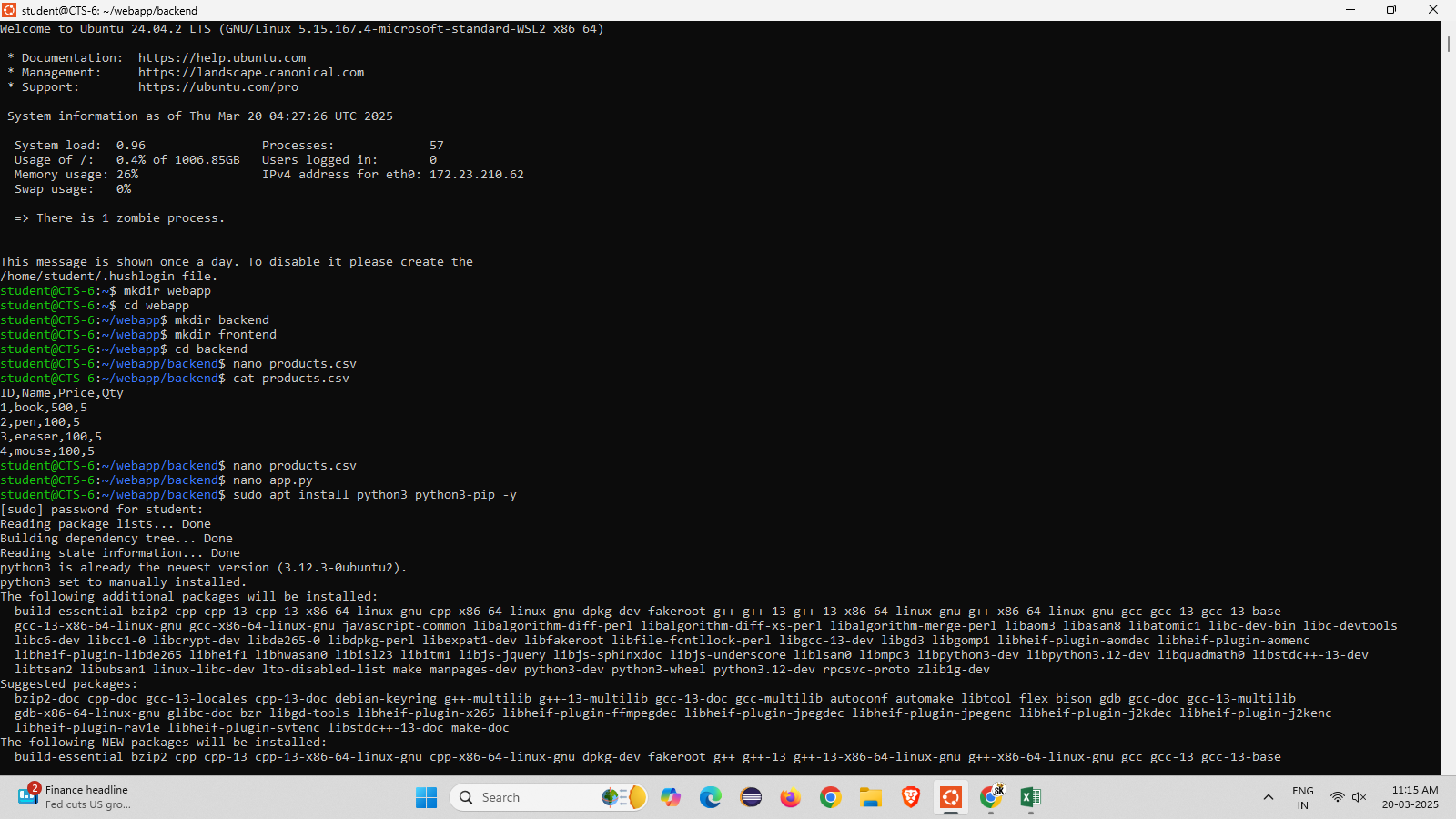
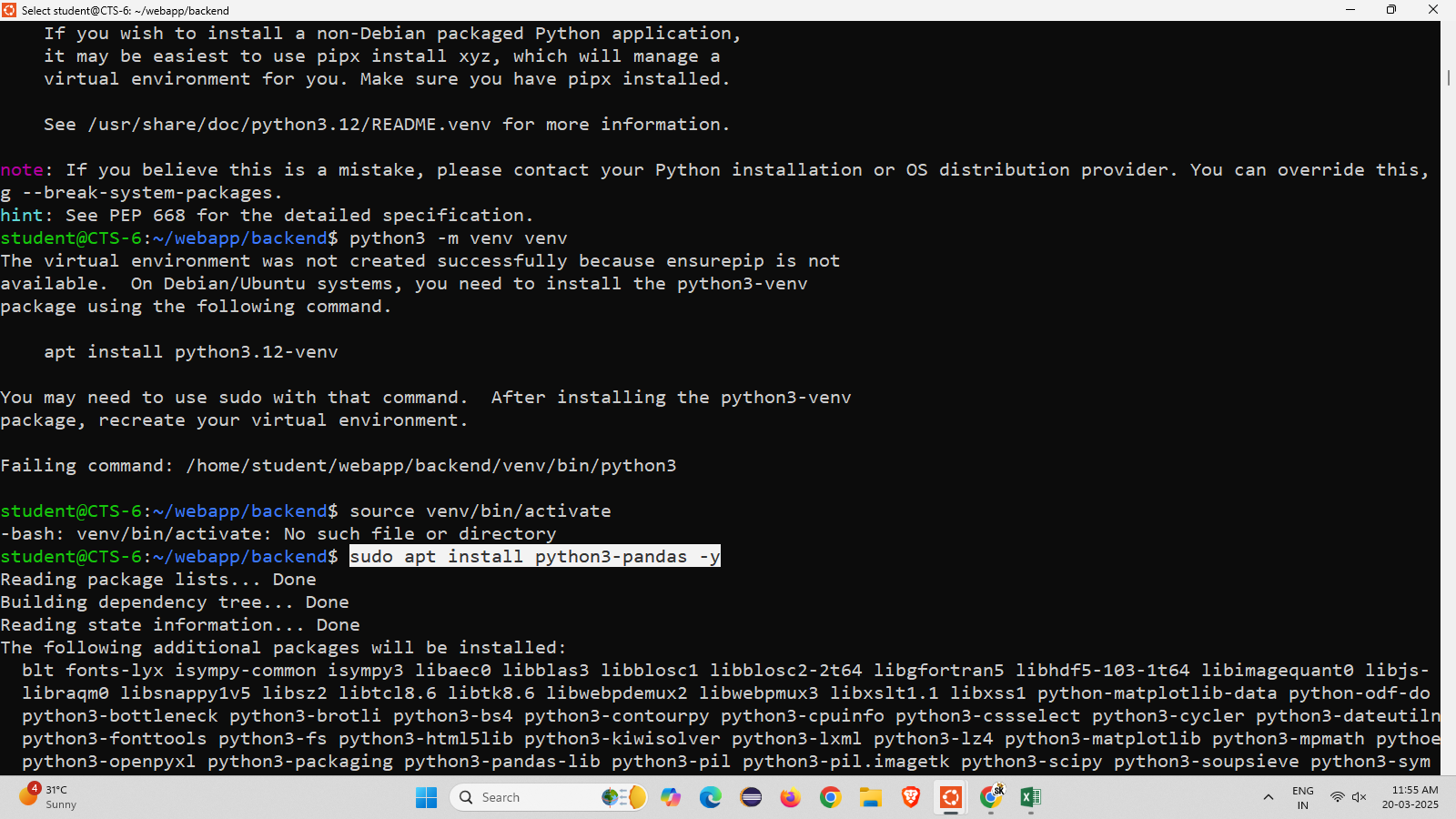
**DEVOPS TRAINING**

