**Create Virtual Machine of Linux and ubuntu with Terraform**

Tools Used:

* Git bash
* AWS CLI
* Terraform

**STEP 1: Open Git Bash**

**STEP 2: Login to AWS login**

**STEP 3: Create Terraform code file**

provider "aws" {

region = "us-east-2" # change to your preferred region

}

resource "aws\_vpc" "main" {

cidr\_block = "10.0.0.0/16"

}

resource "aws\_subnet" "main" {

vpc\_id = aws\_vpc.main.id

cidr\_block = "10.0.1.0/24"

availability\_zone = "us-east-2a"

}

resource "aws\_security\_group" "allow\_ssh" {

name = "allow\_ssh"

description = "Allow SSH access"

vpc\_id = aws\_vpc.main.id

ingress {

from\_port = 22

to\_port = 22

protocol = "tcp"

cidr\_blocks = ["0.0.0.0/0"]

}

egress {

from\_port = 0

to\_port = 0

protocol = "-1"

cidr\_blocks = ["0.0.0.0/0"]

}

}

# Amazon Linux EC2 Instance

resource "aws\_instance" "linux\_vm" {

ami = "ami-058a8a5ab36292159" # Amazon Linux 2 (us-east-1)

instance\_type = "t2.micro"

subnet\_id = aws\_subnet.main.id

key\_name = "1234" # Replace with your key pair name

security\_groups = [aws\_security\_group.allow\_ssh.id]

tags = {

Name = "AmazonLinuxVM"

}

}

# Ubuntu EC2 Instance

resource "aws\_instance" "ubuntu\_vm" {

ami = "ami-04f167a56786e4b09" # Ubuntu Server 20.04 LTS (us-east-1)

instance\_type = "t2.micro"

subnet\_id = aws\_subnet.main.id

key\_name = "1234" # Replace with your key pair name

security\_groups = [aws\_security\_group.allow\_ssh.id]

tags = {

Name = "UbuntuVM"

