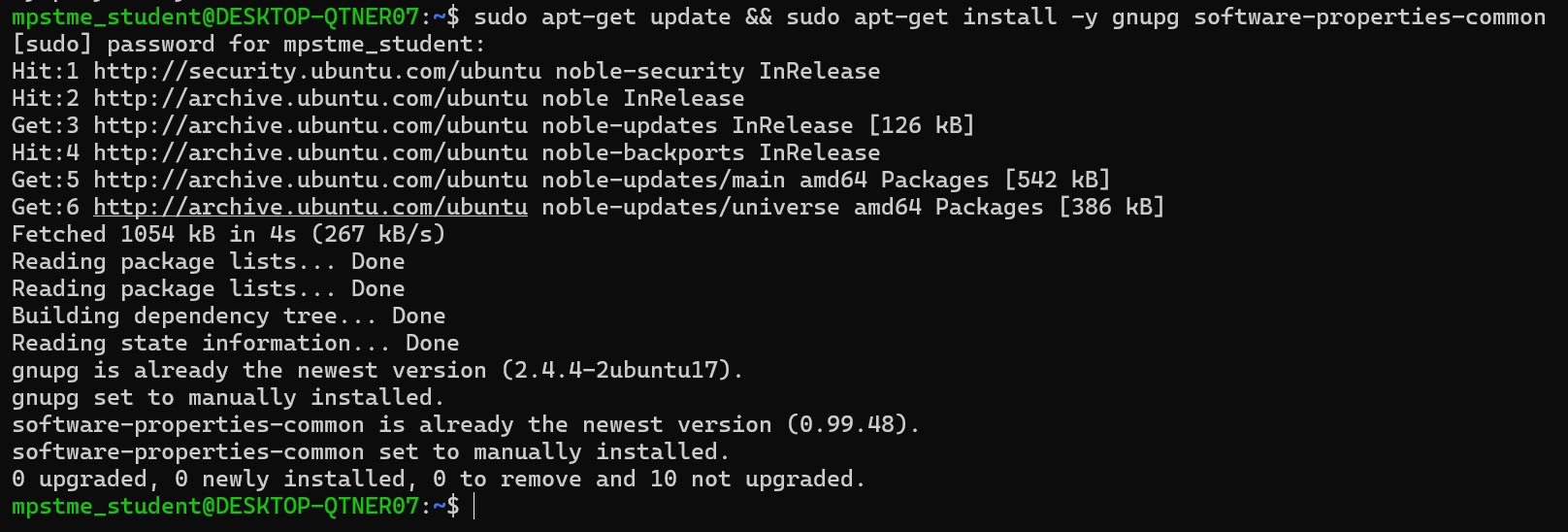
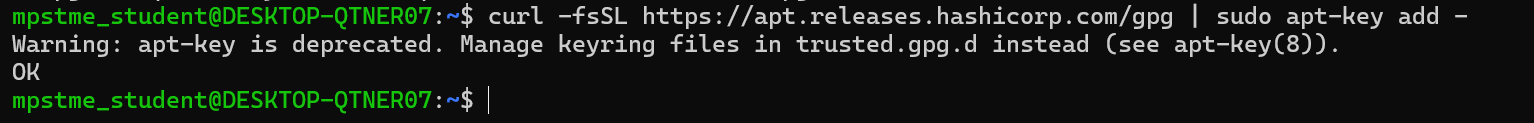
**Installing Terraform and Using it to Manage Nginx Server on Docker Container**

Similar to Ansible, we need WSL Ubuntu machine on which Terraform will be installed and Docker container on Windows to simulate the cloud infrastructure.