**DAY 3 & 4**

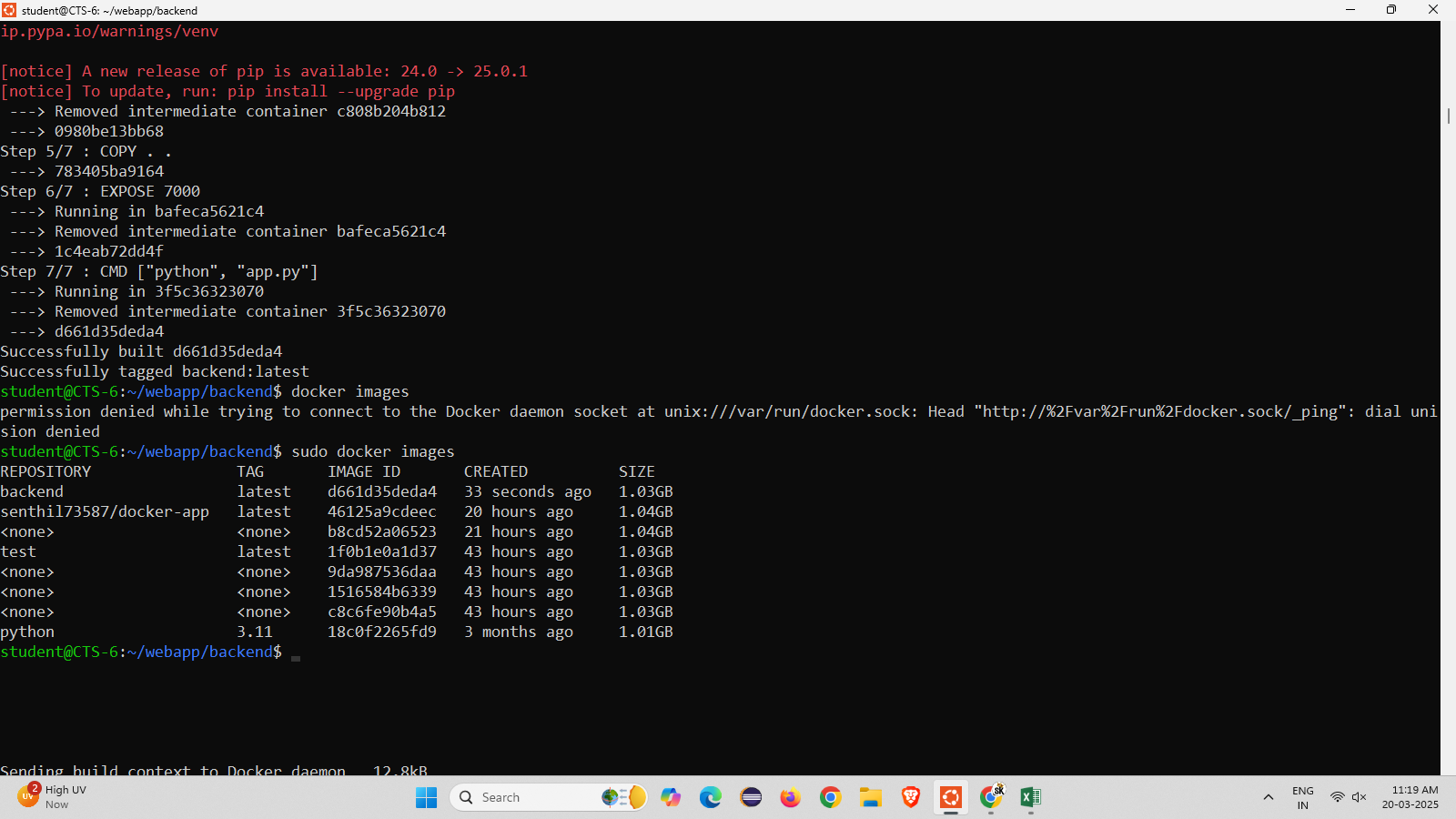
Step 1: create the directory as webapp and make the directory frontend and backend and create a nano products.csv file and check in excel whether the format is correct



Step 2: install the python3-pandas-y libraries into it



Step 3: using sudo docker images command to view list



**App.py**

from flask import Flask

import pandas as pd

app = Flask(\_\_name\_\_)

@app.route("/products", methods=['GET']) # ✅ Fixed 'methods' issue

def read\_data():

df = pd.read\_csv("products.csv")

json\_data = df.to\_json()

return json\_data

if \_\_name\_\_ == "\_\_main\_\_":

app.run(host="0.0.0.0", port=7000)

**Docker File**

FROM python:3.11

WORKDIR /app

COPY requirements.txt . # Ensure pandas is in requirements.txt

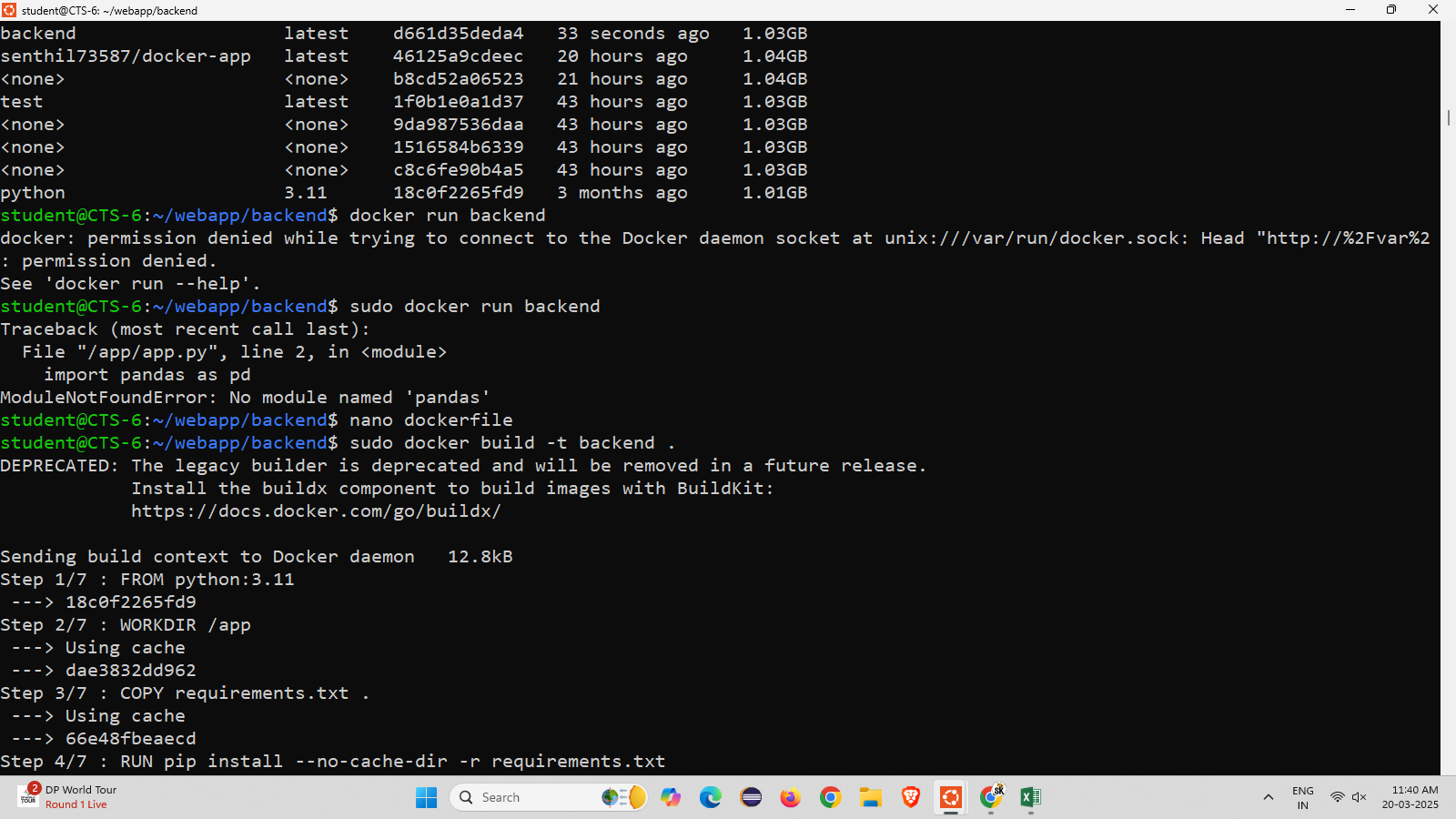
RUN pip install --no-cache-dir -r requirements.txt

COPY . .

