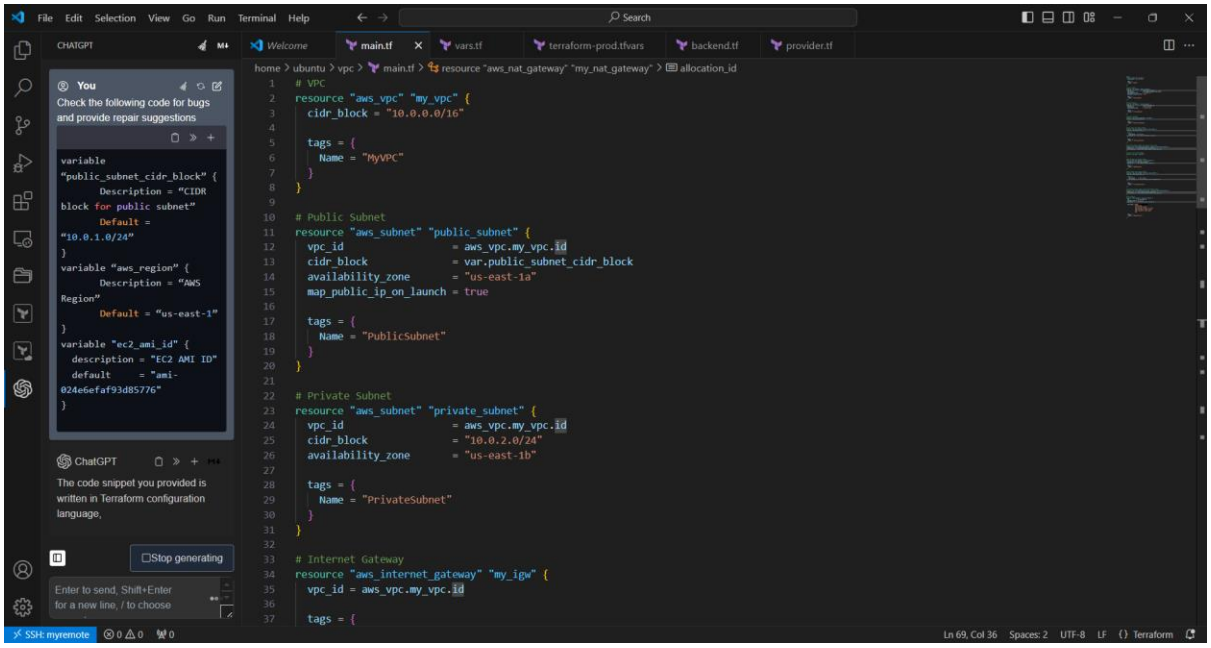
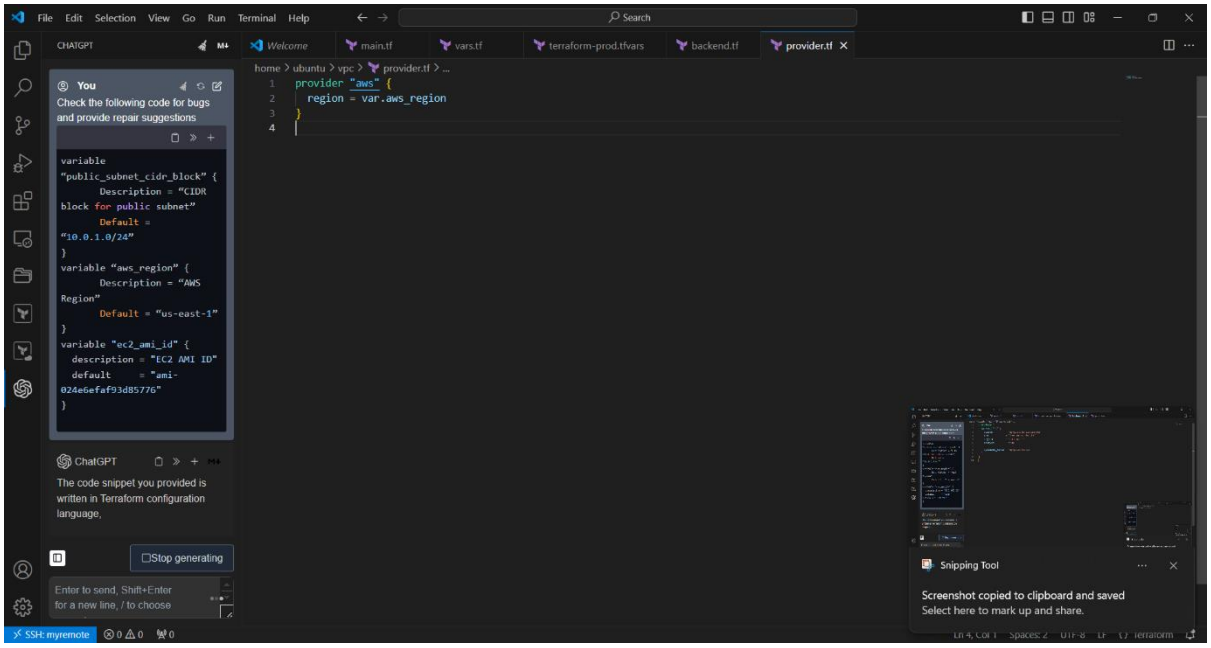
AWS VPC Creation in Terraform