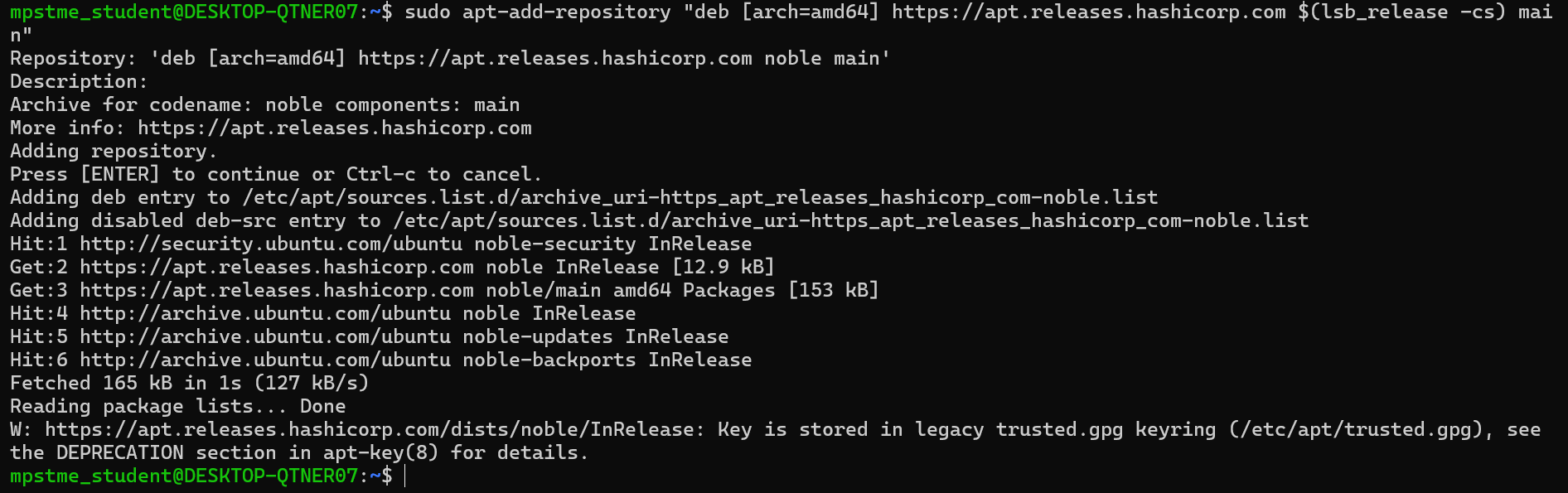
1. On WSL Ubuntu, type: sudo apt-get update && sudo apt-get install -y gnupg software-properties-common , to update package repo



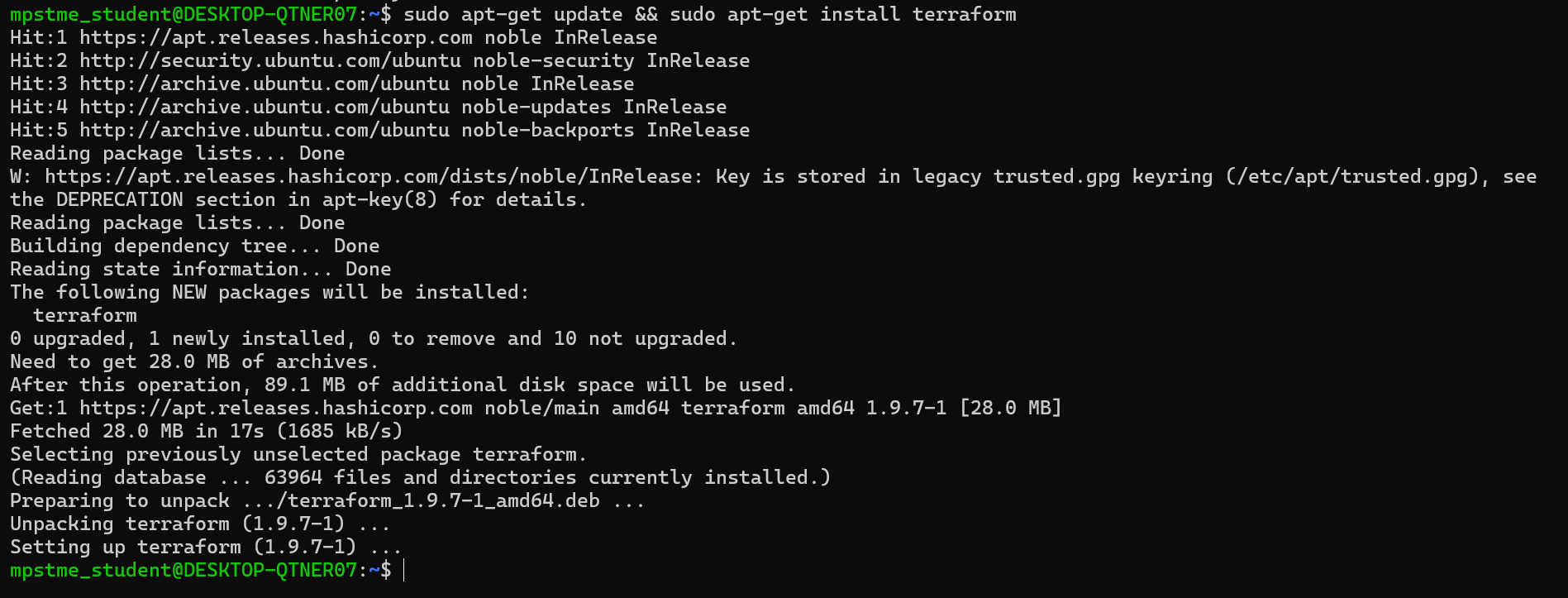
1. Install HashiCorp GPG key: curl -fsSL https://apt.releases.hashicorp.com/gpg | sudo apt-key add –



