Dear Candidate,   
  
Please find the online tasks for the shortlisting process. Read the following instructions carefully and complete the task within a given period of time.

* Complete minimum 10 tasks to get shortlisted for interview round
* This round will test your practical hands on skills w.r.t AWS and DevOps.
* Idea is to test your strengths rather than find your weaknesses. You can try any of the tasks. Need not be in any particular order.
* Once the task is completed kindly upload a final output screenshot after each question.
* Complete the task before 01.00 PM. No task will be accepted once the time is over.
* Only one task can be submitted by one person, so before submitting please review your task.

How to submit the 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

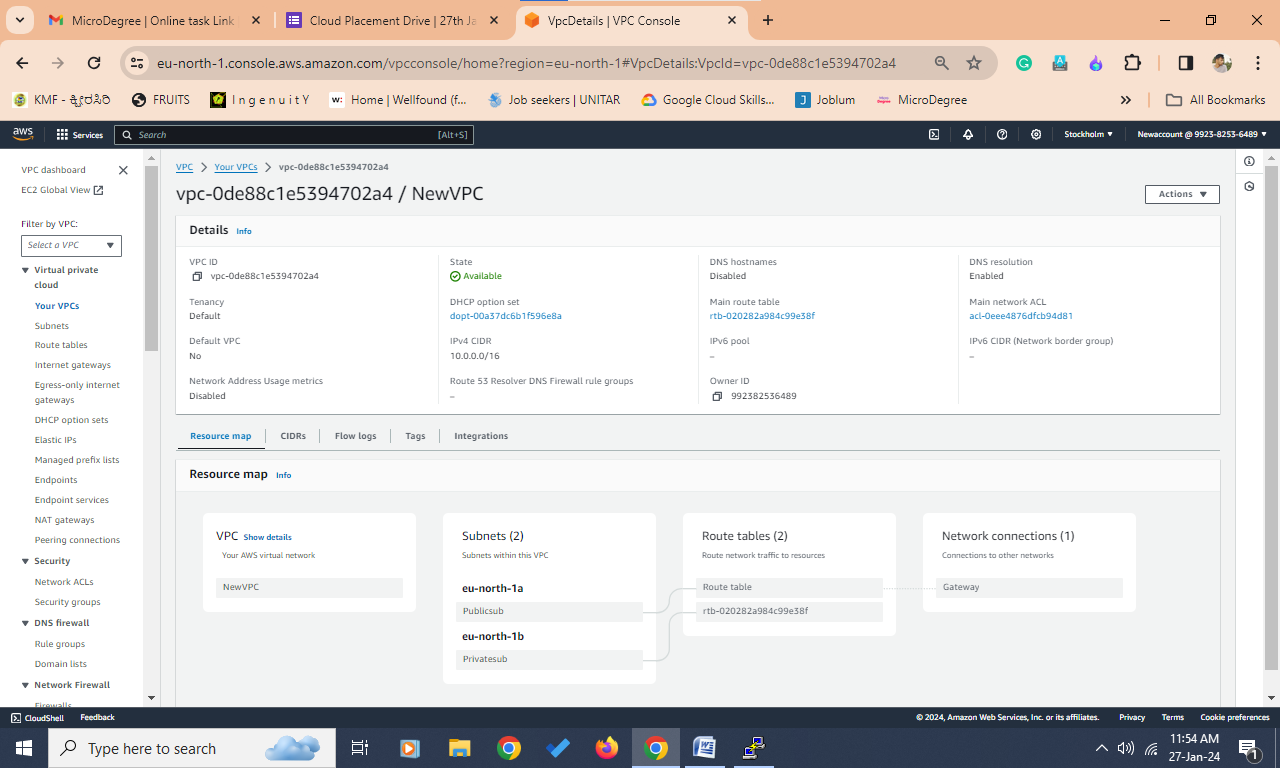
Step 6: Upload the document in Google form

* **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 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)**