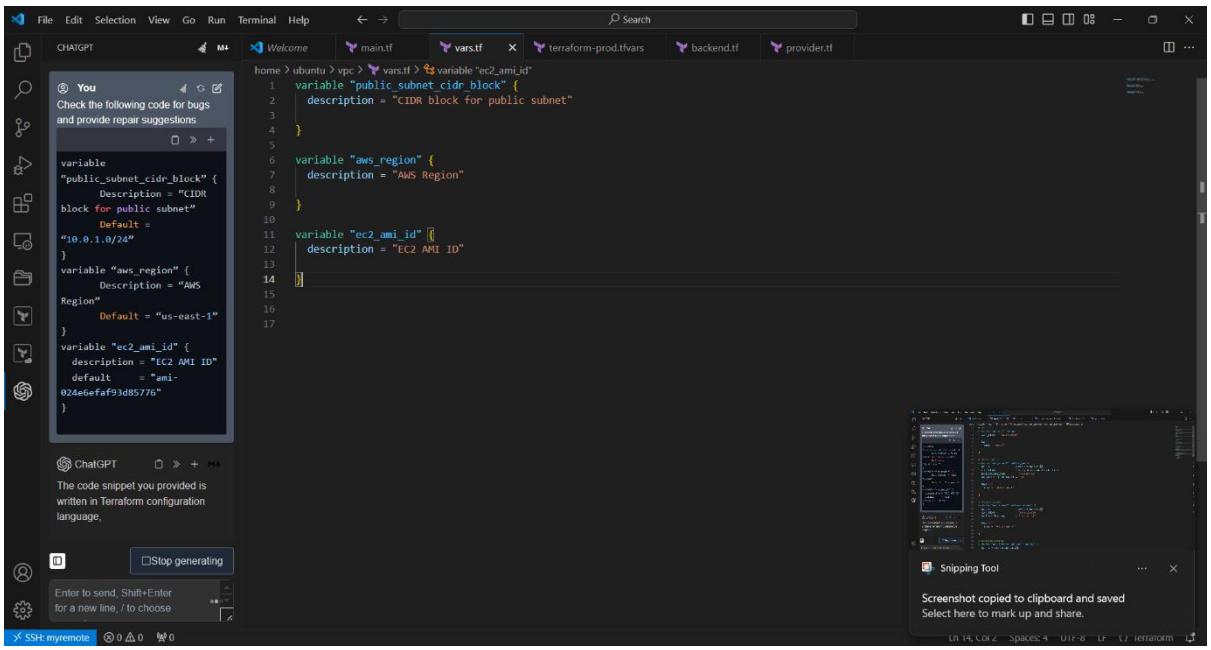
1. Created an own VPC.
2. Created a Public and Private subnet for different Available AZs by assigning different CIDR blocks.
3. Created Internet Gateway & attach it to the VPC.
4. Created Routing table [RT], One as Public & One as Private by associating the appropriate subnets to it.
5. Edited the Public route table's Route alone and map the IGW, not the Private and leave it as it is.
6. Created NAT gateway with new Elastic IP for the internet connection in the Private Subnet. Map it to Private RT.
7. Created an ec2 in public subnet and given the user data to launch an Apache web page in it.
8. Created a s3 bucket to store the terraform statefile in it .
9. Created a DynamoDB table to lock the statefile in s3 bucket.
10. Created the files separately to be reused for dev environment, prod environment etc
    - Main.tf
    - Provider.tf
    - Vars.tf
    - terraform-prod.tfvars



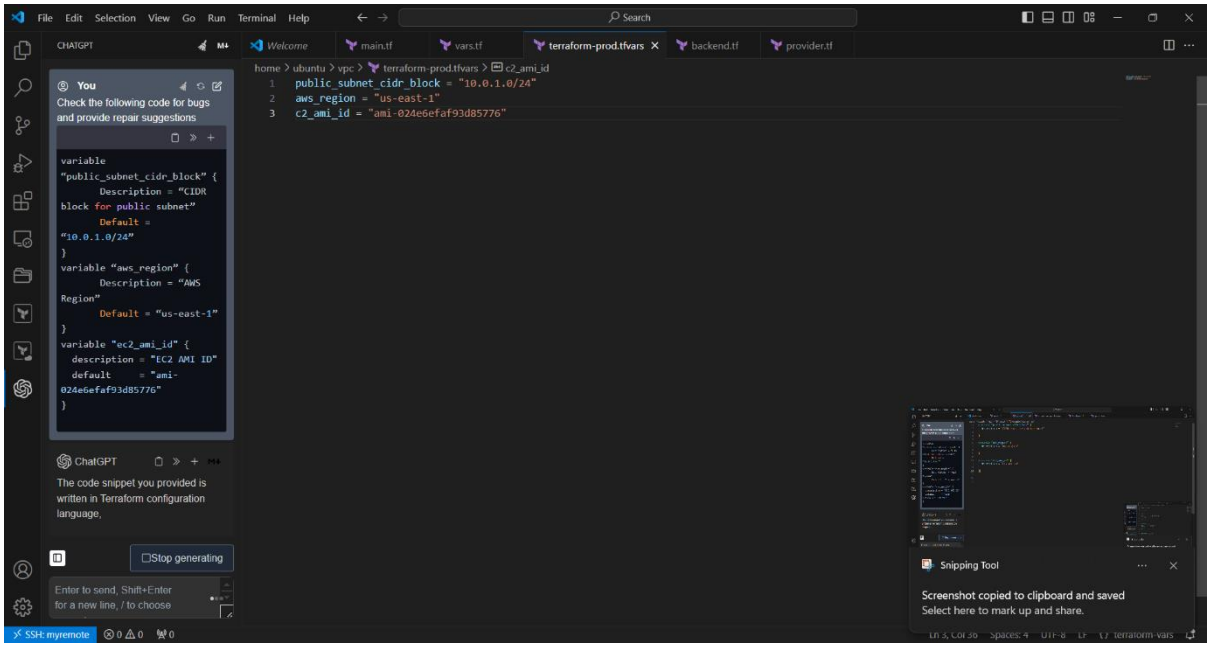
1.created main.tf file with all the resources



