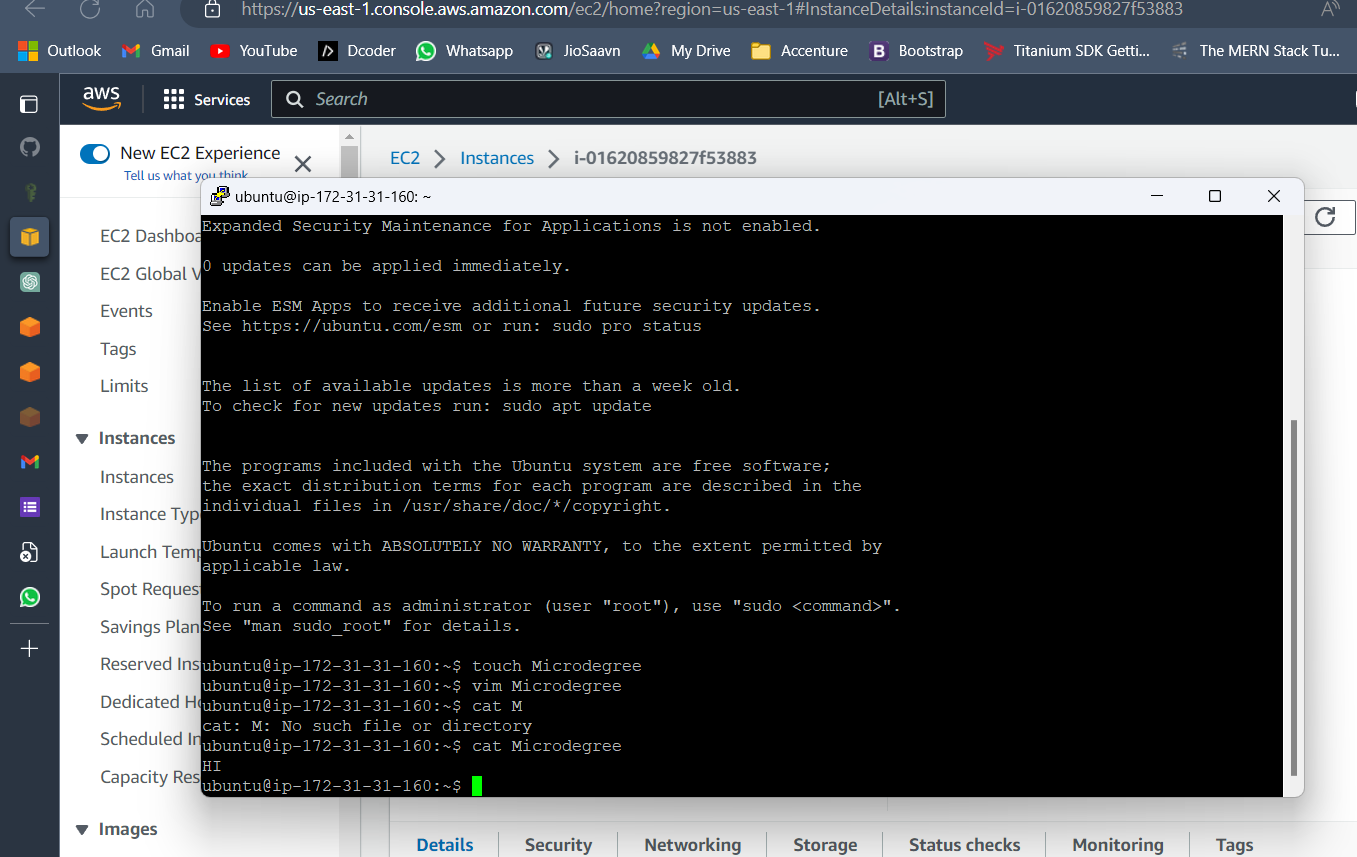
**Open putty, connect public IP and run some commands and show the output. Create Empty file, add some content and edit it in Vim.**