****

**Assignment 2**

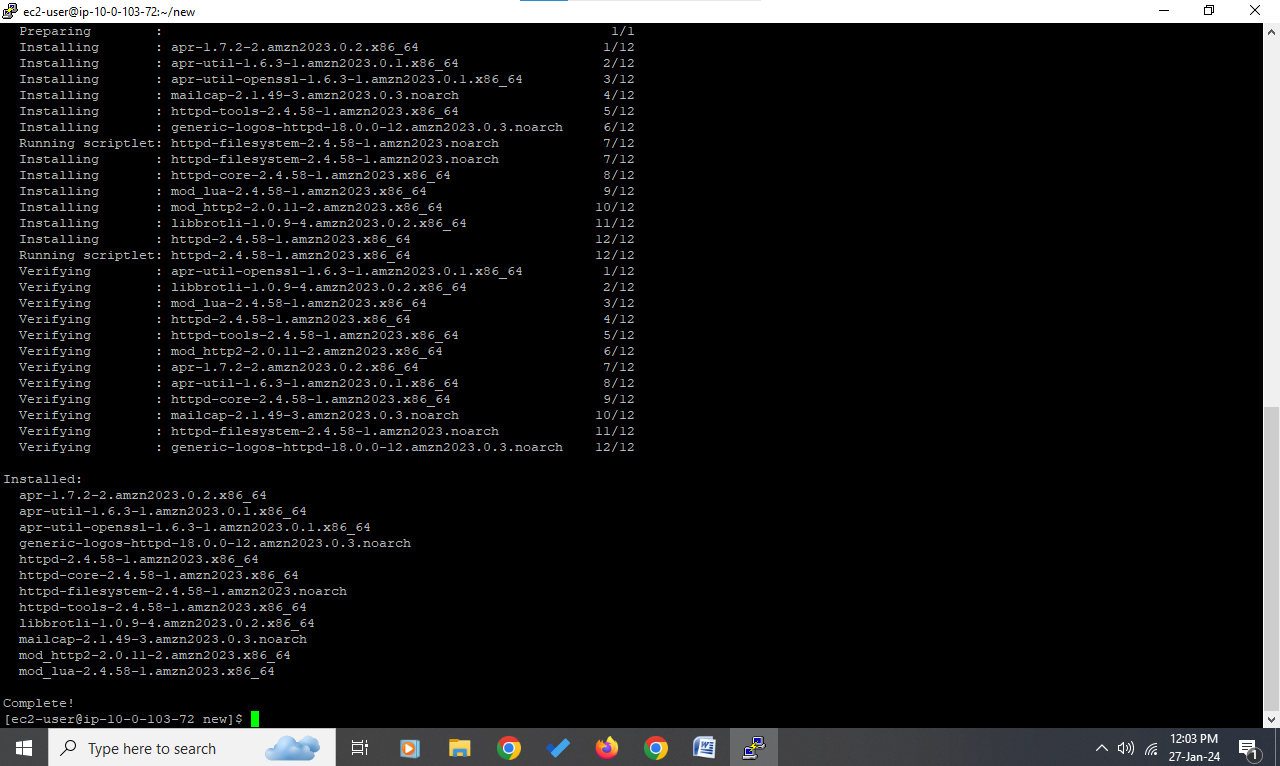
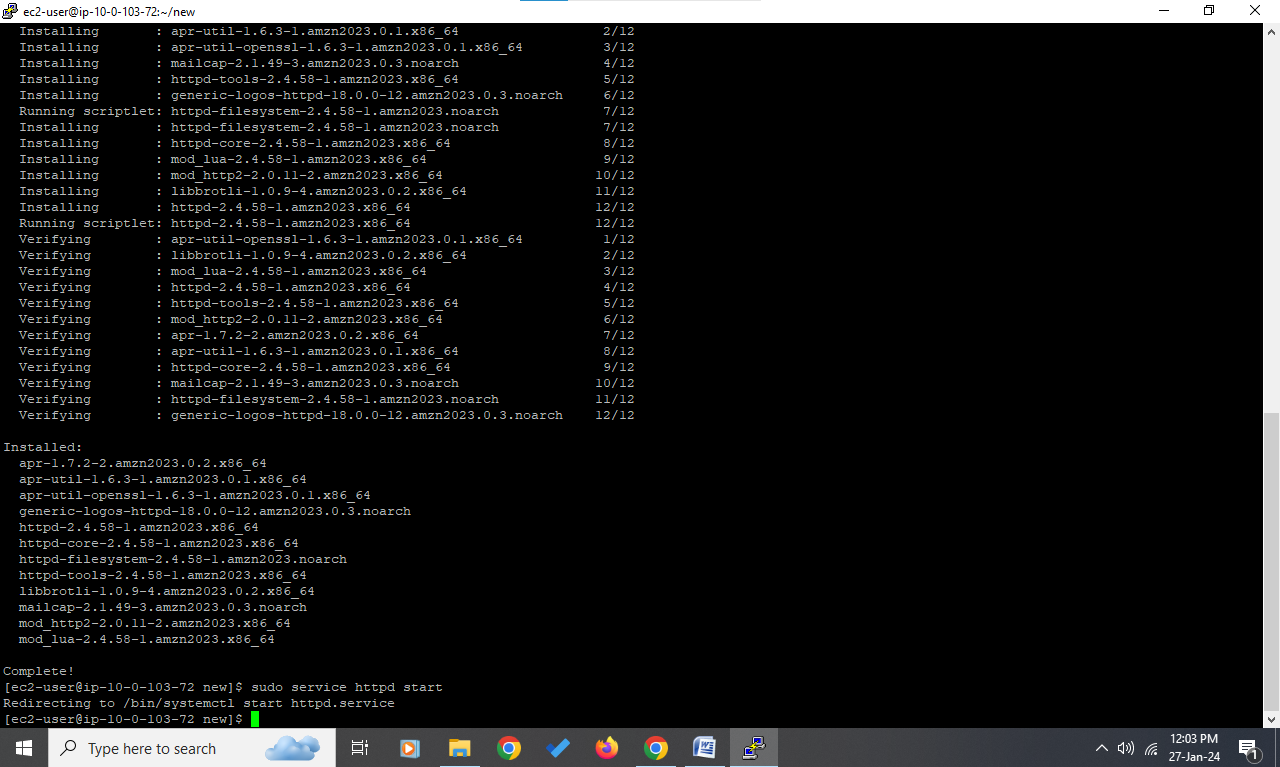
**Associate an Elastic IP to an EC2 Instance using Terraform**