2.created provide.tf to specify cloud provider

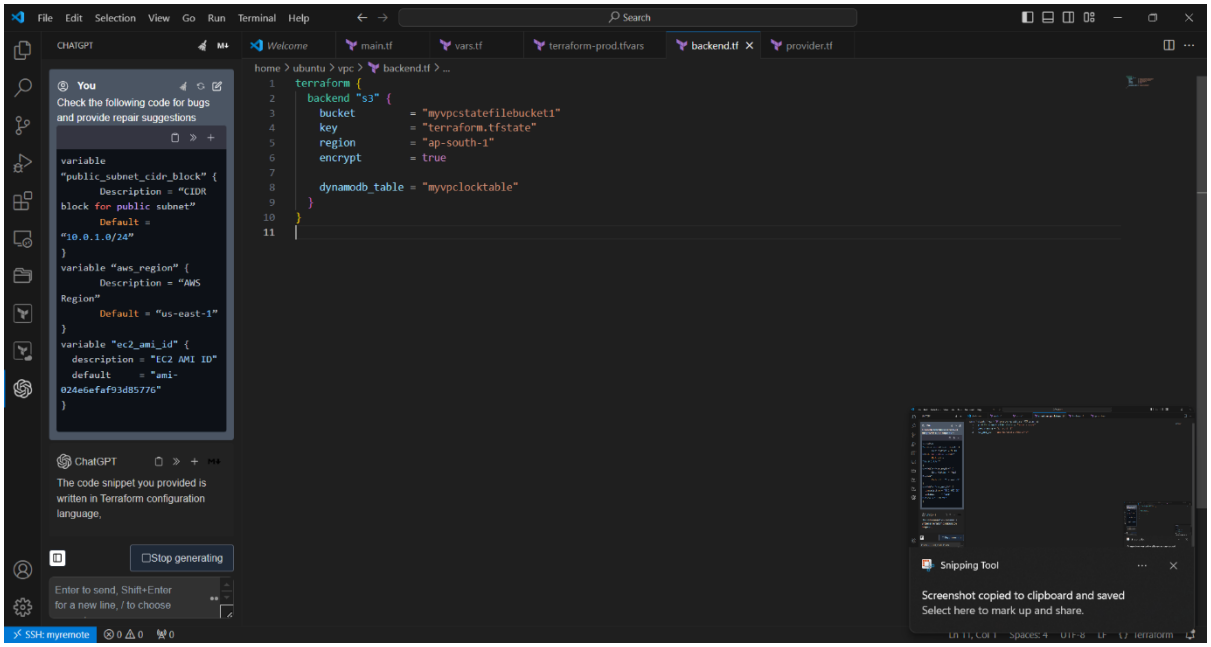


3.created vars.tf to specify variables

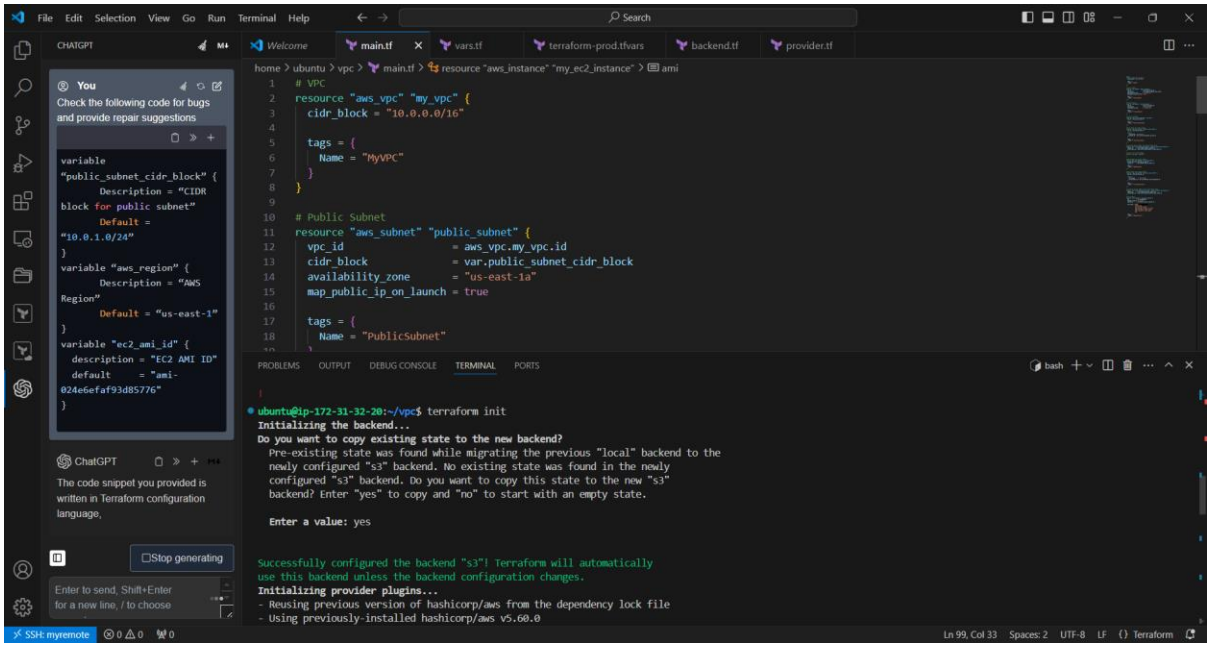


4.created terraform-prod. tfvars to specify the variables for reuse the code for different environments

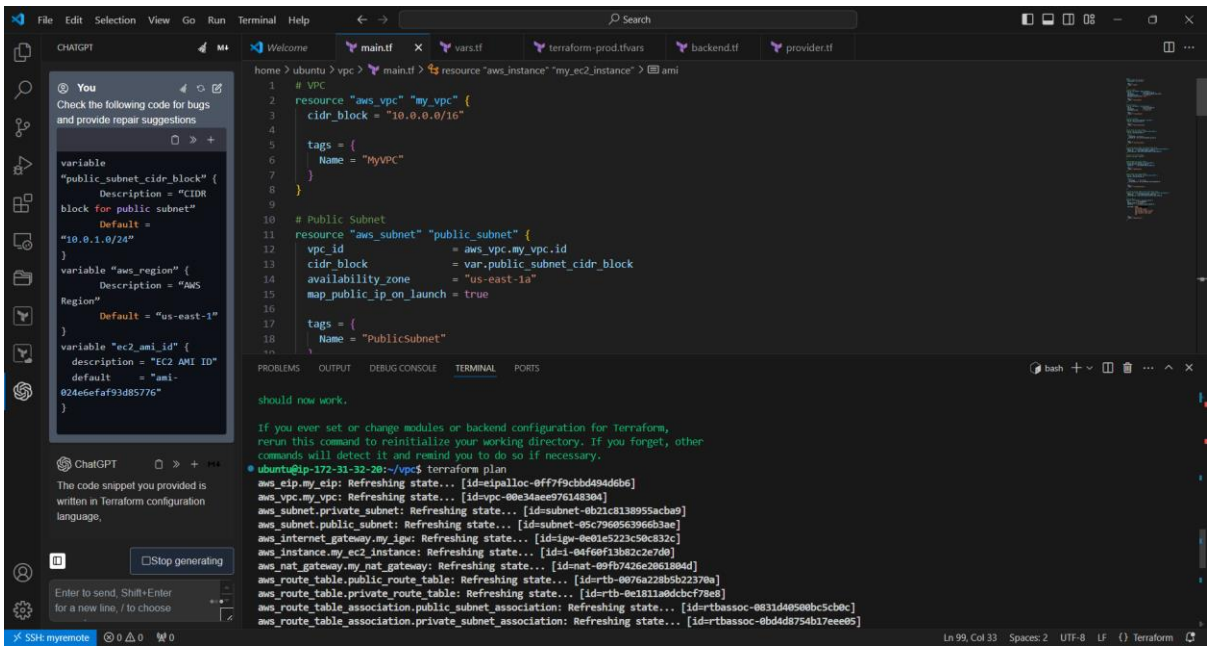
Terraform init –var-file = “terraform-prod. Tfvars” use this command to specify the environment.