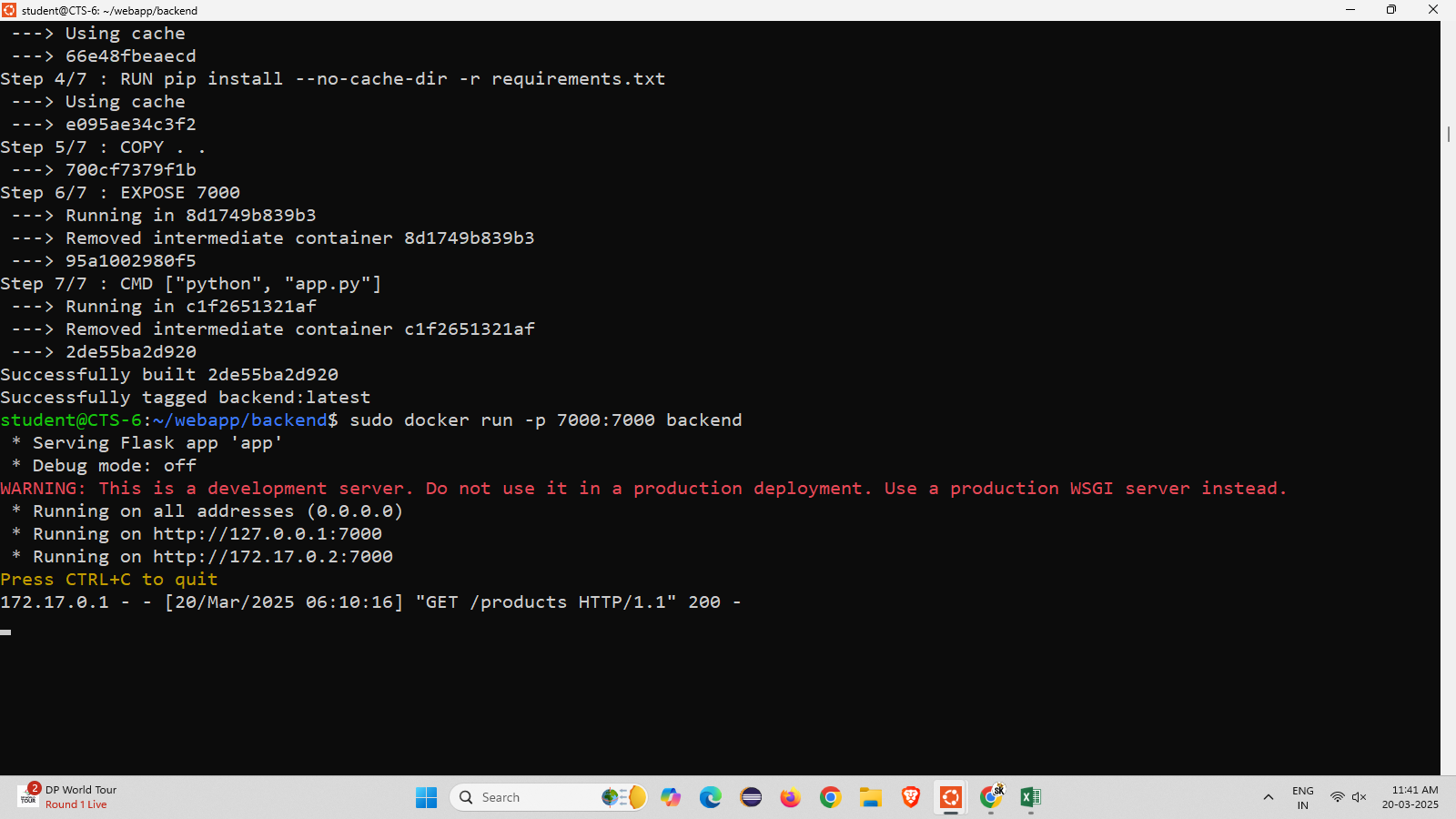
EXPOSE 7000

CMD ["python", "app.py"]

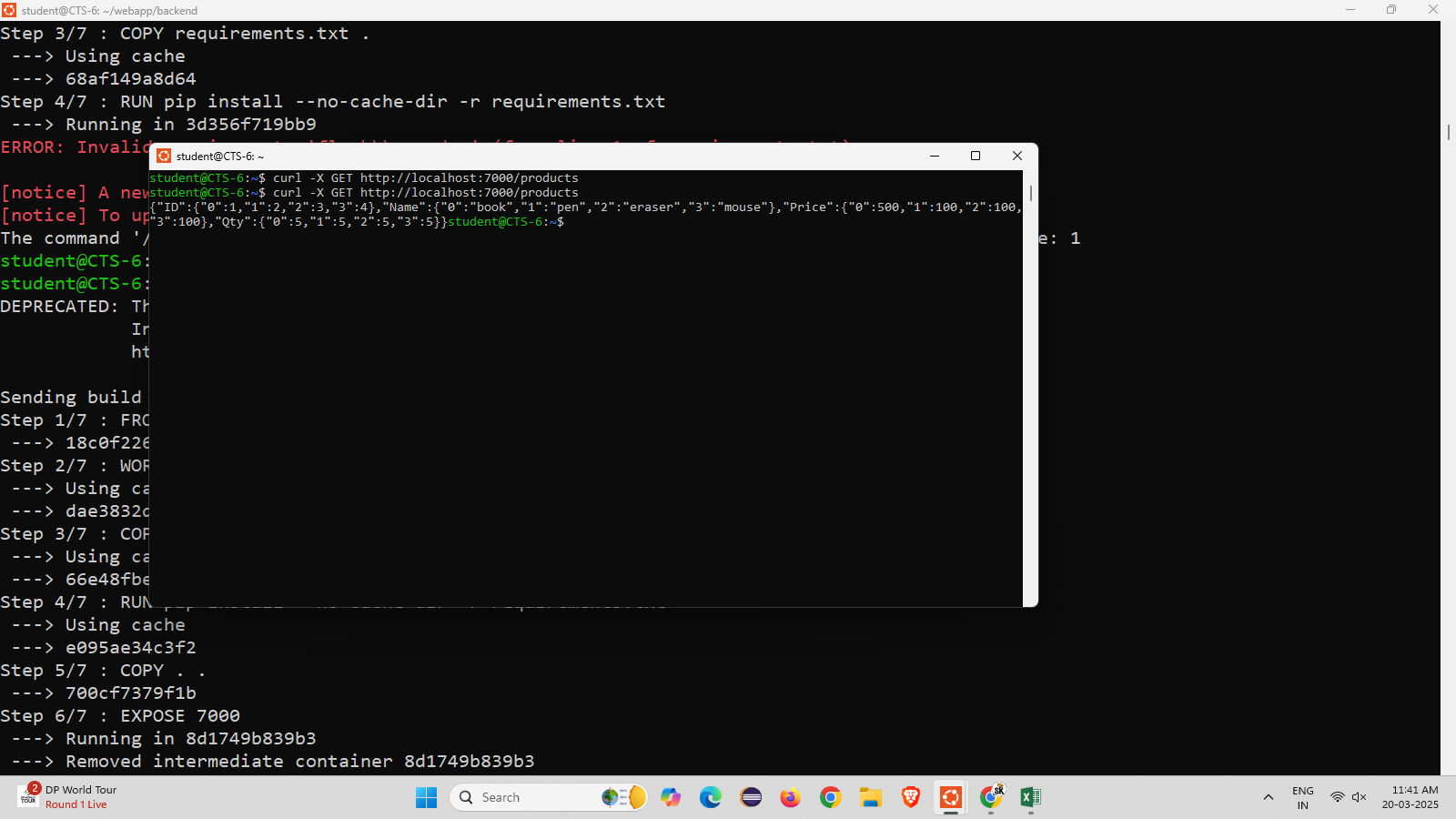
Step 4 : nano dockerfile and nano docker built –t backend . to built the docker