**- Create an ec2 instance using terraform workflow**

**- associate an elastic IP**  
 **Upload the final output Screenshot**

**Assignment 3**

**Configure ec2 Linux machine and install apache configuration**

**  
**

**Upload the final output Screenshot**

**Assignment 4**

**Creating a Basic Amazon S3 Lifecycle Policy**

**- Create an S3 Bucket and Upload an Object**

**- Create a Lifecycle Policy - Create a lifecycle policy that moves objects to Glacier Flexible Retrieval (formerly Amazon Glacier) if they haven't been accessed in the last 30 days.**

**Upload the final output Screenshot**

**Assignment 5**

**Set up Cross-Region S3 Bucket Replication**

**- Create an S3 Bucket and Enable Replication**

**- Test Replication and Observe Results**

**Upload the final output Screenshot**

**Assignment 6**

**Creating a Custom Amazon Machine Image (AMI)**

**- Launch a New EC2 Instance**

**- Install http on the new instance, enable the http service to start at boot.**

**- Create a New AMI from customized instance and name the AMI MicroDegreeWeb**

**- Launch a New Instance Using the Custom AM**

**- Verify that http is running.**

**Upload the final output Screenshot**

**Assignment 7**

**Configure a Load Balancer: Set up an Elastic Load Balancer (ELB) that distributes incoming traffic to the EC2 instances in your Auto Scaling group. Configure health checks to ensure that the ELB only forwards traffic to healthy instances.**

**Upload the final output Screenshot**

**Assignment 8**

**Create a Staging branch in GitHub, push code from local repository to remote, and share commands**

**Upload the final output Screenshot**

**Assignment 9**

**Web Application:**

* **Develop a simple "Hello World" web application (you can use any programming language or framework of your choice).**
* **Package the application into a deployable artifact (e.g., a JAR file, Docker image).**

**Upload the final output Screenshot**

**Assignment 10**

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

**Upload the final output Screenshot**

**Assignment 11**

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

**Upload the final output Screenshot**

**Assignment 12**  
  
**Build a custom docker image using Ubuntu as a base docker image and run the nginx application.   
This docker image should be built using a Dockerfile. Once the docker image is build, start the docker image using the host network and make it accessible on Public IP**

**Upload the final output Screenshot**

**Assignment 13**

**Configure email notifications**

**Description: Configure Jenkins to send email notifications when a build fails.**

**Instructions:**

**Upload the final output Screenshot**

**Assignment 14**

**Setting up a Continuous Integration (CI) Pipeline: Configure a CI pipeline using tools like Jenkins, CircleCI, or AWS CodePipeline to automate the building, testing, and deployment of your application code whenever changes are pushed to the repository.**

**Upload the final output Screenshot**

**Assignment 15**

Scenario: Infrastructure as Code

Your team is managing infrastructure using Infrastructure as Code (IaC) principles. How would you use AWS CloudFormation to provision and manage resources? Provide an example of a CloudFormation template that creates an Amazon S3 bucket and an Amazon DynamoDB table.

Write answer with max 300 words