**Task 1: Dockerize Nginx**

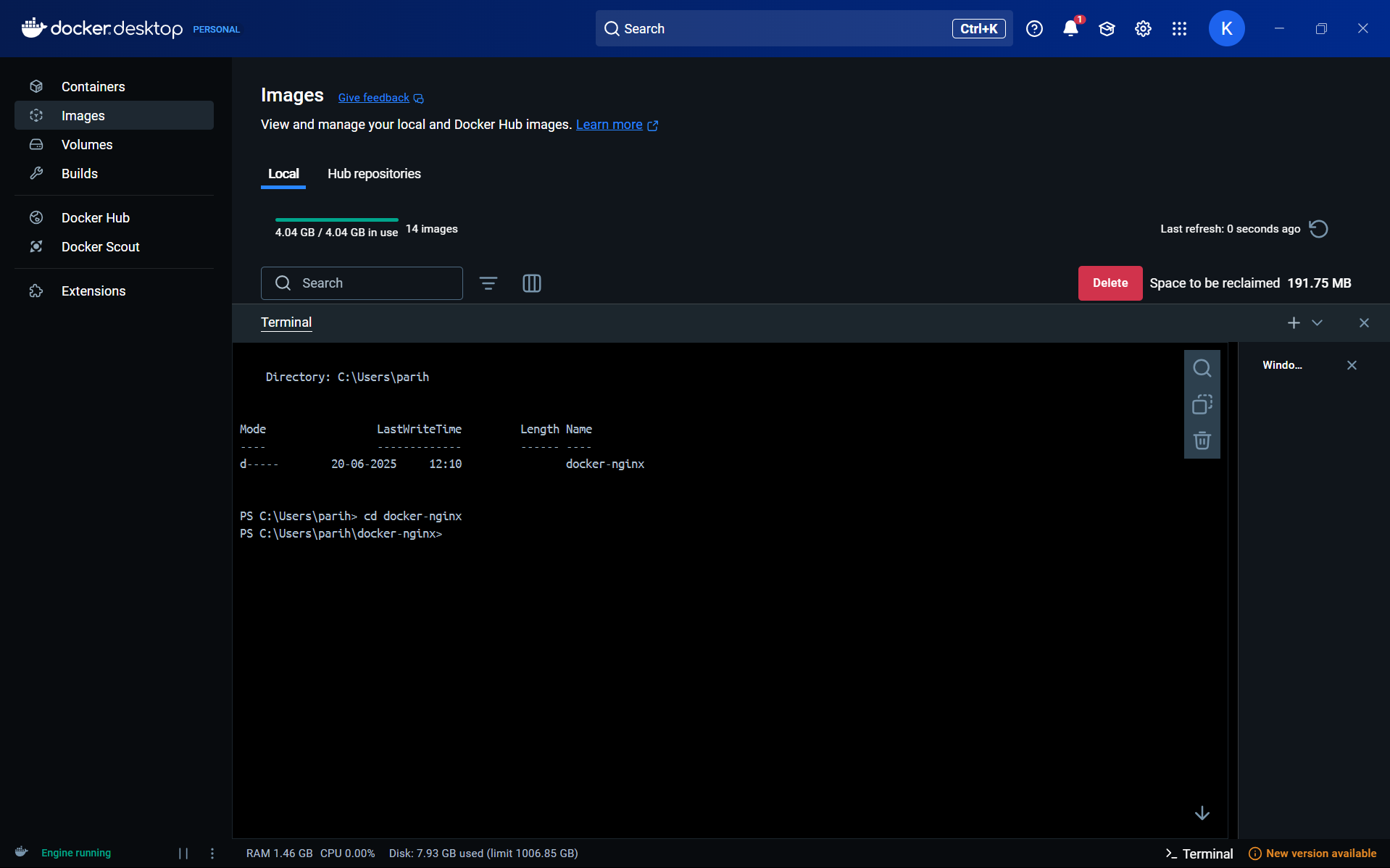
### **Objective:**

Create a Docker image that runs a custom Nginx server with your own index.html file, run it locally, and (optionally) push to Docker Hub.

