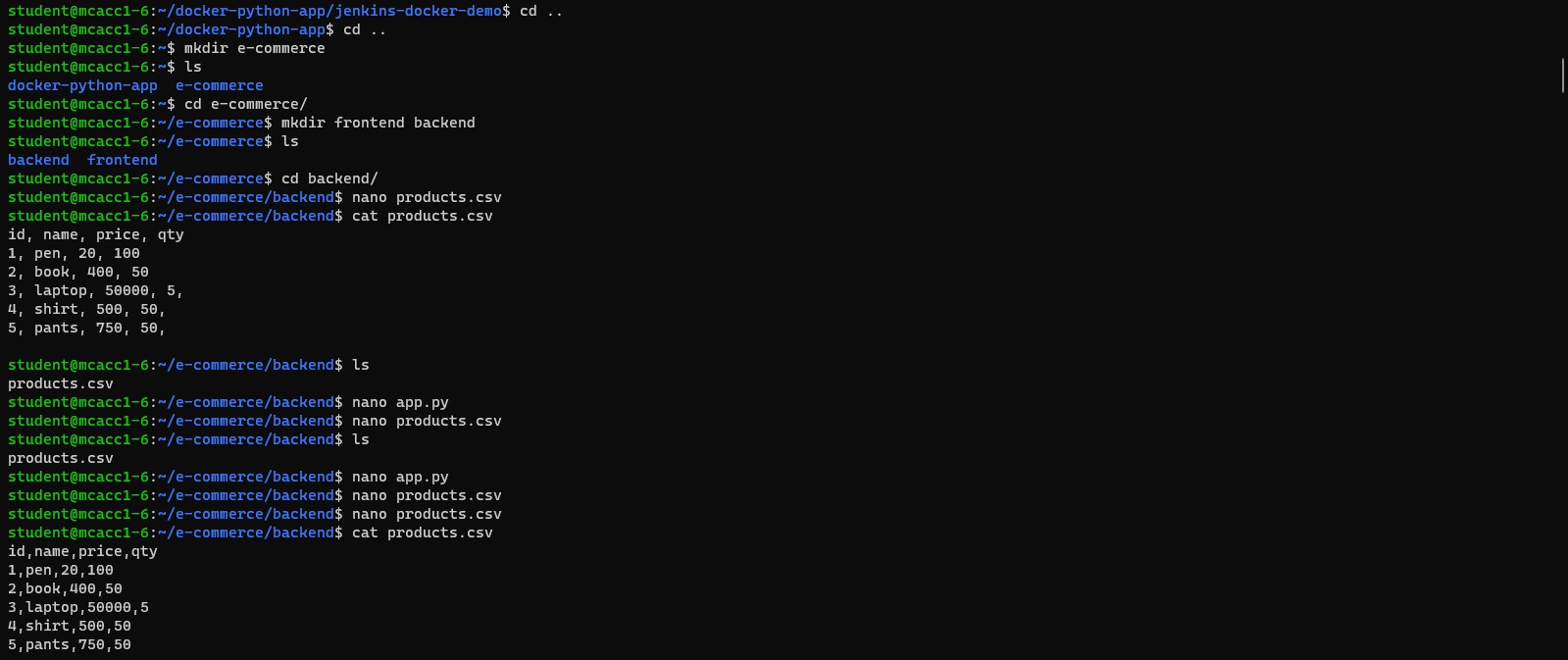
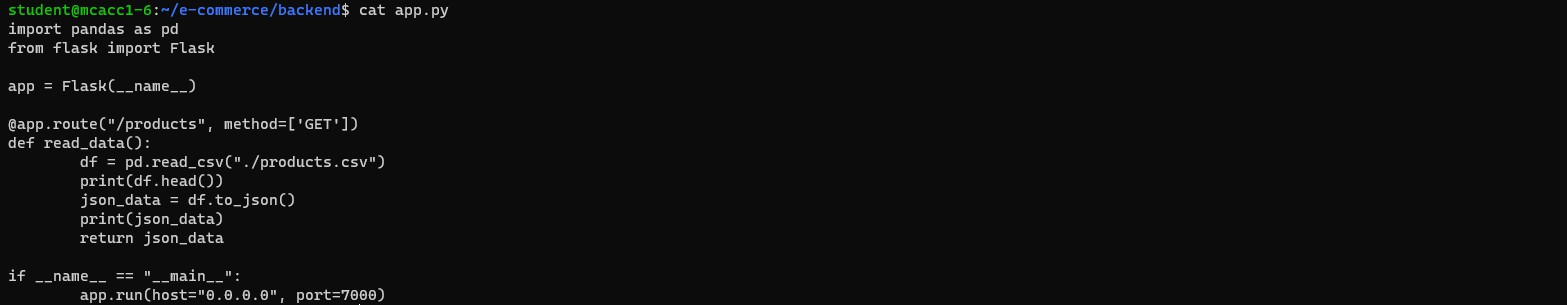
NAME:GOWTHAM[24MCR028]