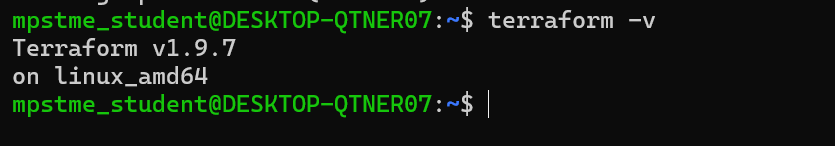
1. Add official HashiCorp Linux Repo: sudo apt-add-repository "deb [arch=amd64] https://apt.releases.hashicorp.com $(lsb\_release -cs) main"



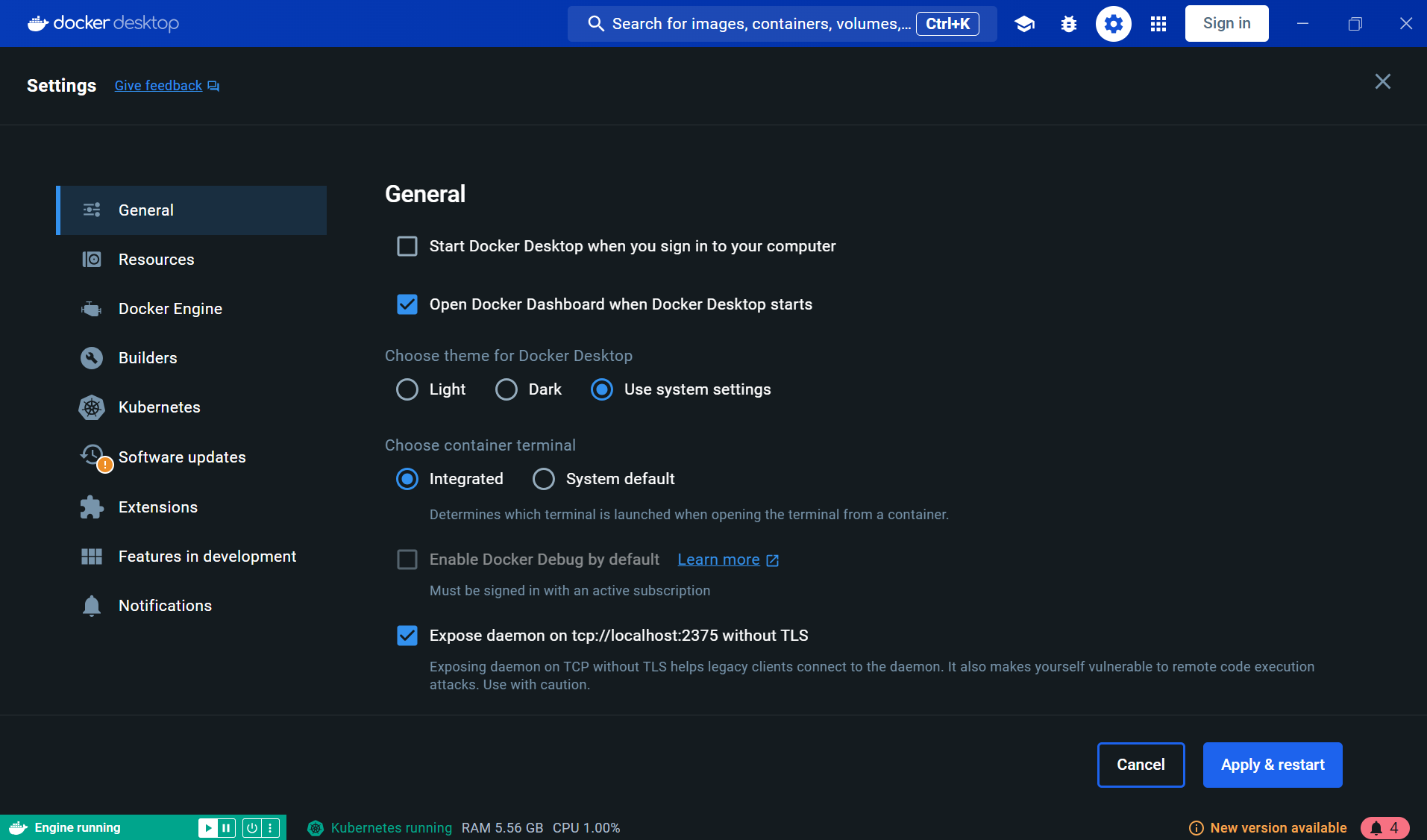
1. Install Terraform: sudo apt-get update && sudo apt-get install terraform