**STEP 1: Create a Project Folder**

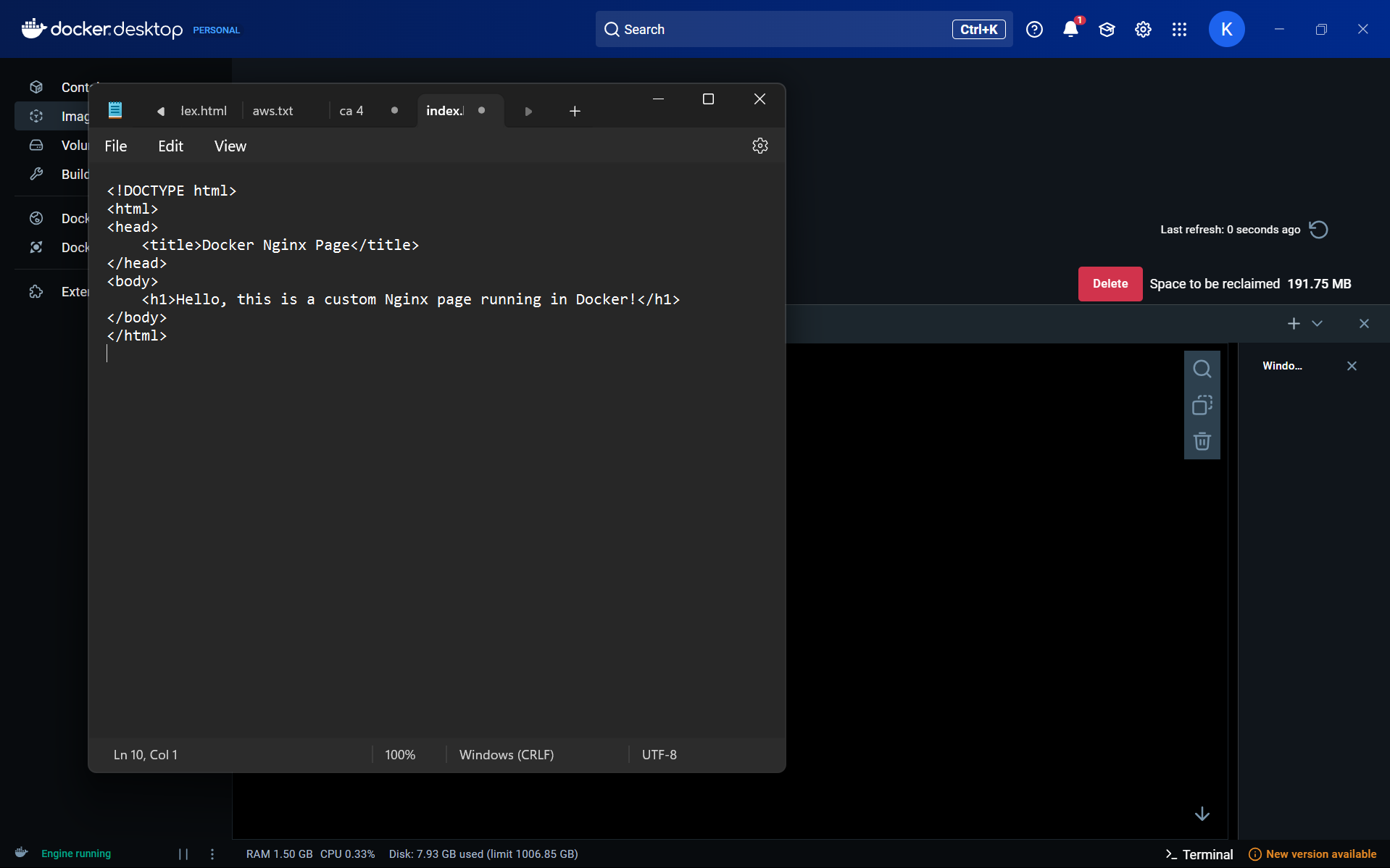
mkdir docker-nginx

cd docker-nginx



**STEP 2: Create index.html**

Notepad index.html



Index.html file contains

<!DOCTYPE html>

<html>

<head>

<title>Docker Nginx Page</title>

</head>

<body>

<h1>Hello, this is a custom Nginx page running in Docker!</h1>