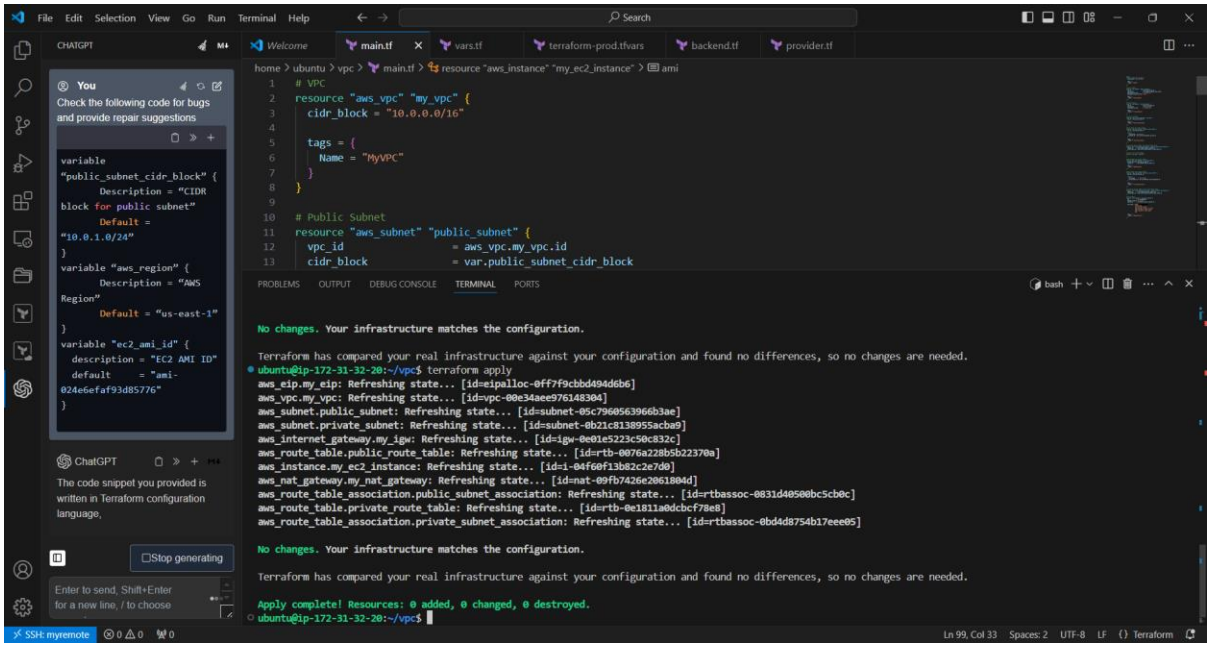
5.created backend.tf file to store the statefile in s3 bucket and locked with DynamoDB