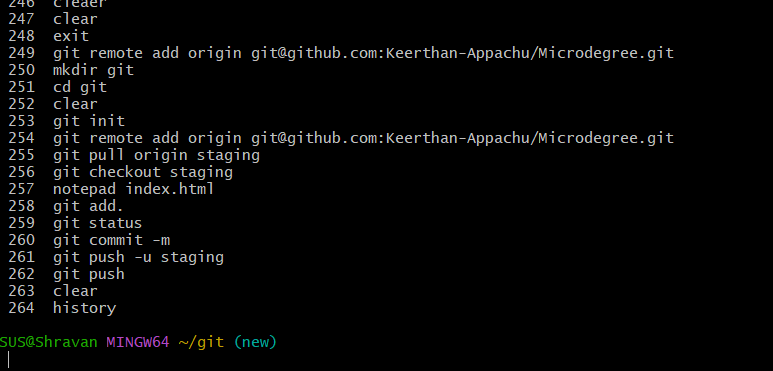


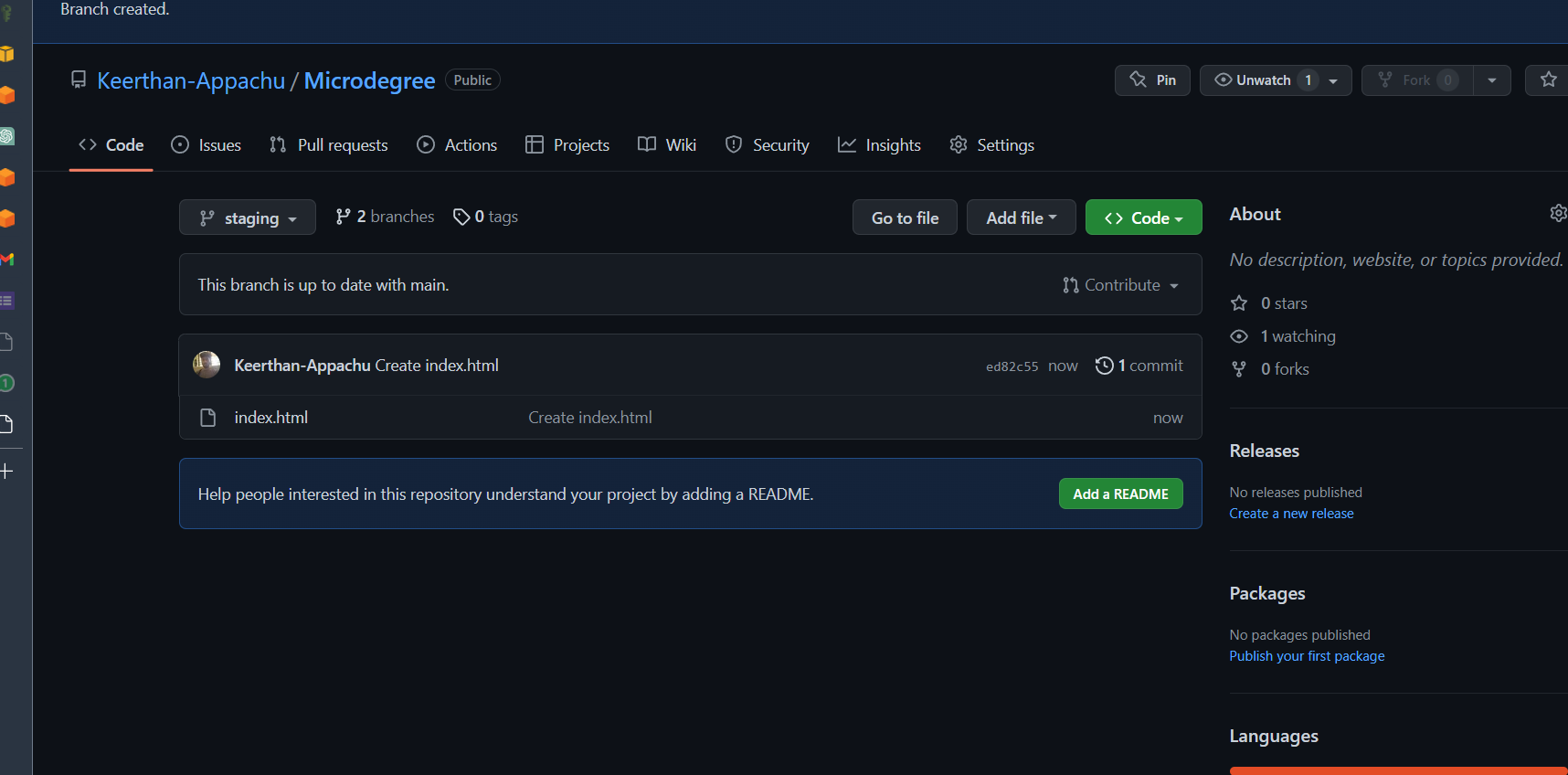


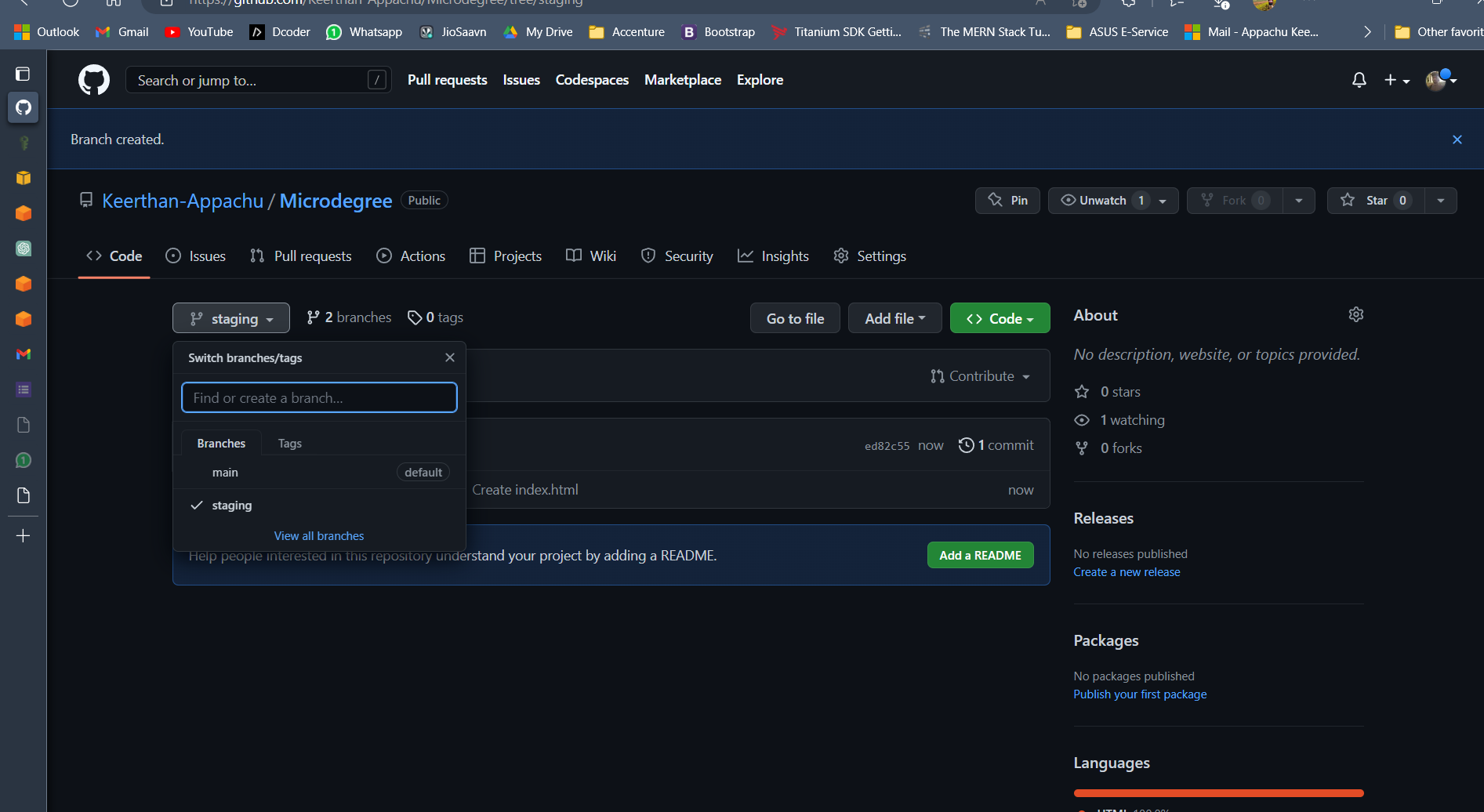


Assignment 6

**Create a Staging branch in GitHub, push code from local repository to remote, and share commands**Upload the final output Screenshot







Assignment 7

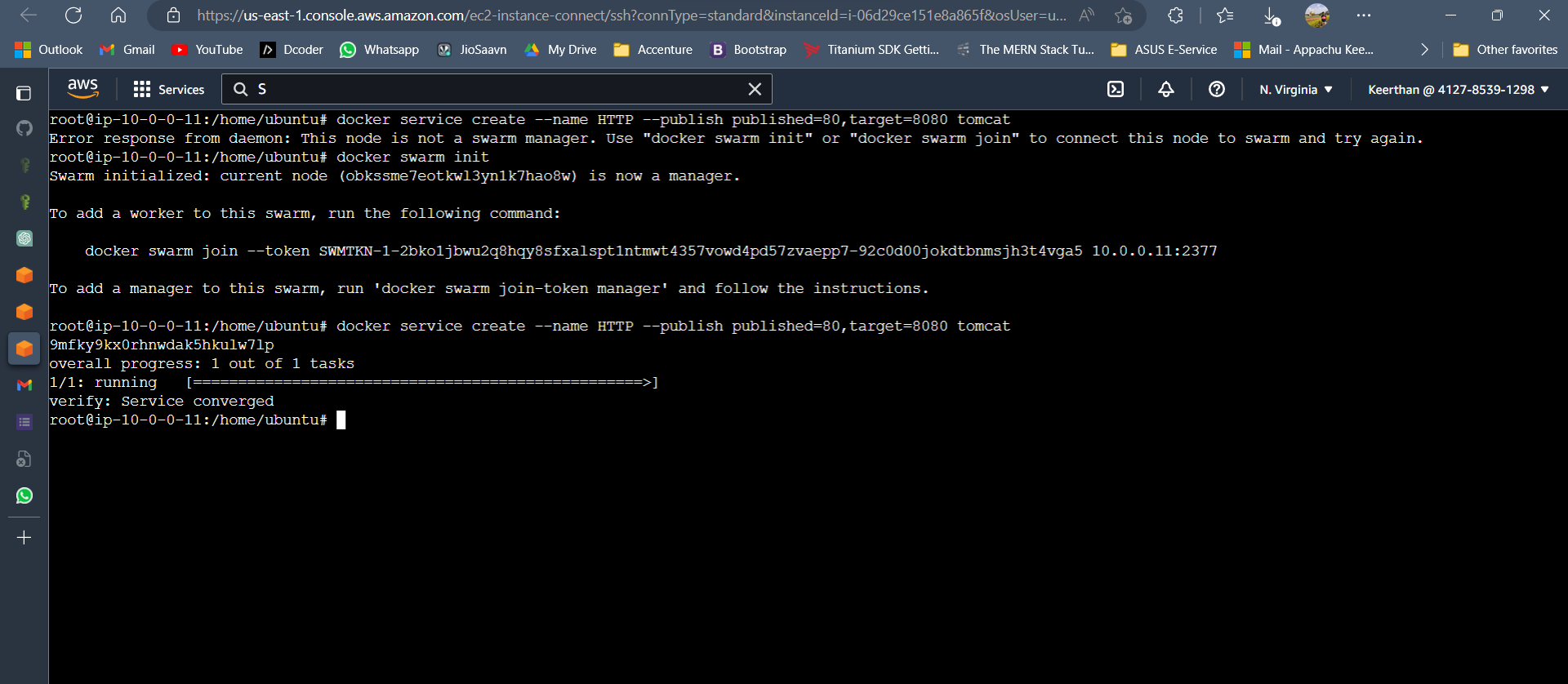
**Create a sample Jenkins CICD pipeline and share output logs and pipeline script**