</body>

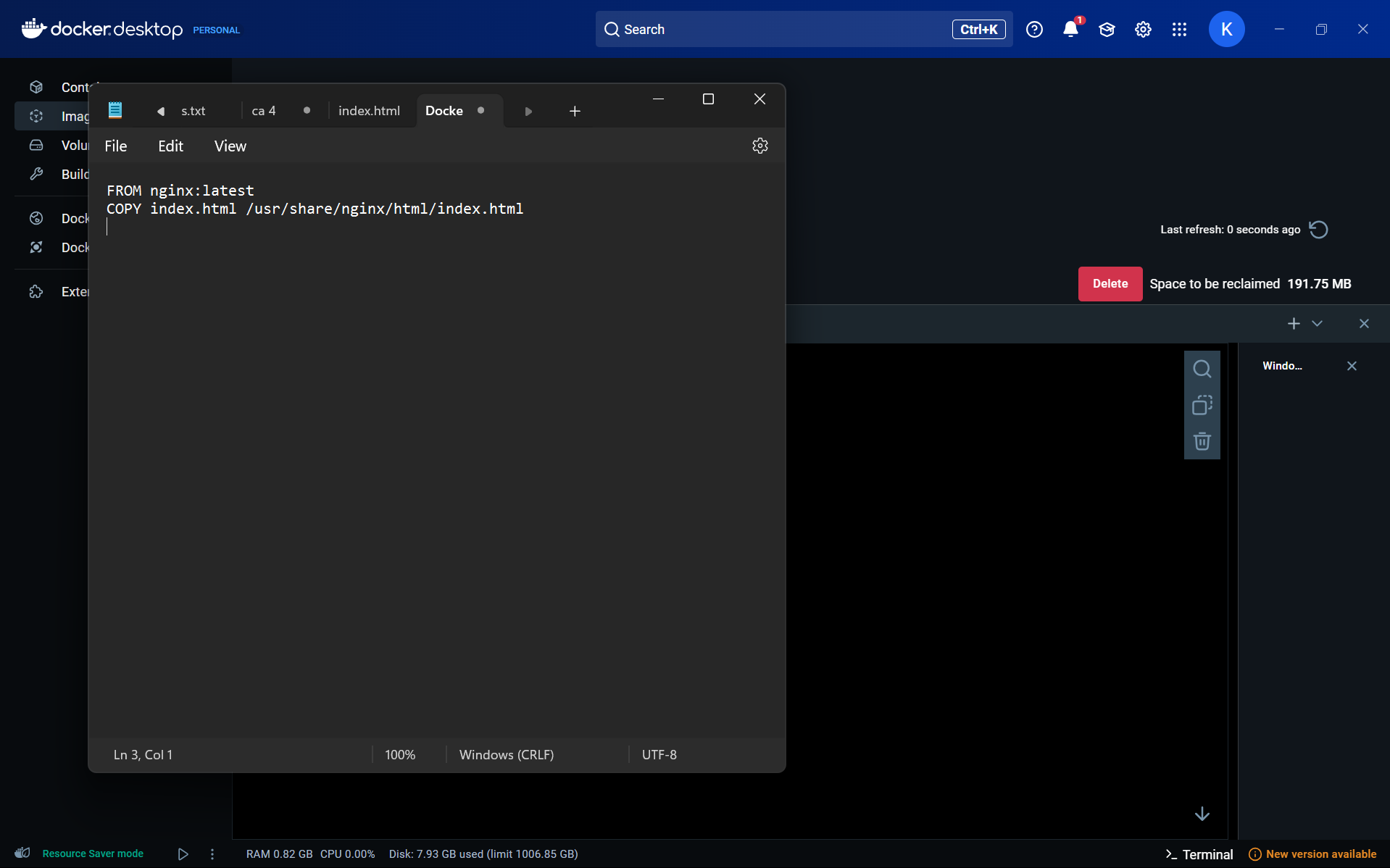
</html>

**STEP 3: Create Dockerfile**

Docker file conatains

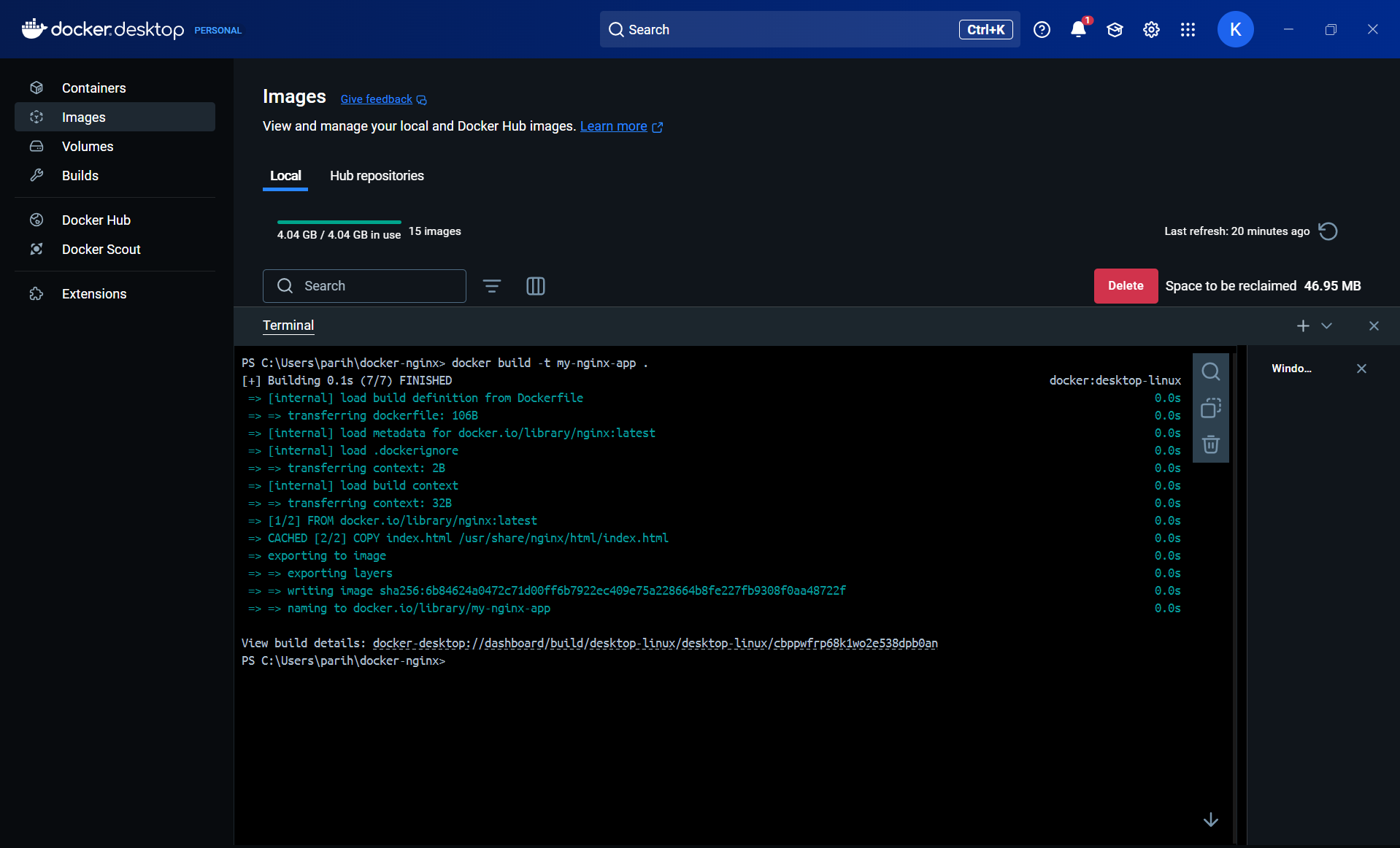
FROM nginx:latest

COPY index.html /usr/share/nginx/html/index.html



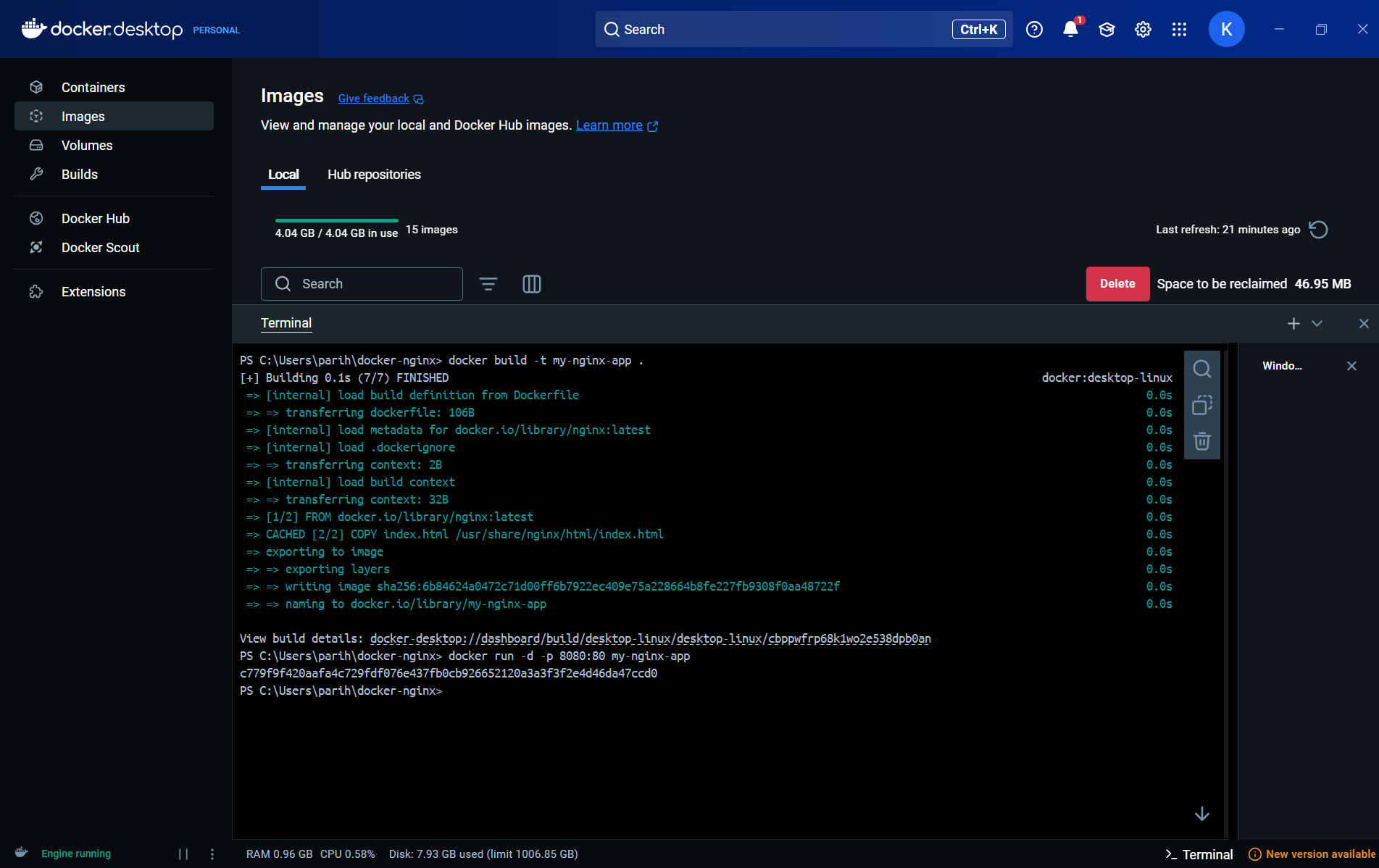
**STEP 4: Build the Docker Image**

docker build -t my-nginx-app .