Day 4: Kubernetes

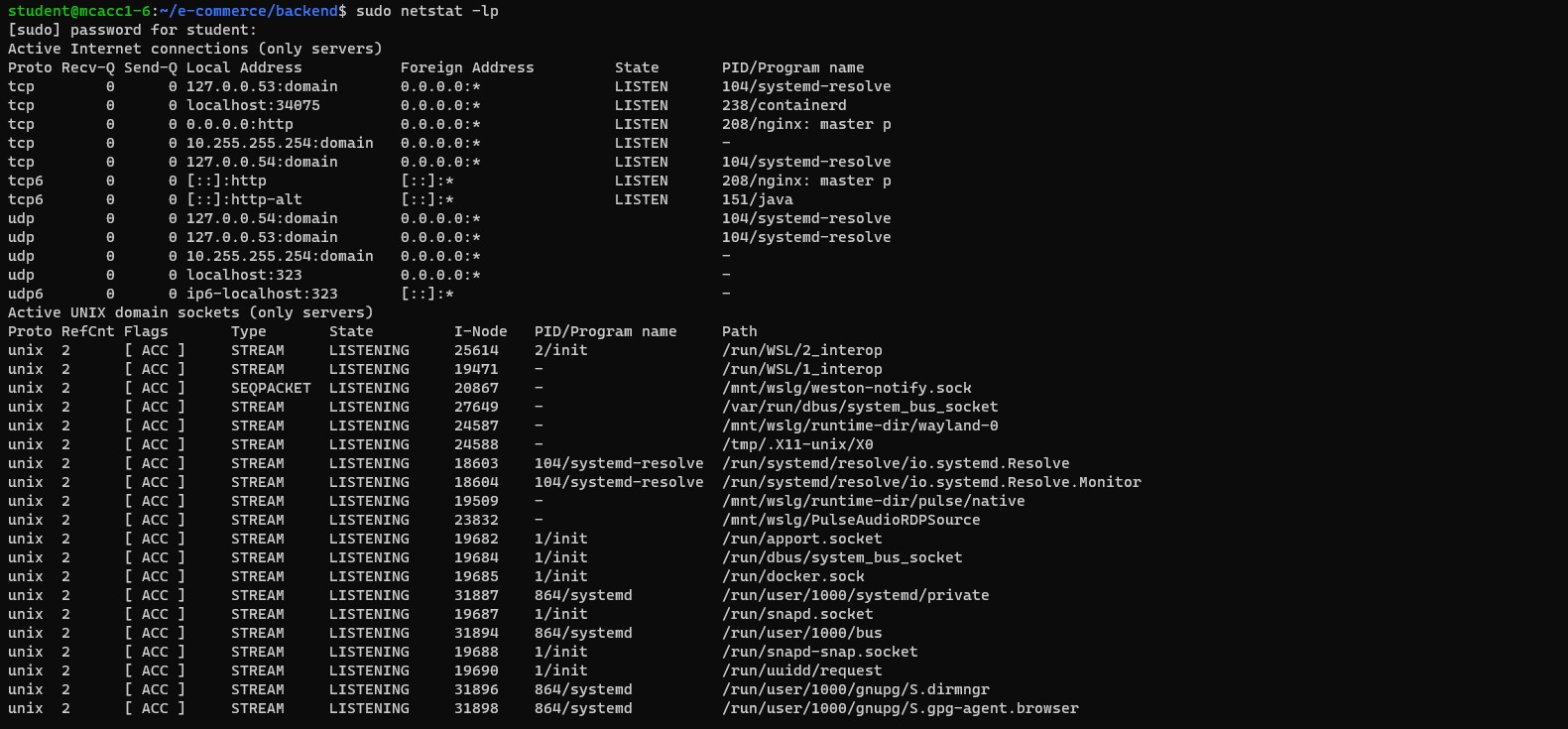
# Create a directory ‘e-commerce’ and its required folders and files