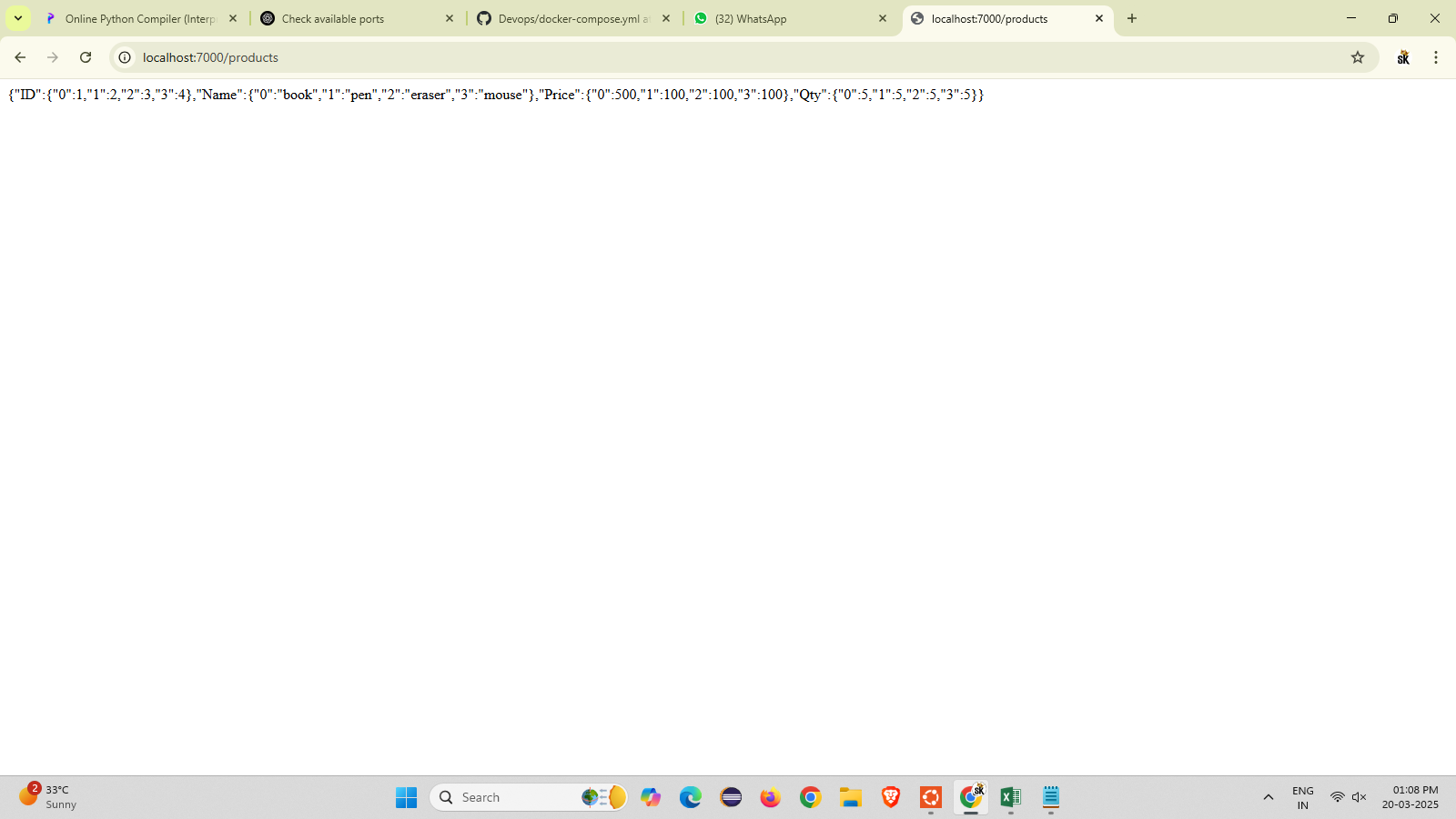
Step 5 : using sudo docker run –p 7000:7000 backend command to run the backend



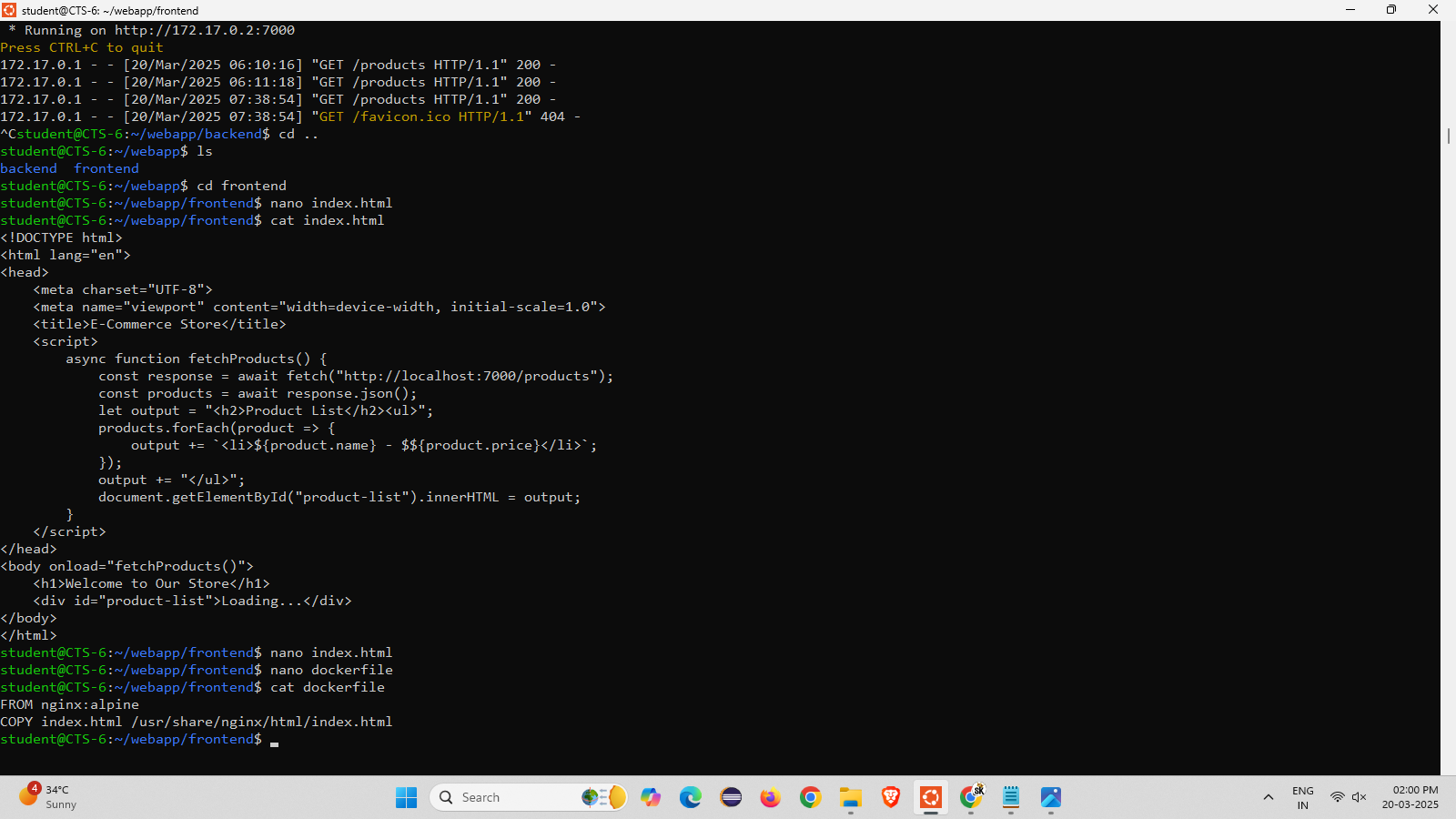
Step 6: open new Ubuntu and run as administration and enter the command as curl –X GET <http://localhost:7000/products>



