

CSI3007 - ADVANCED PYTHON PROGRAMMING

LAB ACTIVITY – 23

E-Commerce Discount and Vendor Info API using FastAPI and Docker

NAME: DHEVISHREE GN

REG NO : 22MID0136

Goal: This project implements a FastAPI-based RESTful service that provides discount and vendor details for a given product in an e-commerce environment. The API accepts a product's details (ID, name, price) via a POST request, processes them, and returns a dynamic discount offer and vendor name.

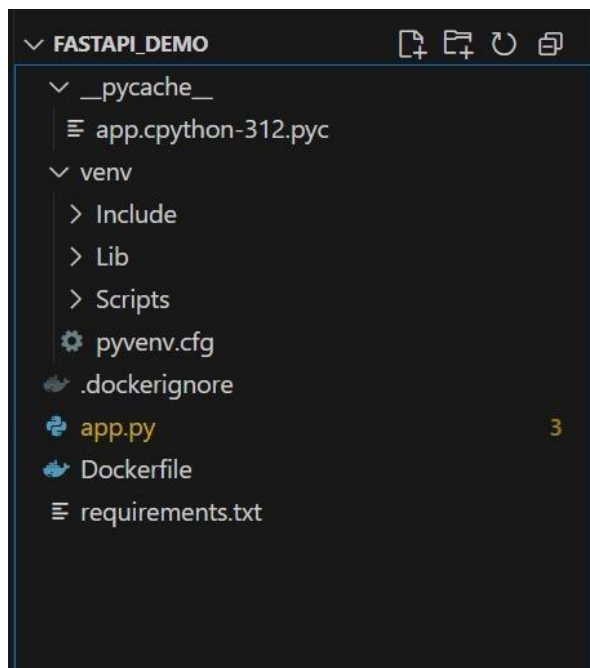
The application is containerized using Docker for portability and uses AWS Lambda compatibility via Mangum to enable serverless deployment.

Reasons to choose FastAPI:

It is one of the fastest Python web frameworks. It Simplifies handling JSON requests and responses using Pydantic models. Uses async/await for non-blocking I/O, improving concurrency.

Project Overview

Project Structure:



Application Code - app.py from fastapi import

FastAPI, HTTPException, Request from

fastapi.middleware.cors import CORSMiddleware from

mangum import Mangum import random

app = FastAPI(title="E-Commerce API", version="1.0")

Enable CORS for frontend integration

app.add_middleware(

CORSMiddleware,

allow_origins=["*"],

allow_credentials=True,

allow_methods=["*"], allow_headers=["*"],

)

--- API Endpoint ---

@app.post("/discount_vendor_info") async

def add_to_cart(request: Request):

try:

data = await request.json()

print(" Received data:", data)

product_id = data.get("product_id")

name = data.get("name") price =

data.get("price")

if not all([product_id, name, price]):

raise HTTPException(status_code=400, detail="Missing product details") #

Generate random discount and vendor discount = random.choice([0, 5, 10, 15, 20])

vendor = random.choice(["TechWorld", "ShopEase", "ElectroMart", "MegaDeals"])

```

    if discount > 0:
        message = f'{discount}% discount available from {vendor} for {name}.'
    else:
        message = f'No current discounts for {name}.'

    return {"message": message}

except Exception as e:
    raise HTTPException(status_code=500, detail=f'Processing failed: {str(e)}')

# --- AWS Lambda handler --- lambda_handler
= Mangum(app)

```

Docker Configuration

Dockerfile

```
FROM public.ecr.aws/lambda/python:3.11
```

```
WORKDIR /var/task
```

```
# Install system dependencies
```

```
RUN yum install -y gcc make rust cargo && \
python3.11 -m pip install --upgrade pip setuptools wheel
```

```
# Install dependencies
```

```
COPY requirements.txt .
```

```
RUN python3.11 -m pip install --no-cache-dir -r requirements.txt
```

```
# Copy source code
```

```
COPY . .
```

```
# Set entry point for AWS Lambda
```

CMD ["ECommerce_API.lambda_handler"]

.dockerignore

To reduce image size and avoid unnecessary files being copied into the Docker image:

Ignore virtual environment

venv/ .env/

ENV/

.venv/

Ignore Python cache and compiled files

__pycache__/

- venv/ — excludes local virtual environment files.
- __pycache__/ — removes unnecessary compiled Python bytecode.
- Keeps Docker image lightweight and clean.

