Advance Python Programming

Lab Task Execution

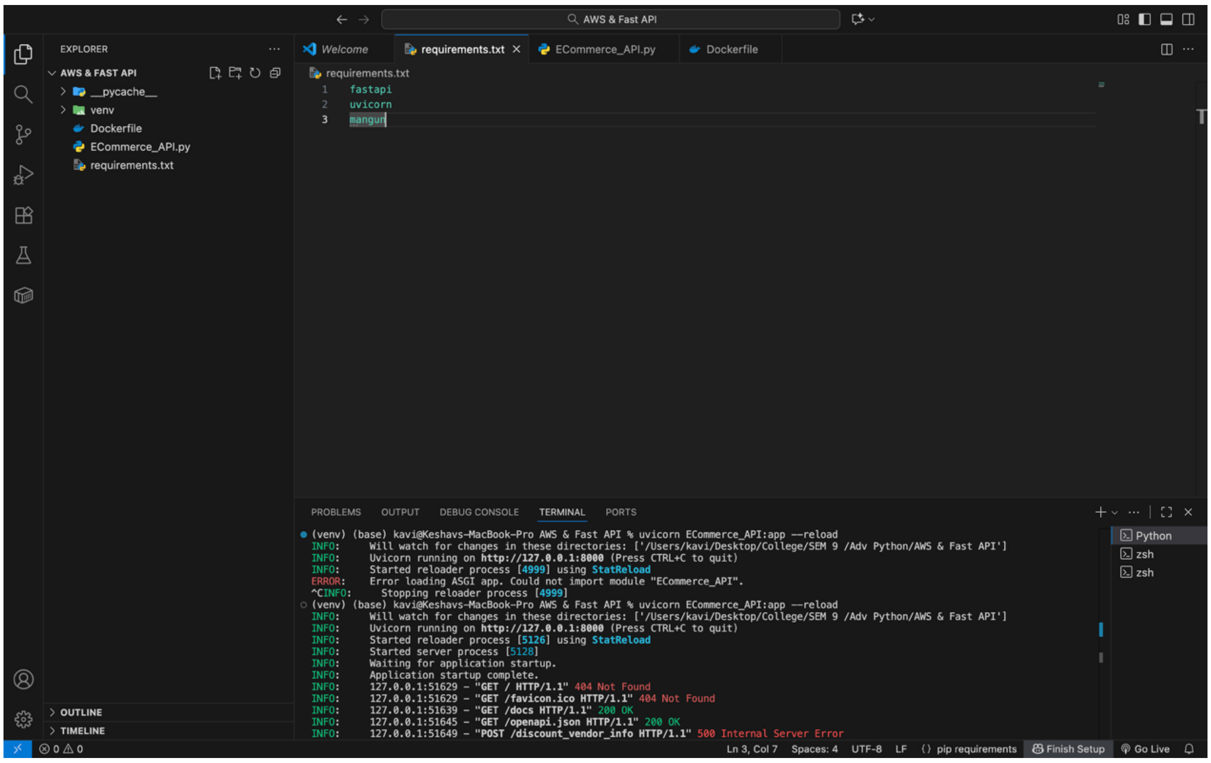
Name-: Keshav Sharma Reg No-: 21MID0182

Flask API execution and creation of the docker image for the same for the lab (07/11/2025).

First create a virtual environment called venv using the below mentioned commands-:
python3 -m venv venv
source venv/bin/activate

Now create a requirements.txt file and download the requirements one-by-one in the terminal
pip install fastapi
pip install magnum

