**Ayush Bhadouria**

**Roll no: A004**

**MCA**

A screenshot of a computer

Description automatically generated

A computer screen shot of a computer screen

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

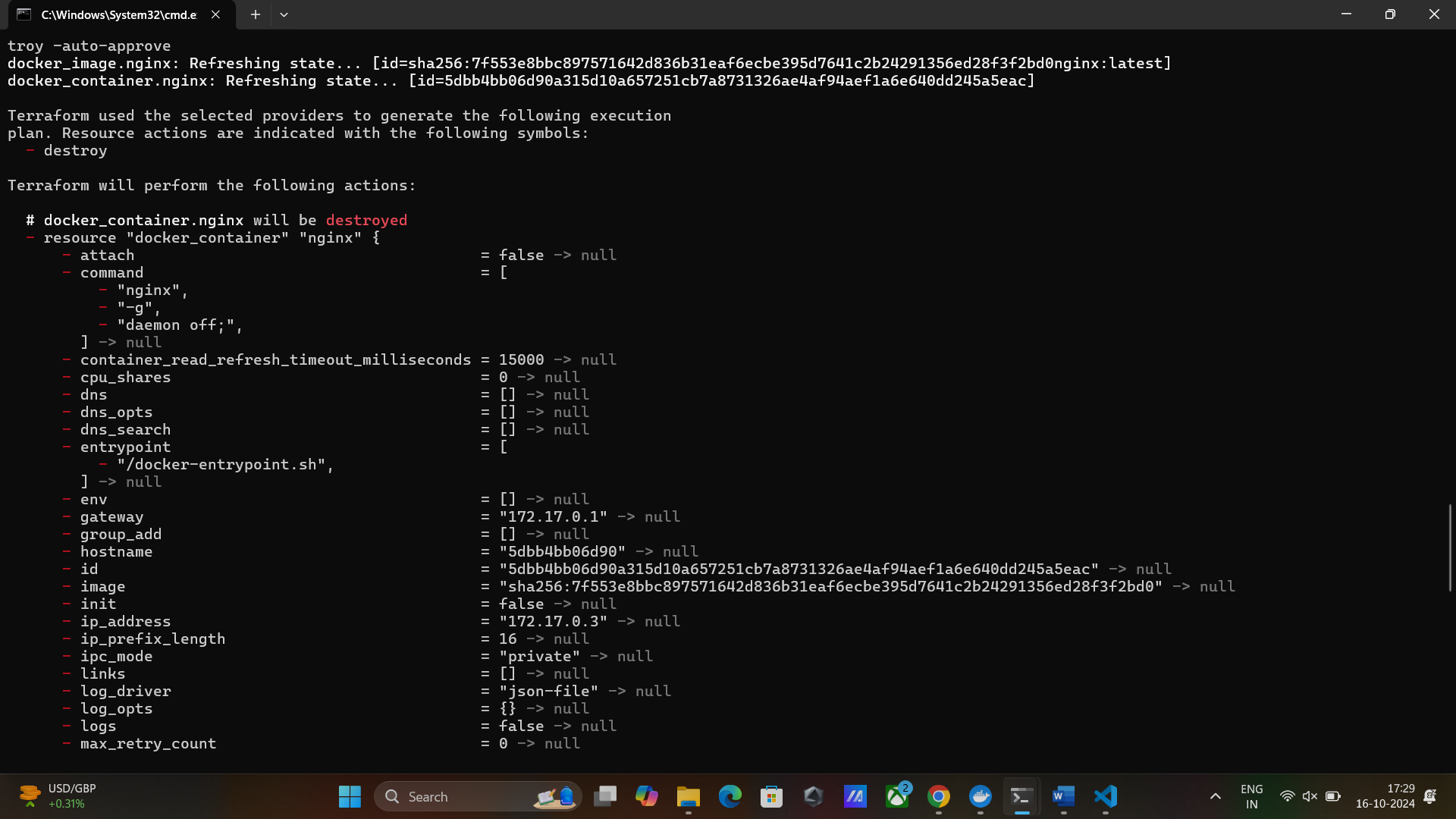
Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated



A screenshot of a computer screen

Description automatically generated

1. **Open Terminal or Command Prompt**:
   * On Linux/macOS, open your terminal.
   * On Windows, you can open Command Prompt or PowerShell.
2. **Run the Command**: Simply enter the following command in your terminal or command prompt to pull the Terraform Docker image:

docker pull hashicorp/terraform

This command pulls the Terraform Docker image from DockerHub to your local machine, allowing you to run Terraform inside a Docker container.

1. **Create a Local Directory for Terraform Configuration**: Create a directory to store your Terraform configuration files, which you’ll map to the Docker container.

mkdir terraform-nginx

cd terraform-nginx

1. **Write the Terraform Configuration**: Create a file named main.tf inside your terraform-nginx directory with the following content:

**main.tf:**

# Specify required providers

terraform {

required\_providers {

docker = {

source = "kreuzwerker/docker"

version = "~> 2.15"

}

}

}

# Configure the Docker provider

provider "docker" {

host = "unix:///var/run/docker.sock"

}

# Pull Nginx image from DockerHub

resource "docker\_image" "nginx" {

name = "nginx:latest"

keep\_locally = false

}

# Create an Nginx container

resource "docker\_container" "nginx" {

image = docker\_image.nginx.latest

name = "nginx\_terraform"

ports {

internal = 80

external = 8080

}

}

Explanation:

The required\_providers block tells Terraform where to get the docker provider and what version to use.

The provider block contains only the host argument to point Terraform to the Docker daemon running on your local machine.

1. **Run Docker with the Mounted Directory**: Now run the Terraform container with your local directory mounted. This way, Terraform will have access to your configuration files and Docker.

docker run -v C:/Users/ayush/Desktop/Devops:/app -w /app -v/var/run/docker.sock:/var/run/docker.sock hashicorp/terraform init

1. **Apply the Terraform Configuration**: After initialization, apply the configuration to create the Nginx container:

**docker run -v C:/Users/ayush/Desktop/Devops:/app -w /app -v //var/run/docker.sock:/var/run/docker.sock hashicorp/terraform apply -auto-approve**

1. **Access Nginx:**

Once the container is running, open your browser and go to http://localhost:8080. You should see the Nginx welcome page.

1. **Clean Up:** To destroy the container created by Terraform.

**docker run -v C:/Users/ayush/Desktop/Devops:/app -w /app -v //var/run/docker.sock:/var/run/docker.sock hashicorp/terraform destroy -auto-approve**

**#Note:**

1. **Replace C:/Users/ayush/Desktop/Devops with the full path to your working directory.**
2. **Ensure you use //var/run/docker.sock for the Docker socket on Windows.**