

# FA24: CMPE-272 Sec 48 - Enterprise SW Plat

## HW #1 - Ansible

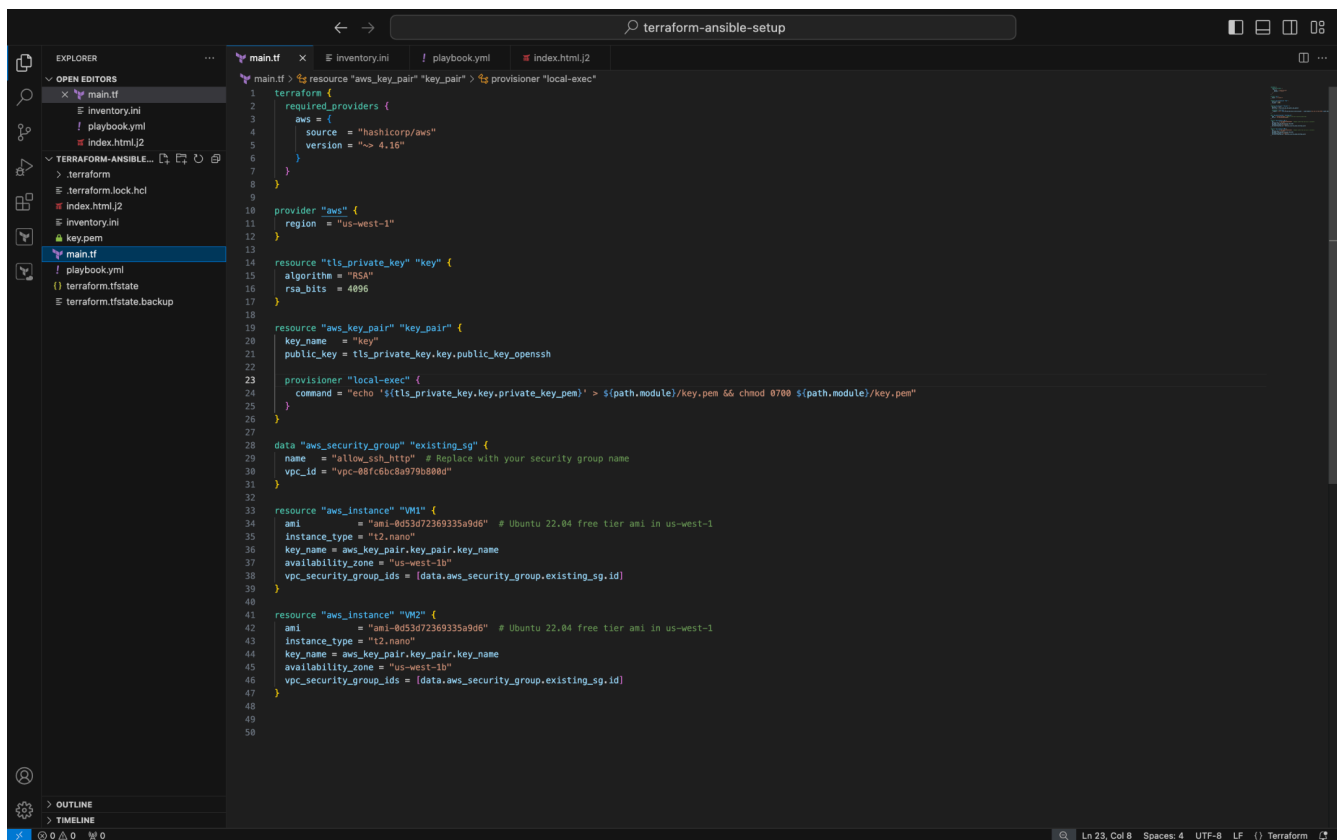
Student Name: Ananya Praveen Shetty

Student ID: 017553276

Submission Date: 09/08/2024

Professor Name: Andrew H. Bond

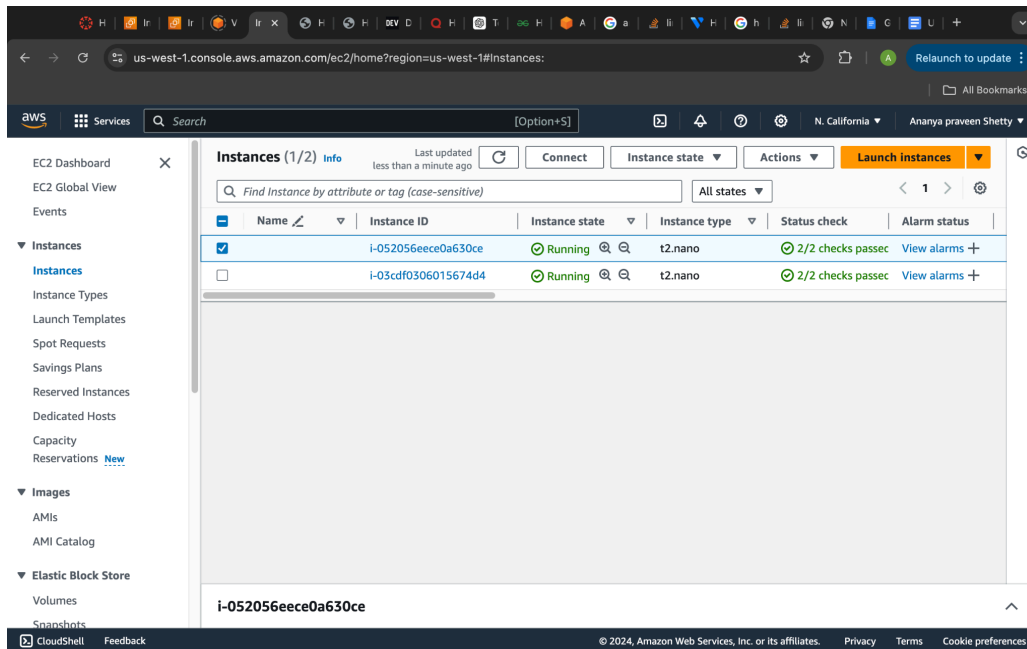
### 1. Configure two VMs, **VM1** and **VM2**



```
1 terraform {
2   required_providers {
3     aws = {
4       source = "hashicorp/aws"
5       version = "~> 4.10"
6     }
7   }
8 }
9
10 provider "aws" {
11   region = "us-west-1"
12 }
13
14 resource "tls_private_key" "key" {
15   algorithm = "RSA"
16   rsa_bits = 4096
17 }
18
19 resource "aws_key_pair" "key_pair" {
20   key_name = "key"
21   public_key = tls_private_key.key.public_key_openssh
22 }
23
24 provisioner "local-exec" {
25   command = "echo '${tls_private_key.key.private_key_pem}' > $(path.module)/key.pem 66 chmod 0700 $(path.module)/key.pem"
26 }
27
28 data "aws_security_group" "existing_sg" {
29   name = "allow_ssh_http" # Replace with your security group name
30   vpc_id = "vpc-08f6b68a979b880d"
31 }
32
33 resource "aws_instance" "VM1" {
34   ami = "ami-0d53d72369335a9d6" # Ubuntu 22.04 free tier ami in us-west-1
35   instance_type = "t2.nano"
36   key_name = aws_key_pair.key_pair.key_name
37   availability_zone = "us-west-1b"
38   vpc_security_group_ids = [data.aws_security_group.existing_sg.id]
39 }
40
41 resource "aws_instance" "VM2" {
42   ami = "ami-0d53d72369335a9d6" # Ubuntu 22.04 free tier ami in us-west-1
43   instance_type = "t2.nano"
44   key_name = aws_key_pair.key_pair.key_name
45   availability_zone = "us-west-1b"
46   vpc_security_group_ids = [data.aws_security_group.existing_sg.id]
47 }
48
49
50
```