Installing Dependencies: Before building the image, install dependencies locally to test

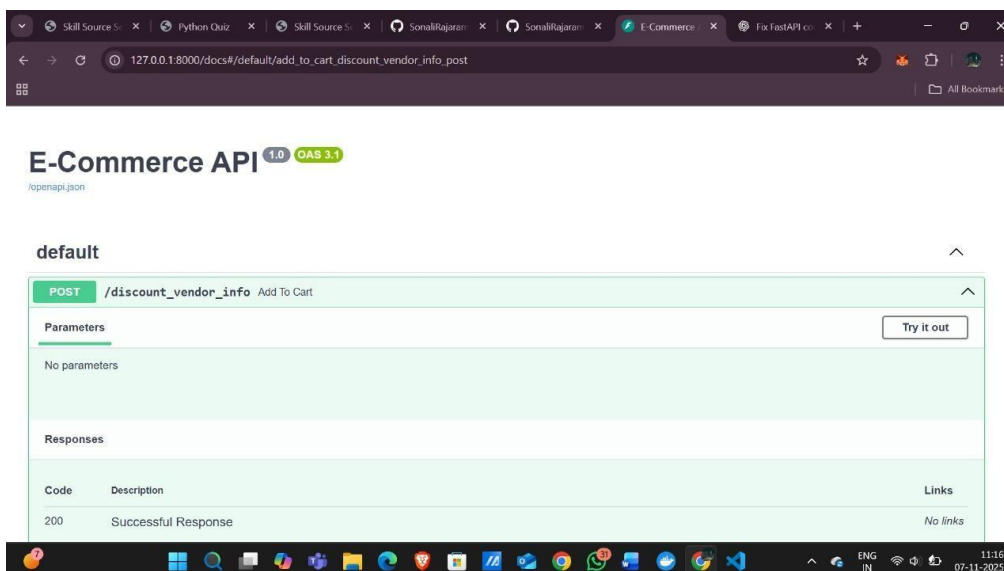
○ `python -m pip install -r requirements.txt`