****Create a **products.csv** file and **app.py**

Install the **pandas** library:



Ensure that the CSV file is read and correctly parsed into **JSON format**.

****To verify the **available port numbers**

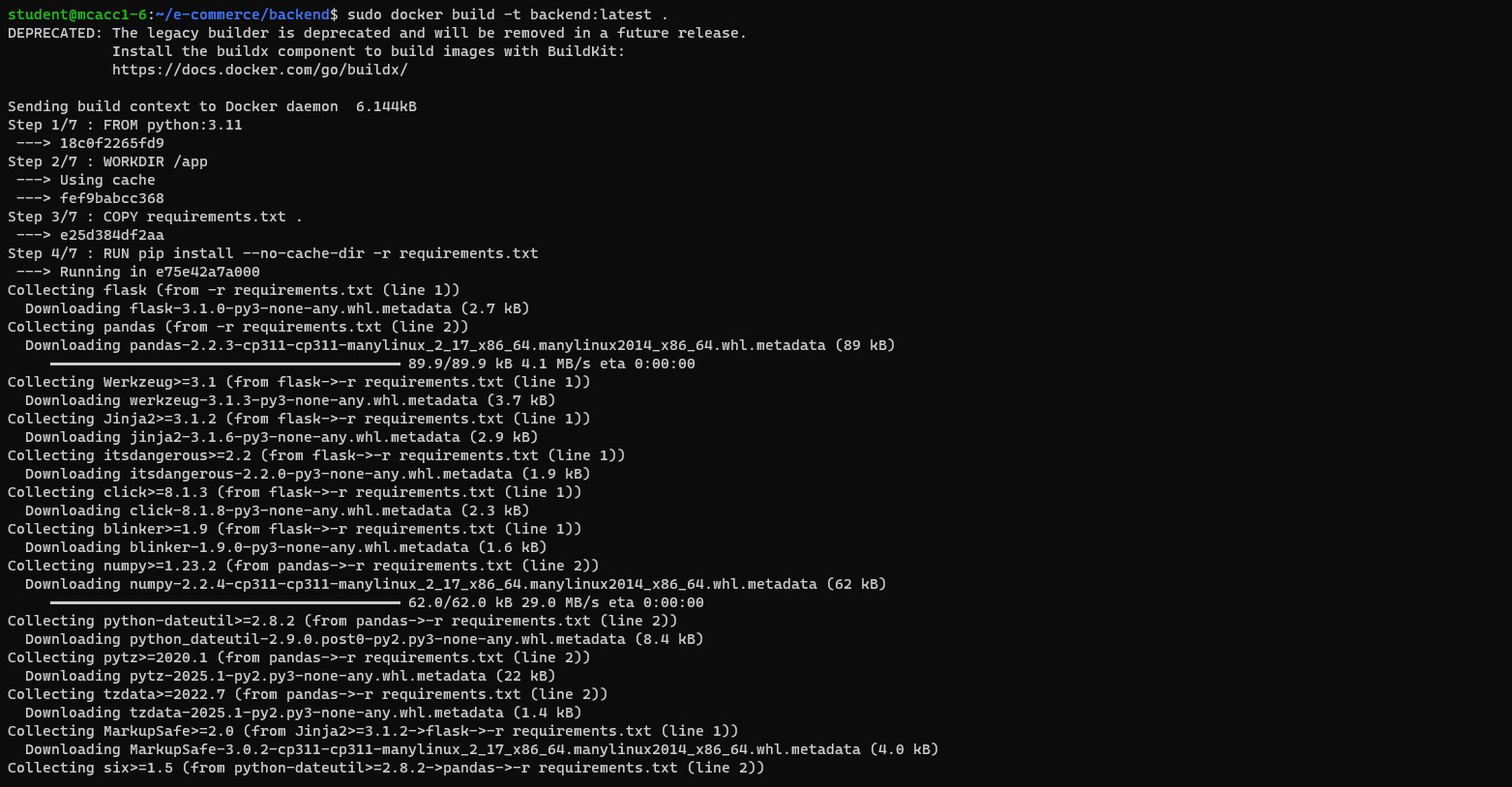
Create **requirements.txt** file