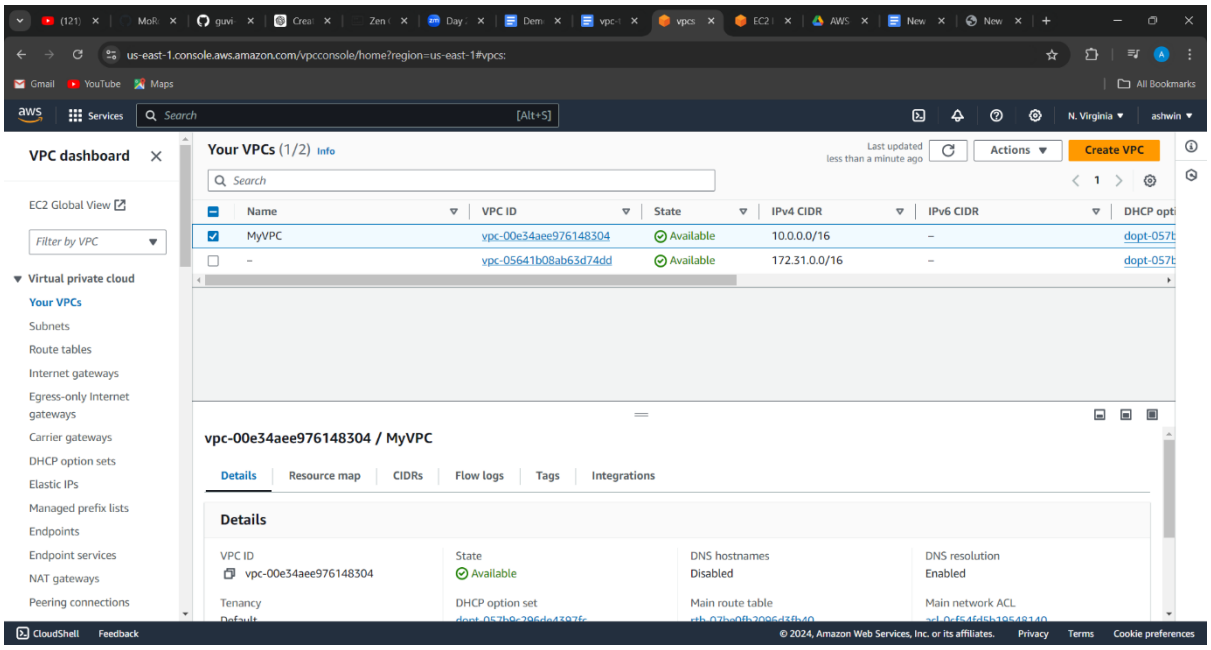
6.terraform init



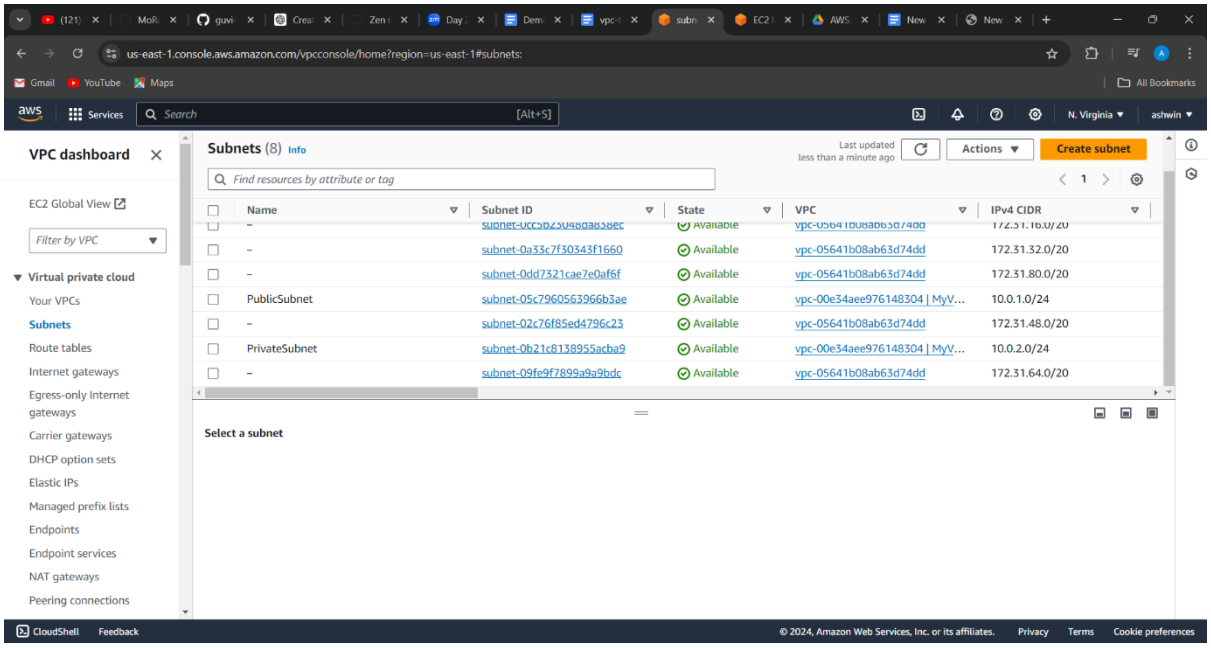
7.terraform plan



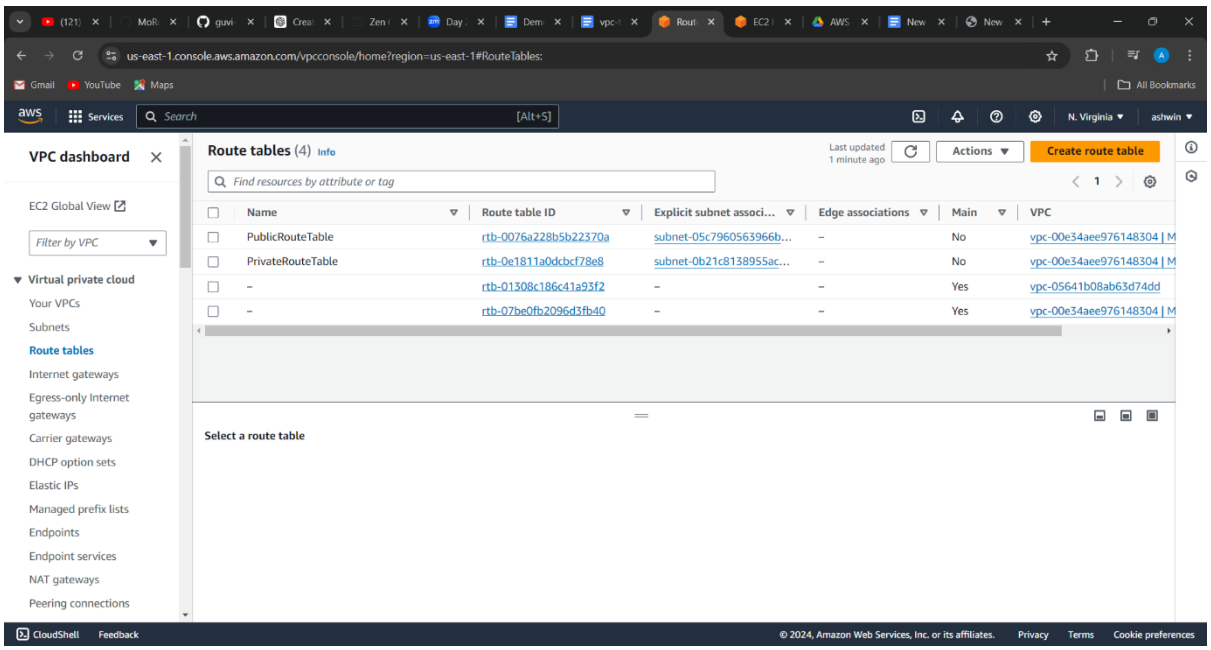
8.terraform apply



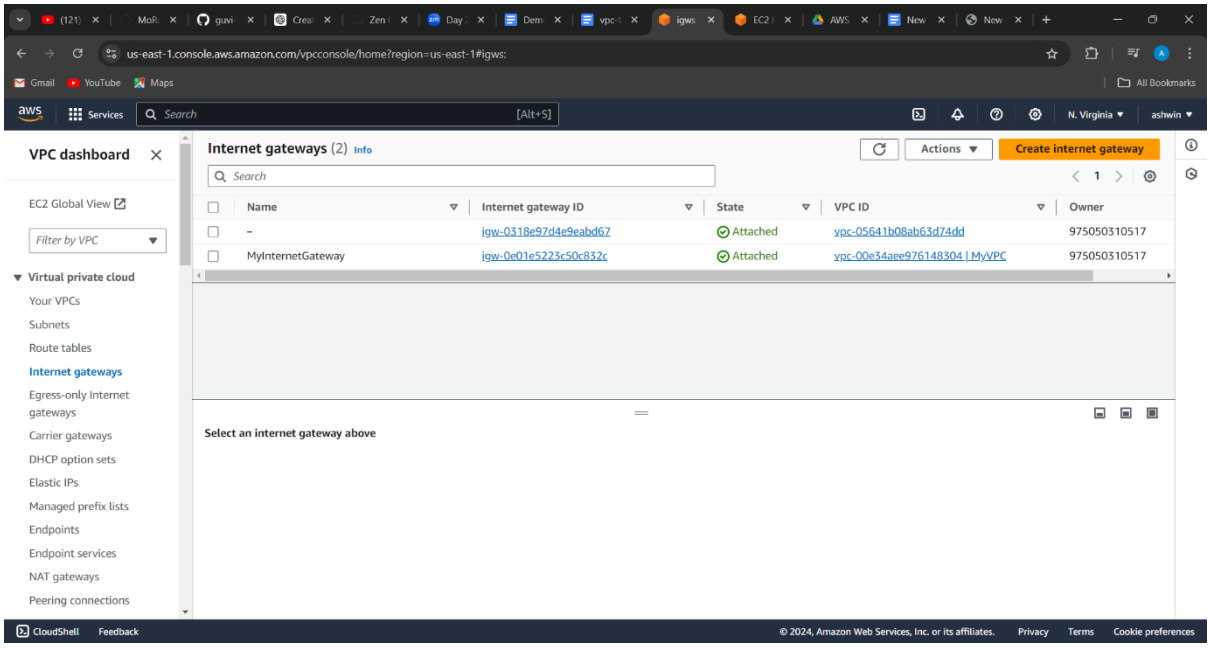