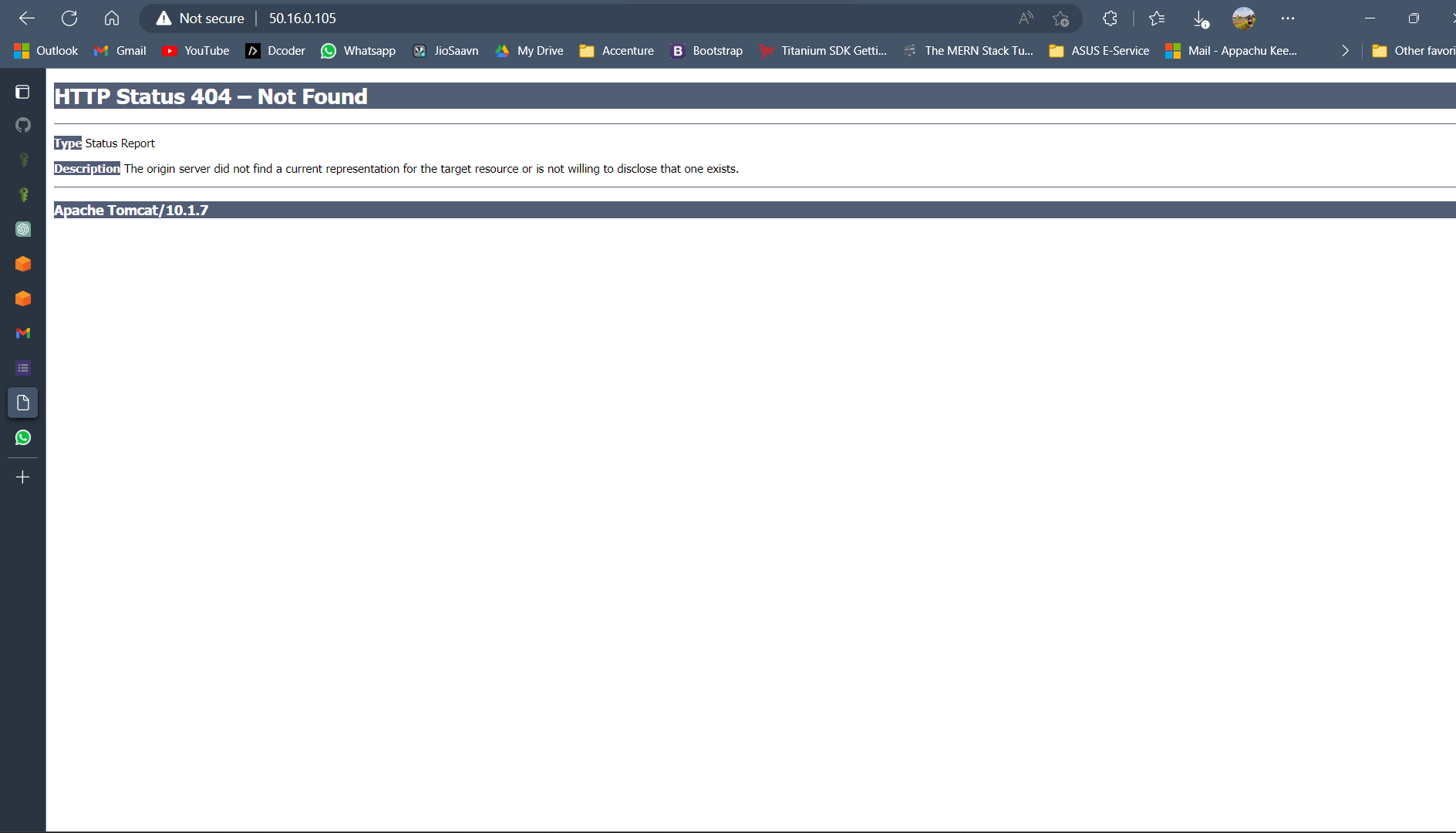
Upload the final output Screenshot

Assignment 8

**Create a docker service with image "tomcat", name "HTTP", port 80**

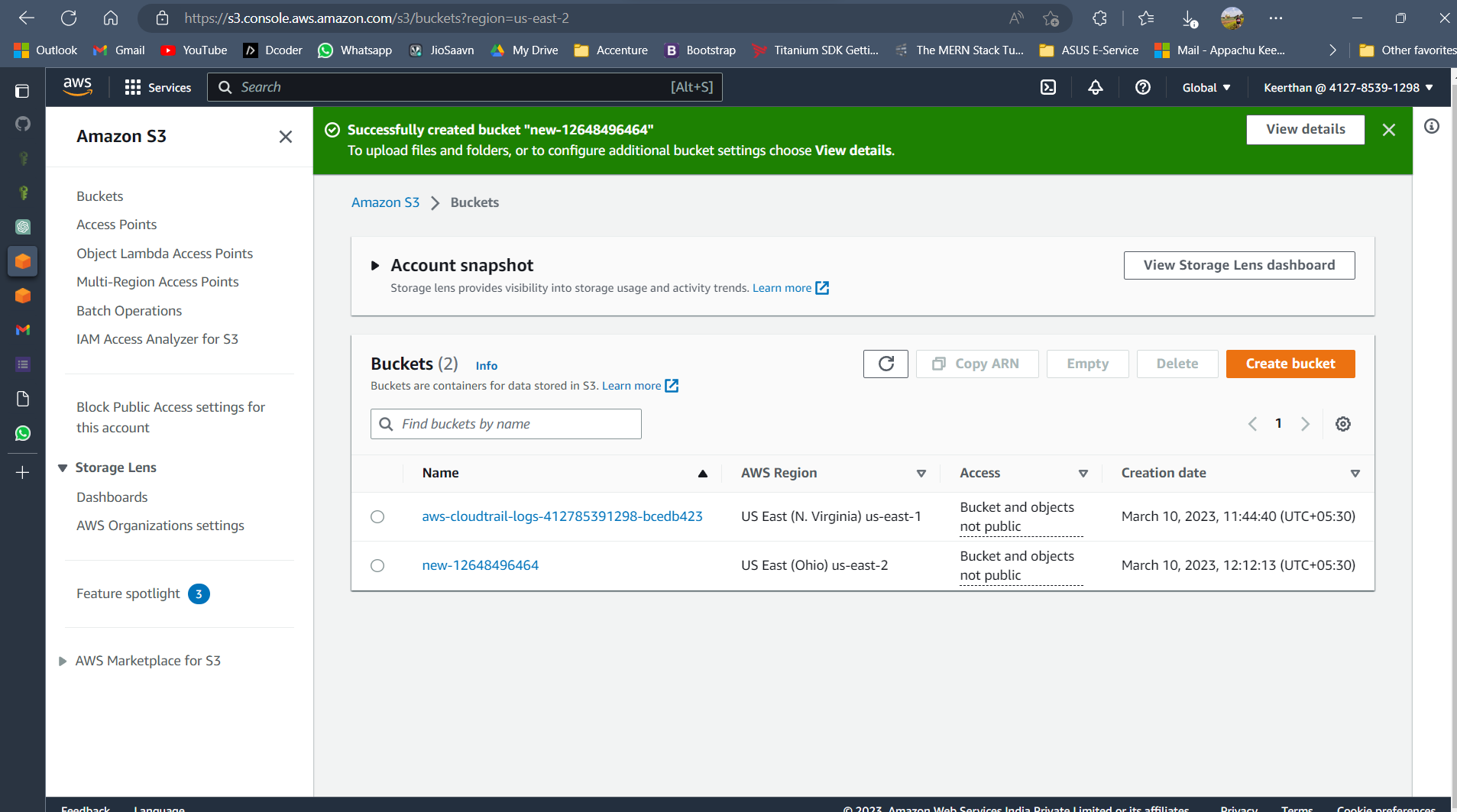
Upload the final output Screenshot