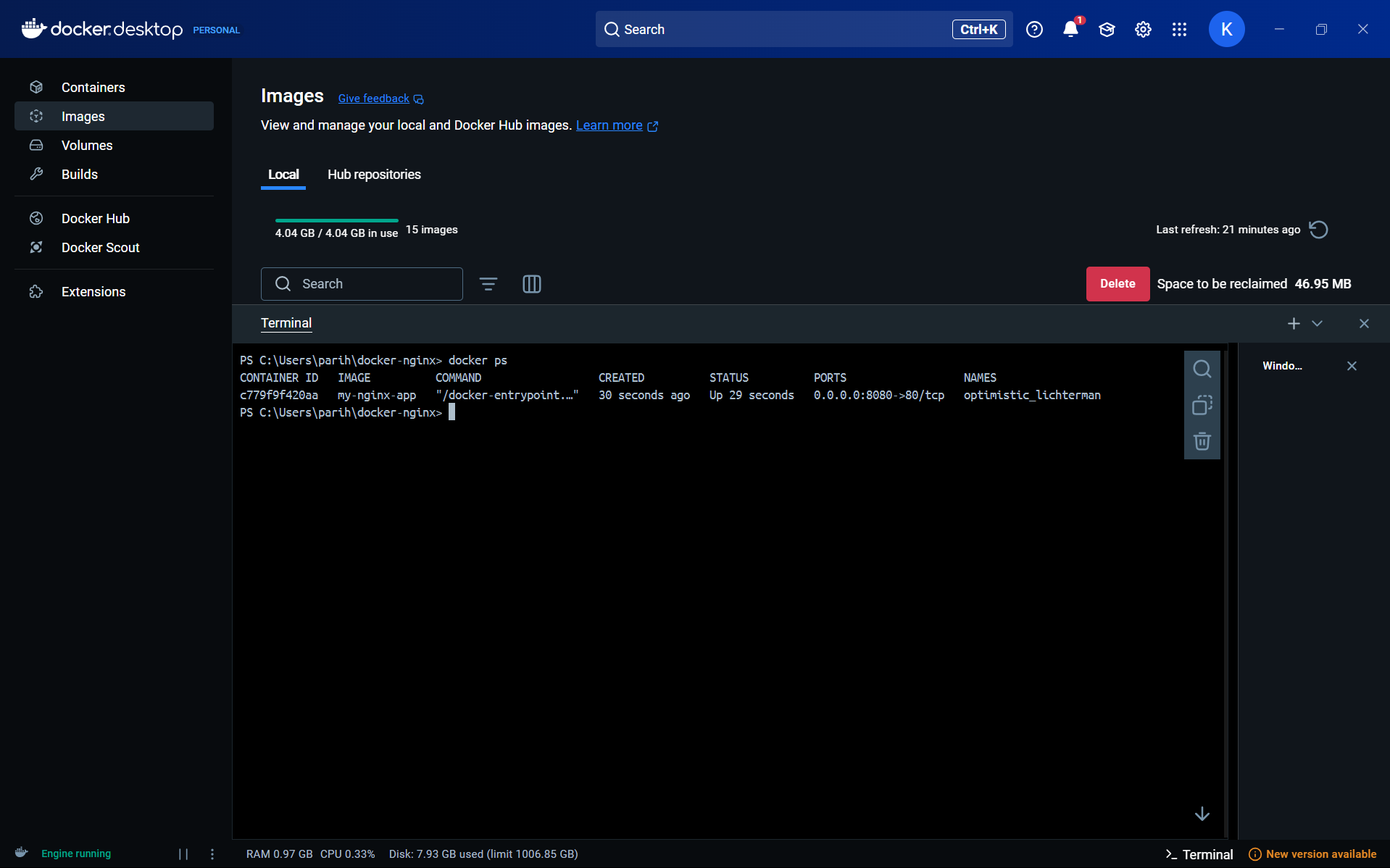


**STEP 5: Run the Docker Container**

docker run -d -p 8080:80 my-nginx-app

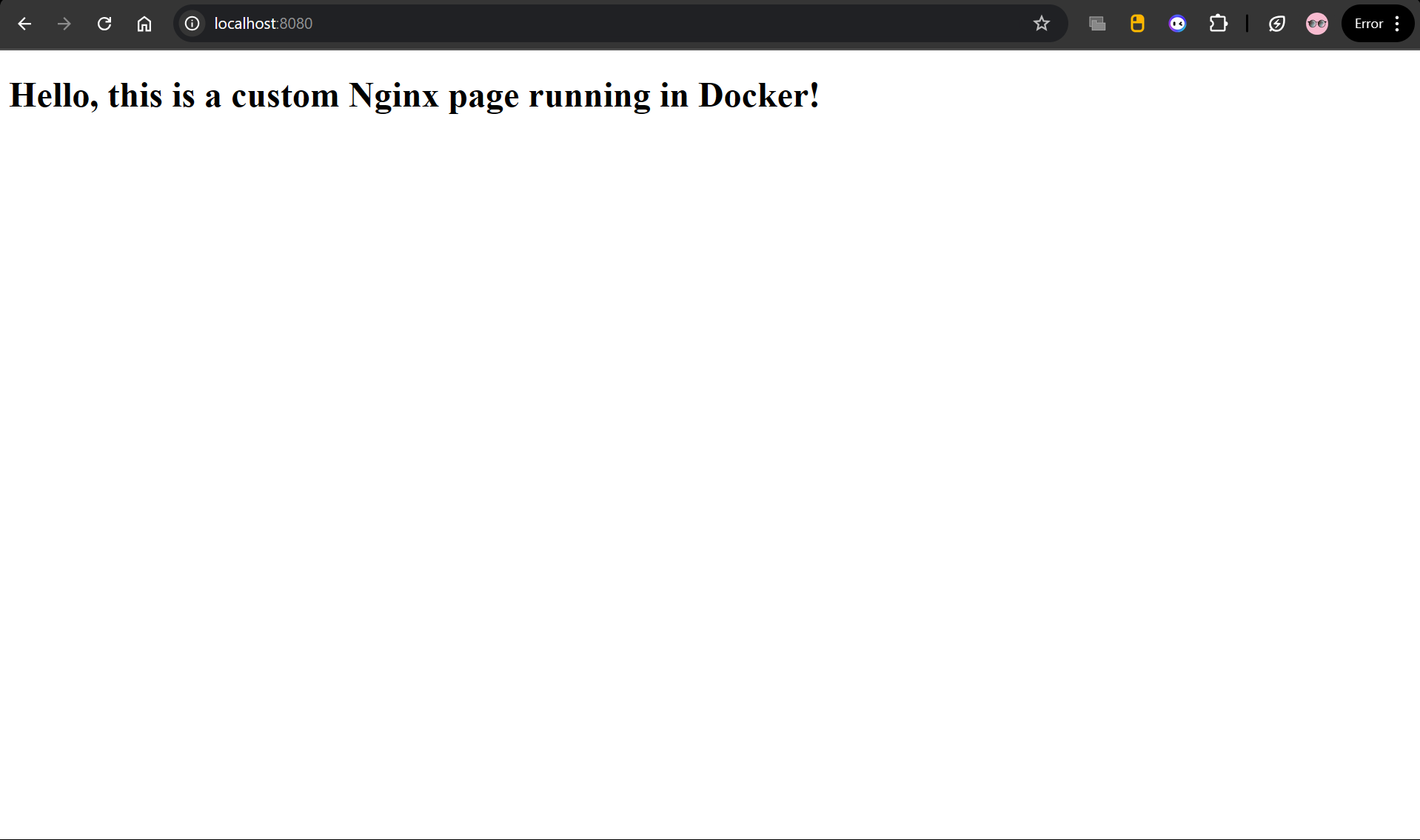


docker ps



**STEP 6: Test in Browser**

http://localhost:8080



**Task 2: Launch EC2 Instance using Terraform**

### **Objective:**

To use Infrastructure as Code (IaC) with Terraform to:

* Launch an EC2 instance using AWS Free Tier (Amazon Linux 2)
* Install and run Nginx on instance launch using user\_data
* Open required ports using a security group
* Output the public IP address for web access

### **Environment Setup**

Before starting the Terraform project, the necessary tools and configurations were installed and verified to ensure a smooth deployment.