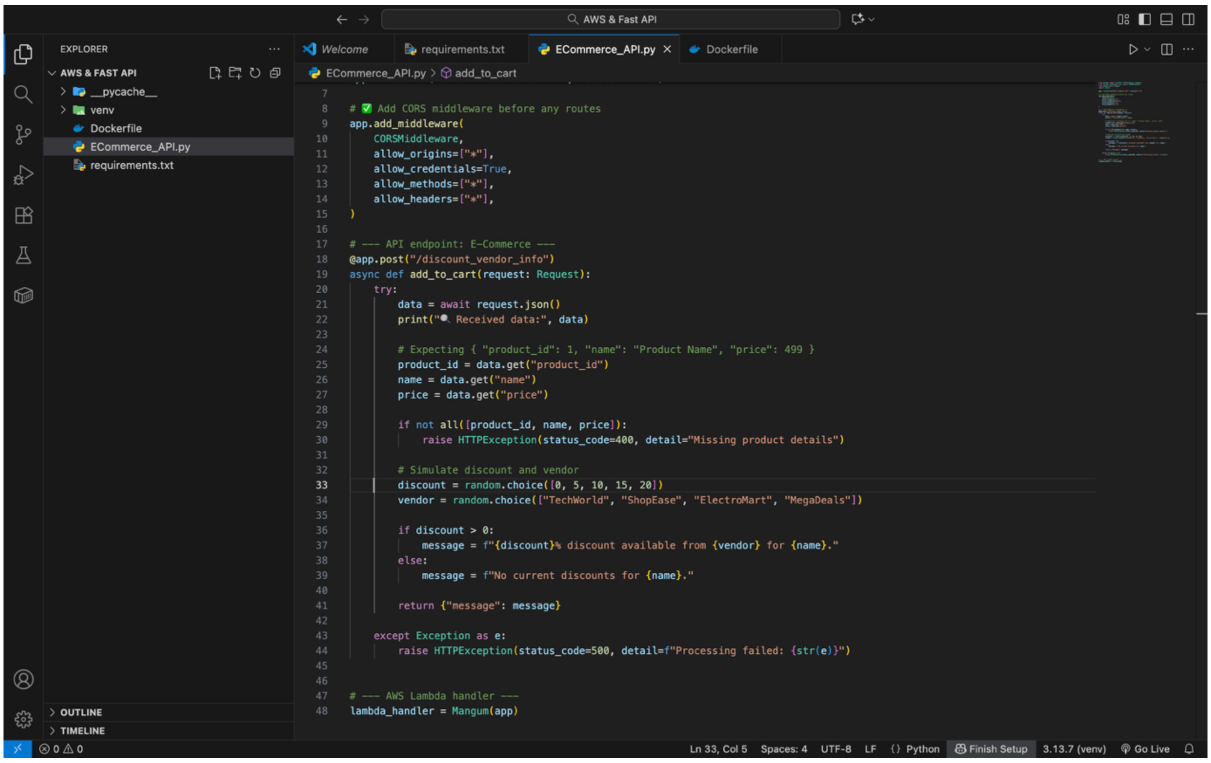
pip install uvicorn



```
fastapi
uvicorn
mangun
```

```
(venv) (base) kavi@Keshavs-MacBook-Pro AWS & Fast API % uvicorn ECommerce_API:app --reload
INFO:     Will watch for changes in these directories: ['/Users/kavi/Desktop/College/SEM 9 /Adv Python/AWS & Fast API']
INFO:     Uvicorn running on http://127.0.0.1:8000 (Press CTRL+C to quit)
INFO:     Started reloader process [4999] using StatReload
ERROR:    Error loading ASGI app. Could not import module "ECommerce_API".
^CINFO:     Stopping reloader process [4999]
(venv) (base) kavi@Keshavs-MacBook-Pro AWS & Fast API % uvicorn ECommerce_API:app --reload
INFO:     Will watch for changes in these directories: ['/Users/kavi/Desktop/College/SEM 9 /Adv Python/AWS & Fast API']
INFO:     Uvicorn running on http://127.0.0.1:8000 (Press CTRL+C to quit)
INFO:     Started reloader process [5126] using StatReload
INFO:     Started server process [5128]
INFO:     Waiting for application startup.
INFO:     Application startup complete.
INFO:     127.0.0.1:51629 - "GET / HTTP/1.1" 404 Not Found
INFO:     127.0.0.1:51629 - "GET /favicon.ico HTTP/1.1" 404 Not Found
INFO:     127.0.0.1:51639 - "GET /docs HTTP/1.1" 200 OK
INFO:     127.0.0.1:51645 - "GET /openapi.json HTTP/1.1" 200 OK
INFO:     127.0.0.1:51649 - "POST /discount_vendor_info HTTP/1.1" 500 Internal Server Error
```