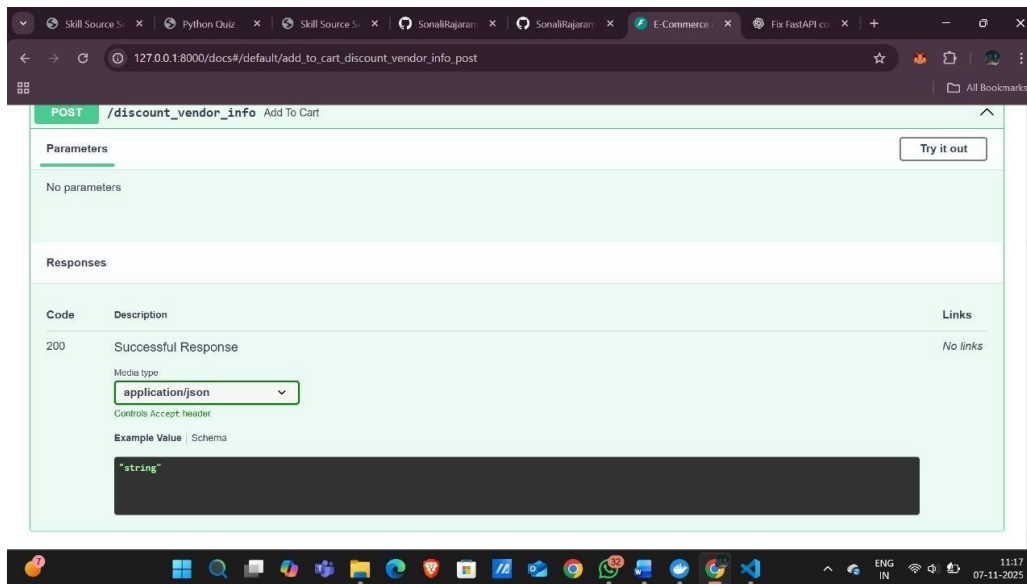
```
PROBLEMS 9 OUTPUT DEBUG CONSOLE TERMINAL PORTS SPELL CHECKER 6 python + - - - - - x
PS C:\Users\sonal\OneDrive\Desktop\FastAPI_Demo> python -m venv venv
PS C:\Users\sonal\OneDrive\Desktop\FastAPI_Demo> python -m pip install -r requirements.txt
Defaulting to user installation because normal site-packages is not writeable
WARNING: Ignoring invalid distribution ~umpy (C:\Users\sonal\AppData\Roaming\Python\Python312\site-packages)
Requirement already satisfied: fastapi in c:\users\sonal\appdata\roaming\python\python312\site-packages (from -r requirements.txt (line 1)) (0.115.0)
Requirement already satisfied: uvicorn in c:\users\sonal\appdata\roaming\python\python312\site-packages (from -r requirements.txt (line 2)) (0.30.6)
Collecting mangum (from -r requirements.txt (line 3))
  Downloading mangum-0.19.0-py3-none-any.whl.metadata (3.6 kB)
Requirement already satisfied: starlette<0.39.0,>=0.37.2 in c:\users\sonal\appdata\roaming\python\python312\site-packages (from fastapi->-r requirements.txt (line 1)) (0.38.6)
Requirement already satisfied: pydantic!=1.8,!>=1.8.1,!>=2.0.0,!>=2.0.1,!>=2.1.0,<3.0.0,>=1.7.4 in c:\users\sonal\appdata\roaming\python\python312\site-packages (from fastapi->-r requirements.txt (line 1)) (2.10.6)
Requirement already satisfied: typing-extensions>=4.8.0 in c:\users\sonal\appdata\roaming\python\python312\site-packages (from fastapi->-r requirements.txt (line 1)) (4.12.2)
Requirement already satisfied: click>=7.0 in c:\users\sonal\appdata\roaming\python\python312\site-packages (from uvicorn->-r requirements.txt (line 2)) (8.1.8)
Requirement already satisfied: h11>=0.8 in c:\users\sonal\appdata\roaming\python\python312\site-packages (from uvicorn->-r requirements.txt (line 2)) (0.14.0)
Requirement already satisfied: colorama in c:\users\sonal\appdata\roaming\python\python312\site-packages (from click>=7.0->uvicorn->-r requirements.txt (line 2)) (0.4.6)
Requirement already satisfied: annotated-types>=0.6.0 in c:\users\sonal\appdata\roaming\python\python312\site-packages (from pydantic!=1.8,!>=1.8.1,!>=2.0.0,!>=2.0.1,!>=2.1.0,<3.0.0,>=1.7.4->fastapi->-r requirements.txt (line 1)) (0.7.0)
Requirement already satisfied: pydantic-core==2.27.2 in c:\users\sonal\appdata\roaming\python\python312\site-packages (from pydantic!=1.8,!>=1.8.1,!>=2.0.0,!>=2.0.1,!>=2.1.0,<3.0.0,>=1.7.4->fastapi->-r requirements.txt (line 1)) (2.27.2)
Requirement already satisfied: anyio<5,>=3.4.0 in c:\users\sonal\appdata\roaming\python\python312\site-packages (from starlette<0.39.0,>=0.37.2->fastapi->-r requirements.txt (line 1)) (4.8.0)
Requirement already satisfied: idna>=2.8 in c:\users\sonal\appdata\roaming\python\python312\site-packages (from anyio<5,>=3.4.0->starlette<0.39.0,>=0.37.2->fastapi->-r requirements.txt (line 1)) (3.10)
Requirement already satisfied: sniffio=1.1 in c:\users\sonal\appdata\roaming\python\python312\site-packages (from a
```