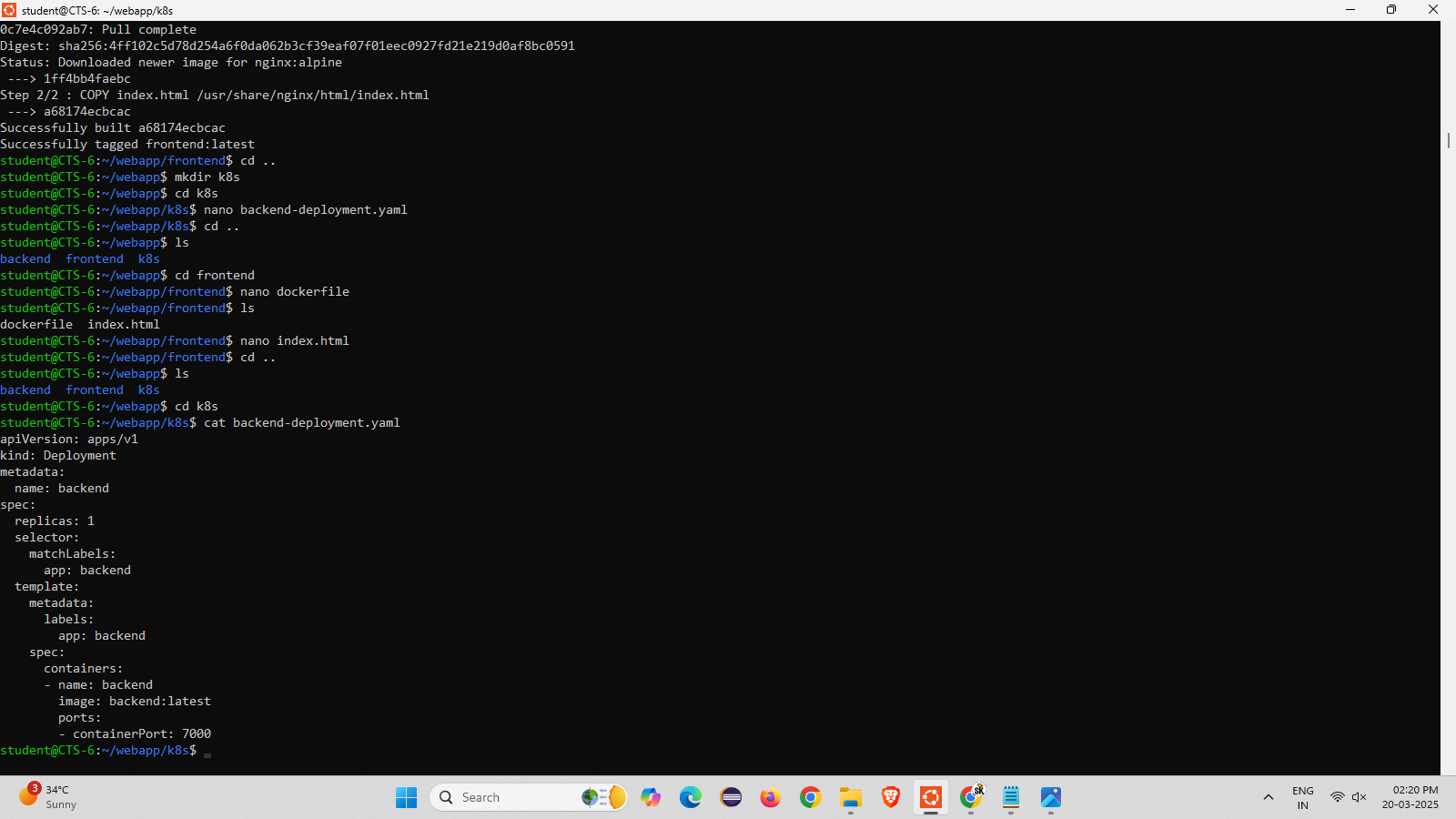
Step 7: go to the browser and enter the url it displays the backend the backend



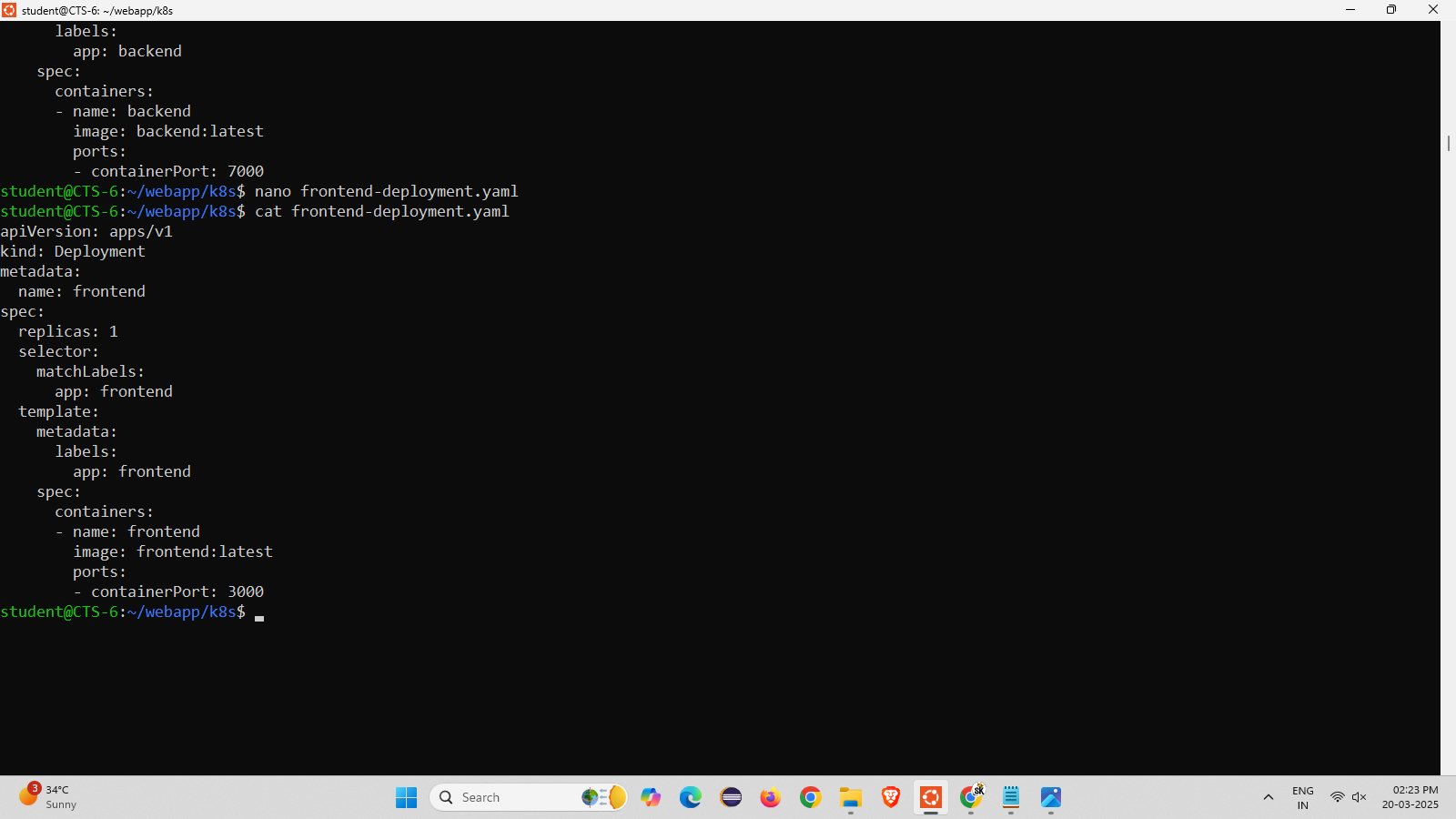
Step 8: Change the repository to frontend and create a nano index.html file and enter the code