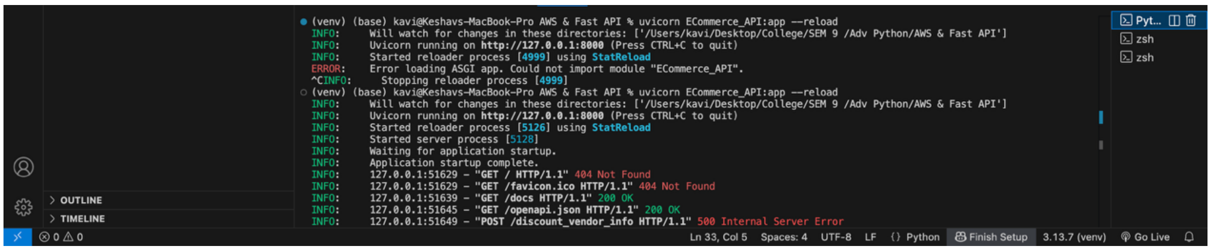
Now create Ecommerce_API.py file



```python
# ✅ Add CORS middleware before any routes
app.add_middleware(
    CORSMiddleware,
    allow_origins=["*"],
    allow_credentials=True,
    allow_methods=["*"],
    allow_headers=["*"],
)

# --- API endpoint: E-Commerce ---
@app.post("/discount_vendor_info")
async def add_to_cart(request: Request):
    try:
        data = await request.json()
        print("🔍 Received data:", data)

        # Expecting { "product_id": 1, "name": "Product Name", "price": 499 }
        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")

        # Simulate 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)
```