1. **Terraform Installation**

**Terraform –version**

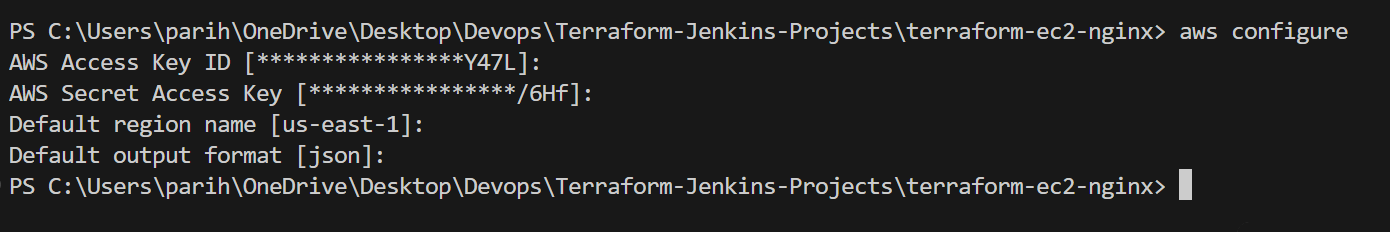
**2. Visual Studio Code Setup**

**Already download and installed extension by hasicorp terraform**

**3. AWS CLI Installation and Configuration**

**Already configure setup aws cli in vs code**

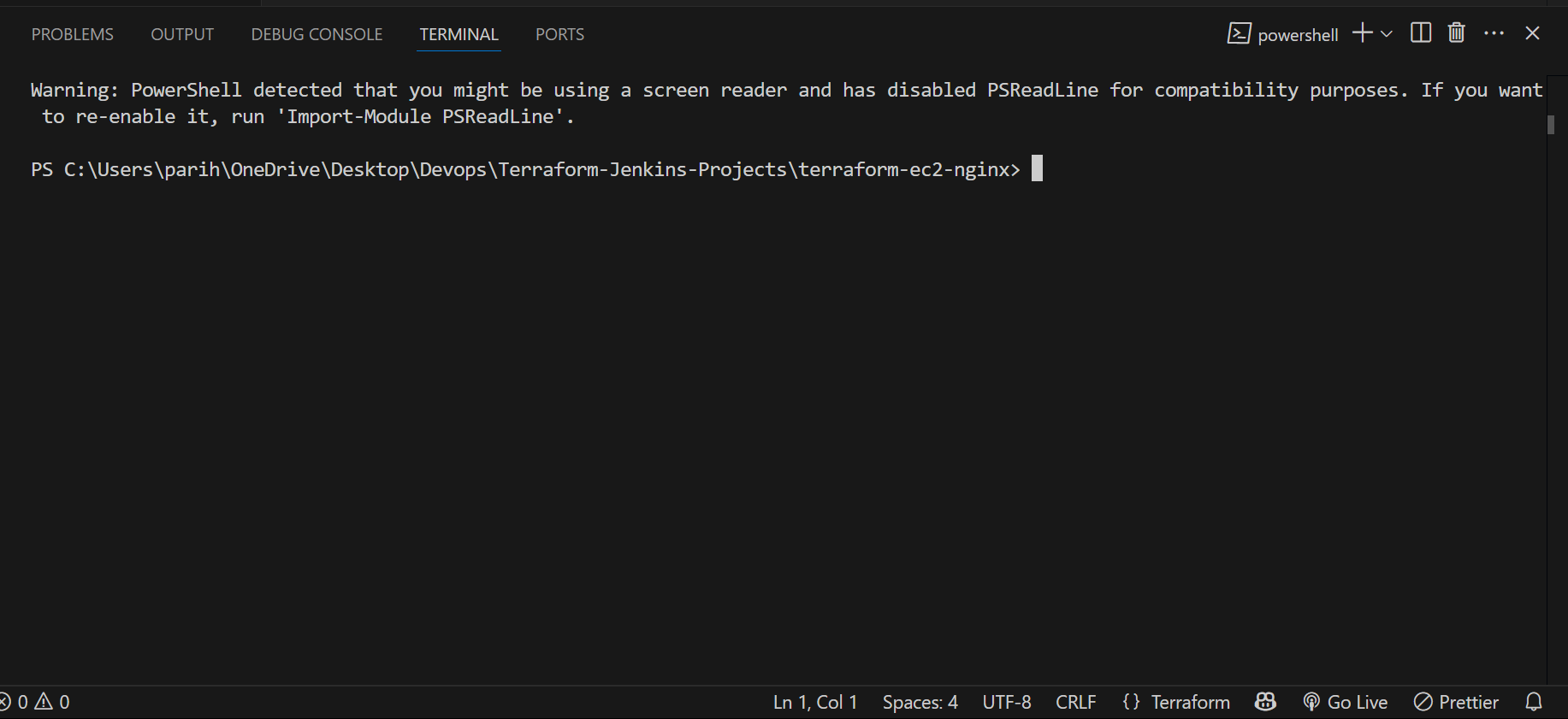
**Aws configure**

****

**Step 1: Create a New Project Directory**

**mkdir terraform-ec2-nginx**

**cd terraform-ec2-nginx**



**Step 2: Create main.tf File**

provider "aws" {

region = "us-east-1"

}

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

name = "allow\_ssh\_http"

description = "Allow SSH and HTTP traffic"

ingress {

from\_port = 22

to\_port = 22

protocol = "tcp"

cidr\_blocks = ["0.0.0.0/0"]

}

ingress {

from\_port = 80

to\_port = 80

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"]

}

}

resource "aws\_instance" "nginx\_server" {

ami = "ami-0c02fb55956c7d316"

instance\_type = "t2.micro"

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

user\_data = <<-EOF

#!/bin/bash

yum update -y

amazon-linux-extras install nginx1 -y

systemctl start nginx

systemctl enable nginx

EOF

tags = {

Name = "Terraform-Nginx-Server"