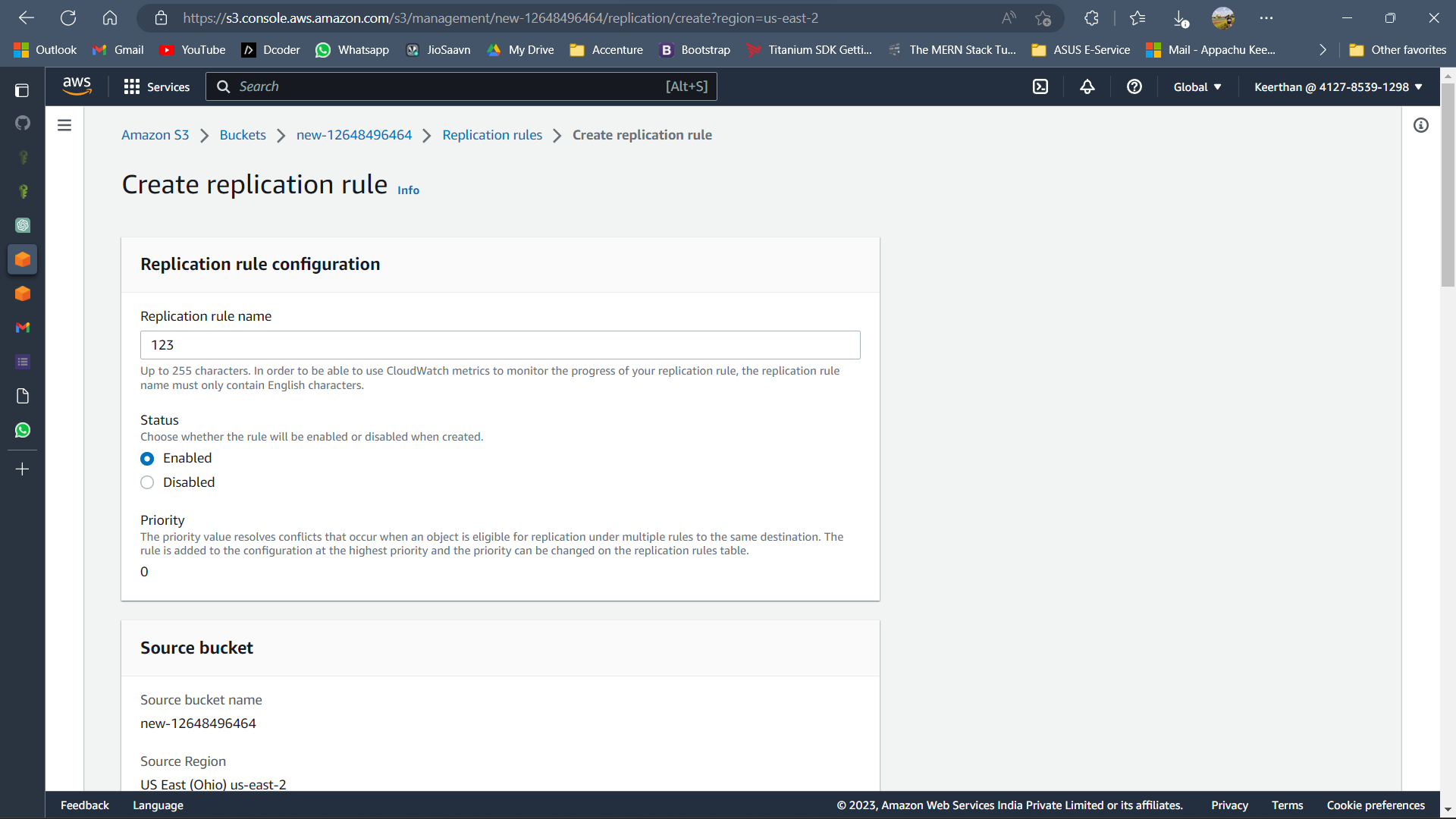


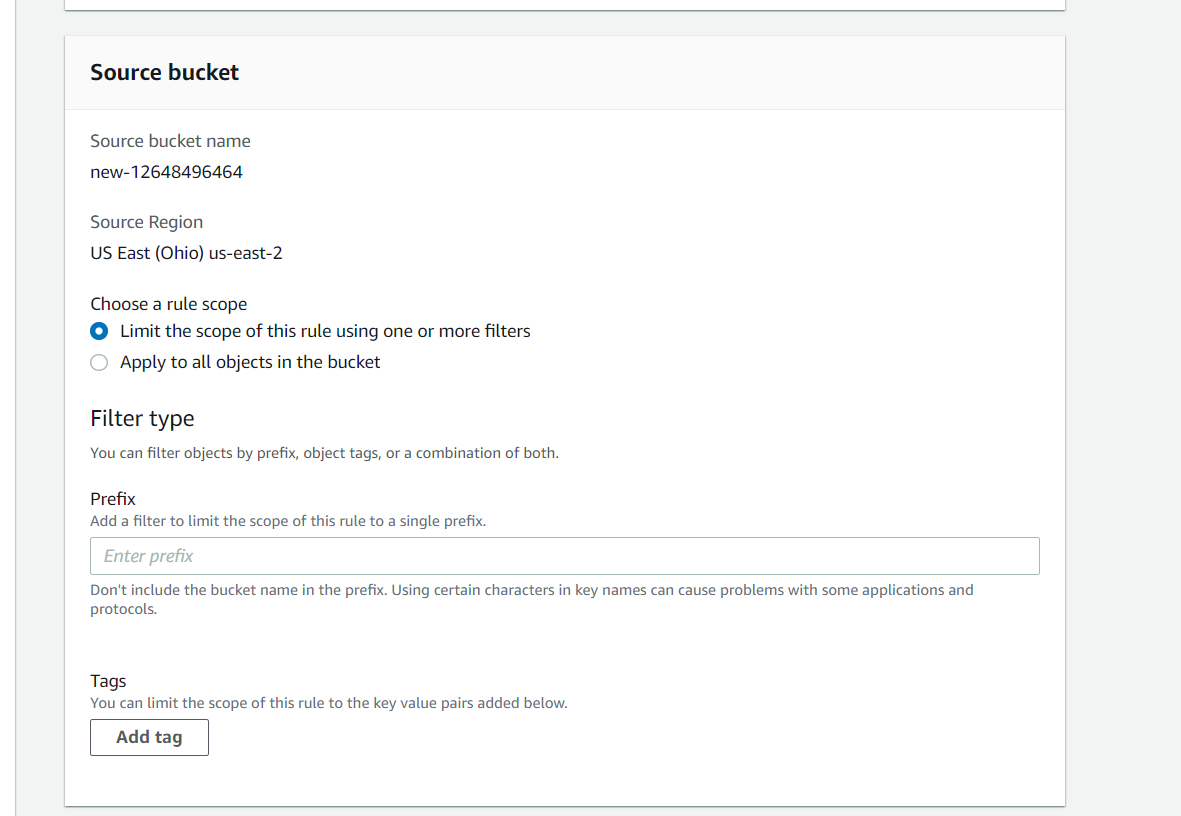


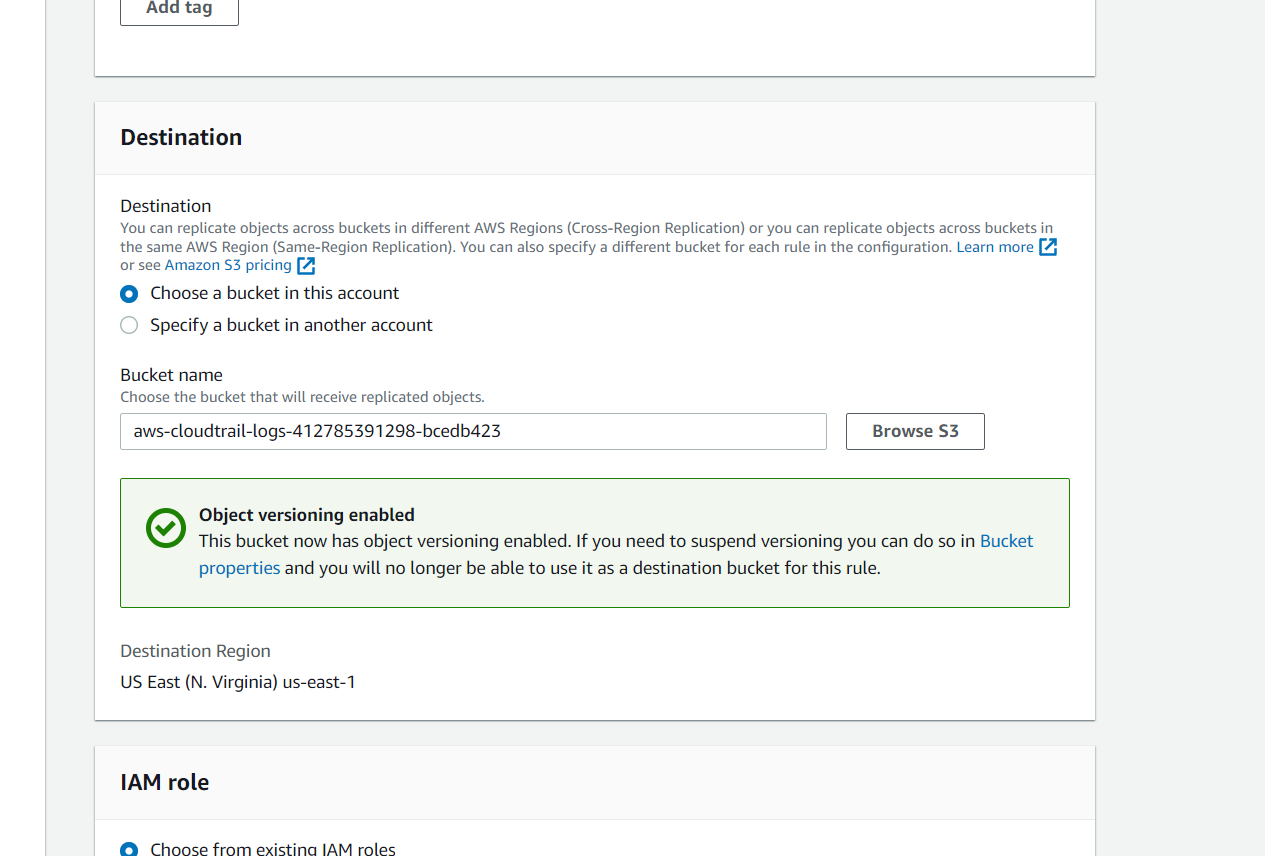
Assignment 9

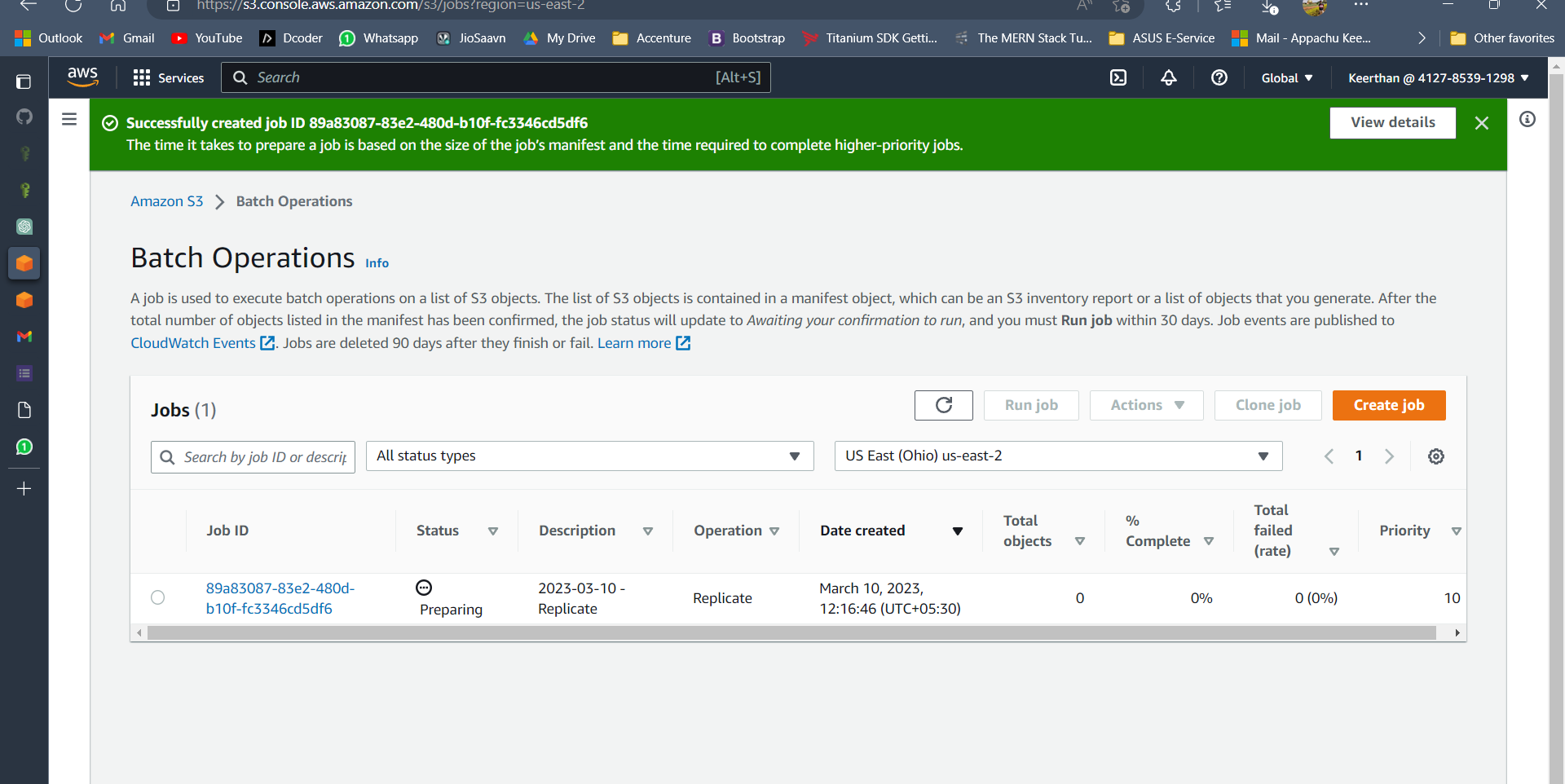
**Set up Cross-Region S3 Bucket Replication, test replication, and observe results**  
  
