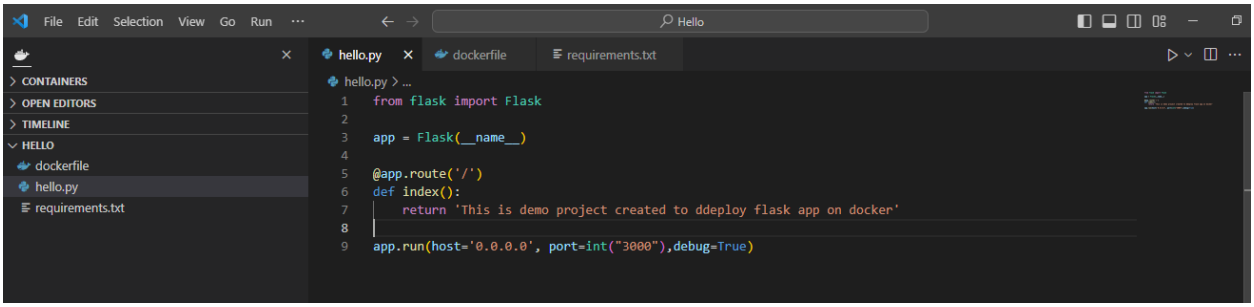
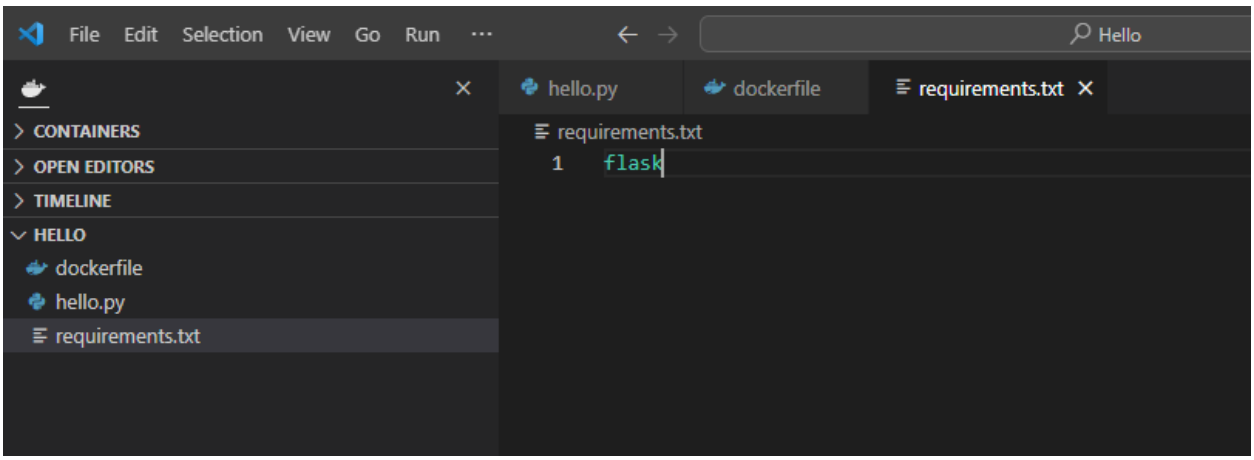
DOCKER ASSIGNMENT