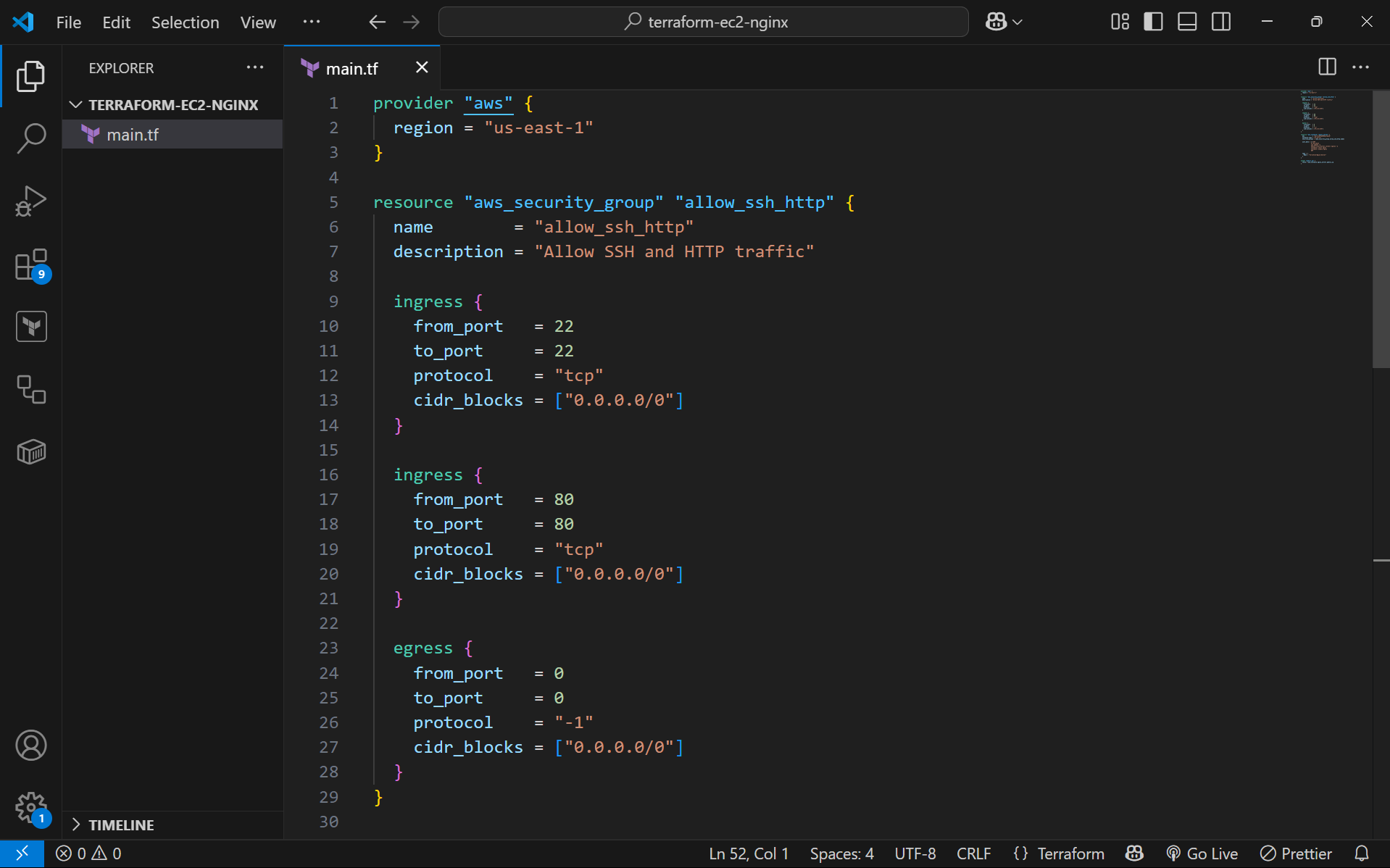
}

}

output "public\_ip" {

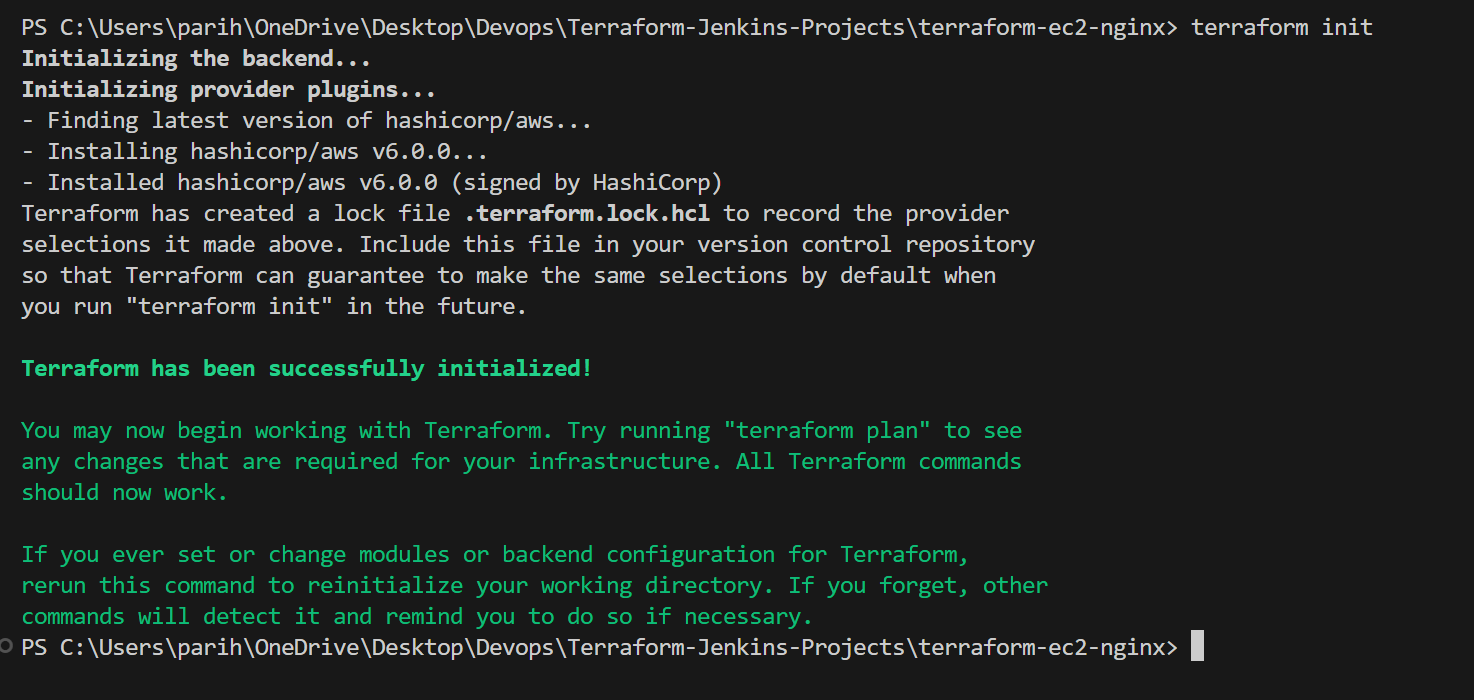
value = aws\_instance.nginx\_server.public\_ip