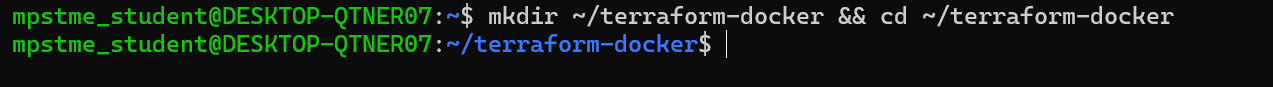
1. Check if Terraform is properly installed: terraform -v



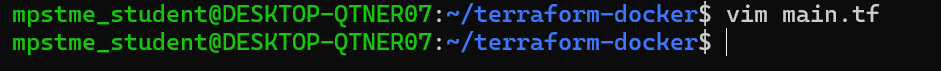
1. Terraform has been successfully installed. Now to configure Docker Desktop to integrate with WSL environment.
2. Open Docker Desktop -> Settings -> General -> **Expose daemon on tcp://localhost:2375 without TLS**. -> Check the box to expose Docker via TCP -> Apply and Restart



1. To have Terraform work with Docker containers, we need to setup Terraform Docker Provider
2. Create new directory inside WSL Ubuntu



1. Create a file: vim main.tf



1. Paste this content:

terraform {

required\_providers {

docker = {

source = "kreuzwerker/docker"

version = "~> 2.21.0"

}

}

}

provider "docker" {

host = "tcp://localhost:2375"

}

resource "docker\_image" "nginx" {

name = "nginx:latest"

}

resource "docker\_container" "nginx" {

image = docker\_image.nginx.latest

name = "nginx\_container"

ports {

internal = 80

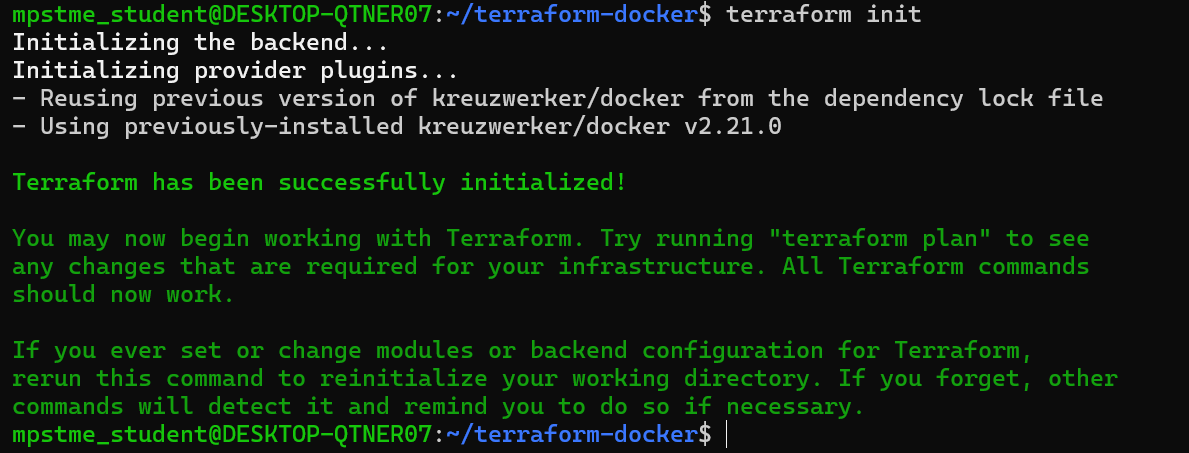
external = 8080

}

}



1. Press ESC and then type :wq to save and exit vim
2. Initialize Terraform project: terraform init



1. Now to make Terraform create the Docker container with nginx: terraform apply