The requirements.txt file is used in Python projects to list all the dependencies (packages) that the application needs to run.

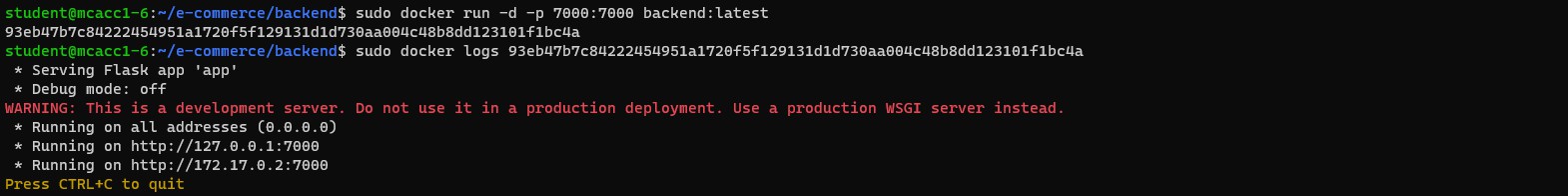
Create **docker-compose.yml** file

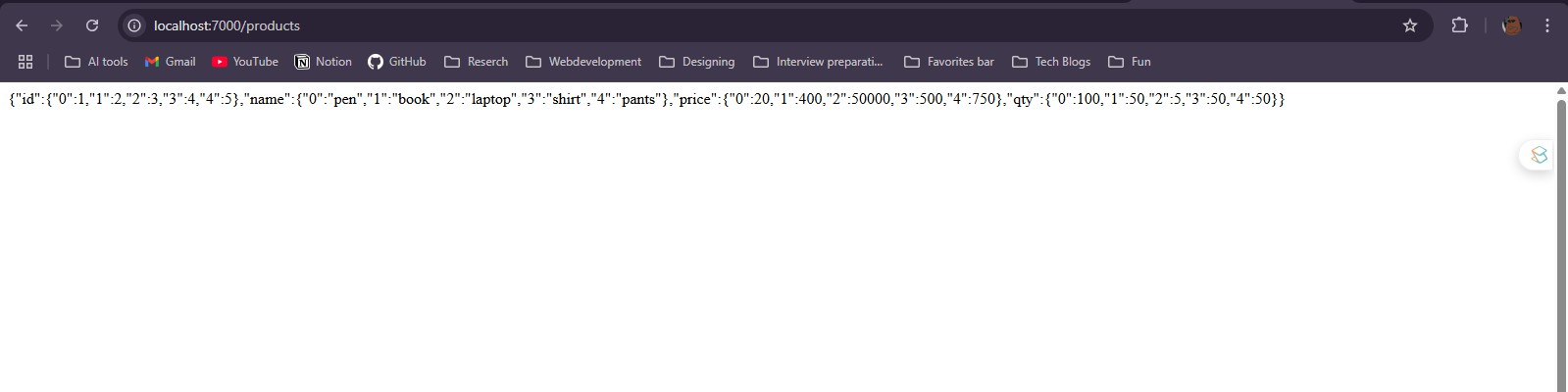
docker-compose.yml is a YAML configuration file used to define and run multi-container Docker applications.