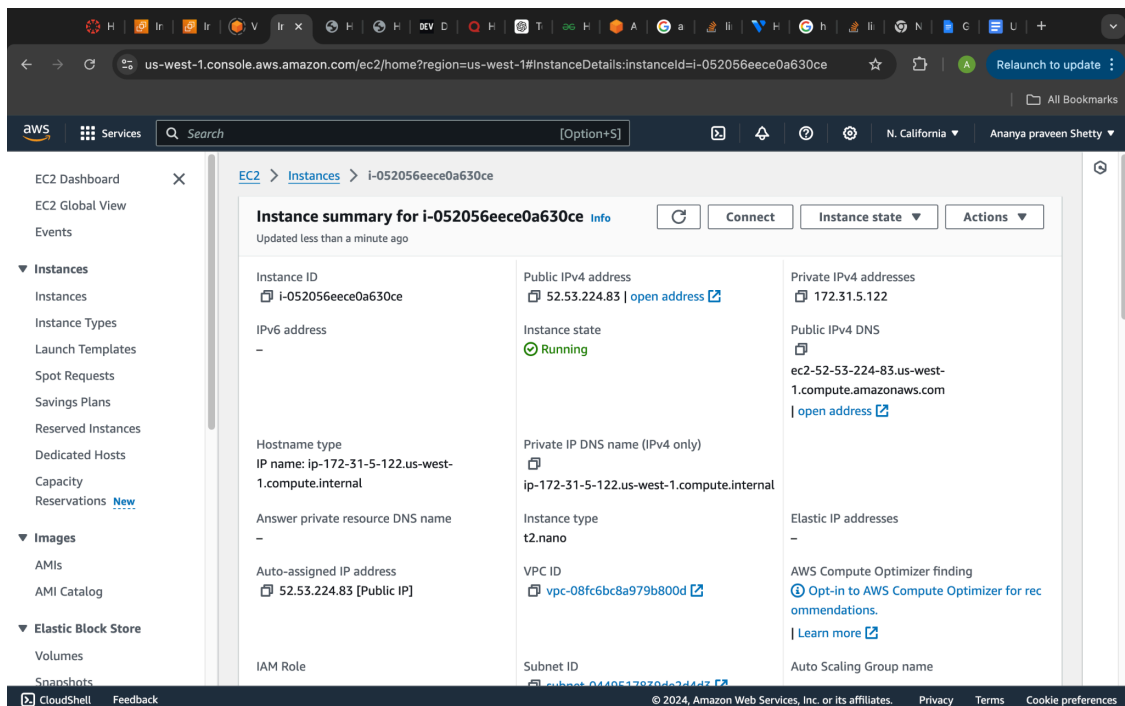
In this terraform file main.tf i have configured two VMs namely VM1 and VM2. These instances will be shown up in the aws console.