Now run the Ecommerce_API.py by using the below code in terminal

uvicorn ECommerce_API:app --reload

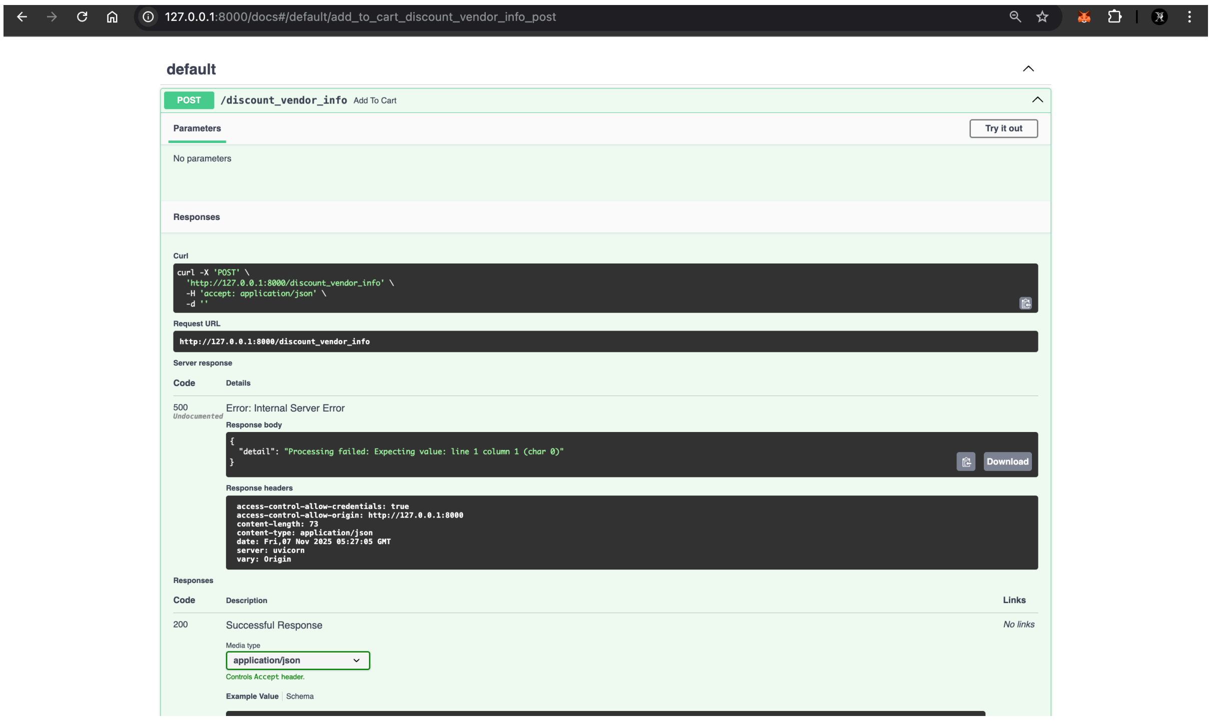


```
(venv) (base) kavi@Keshavs-MacBook-Pro AWS & Fast API % uvicorn ECommerce_API:app --reload
INFO:     Will watch for changes in these directories: ['/Users/kavi/Desktop/College/SEM 9 /Adv Python/AWS & Fast API']
INFO:     Uvicorn running on http://127.0.0.1:8000 (Press CTRL+C to quit)
INFO:     Started reloader process [4999] using StatReload
ERROR:    Error loading ASGI app. Could not import module "ECommerce_API".
^CINFO:     Stopping reloader process [4999]
(venv) (base) kavi@Keshavs-MacBook-Pro AWS & Fast API % uvicorn ECommerce_API:app --reload
INFO:     Will watch for changes in these directories: ['/Users/kavi/Desktop/College/SEM 9 /Adv Python/AWS & Fast API']
INFO:     Uvicorn running on http://127.0.0.1:8000 (Press CTRL+C to quit)
INFO:     Started reloader process [5126] using StatReload
INFO:     Started server process [5128]
INFO:     Waiting for application startup.
INFO:     Application startup complete.
INFO:     127.0.0.1:51629 - "GET / HTTP/1.1" 404 Not Found
INFO:     127.0.0.1:51629 - "GET /favicon.ico HTTP/1.1" 404 Not Found
INFO:     127.0.0.1:51639 - "GET /docs HTTP/1.1" 200 OK
INFO:     127.0.0.1:51645 - "GET /openapi.json HTTP/1.1" 200 OK
INFO:     127.0.0.1:51649 - "POST /discount_vendor_info HTTP/1.1" 500 Internal Server Error
```

Click on the http://127.0.0.1:8000/ link to open the flask api site

The in the url add (/docs) to get the below output

Now click on Try it out


127.0.0.1:8000/docs#/default/add_to_cart_discount_vendor_info_post
default
POST
Try it out
Download

/discount_vendor_info Add To Cart

Parameters

No parameters

Responses

Curl

```
curl -X 'POST' \
  'http://127.0.0.1:8000/discount_vendor_info' \
  -H 'accept: application/json' \
  -d ''
```

Request URL

```
http://127.0.0.1:8000/discount_vendor_info
```

Server response

Code Details

500 *Undocumented* Error: Internal Server Error

Response body

```
{
  "detail": "Processing failed: Expecting value: line 1 column 1 (char 0)"
}
```

Response headers

```
access-control-allow-credentials: true
access-control-allow-origin: http://127.0.0.1:8000
content-length: 73
content-type: application/json
date: Fri,07 Nov 2025 05:27:05 GMT
server: uvicorn
vary: Origin
```