9.vpc has created



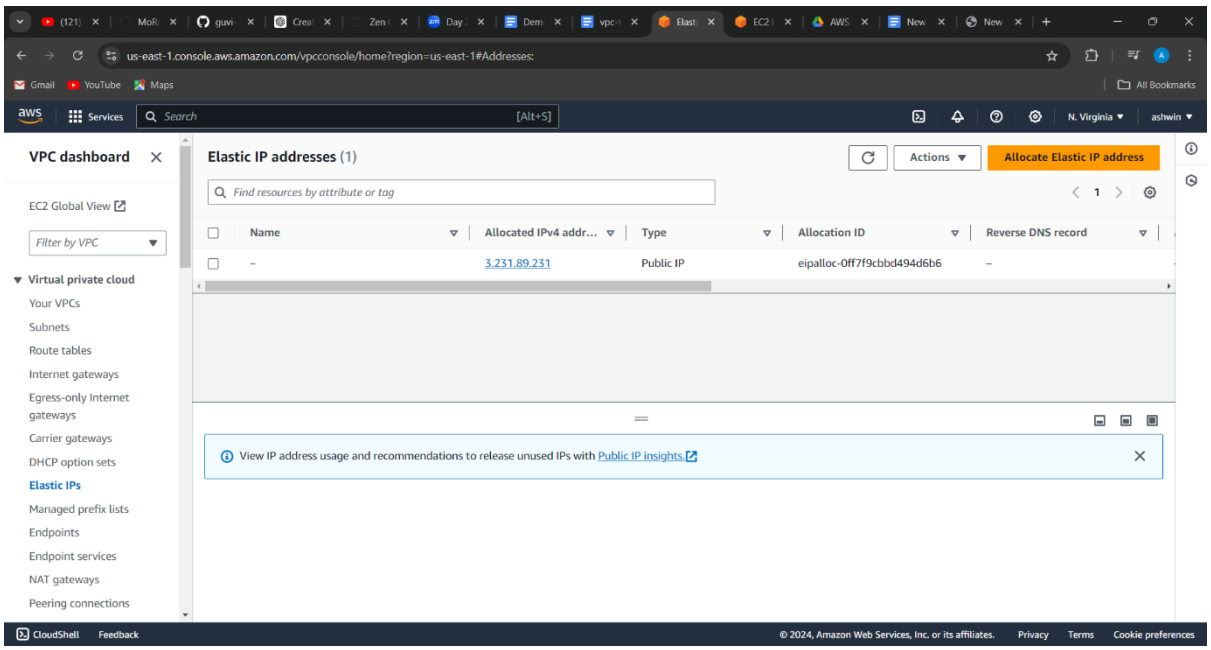
10 created subnets public and private in different availability zones



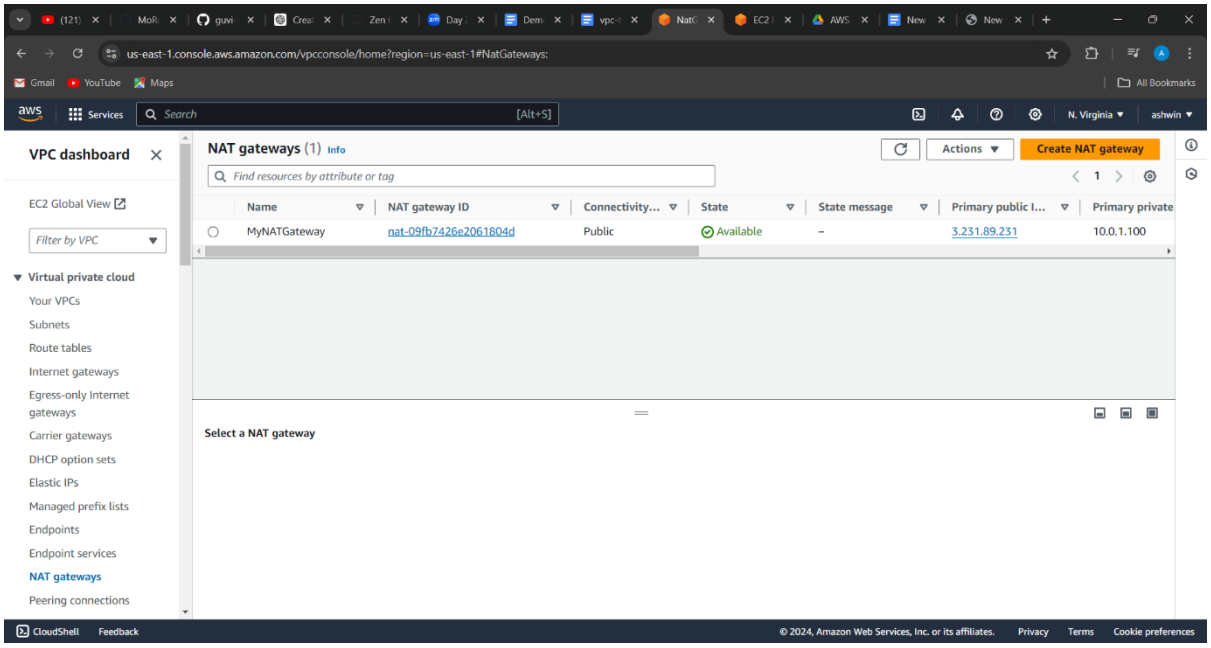
11.created route tables for public and private.