## 2.AWS console showing two running instances.

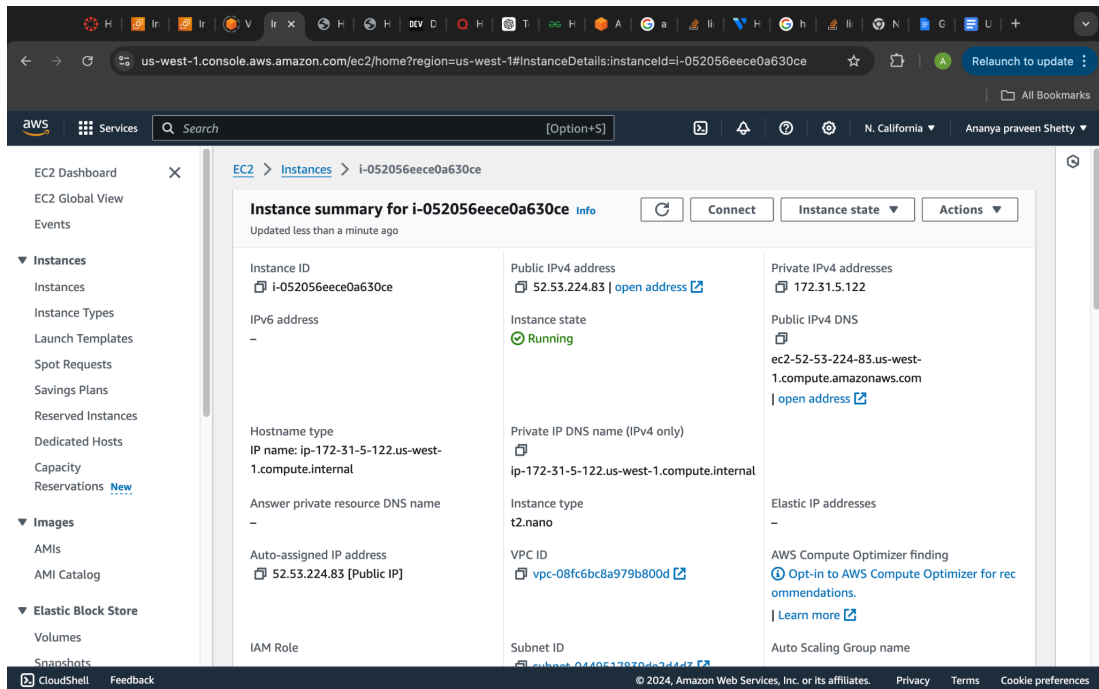


3.We need to note down the Public ip address of both VMs as we need it to configure web servers on them.