Running the FastAPI App Locally: Run the app on your local system

○ python -m uvicorn app:app --reload

```
PS C:\Users\sonal\OneDrive\Desktop\FastAPI_Demo> python -m uvicorn app:app --reload
INFO: 127.0.0.1:20632 - "GET / HTTP/1.1" 404 Not Found
INFO: 127.0.0.1:20632 - "GET /favicon.ico HTTP/1.1" 404 Not Found
INFO: 127.0.0.1:15086 - "GET / HTTP/1.1" 404 Not Found
INFO: 127.0.0.1:22892 - "GET /doc HTTP/1.1" 404 Not Found
INFO: 127.0.0.1:31346 - "GET /doc/ HTTP/1.1" 404 Not Found
INFO: 127.0.0.1:53592 - "GET /docs HTTP/1.1" 200 OK
INFO: 127.0.0.1:54270 - "GET /openapi.json HTTP/1.1" 200 OK
INFO: 127.0.0.1:11296 - "POST /discount_vendor_info HTTP/1.1" 500 Internal Server Error
INFO: 127.0.0.1:50795 - "POST /discount_vendor_info HTTP/1.1" 500 Internal Server Error
INFO: 127.0.0.1:31036 - "GET /docs HTTP/1.1" 200 OK
INFO: 127.0.0.1:31036 - "GET /openapi.json HTTP/1.1" 200 OK
Received data: {'product_id': 1, 'name': 'Beige Living Room Sofa', 'price': 499}
INFO: 127.0.0.1:33996 - "POST /discount_vendor_info HTTP/1.1" 200 OK
```





Testing with PowerShell (Invoke-RestMethod)

- `Invoke-RestMethod -Uri "http://127.0.0.1:8000/discount_vendor_info" -Method POST -Headers @{ "Content-Type" = "application/json" } -Body '{"product_id": 1, "name": "Beige Living Room Sofa", "price": 499}'`

```
PS C:\Users\sonal\OneDrive\Desktop\FastAPI_Demo> Invoke-RestMethod -Uri "http://127.0.0.1:8000/discount_vendor_info"
-Method POST -Headers @{ "Content-Type" = "application/json" } -Body '{"product_id": 1, "name": "Beige Living Room
Sofa", "price": 499}'

message
-----
15% discount available from MegaDeals for Beige Living Room Sofa.

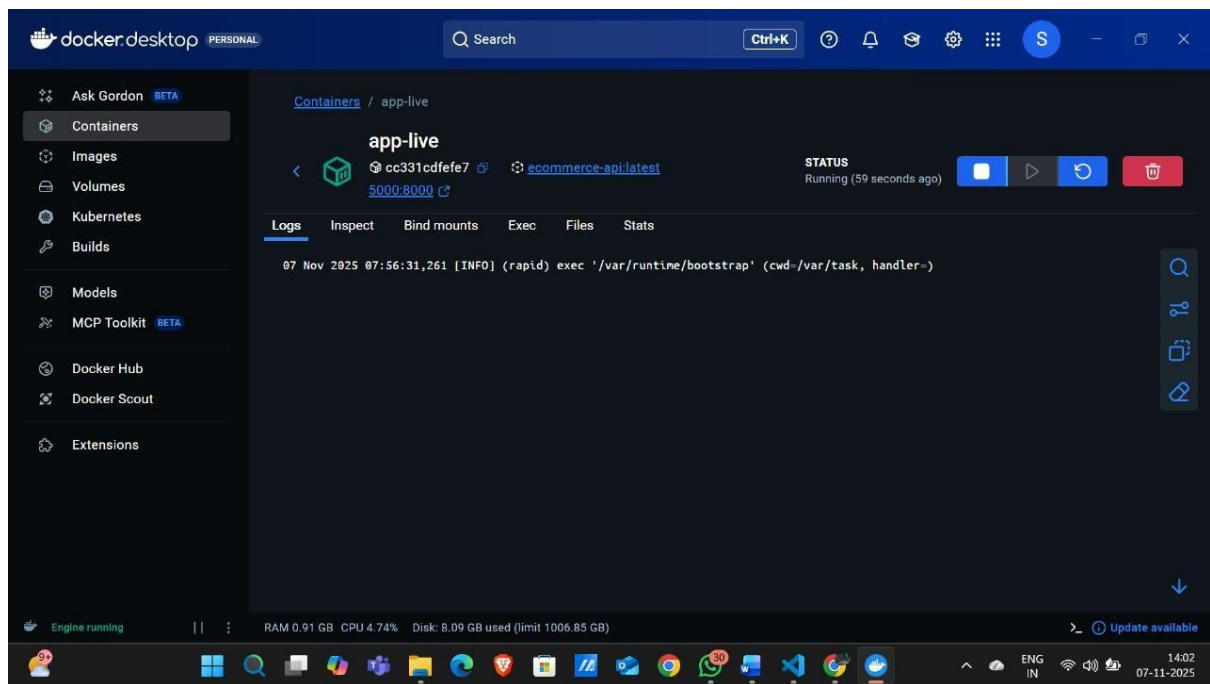
PS C:\Users\sonal\OneDrive\Desktop\FastAPI_Demo>
```