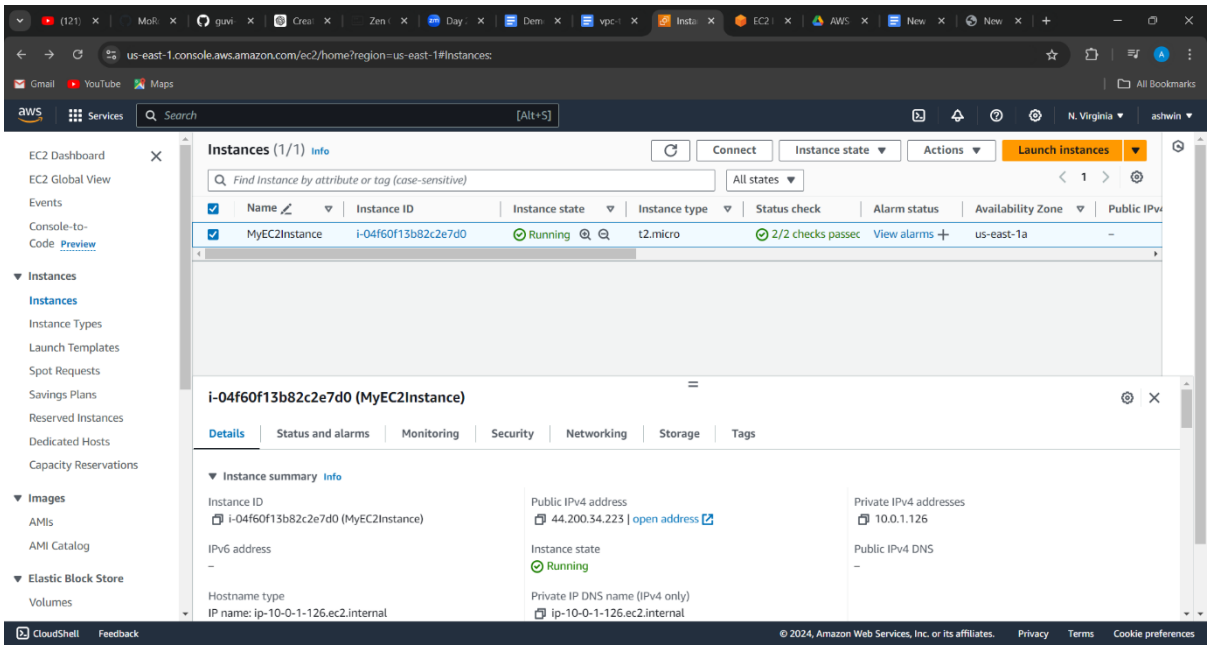
12.created igw and attached to vpc and public route table .



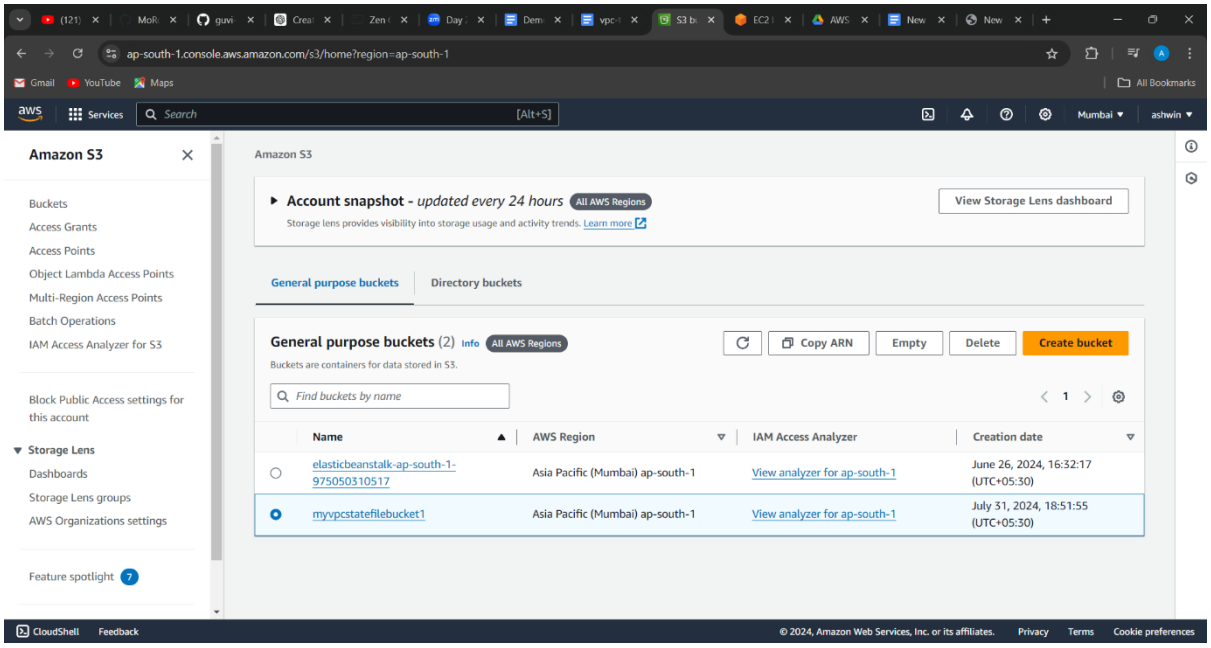
13.created a elastic ip to create NAT gateway



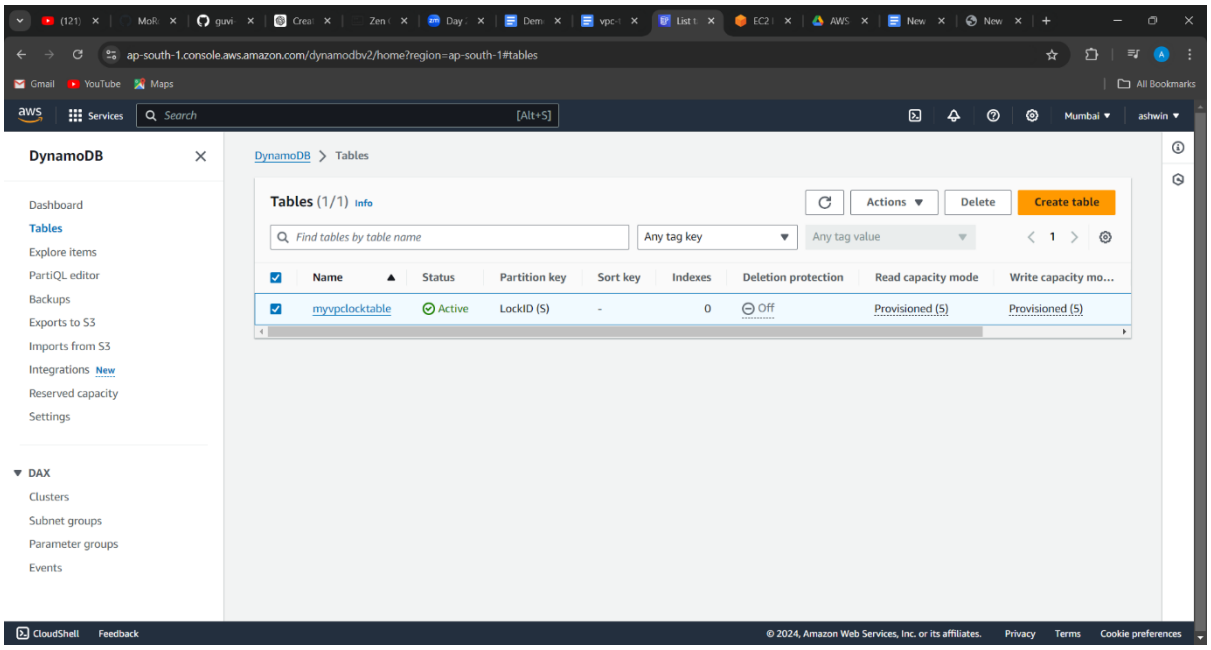
14.created a nat gateway and attached to private route table