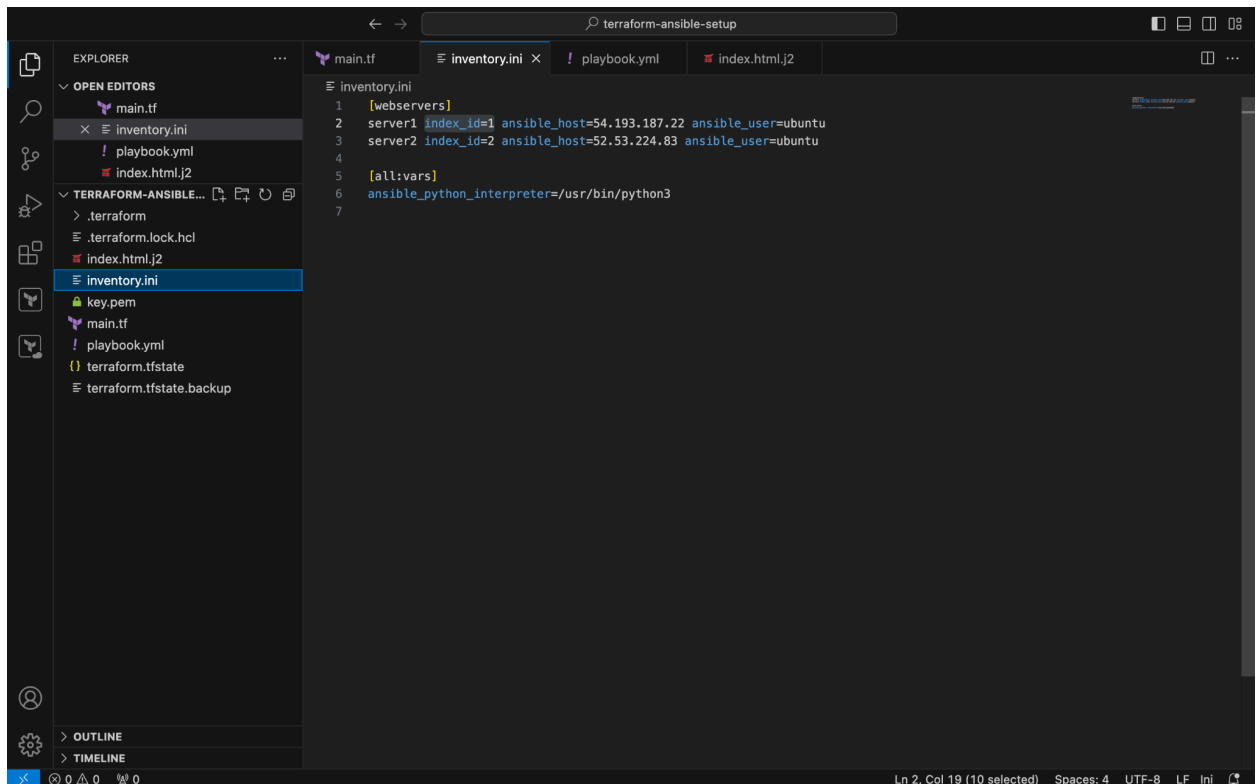
Below is VM1



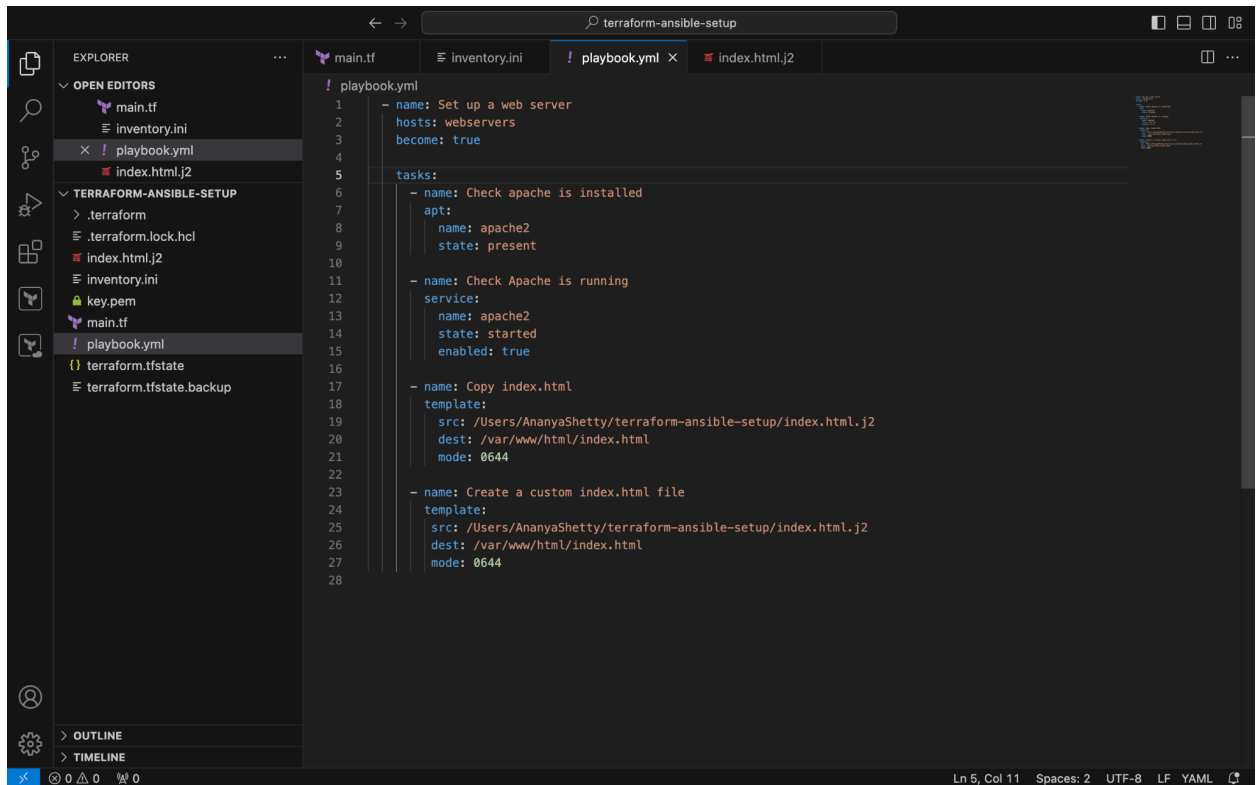
VM2:



4. We need to install Ansible in the local machine. We need to create an `inventory.ini` file where we need to list both the servers with `server_id`, `ansible_host` and `ansible_user`.



