Task 1: Enable Required GCP APIs

Step 1: Install Ansible on the VM

```
sudo apt update
sudo apt install -y ansible
ansible --version
gcloud services enable compute.googleapis.com
gcloud services enable iam.googleapis.com
gcloud services enable cloudresourcemanager.googleapis.com
```

Task 2: Authenticate Terraform with GCP

Step 1: Authenticate with GCP

```
gcloud auth login
gcloud config set project ""
gcloud auth application-default login
ssh-keygen -t rsa -f ~/.ssh/id_rsa -C terraform -N ""
```

Task 3: Provision a GCE VM using Terraform

Step 1: Navigate to Terraform Directory

```
cd ~/Desktop/Project/terraform
project_id = "your-project-id"
resource "google_compute_instance" "apache_vm" {
 name        = "apache-vm"
 machine_type = "e2-medium"
 zone        = var.zone
 boot_disk {
  initialize_params {
   image = "ubuntu-os-cloud/ubuntu-2204-lts"
  }
```

```
  }
  network_interface {
    network = "default"
    access_config {}
  }
  tags = ["http-server"]
  metadata = {
    ssh-keys = "${var.ssh_user}:${file(var.public_key_path)}"
  }
}
resource "google_compute_firewall" "allow_http" {
  name    = "allow-http"
  network = "default"
  allow {
    protocol = "tcp"
    ports    = ["80"]
  }
  source_ranges = ["0.0.0.0/0"]
  target_tags   = ["http-server"]
}

terraform init
terraform apply
gcloud compute instances list
```

Task 4: Configure Web Server using Ansible

Step 1: Navigate to Ansible Directory

```
cd ~/Desktop/Project/ansible
 ansible_user= ansible_ssh_private_key_file=~/.ssh/id_rsa
- hosts: web
  become: yes
  tasks:
    - name: Update APT cache
      apt:
        update_cache: yes
    - name: Install Apache
      apt:
        name: apache2
        state: present
    - name: Copy custom index.html
      copy:
        src: index.html
        dest: /var/www/html/index.html
        owner: www-data
        group: www-data
        mode: '0644'
    - name: Ensure Apache is running
      service:
        name: apache2
        state: started
        enabled: yes
```

ansible-playbook -i inventory.ini site.yml

Task 5: Verify Apache Landing Page

Step 1: Open Browser

Visit:

http://<EXTERNAL_VM_IP>

Step 2: Confirm Output

You should see:

Welcome your-name

1. Checking the GCE Virtual Machine Name *(15 marks)*

gcloud compute instances list --filter="name=apache-vm"

2. Checking the GCE Instance Type *(5 marks)*

gcloud compute instances describe apache-vm --zone= --format="value(machineType)"

3. Checking the OS Image Used *(5 marks)*

gcloud compute instances describe apache-vm --zone= --
format="value(disks[0].licenses)"

4. Checking if the Tag is Attached to the VM *(5 marks)*

gcloud compute instances describe apache-vm --zone= --format="value(tags.items)"

5. Checking if SSH Key Metadata is Set for VM *(10 marks)*

gcloud compute instances describe apache-vm --zone= --format="value(metadata.items)"

6. Checking the Firewall Rule Name *(15 marks)*

gcloud compute firewall-rules list --filter="name=allow-http"

7. Checking the Source Range Allowed for Firewall *(5 marks)*

gcloud compute firewall-rules describe allow-http --format="value(sourceRanges)"

8. Checking if the VM is Reachable via Public IP *(20 marks)*

gcloud compute instances describe apache-vm --zone= --
format="value(networkInterfaces[0].accessConfigs[0].natIP)"

ping <EXTERNAL_VM_IP>

9. Checking if the Landing Page is Up and Running *(20 marks)*

curl http://<EXTERNAL_VM_IP>