## Build Docker image

Sudo docker build -t backend:latest

Run the docker:

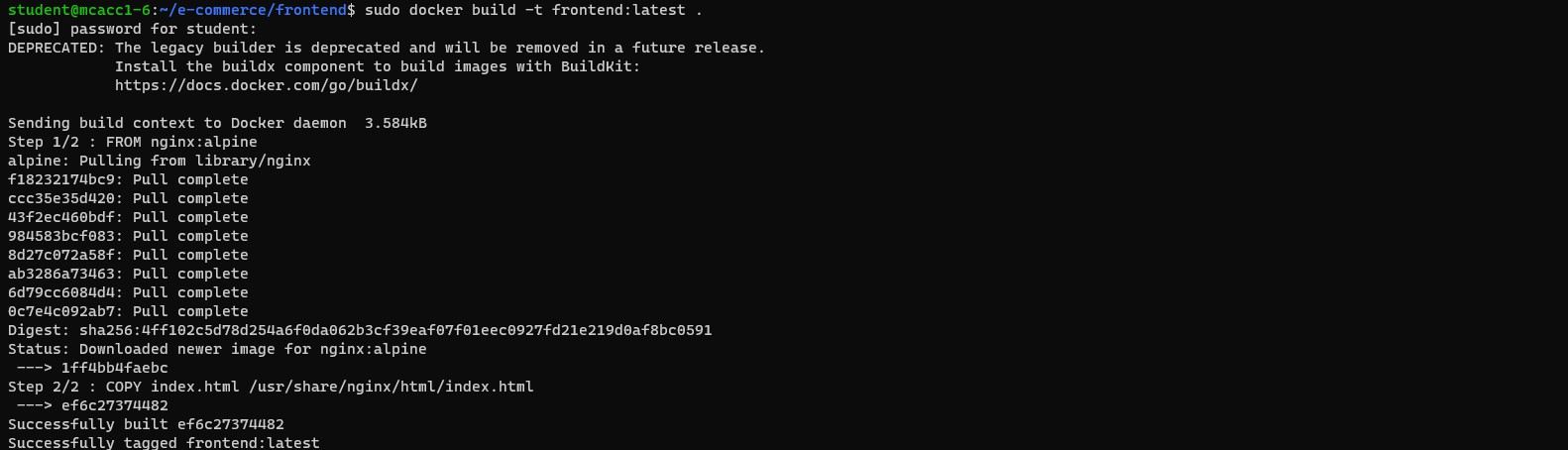
sudo docker run -d -p 7000:7000 backend:latest sudo docker logs <Generated number>

Run the application in the 7000/products

The JSON data is displayed at our port: 7000/products.

## Create a container in frontend

****Create **index.html file** and **Dockerfile**

**Build the image** using the command: sudo docker build -t frontend:latest.

# Kubernetes Deployment YAML Files

Create **backend-deployment.yaml file** and **frontend-deployment.yaml** in a folder k8s These files define how our application should be deployed in the cluster.