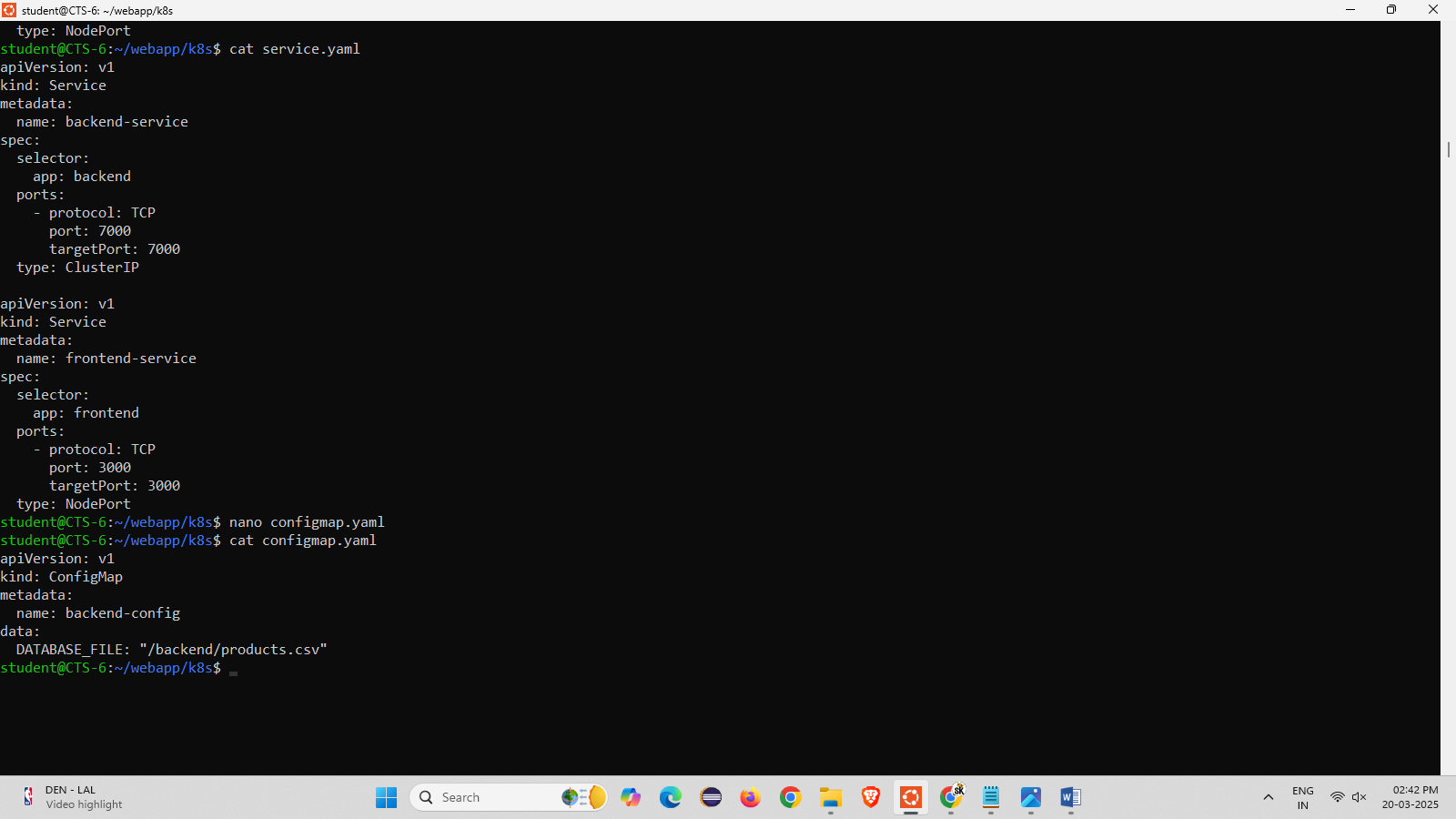
Step 9: change the directory and and make the directory k8s in that create a nano backend-deployment.yaml and add the code



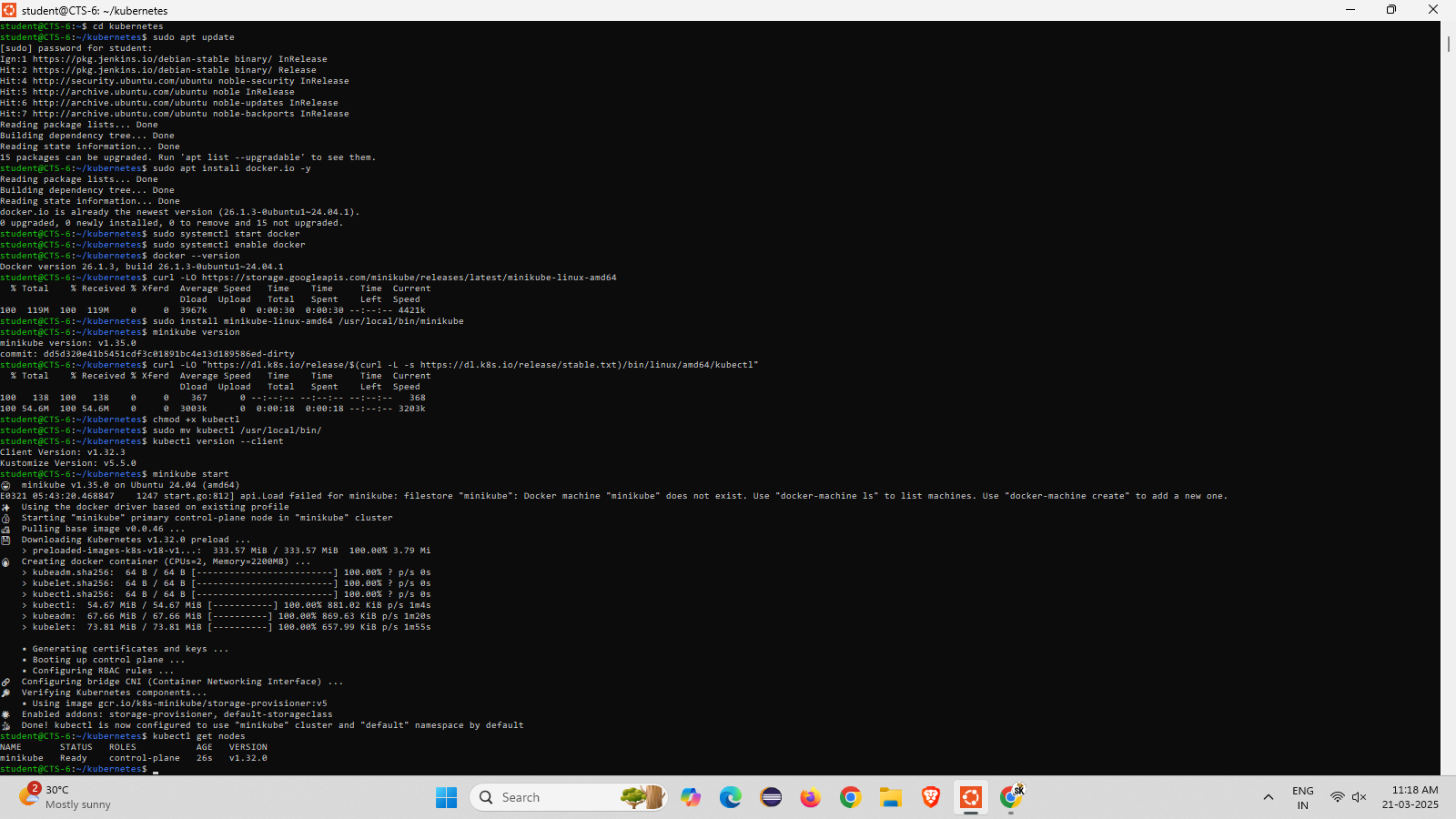
Step 10: create another nano frontend-deployment.yaml file and add the code