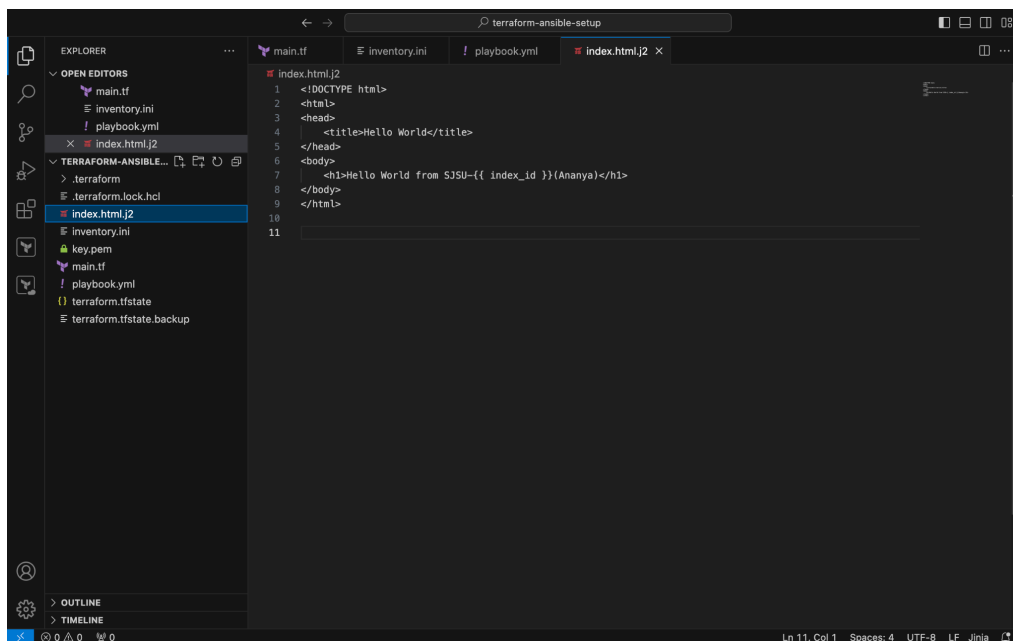
5.Next we need to create a playbook.yml file where we need to add all the tasks that the ansible should do.For example, start apache server, check status,create an index.html file in a given directory.



The screenshot shows the VS Code editor with the 'playbook.yml' file open. The file contains the following Ansible playbook:

```
1  - name: Set up a web server
2    hosts: webserver
3    become: true
4
5  tasks:
6
7    - name: Check apache is installed
8      apt:
9        name: apache2
10       state: present
11
12   - name: Check Apache is running
13     service:
14       name: apache2
15       state: started
16       enabled: true
17
18   - name: Copy index.html
19     template:
20       src: /Users/AnanyaShetty/terraform-ansible-setup/index.html.j2
21       dest: /var/www/html/index.html
22       mode: 0644
23
24   - name: Create a custom index.html file
25     template:
26       src: /Users/AnanyaShetty/terraform-ansible-setup/index.html.j2
27       dest: /var/www/html/index.html
28       mode: 0644
```