1. Hello world using flask and deployment using Docker

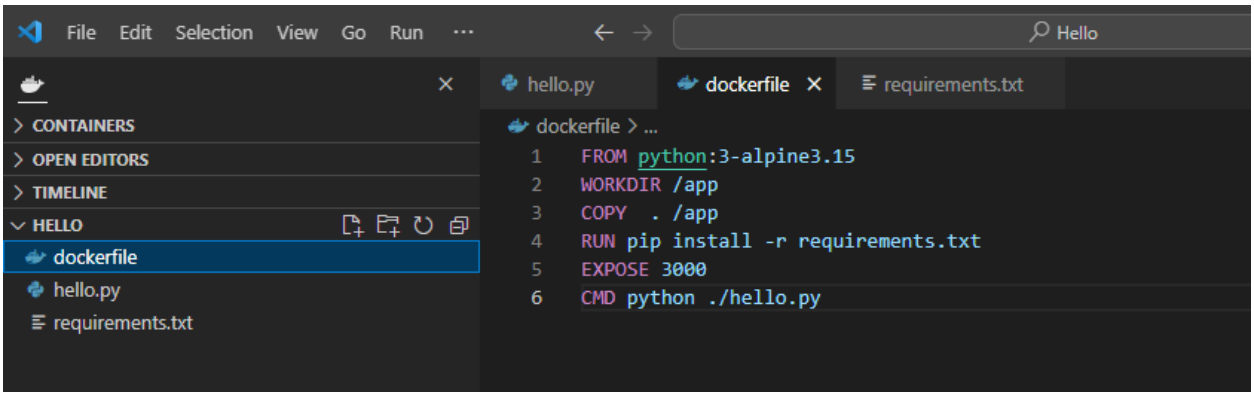
STEP 1: CREATED APP



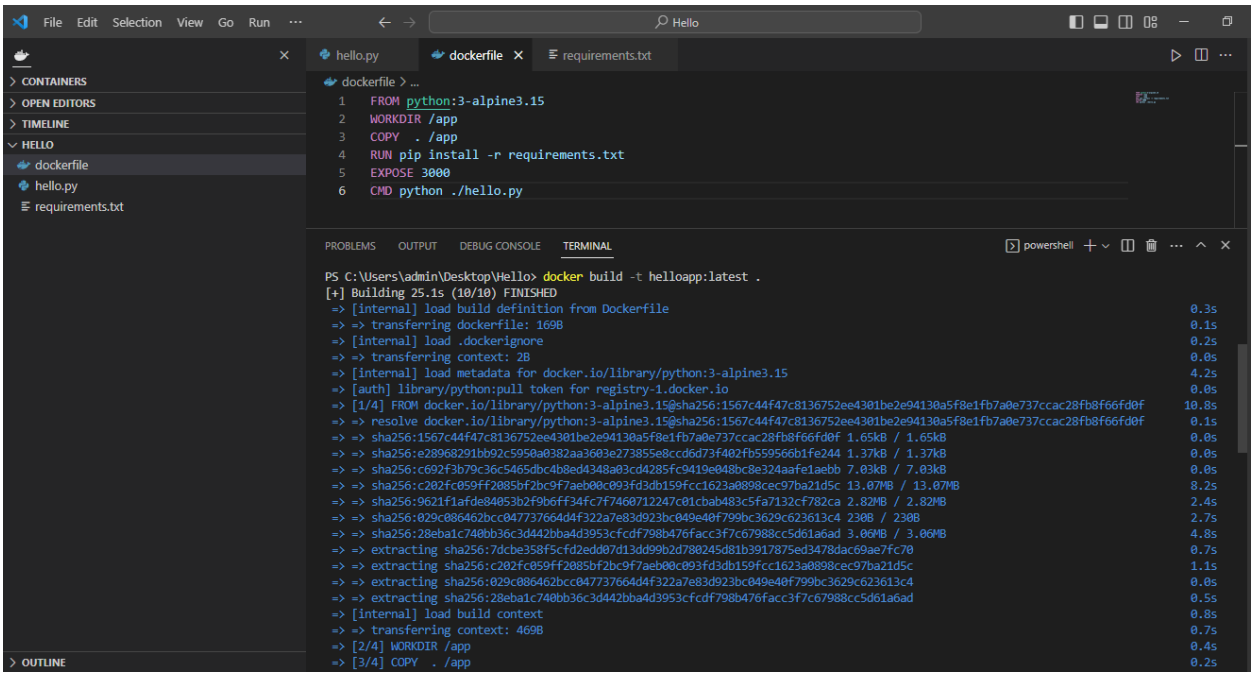
STEP 2: CREATED REQUIREMENT FILE



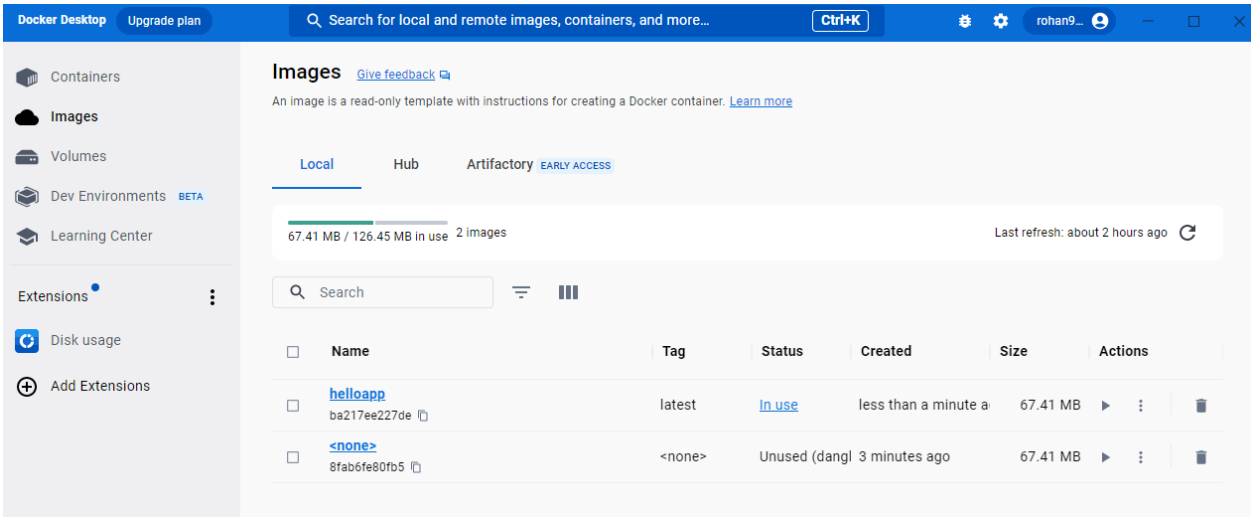