Building and Running Docker Image

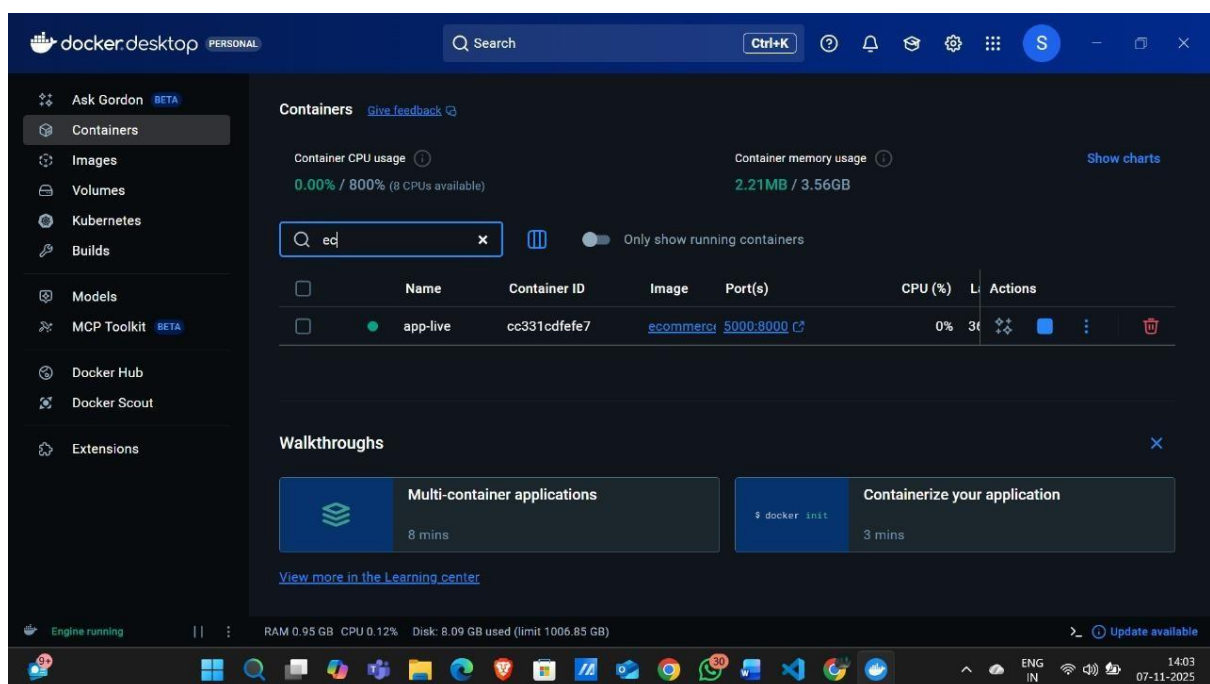
```
PS C:\Users\sonal\OneDrive\Desktop\FastAPI_Demo> docker build --network=host -t ecommerce-api .
[+] Building 968.5s (11/11) FINISHED
=> [internal] load build definition from Dockerfile 0.1s
=> => transferring dockerfile: 469B 0.0s
=> [internal] load metadata for public.ecr.aws/lambda/python:3.11 4.5s
=> [internal] load .dockerignore 0.1s
=> => transferring context: 154B 0.0s
=> [internal] load build context 0.0s
=> => transferring context: 125B 0.0s
=> [1/6] FROM public.ecr.aws/lambda/python:3.11@sha256:462443cde0453d1bb1aa8e3f50923fd26a8f458d3aa506eedb11ea037b 0.1s
=> => resolve public.ecr.aws/lambda/python:3.11@sha256:462443cde0453d1bb1aa8e3f50923fd26a8f458d3aa506eedb11ea037b 0.1s
=> CACHED [2/6] WORKDIR /var/task 0.0s
=> [3/6] RUN yum install -y gcc make rust cargo && python3.11 -m pip install --upgrade pip setuptools wheel 723.2s
=> [4/6] COPY requirements.txt . 0.2s
=> [5/6] RUN python3.11 -m pip install --no-cache-dir -r requirements.txt 121.9s
=> [6/6] COPY . . 0.5s
=> exporting to image 117.0s
=> => exporting layers 95.4s
=> => exporting manifest sha256:bac7f2727369dba525527d9cec91574a19ab422333af9286f8f9b8aad9f420d8 0.2s