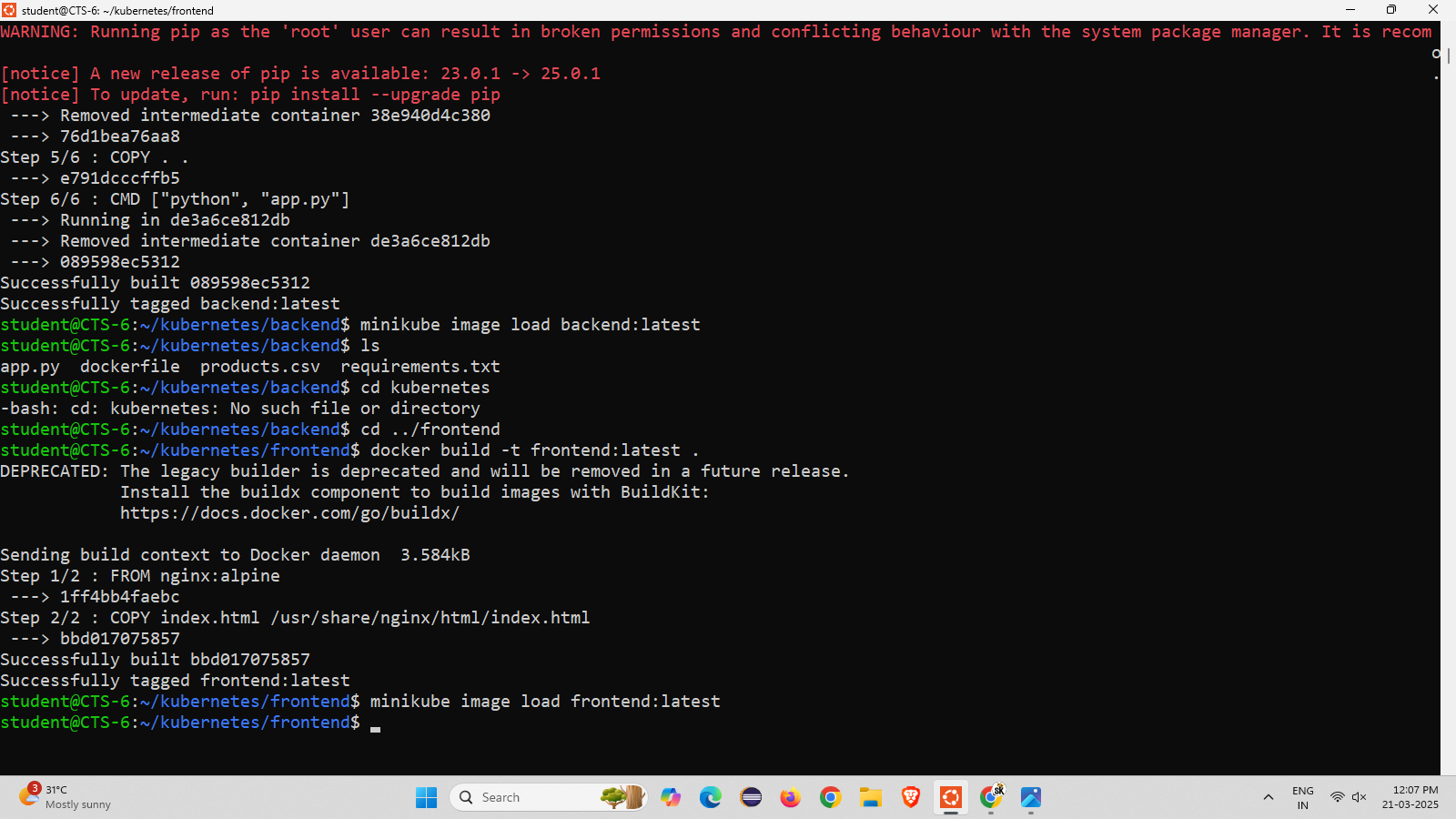
Step 11: create another nano service.yaml and configmap.yaml file and add the code



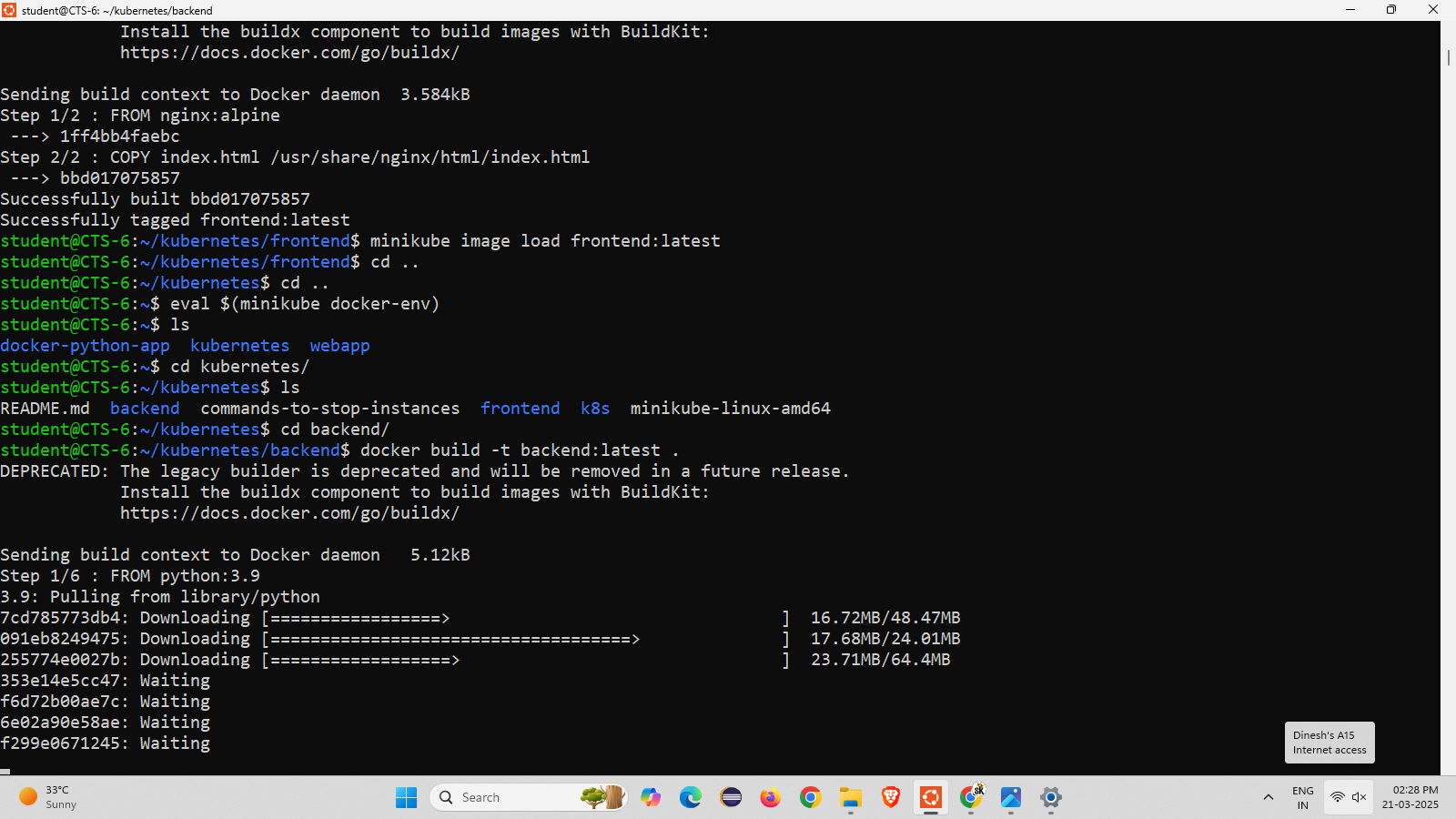
Step 12: clone the kubernetes github repository and run the following commands