Upload the final output Screenshot











Assignment 10

**Create playbook "webservers.yaml" and install httpd, start and enable the http service**

**Working with Docker Images**

**Pull the latest "httpd" image**

**Pull the latest "alpine" image**

**Verify images pulled and create 2 containers in each server**

Upload the final output Screenshot  
  
Assignment 11

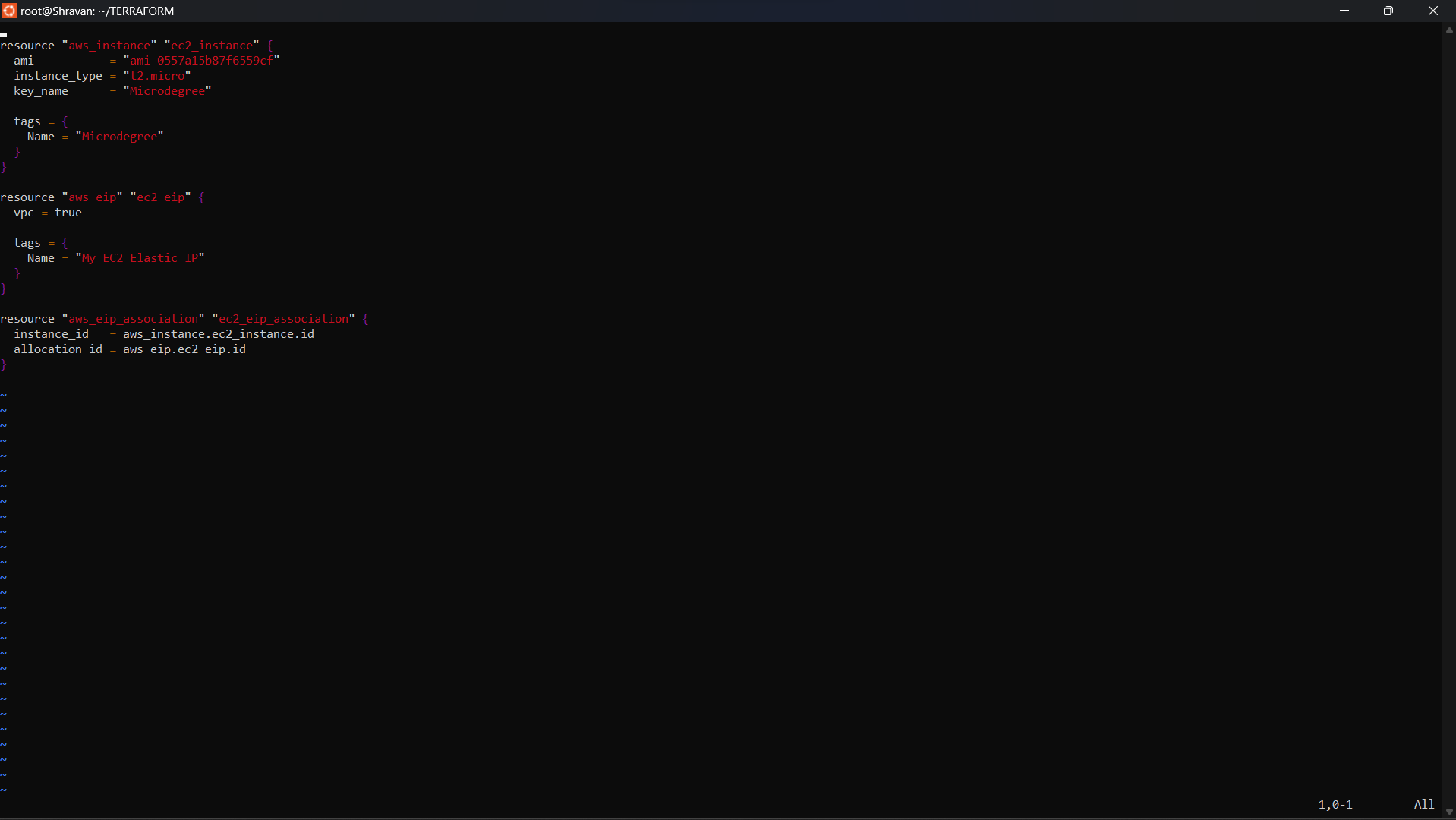
**Deploy Apache Web Server using Terraform IaC**

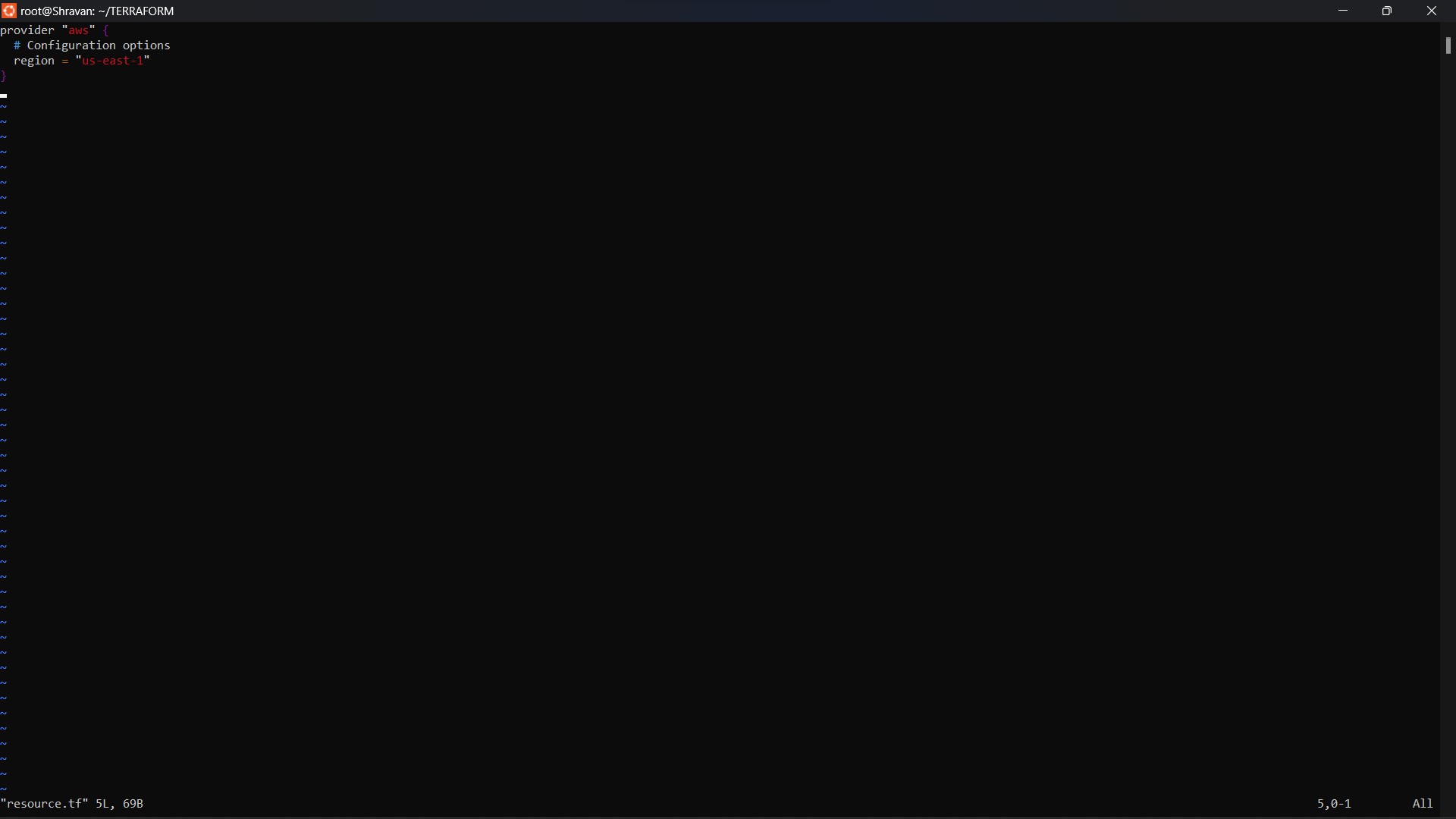
**Associate an Elastic IP to an EC2 instance using Terraform**

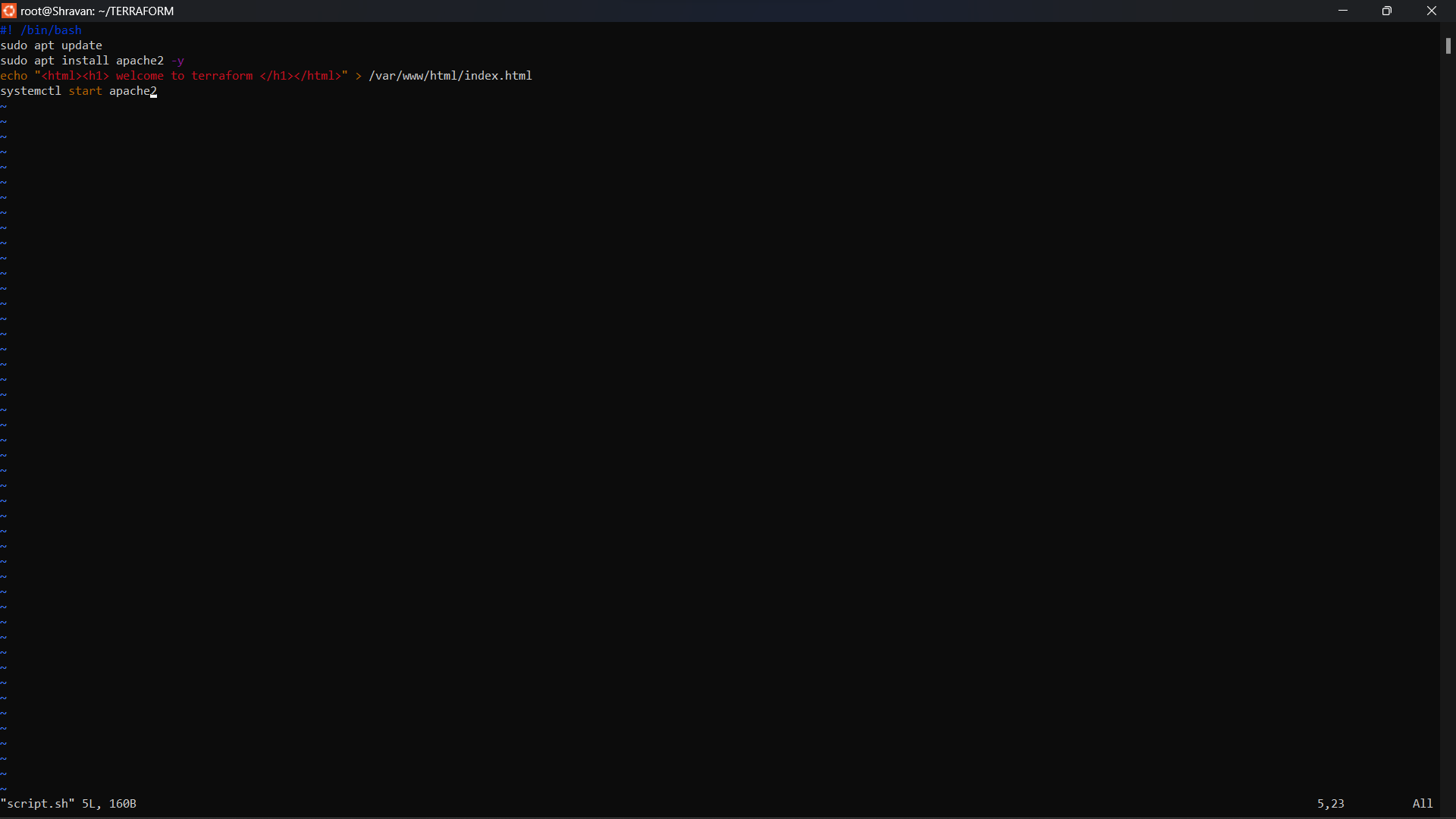
**Create an EC2 instance using Terraform workflow**