}



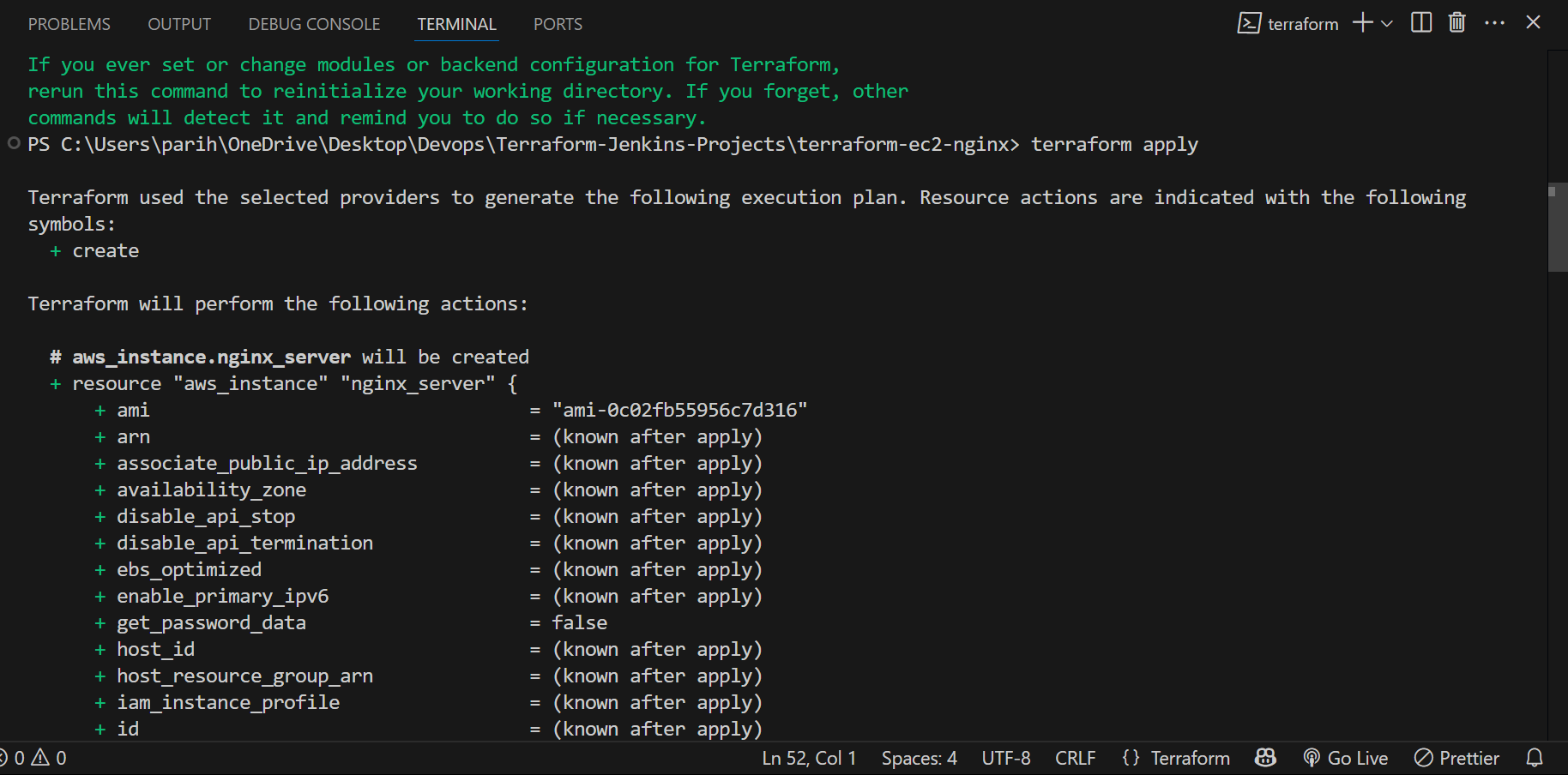
**Step 3: Initialize Terraform**

terraform init



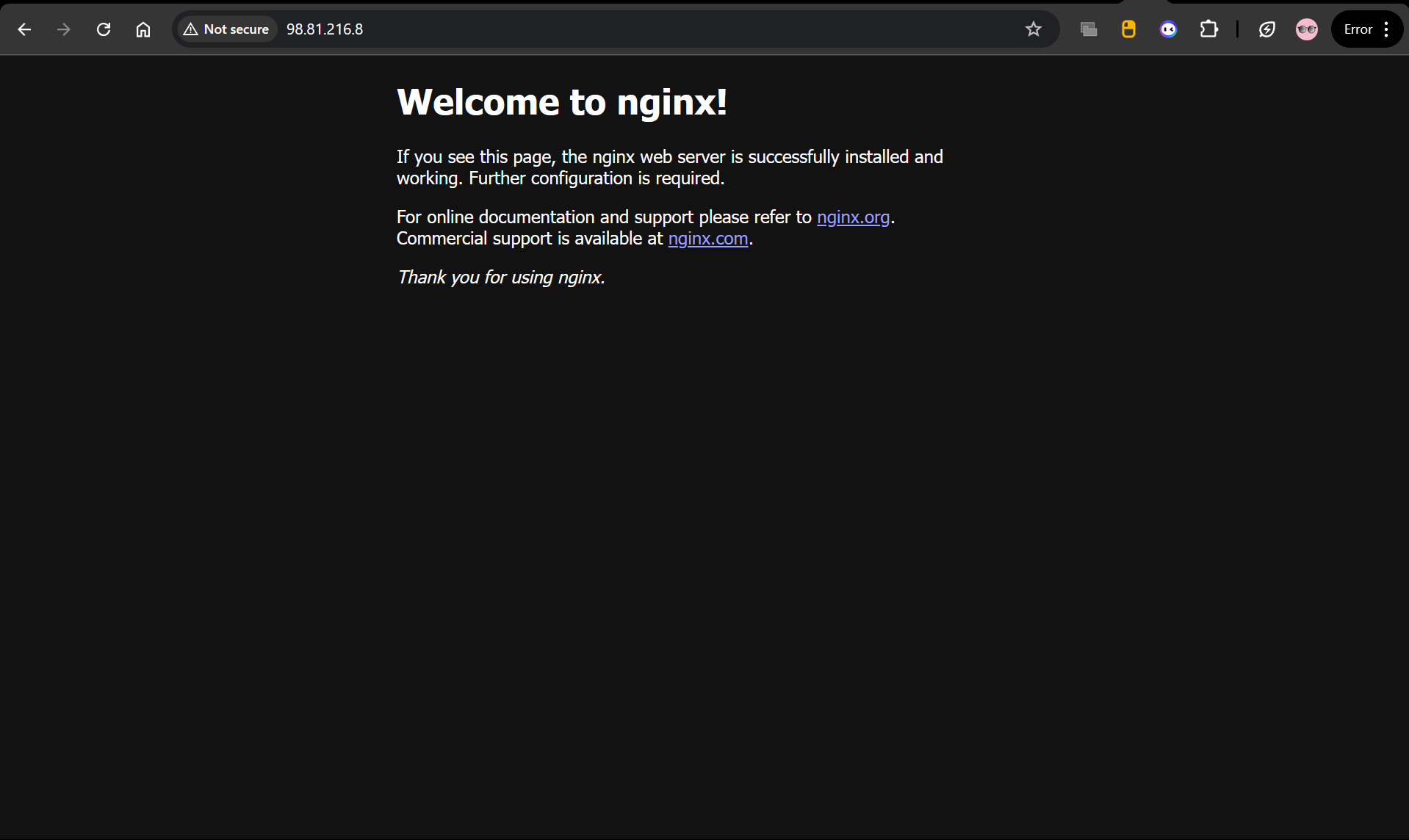
**Step 4: Apply the Configuration**

**terraform apply**



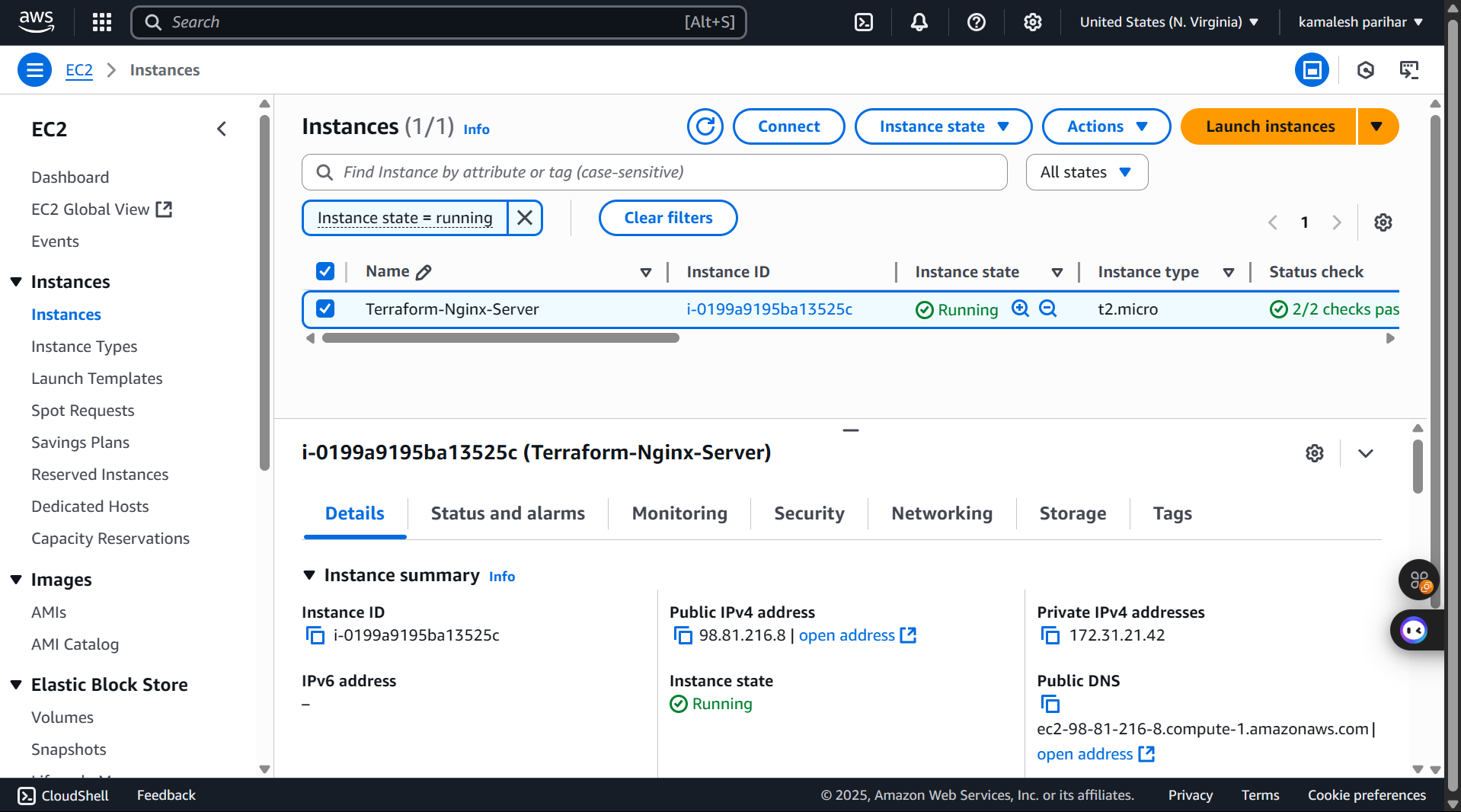
**Step 5: Access EC2 Instance in Browser**