}

}

**STEP 4: Run terraform commands**

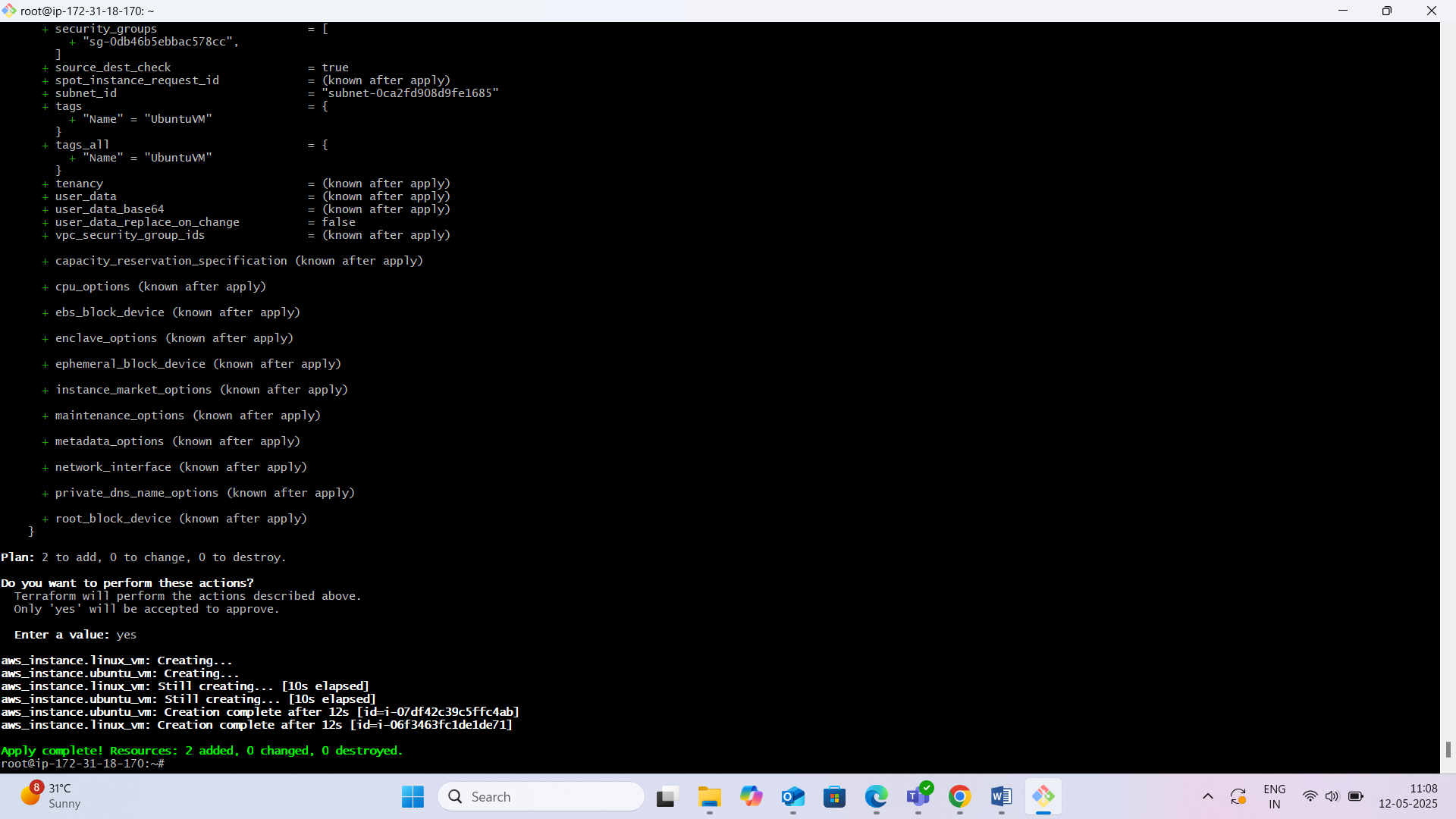
terraform init

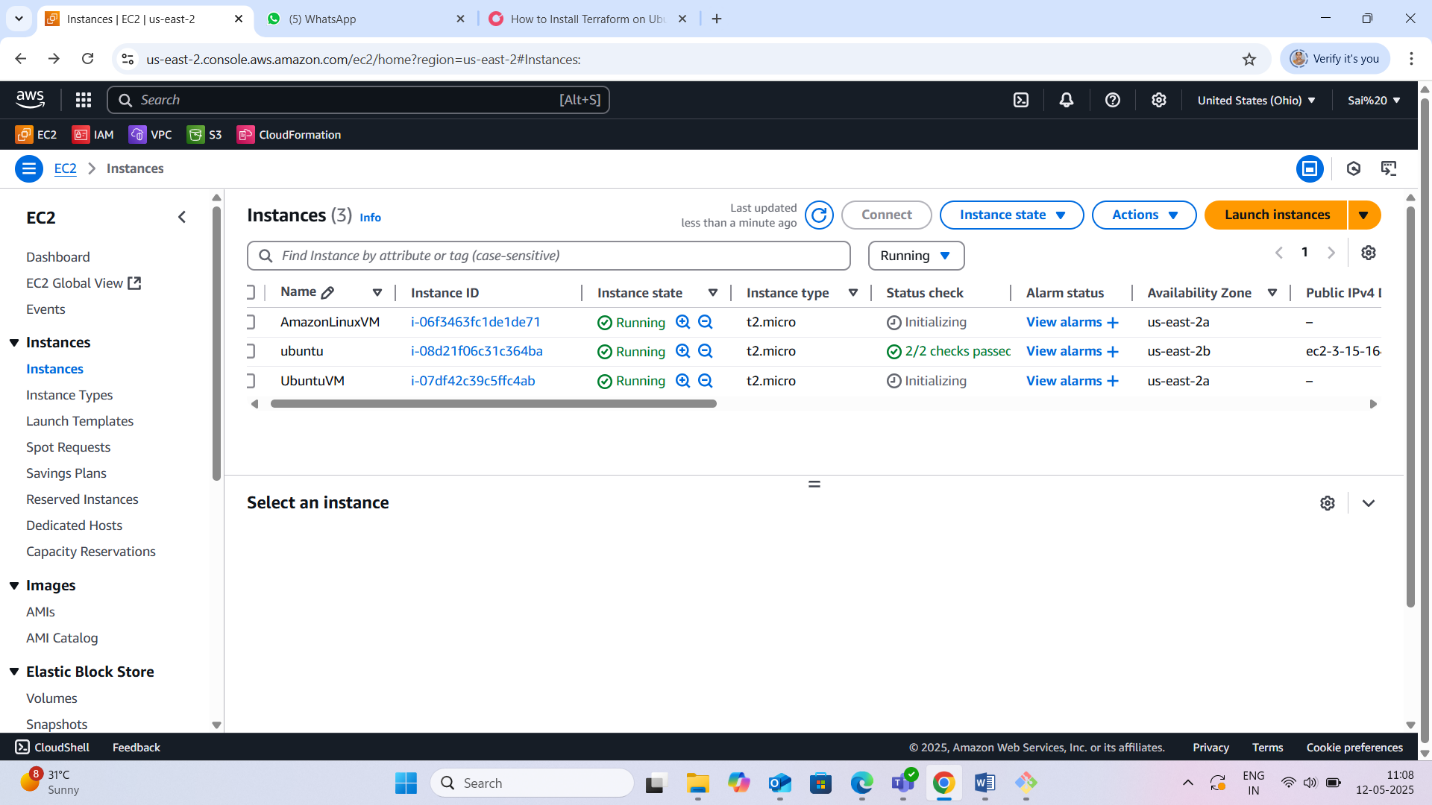
terraform plan

terraform apply

**STEP 5: Output**

Now you can see the output as attached screenshot





**STEP 6: Push to GitHub**

1. git init

2. git remote add origin <your-repo-link>

3. git add .

4. git commit -m "Active users listing using Terraform"

5. git push -u origin main

**STEP 7: Submit the following**

Terraform file

Notepad Documentation

Screenshot

GitHub code link