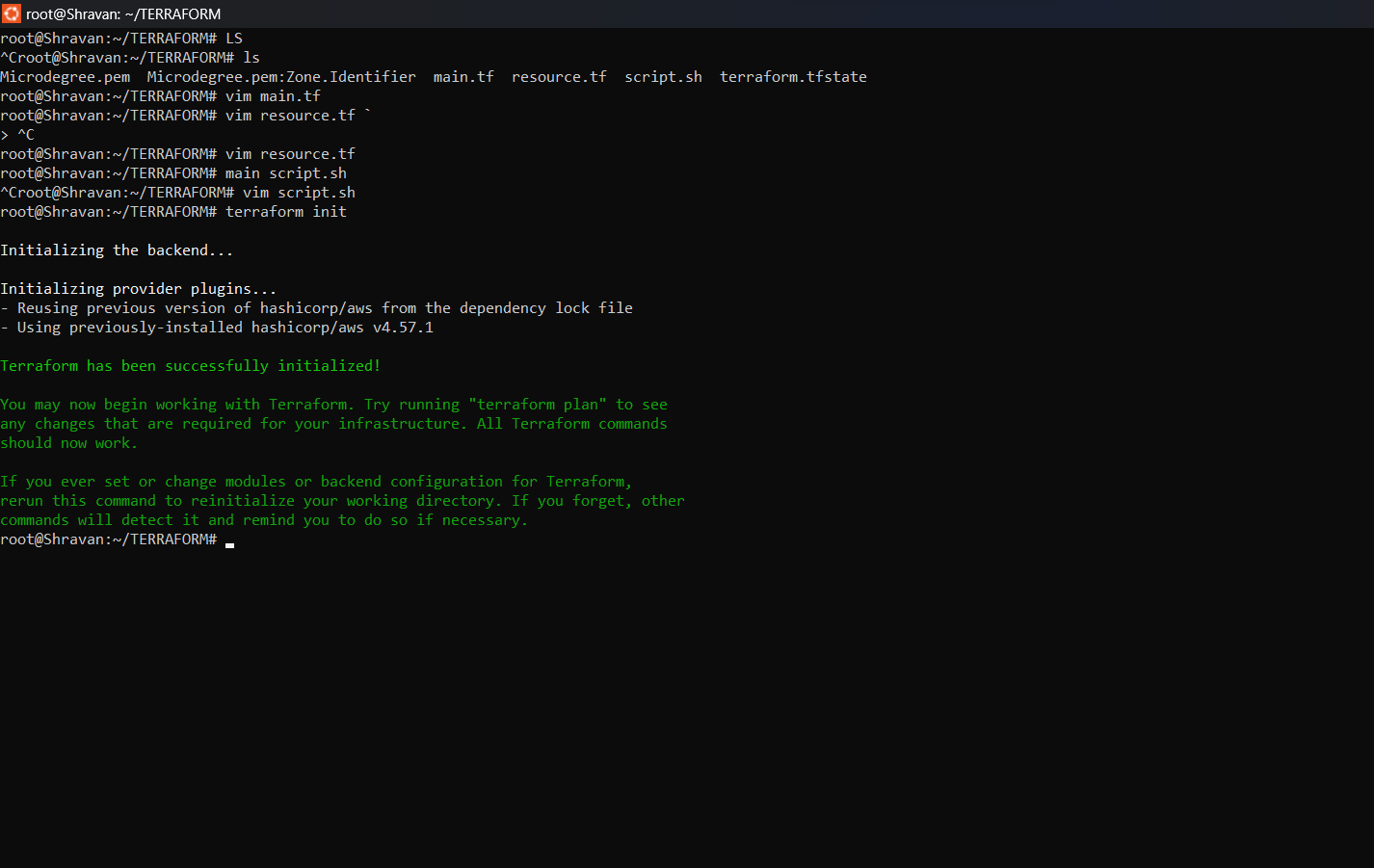
**Associate an Elastic IP.**

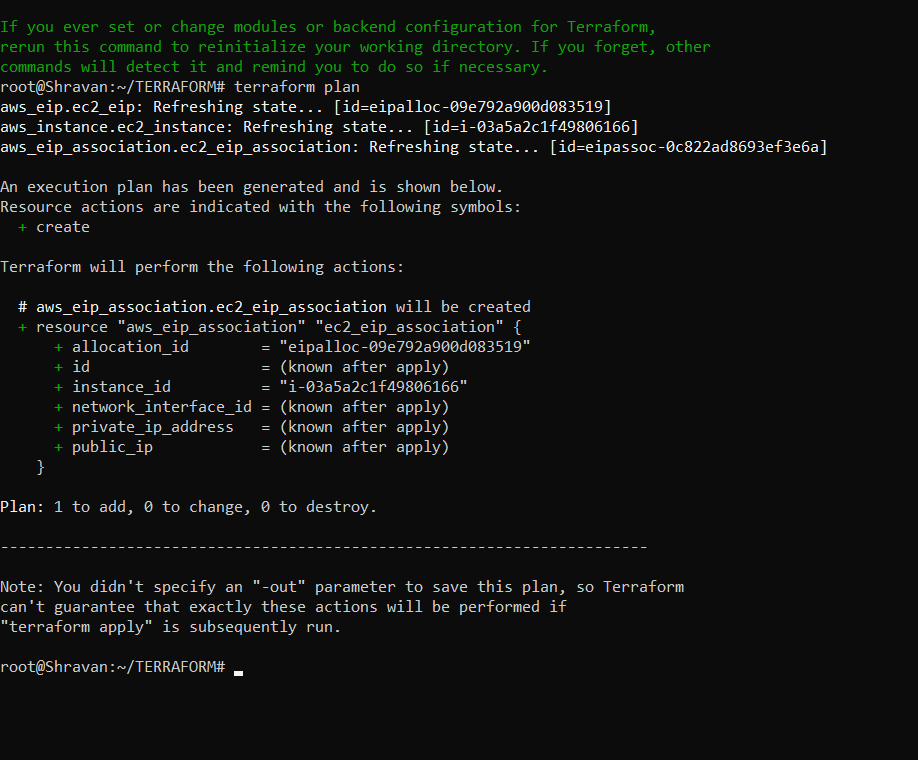
Upload the final output Screenshot

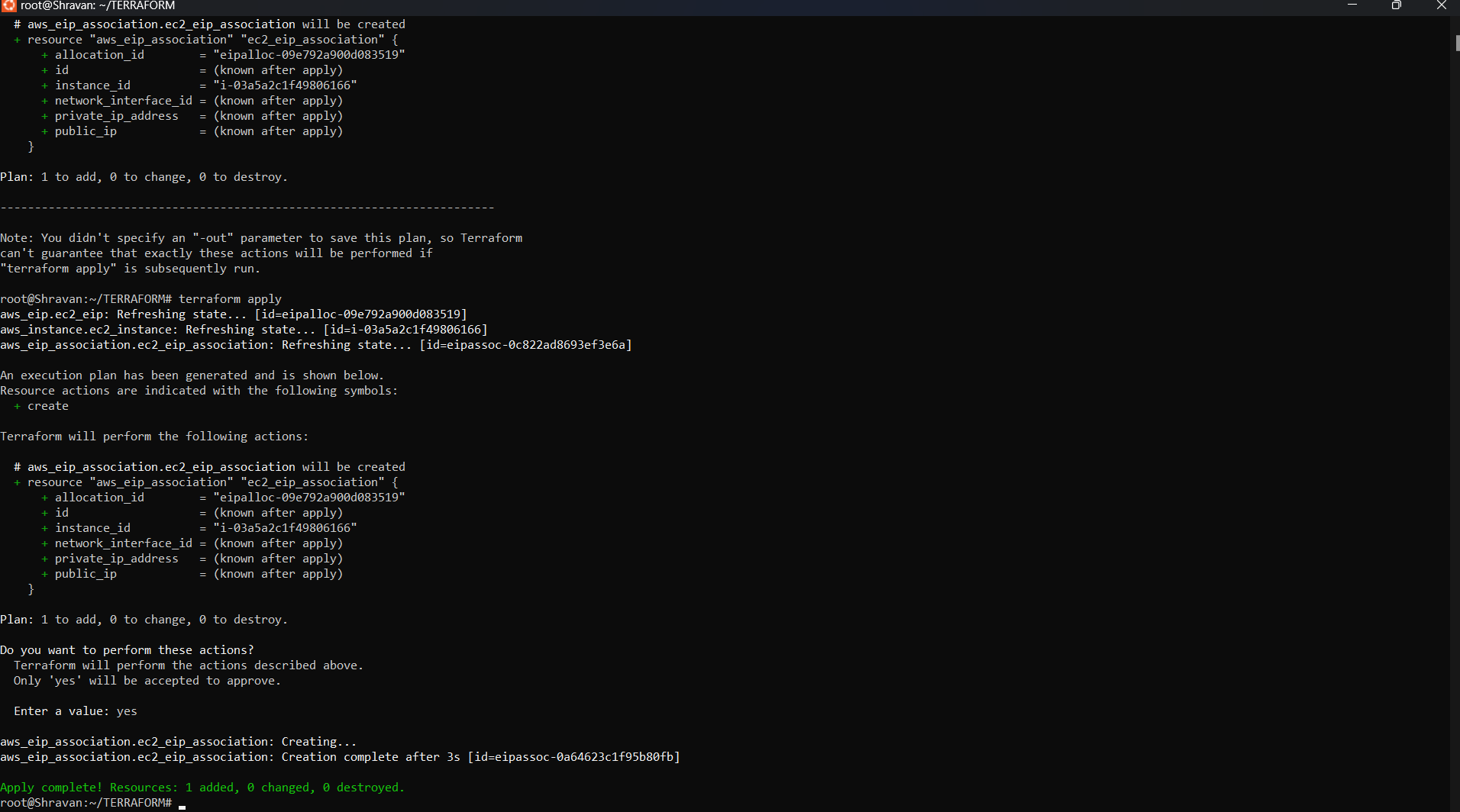


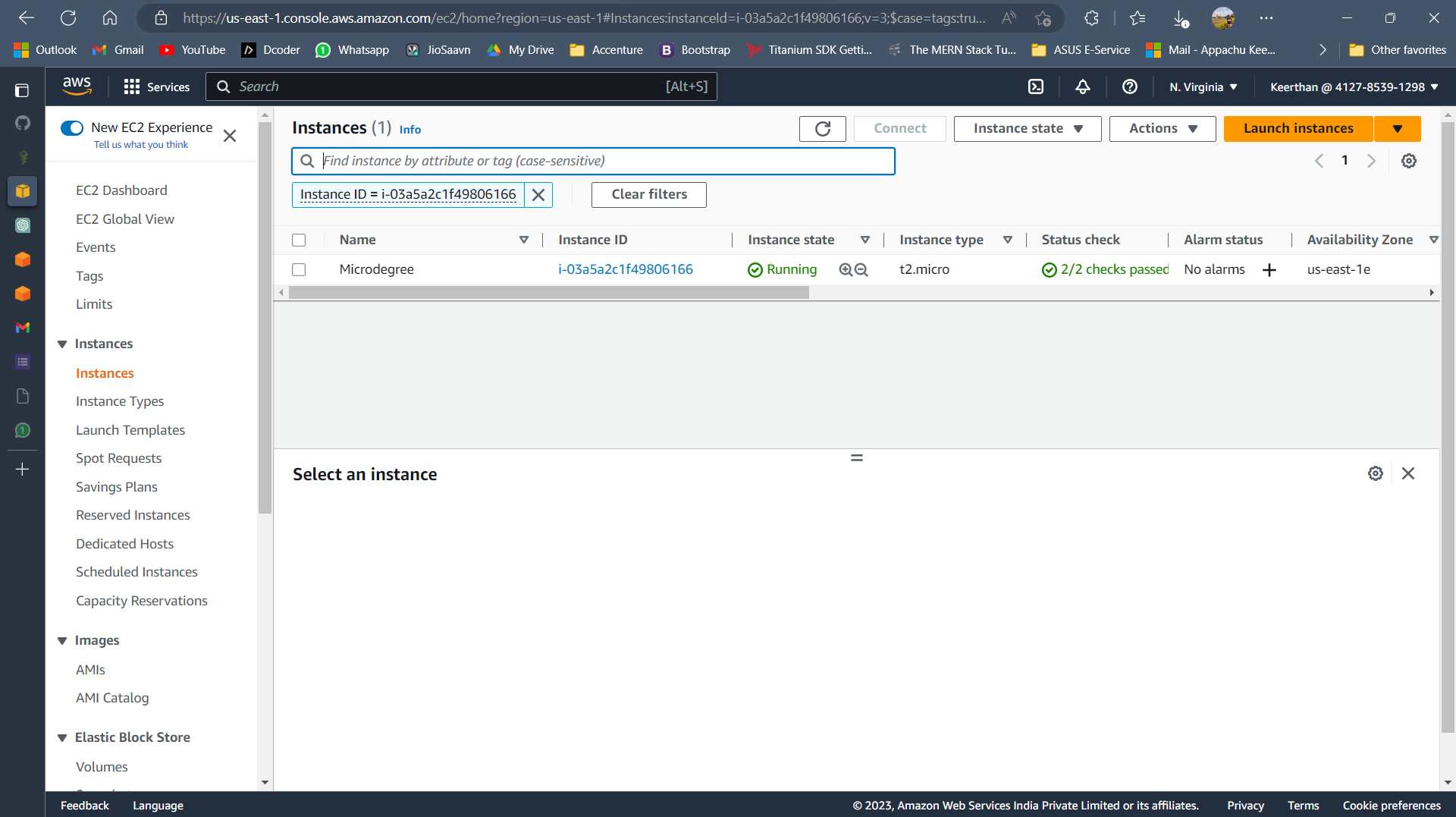


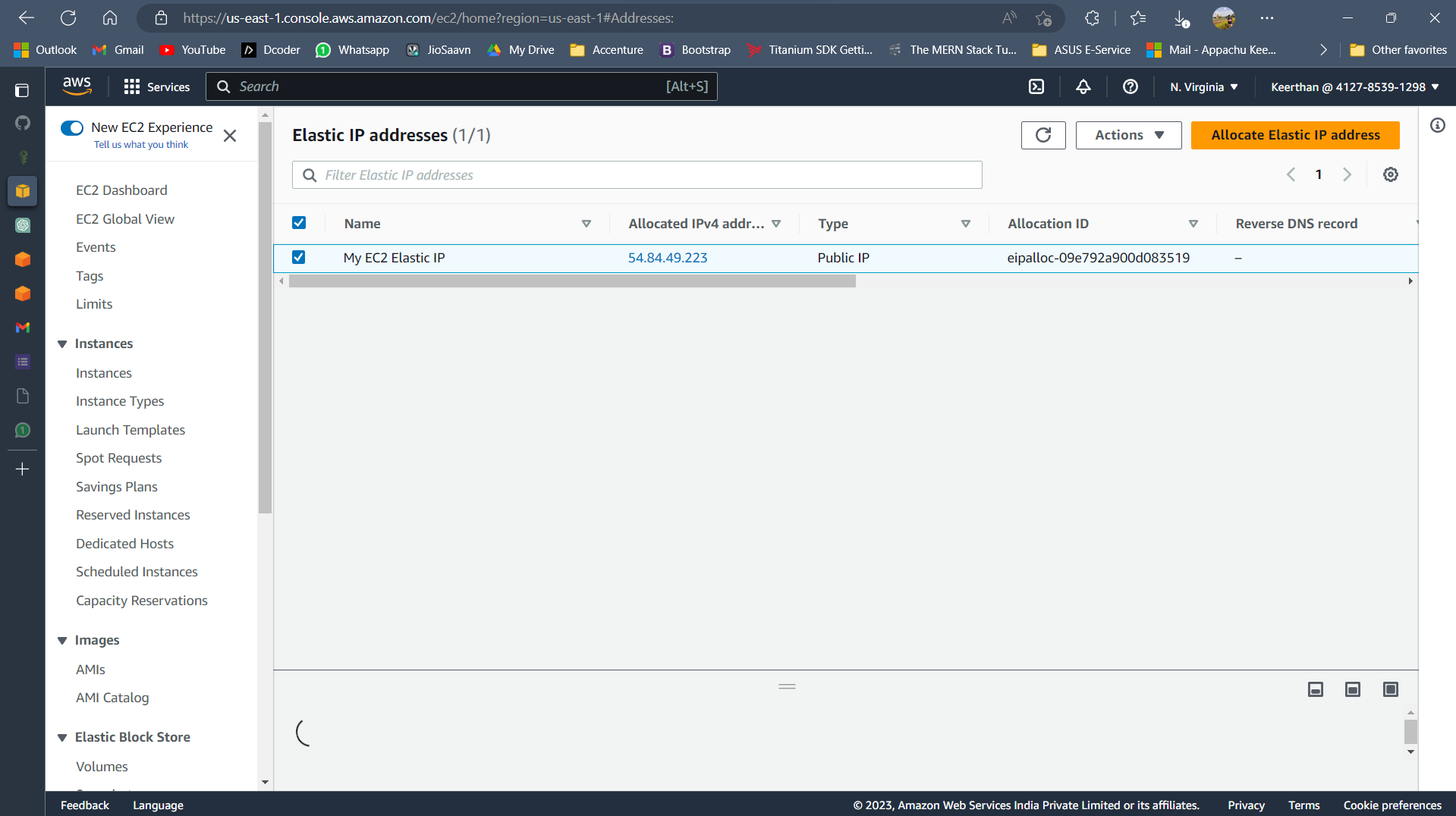






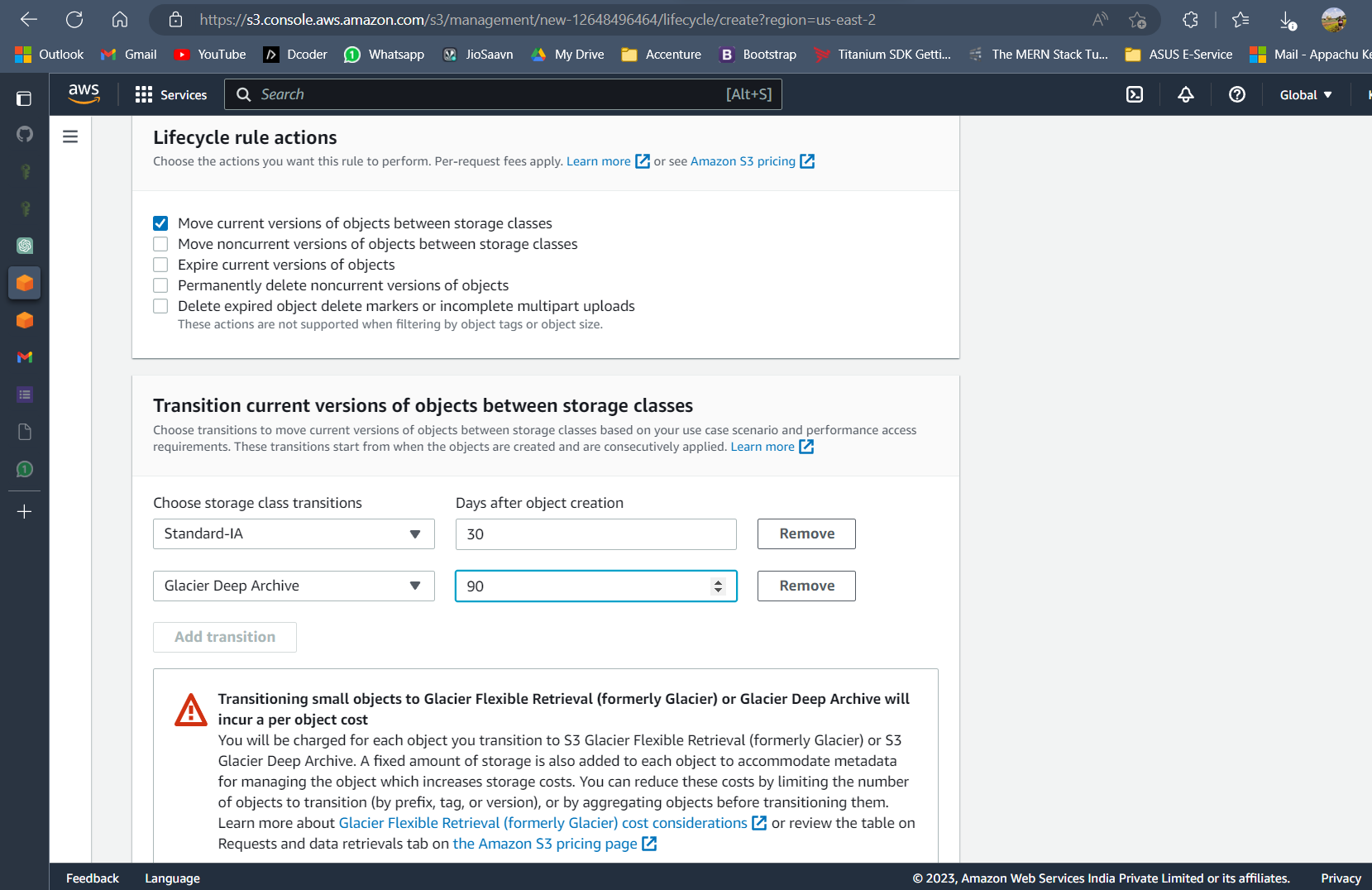


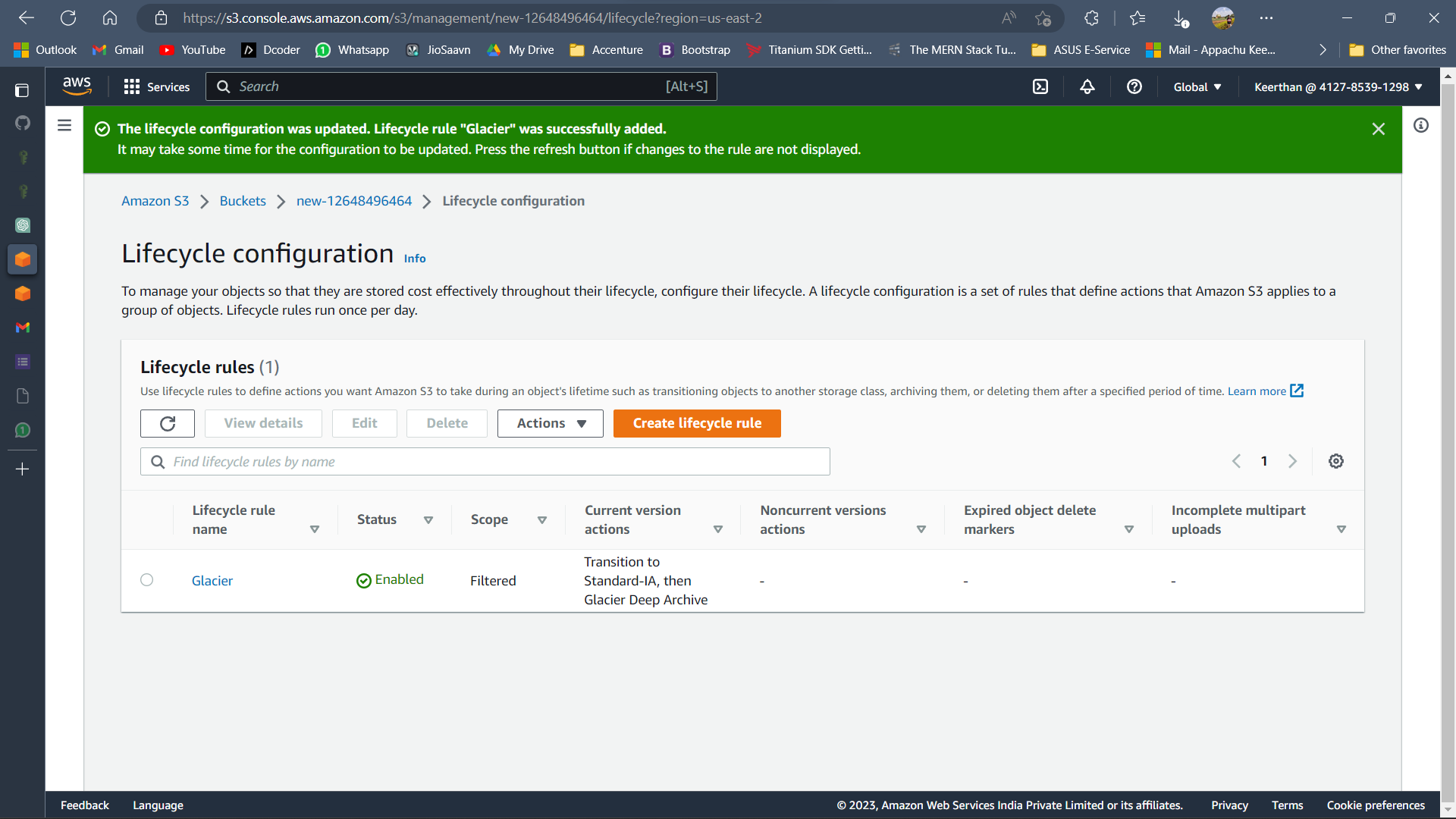




Assignment 12