**Url - http://98.81.216.8/**

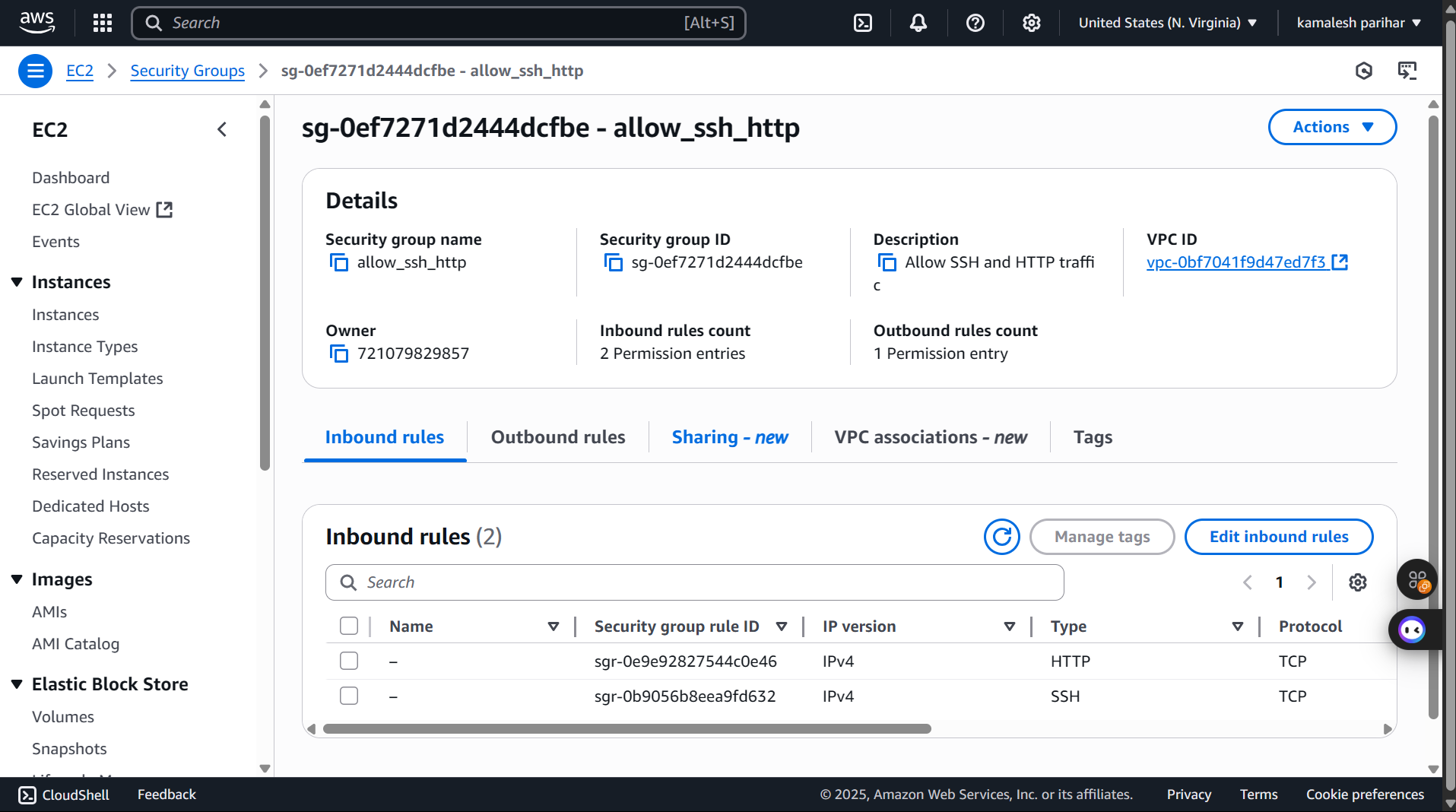
****

**Step 6 AWS infrastructure**

Running Ec2 Instance

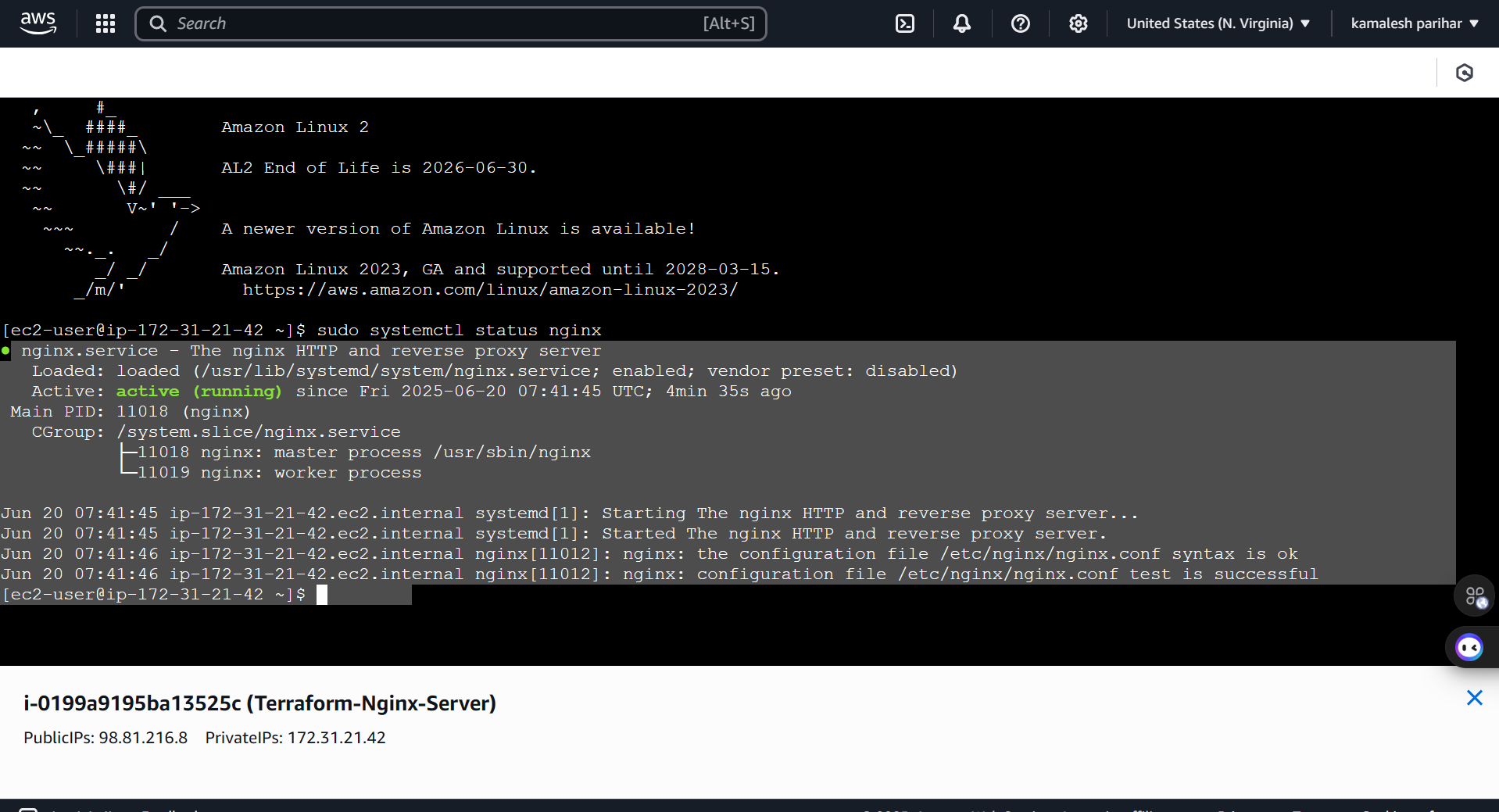


Security group details created with terraform



Ec2 instance connect with nginx server and run

sudo systemctl start nginx



Lastly vs code ss of all directories and files