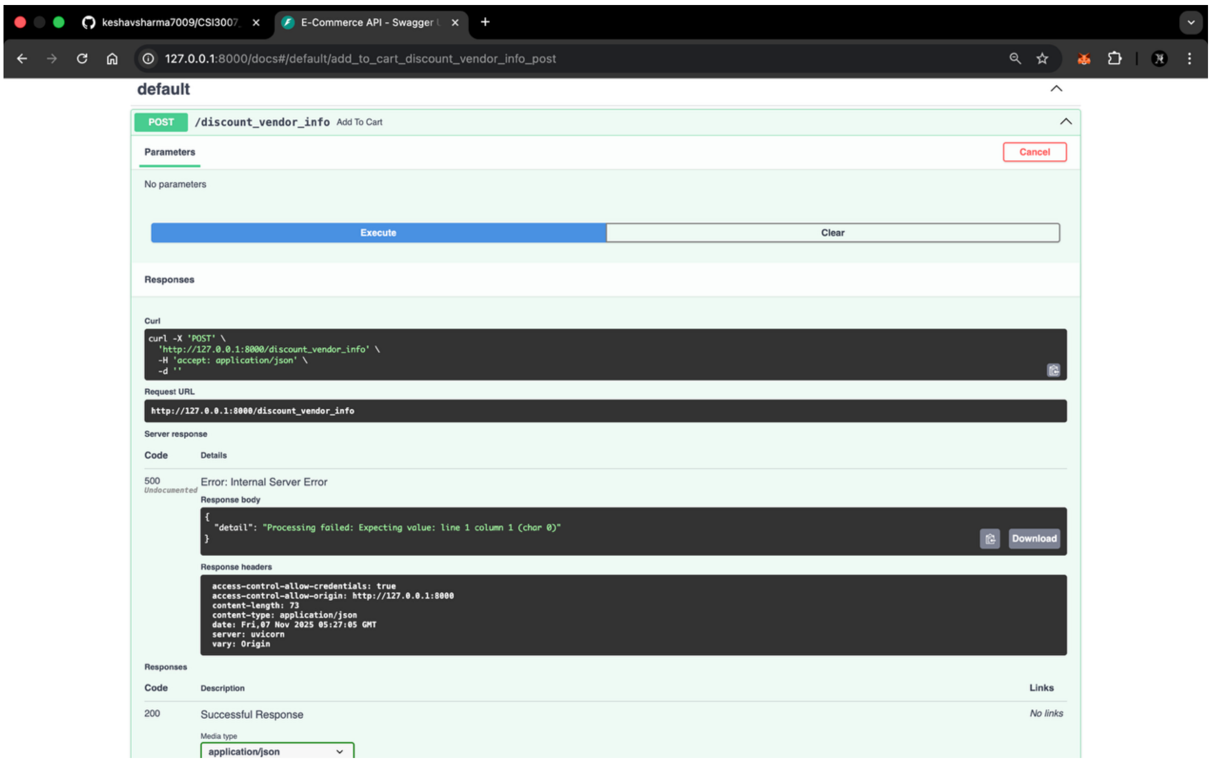
Responses

Code	Description	Links
200	Successful Response	*No links*

Media type: application/json

Controls Accept header.

Example Value | Schema



/discount_vendor_info Add To Cart

Parameters

No parameters

Responses

Curl

```
curl -X 'POST' \
  'http://127.0.0.1:8000/discount_vendor_info' \
  -H 'accept: application/json' \
  -d ''
```

Request URL

```
http://127.0.0.1:8000/discount_vendor_info
```

Server response

Code Details

500 *Undocumented* Error: Internal Server Error

Response body

```
{
  "detail": "Processing failed: Expecting value: line 1 column 1 (char 0)"
}
```

Response headers

```
access-control-allow-credentials: true
access-control-allow-origin: http://127.0.0.1:8000
content-length: 73
content-type: application/json
date: Fri,07 Nov 2025 05:27:05 GMT
server: uvicorn
vary: Origin
```