=> => exporting config sha256:3f205e9a07cc4b248b571cfe8d87b511970936bedb54c2f7fc2344855fa7083a 0.1s
=> => exporting attestation manifest sha256:86b267812c274015bfb4e6c3ad3dc135e28e2bdb793389831238cd23317397f7 0.1s
=> => exporting manifest list sha256:e6938f3550a15e32112ce1df087d27d6e78f0d22e4efa7b3c07219f402e8631c 0.1s
=> => naming to docker.io/library/ecommerce-api:latest 0.1s
=> => unpacking to docker.io/library/ecommerce-api:latest 20.7s

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/rrolio35a2bwto5db4xonsx4p
PS C:\Users\sonal\OneDrive\Desktop\FastAPI_Demo> docker run -d -p 5000:8000 --name app-live ecommerce-api
cc331cdfe73f45fee1250238f2d827a1e05634240a005ed1de7bcb56b3d7fd
PS C:\Users\sonal\OneDrive\Desktop\FastAPI_Demo>
```

Docker Image:



Docker Container:



Troubleshooting:

- Error: Invoke-RestMethod -Uri "http://127.0.0.1:8080/discount_vendor_info" -Method POST -Headers @{ "Content-Type" = "application/json" } -Body '{"product_id": 1, "name": "Beige Living Room Sofa", "price": 499}'
- Resolved: Invoke-RestMethod -Uri "http://127.0.0.1:8000/discount_vendor_info" -Method POST -Headers @{ "Content-Type" = "application/json" } -Body '{"product_id": 1, "name": "Beige Living Room Sofa", "price": 499}'
(wrong port - 8080)