STEP 3: CREATED DOCKER FILE

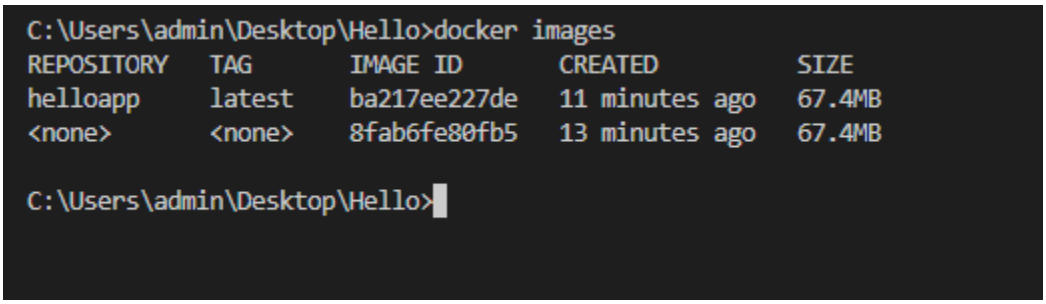


STEP 4: CREATNG DOCKER IMAGE

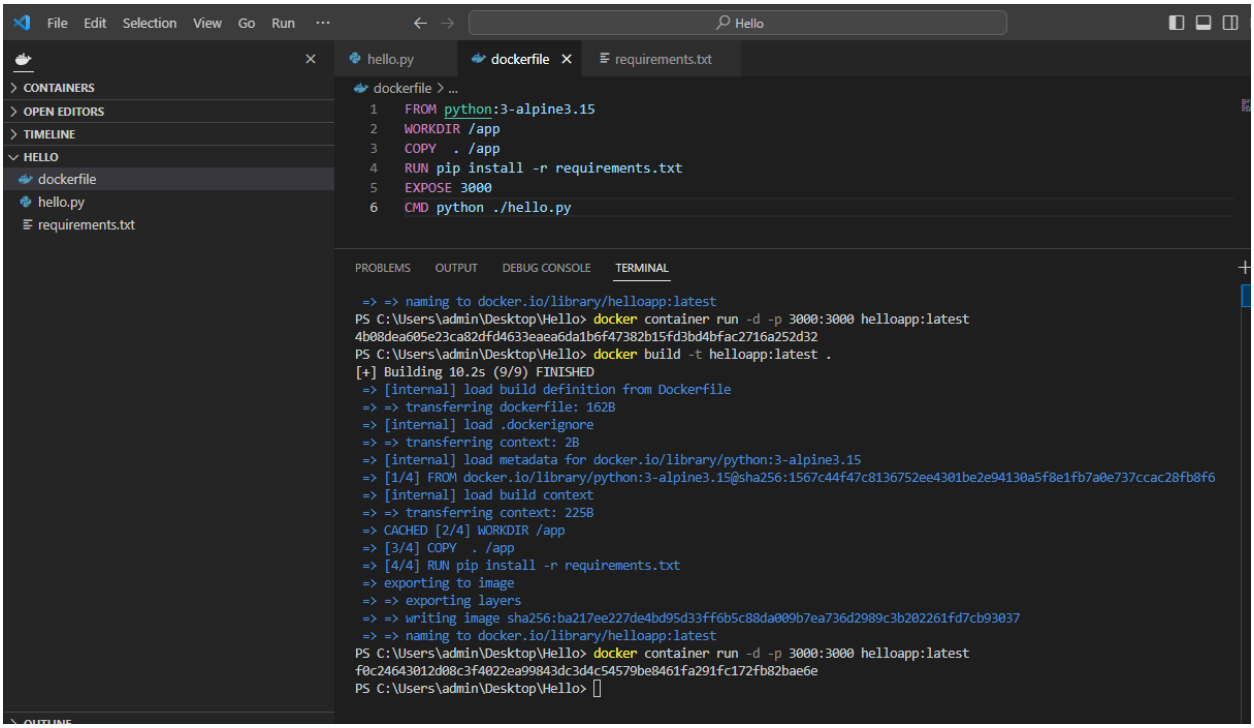


STEP 5: CHECKING DOCKER IMAGES

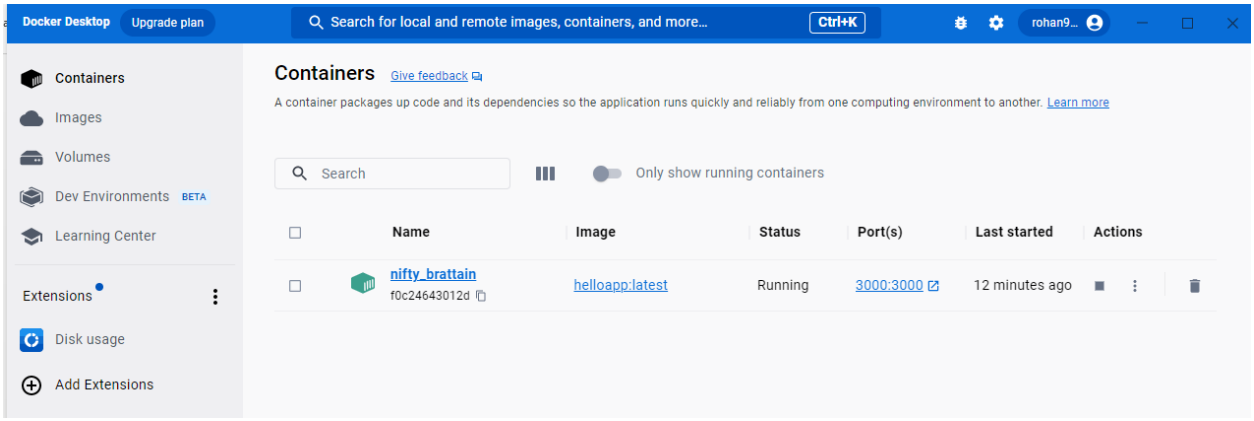




STEP 6: CREATING CONTAINER