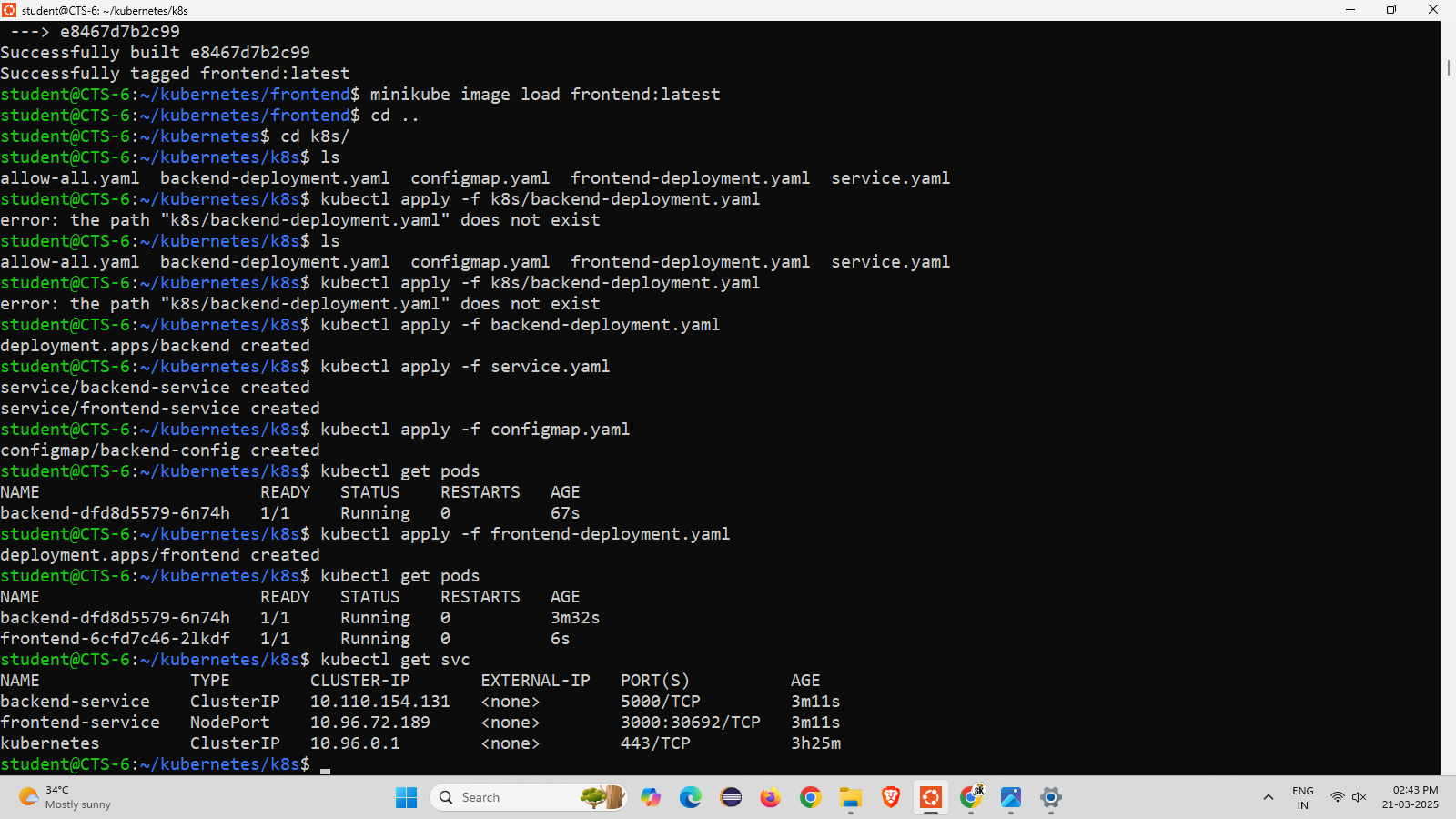
Step 13 : go to the kubernetes backend and frontend directory and type the command to load an image



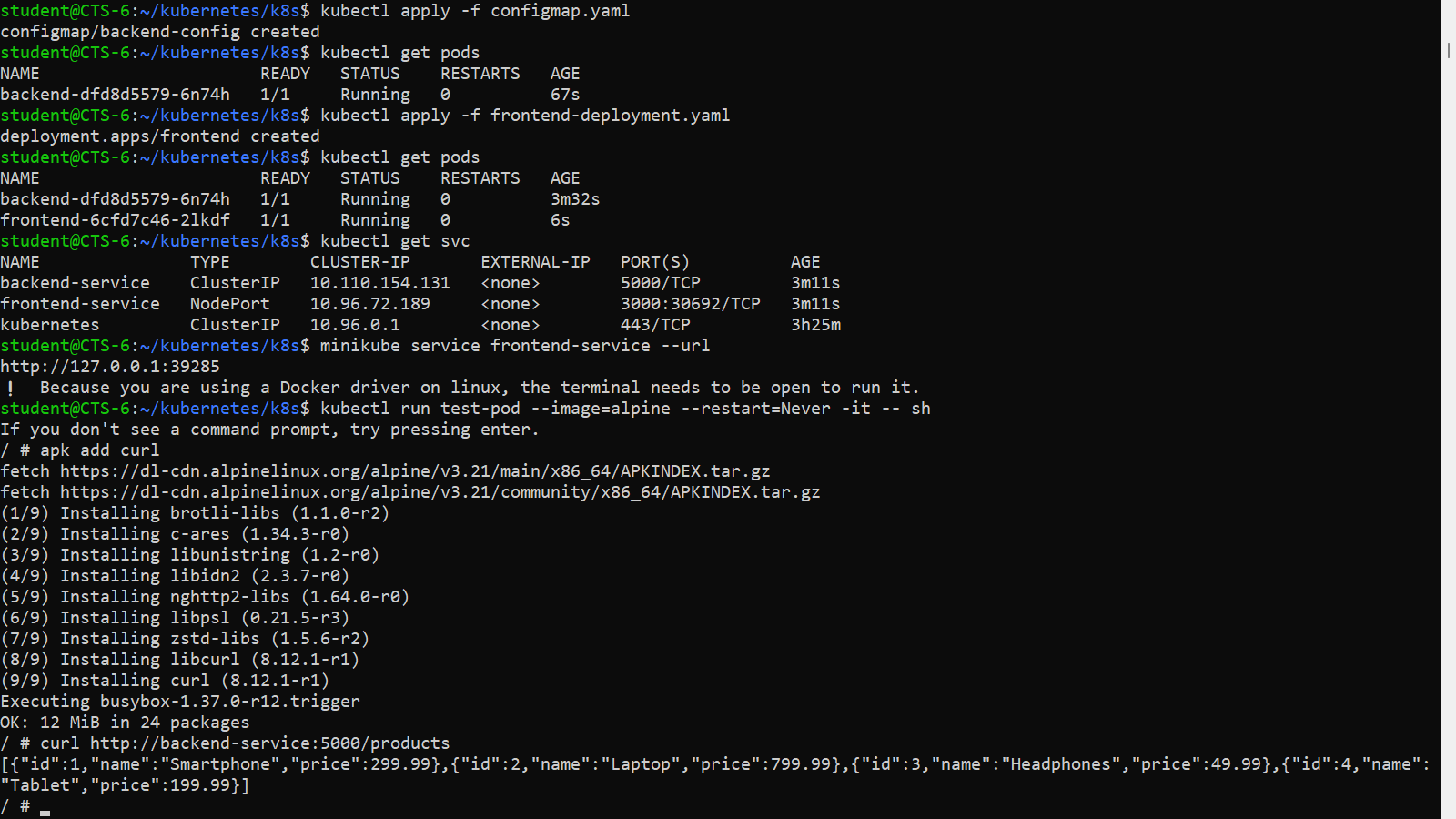
Step 14 : Come back to root directory setup minikube docker-env and build the frontend and backend



Step 15 : open k8s directory and list the files into it and kubectl apply –f commands and initialize get pods , get svc



Step 16 : using minikube service frontend-service –url it will displays the https:// ip address and using curl command to run in the terminal.



Step 16 : Enter the given ip address in the browser and get the output.