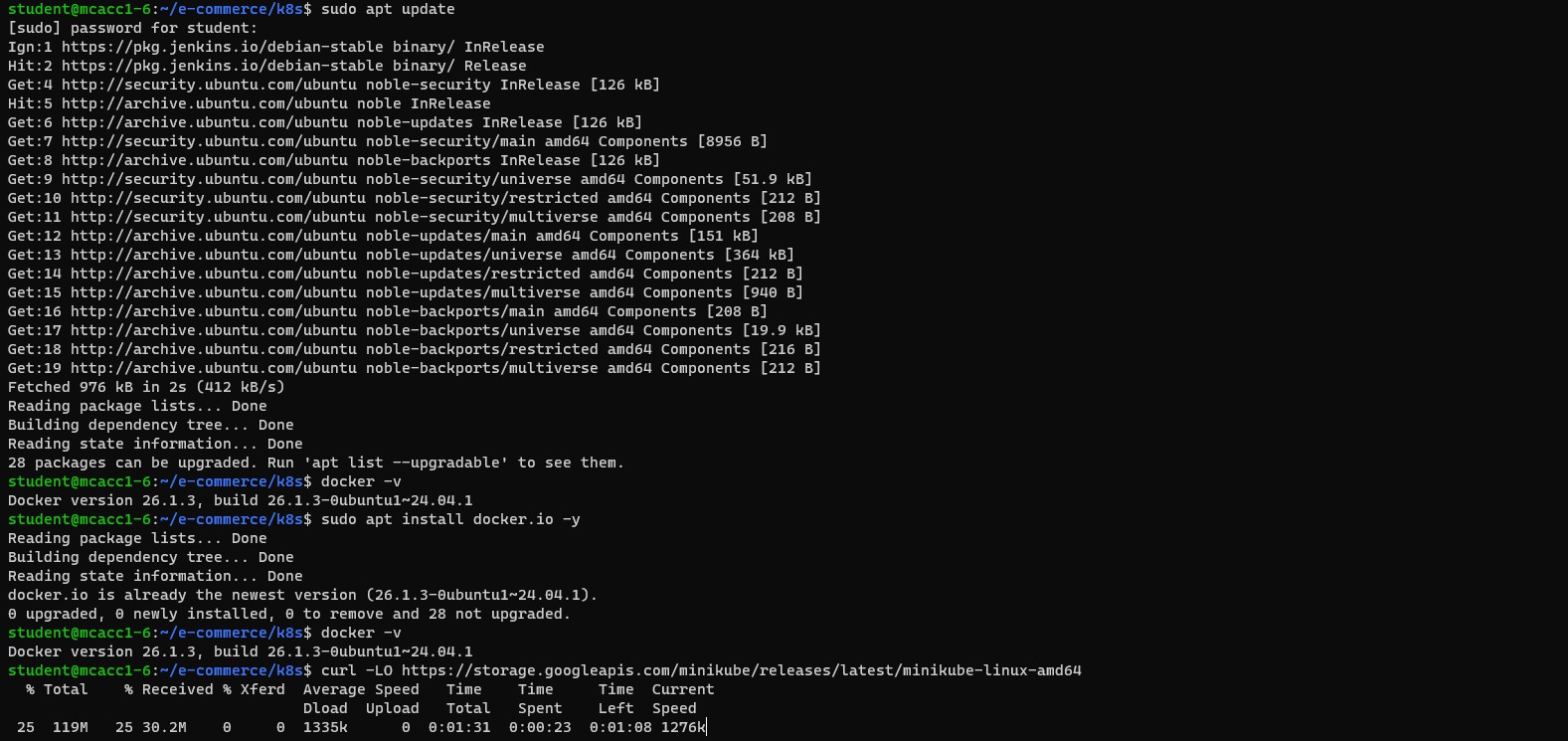
Create **service.yaml** file

It exposes our application within or outside the cluster.

Create **configmap.yaml** file

Stores configuration data as key-value pairs.