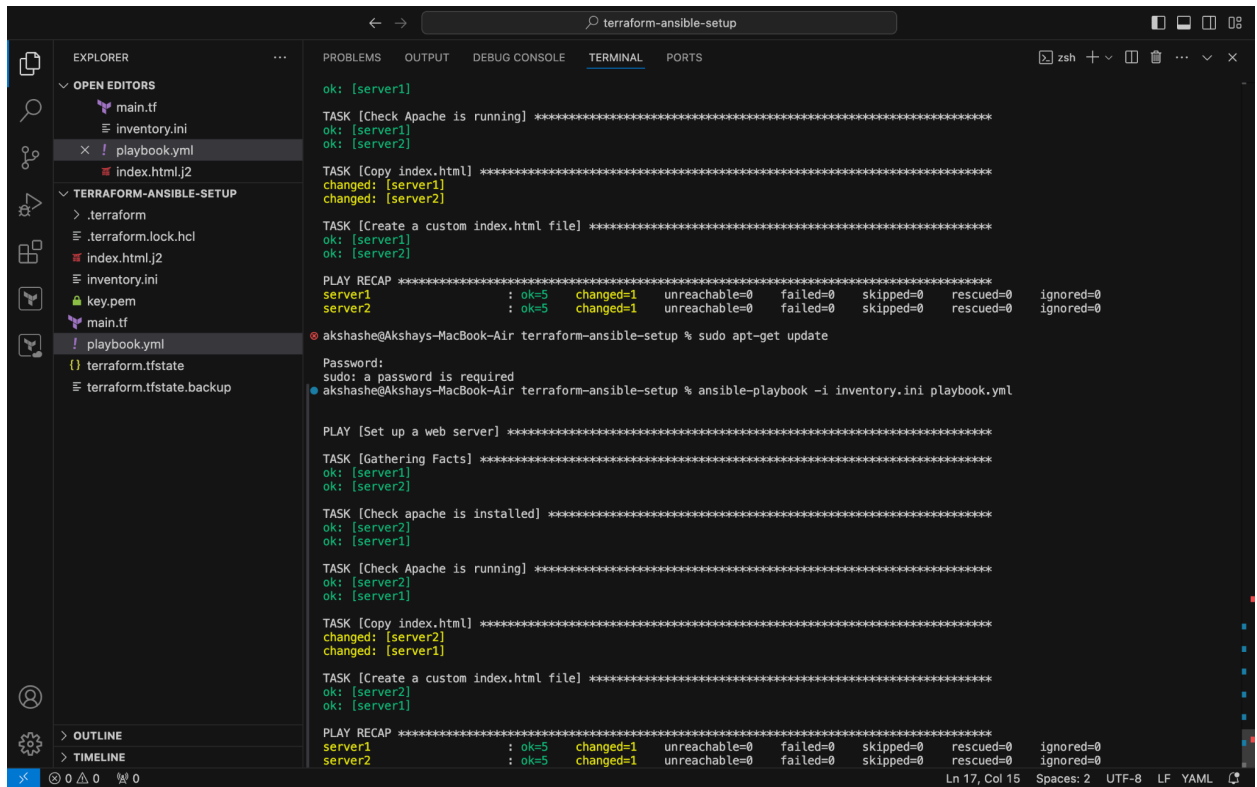
6.Create index.html file and add the contents.



The screenshot shows the VS Code editor with the 'index.html.j2' file open. The file contains the following Jinja2 template:

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4    <title>Hello World</title>
5  </head>
6  <body>
7    <h1>Hello World from SJSU-{{ index_id }}(Ananya)</h1>
8  </body>
9  </html>
```

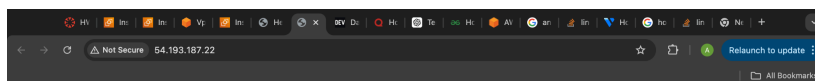
7. Now run the command **ansible-playbook -i inventory.ini playbook.yml**  
To get the output below:



```
ok: [server1]
TASK [Check Apache is running] *****
ok: [server1]
ok: [server2]
TASK [Copy index.html] *****
changed: [server1]
changed: [server2]
TASK [Create a custom index.html file] *****
ok: [server1]
ok: [server2]
PLAY RECAP *****
server1 : ok=5 changed=1 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
server2 : ok=5 changed=1 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
akshashe@Akshays-MacBook-Air terraform-ansible-setup % sudo apt-get update
Password:
sudo: a password is required
akshashe@Akshays-MacBook-Air terraform-ansible-setup % ansible-playbook -i inventory.ini playbook.yml