Responses

Code	Description	Links
200	Successful Response	*No links*

Media type: application/json

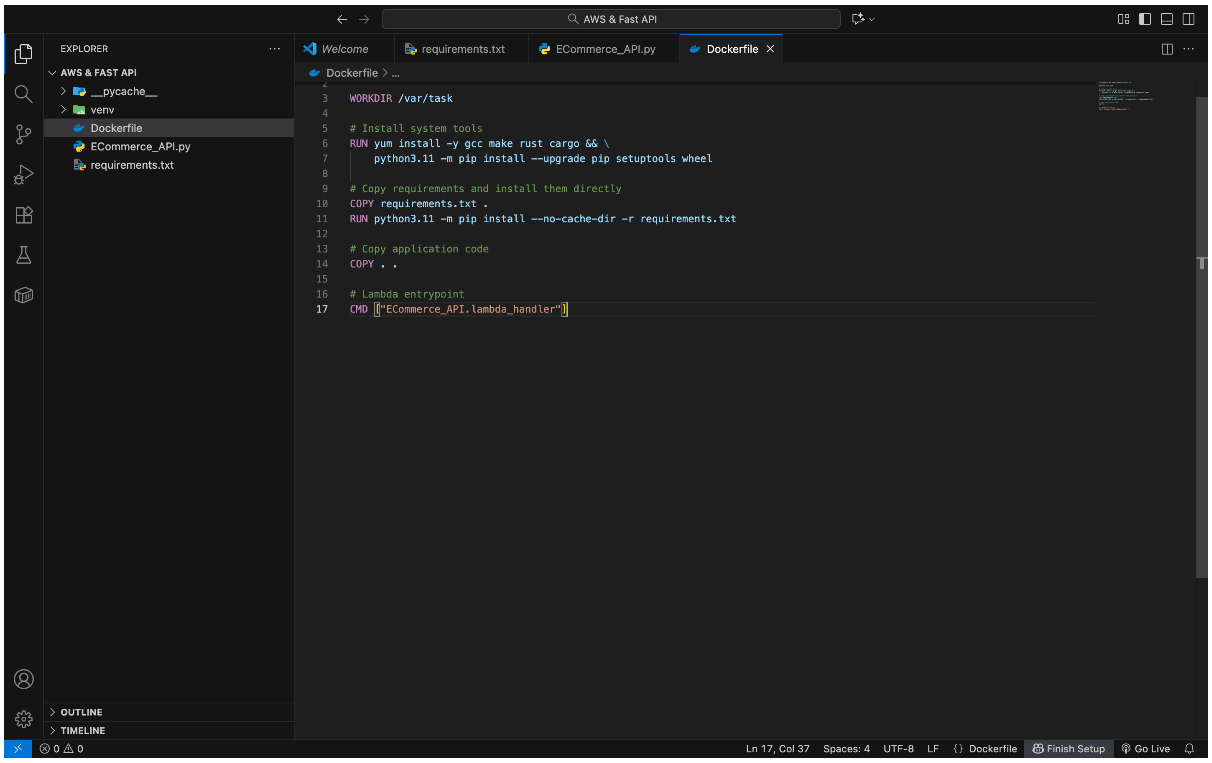
Now create a new terminal and copy paste the below mentioned text to get an output-:

curl -X 'POST' \ 'http://127.0.0.1:8000/discount_vendor_info'
- H 'accept: application/json' \
- d '{ "product_id": 1, "name": "Product Name" , "price": 499 }'

You will get below mentioned output (output can vary but the syntax will be same)

{"message":"5% discount available from ElectroMart for Product Name."}

Now create a Dockerfile in the same folder as you have created the requirements.txt and Ecommerce_API.py files