Install minikube

Minikube is a tool that allows you to run a Kubernetes cluster locally on our machine. It is designed for developers who want to test and experiment with Kubernetes without needing a full-scale cloud-based cluster.

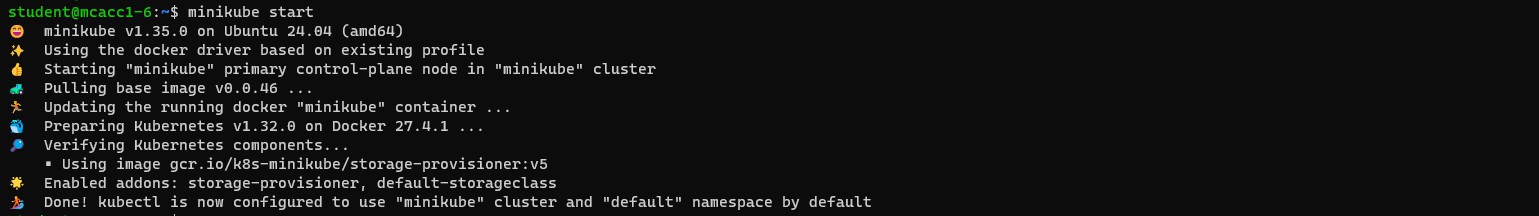
Install kubectl

kubectl is the command-line tool used to interact with a Kubernetes cluster. It allows you to deploy applications, inspect and manage cluster resources, and troubleshoot issues.

Grant permission for **kubectl chmod +x kubectl**

Move to kubectl to root

Check the **minikube** and **kubectl** installed properly

****Start minicube: **minikube start**

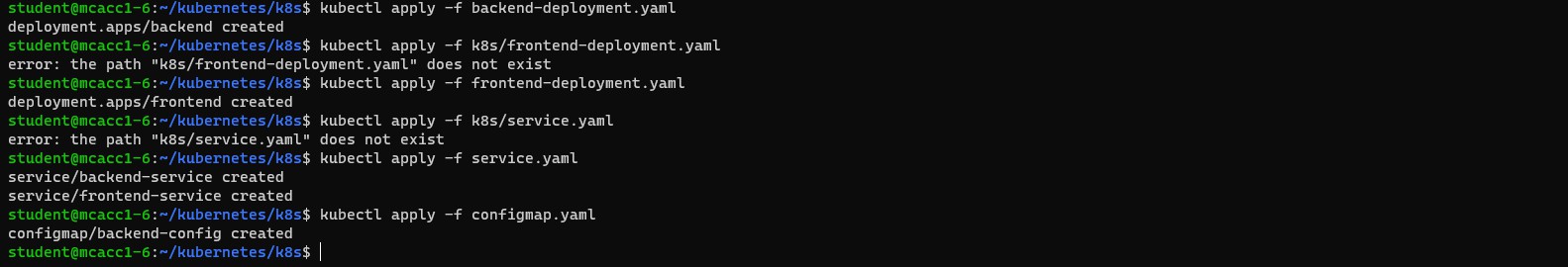
**Verify minikube is running**

**Load the image to the minikube Befor loading images**

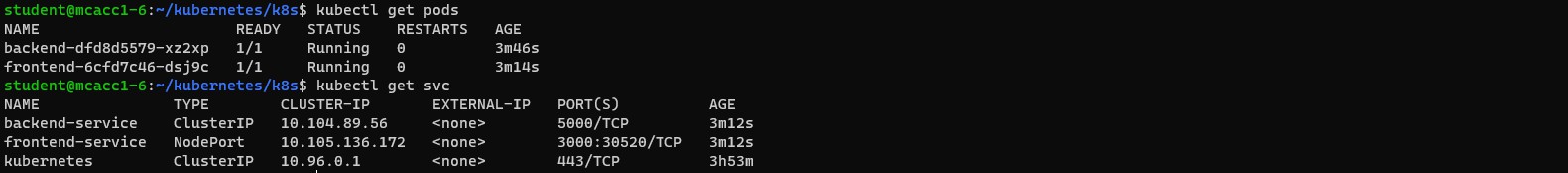
**Perform this commend:** eval $(minikube docker-env) minikube image load frontend:latest