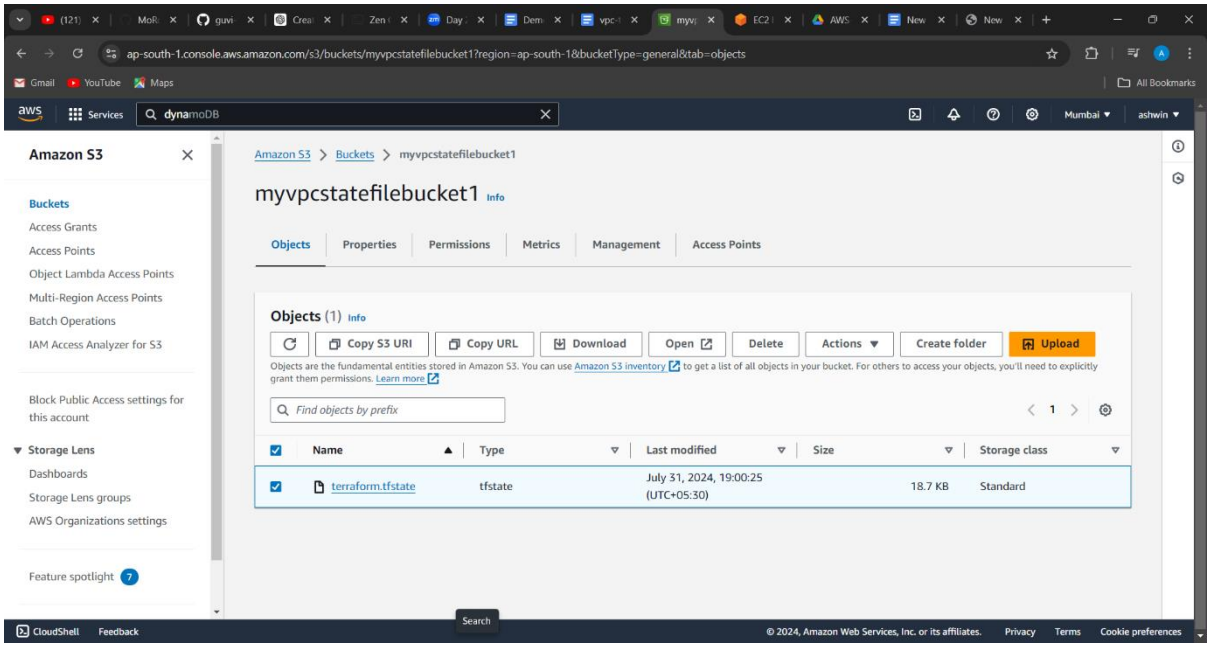
15. A ec2 machine has created in public subnet of the vpc



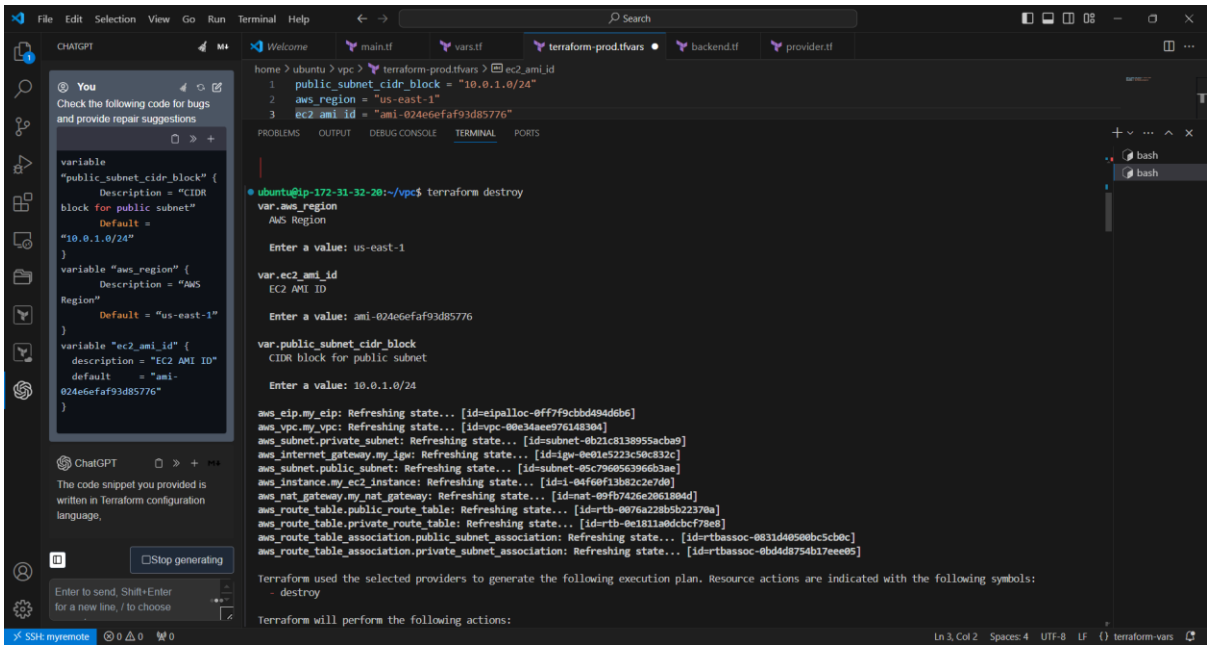
16. s3 bucket is created to store statefile



17.A dynamodb table has been created to lock the statefile



18.terraform statefile has been pushed to s3 bucket



19.terraform destroy it has asking for some variables to be destroyed because of DynamoDB lock