PLAY [Set up a web server] *****
TASK [Gathering Facts] *****
ok: [server1]
ok: [server2]
TASK [Check apache is installed] *****
ok: [server2]
ok: [server1]
TASK [Check Apache is running] *****
ok: [server2]
ok: [server1]
TASK [Copy index.html] *****
changed: [server2]
changed: [server1]
TASK [Create a custom index.html file] *****
ok: [server2]
ok: [server1]
PLAY RECAP *****
server1 : ok=5 changed=1 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
server2 : ok=5 changed=1 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
```

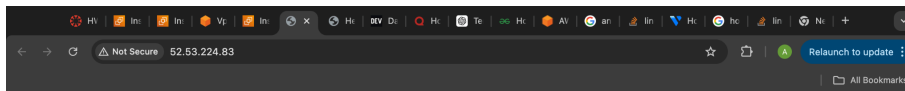
7. Checking a web page that is accessible from a web browser, and displays the message: “Hello World from SJSU-X” where X is 1 or 2 depending on which web server instance, VM1 or VM2.



**Hello World from SJSU-1(Ananya)**



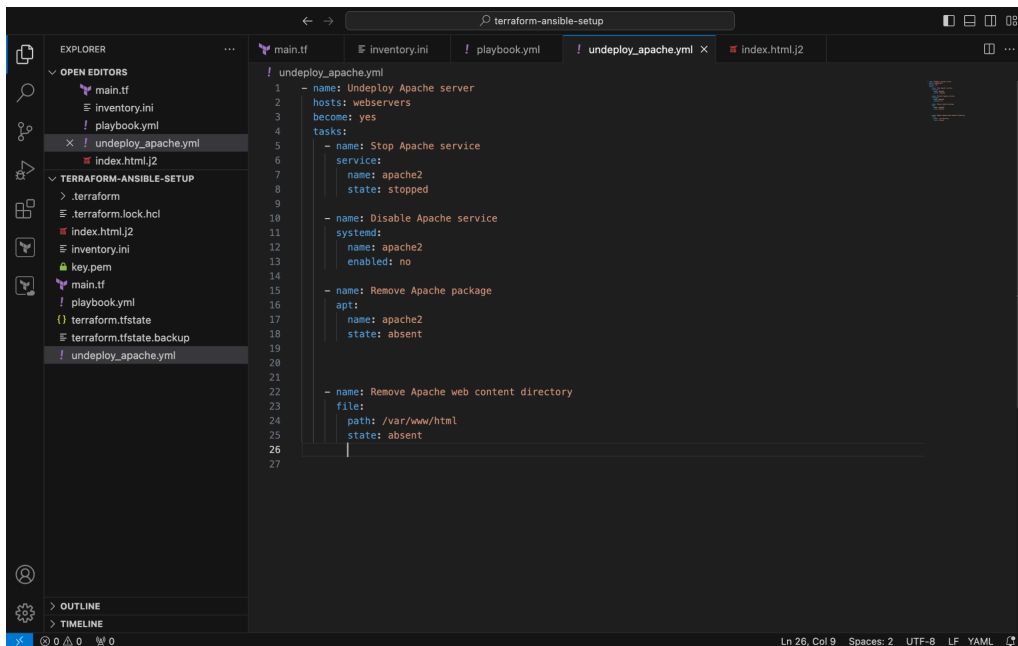
## 2nd VM:



**Hello World from SJSU-2(Ananya)**

This is how I used Ansible as a configuration management tool to deploy apache web servers on both the VMs without any difficulty. I used Ansible to get the task of starting apache, running apache and writing index.html file in a couple of mins.

8. Now it's time to un-deploy the apache server. With ansible we can list the tasks in apache\_undeploy.yml to un-deploy the apache server.



Output :

```
● akshashe@Akshays-MacBook-Air terraform-ansible-setup % ansible-playbook -i inventory.ini undeploy_apache.yml

PLAY [Undeploy Apache server] *****

TASK [Gathering Facts] *****
ok: [server1]
ok: [server2]

TASK [Stop Apache service] *****
[WARNING]: The service (apache2) is actually an init script but the system is managed by systemd
ok: [server1]
ok: [server2]

TASK [Disable Apache service] *****
ok: [server2]
ok: [server1]

TASK [Remove Apache package] *****
ok: [server2]
ok: [server1]

TASK [Remove Apache web content directory] *****
changed: [server1]
changed: [server2]

PLAY RECAP *****
server1      : ok=5    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
server2      : ok=5    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

---

Below is the github link repository of all the codes :

<https://github.com/ananya101001/Homework-Ansible>

---

Thank you !