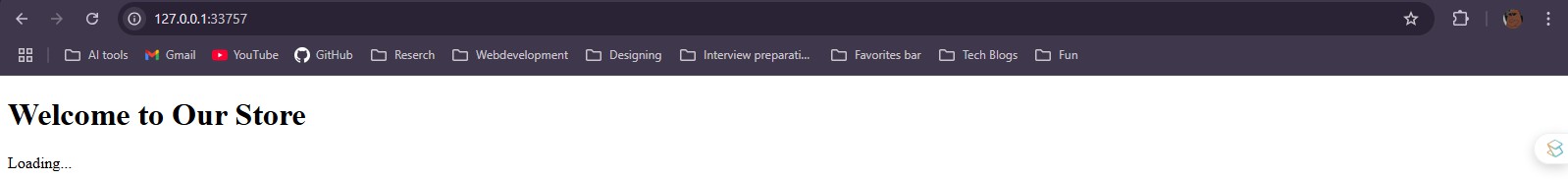
minikube image load backend:latest

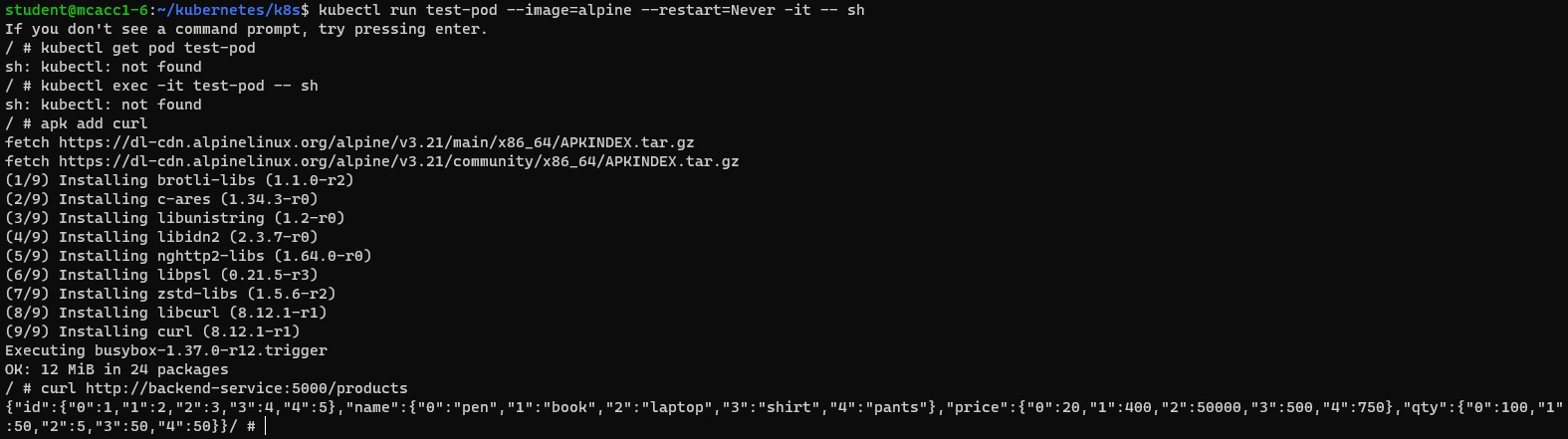
**Check the images are loaded**

Commands are used to deploy your application components (backend and frontend), expose them through a service, and provide them with the necessary configuration via a ConfigMap.

These commands are used to list and inspect the running resources in your Kubernetes cluster:

**kubectl get pods kubectl get svc**

**To test Frontend**

**To Test backend**