**Configure lifecycle s3 to glacier**



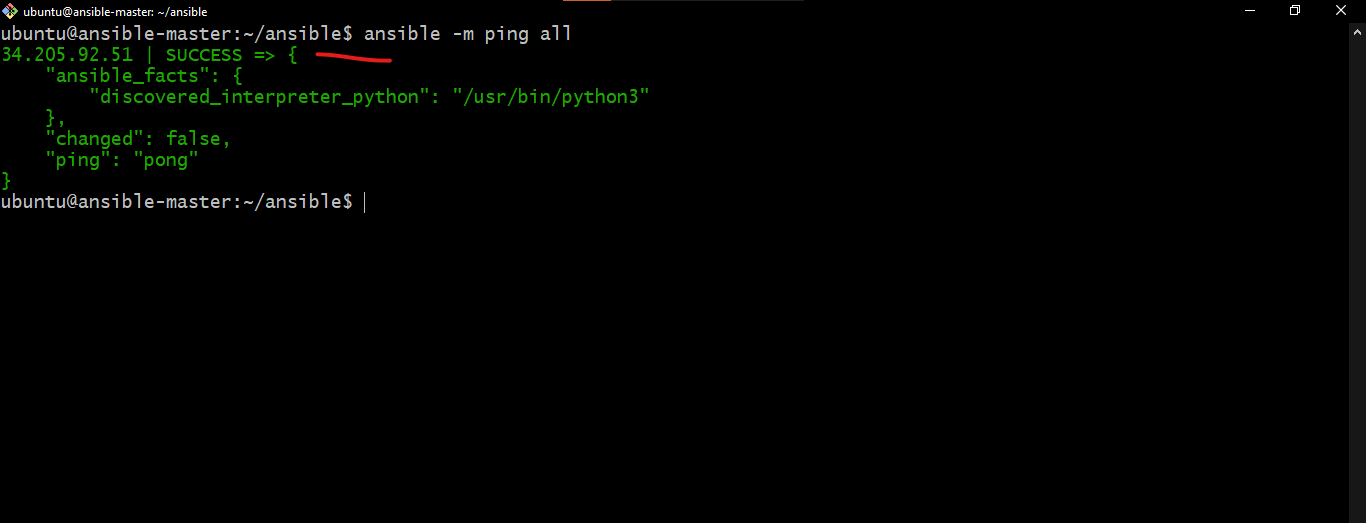


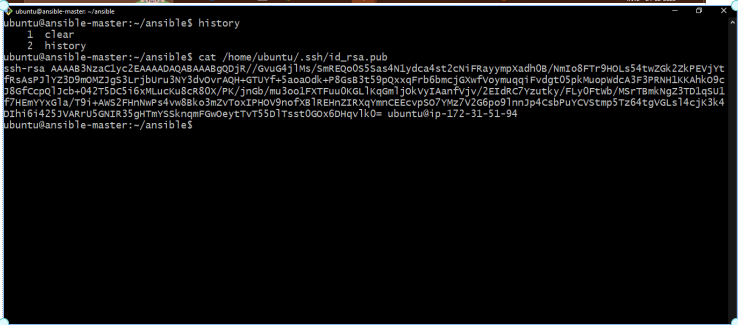
Assignment 13  
  
**Configure web servers using Ansible**

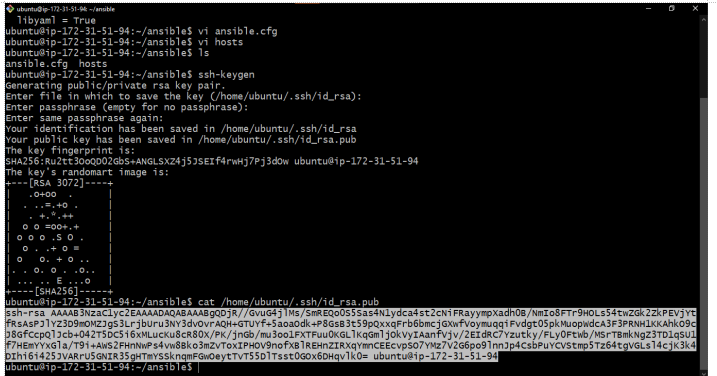
Upload the final output Screenshot

Assignment 14  
  
**Verify connectivity in the environment using "ansible -m ping all"**

Upload the final output Screenshot







Assignment 15  
  
**Create a basic python or PHP web server that says “Hello World”**

**Create a docker container for it**

**Host it on an EC2 or free Azure server.**

**As a part of submission,   
Provide IP address/Hostname of the server, which displays “Hello world”  
Provide docker repo or docker code details.**

Upload the final output Screenshot