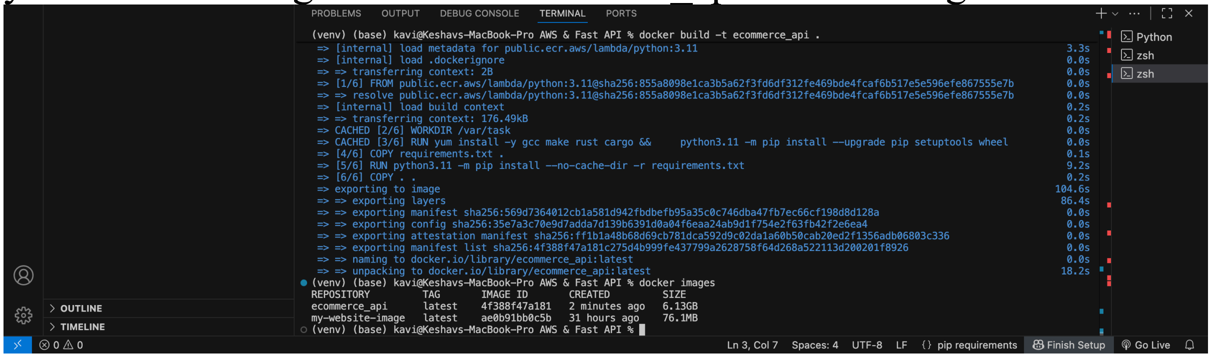
```
WORKDIR /var/task

# Install system tools
RUN yum install -y gcc make rust cargo && \
    python3.11 -m pip install --upgrade pip setuptools wheel

# Copy requirements and install them directly
COPY requirements.txt .
RUN python3.11 -m pip install --no-cache-dir -r requirements.txt

# Copy application code
COPY . .

# Lambda entrypoint
CMD ["ECommerce_API.lambda_handler"]
```

Now open a new terminal do not close any of the previous terminals And type the below mentioned code in the new terminal

docker build -t ecommerce_api .

After it is completely installed type another code in the same terminal

Docker images you will see all the docker images that are present in your docker along with the ecommerce_api docker image



```
(venv) (base) kavi@Keshavs-MacBook-Pro AWS & Fast API % docker build -t ecommerce_api .
 => [internal] load metadata for public.ecr.aws/lambda/python:3.11                                                              3.3s
 => [internal] load .dockerignore                                                                                               0.0s
 => => transferring context: 2B                                                                                                 0.0s
 => [1/6] FROM public.ecr.aws/lambda/python:3.11@sha256:855a8098e1ca3b5a62f3fd6df312fe469bde4fcaf6b517e5e596efe867555e7b        0.0s
 => => resolve public.ecr.aws/lambda/python:3.11@sha256:855a8098e1ca3b5a62f3fd6df312fe469bde4fcaf6b517e5e596efe867555e7b        0.0s
 => [internal] load build context                                                                                               0.2s
 => => transferring context: 176.49kB                                                                                           0.2s
 => CACHED [2/6] WORKDIR /var/task                                                                                              0.0s
 => CACHED [3/6] RUN yum install -y gcc make rust cargo &&     python3.11 -m pip install --upgrade pip setuptools wheel         0.0s
 => [4/6] COPY requirements.txt .                                                                                               0.1s
 => [5/6] RUN python3.11 -m pip install --no-cache-dir -r requirements.txt                                                      9.2s
 => [6/6] COPY . .                                                                                                              0.2s
 => exporting to image                                                                                                        104.6s
 => => exporting layers                                                                                                        86.4s
 => => exporting manifest sha256:569d7364012cb1a581d942fbdbefb95a35c0c746dba47fb7ec66cf198d8d128a                               0.0s
 => => exporting config sha256:35e7a3c70e9d7adda7d139b6391d0a04f6eaa24ab9d1f754e2f63fb42f2e6ea4                                 0.0s
 => => exporting attestation manifest sha256:ff1b1a48b68d69cb781dca592d9c02da1a60b50cab20ed2f1356adb06803c336                   0.0s
 => => exporting manifest list sha256:4f388f47a181c275d4b999fe437799a2628758f64d268a522113d200201f8926                          0.0s
 => => naming to docker.io/library/ecommerce_api:latest                                                                         0.0s
 => => unpacking to docker.io/library/ecommerce_api:latest                                                                     18.2s
● (venv) (base) kavi@Keshavs-MacBook-Pro AWS & Fast API % docker images
REPOSITORY        TAG       IMAGE ID       CREATED         SIZE
ecommerce_api     latest    4f388f47a181   2 minutes ago   6.13GB
my-website-image  latest    ae0b91bb0c5b   31 hours ago    76.1MB
○ (venv) (base) kavi@Keshavs-MacBook-Pro AWS & Fast API % 
```