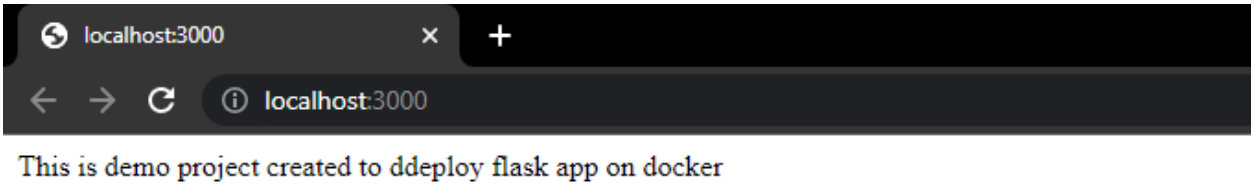
STEP 7: CHECKING CONTAINER




```
C:\Users\admin\Desktop\Hello>docker ps
CONTAINER ID   IMAGE             COMMAND                  CREATED          STATUS          PORTS                    NAM
ES
f0c24643012d   helloapp:latest   "/bin/sh -c 'python …"   12 minutes ago   Up 12 minutes   0.0.0.0:3000->3000/tcp   nif
ty_brattain

C:\Users\admin\Desktop\Hello>
```

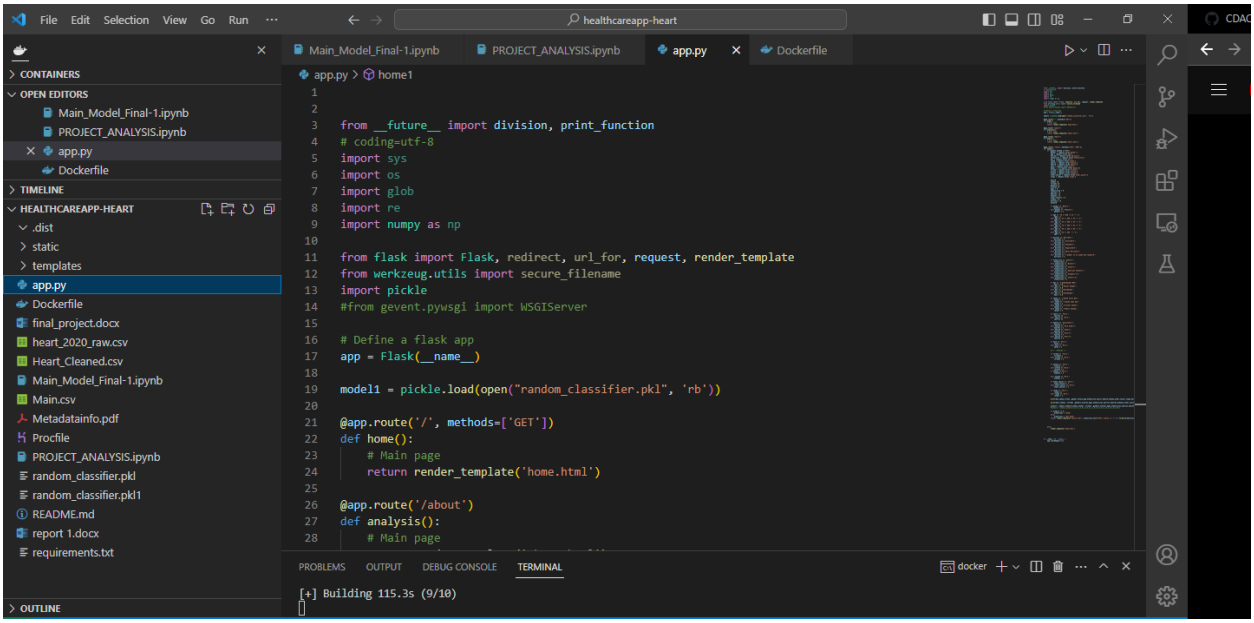
STEP 8: OPENING OR RUNNING CONTAINER



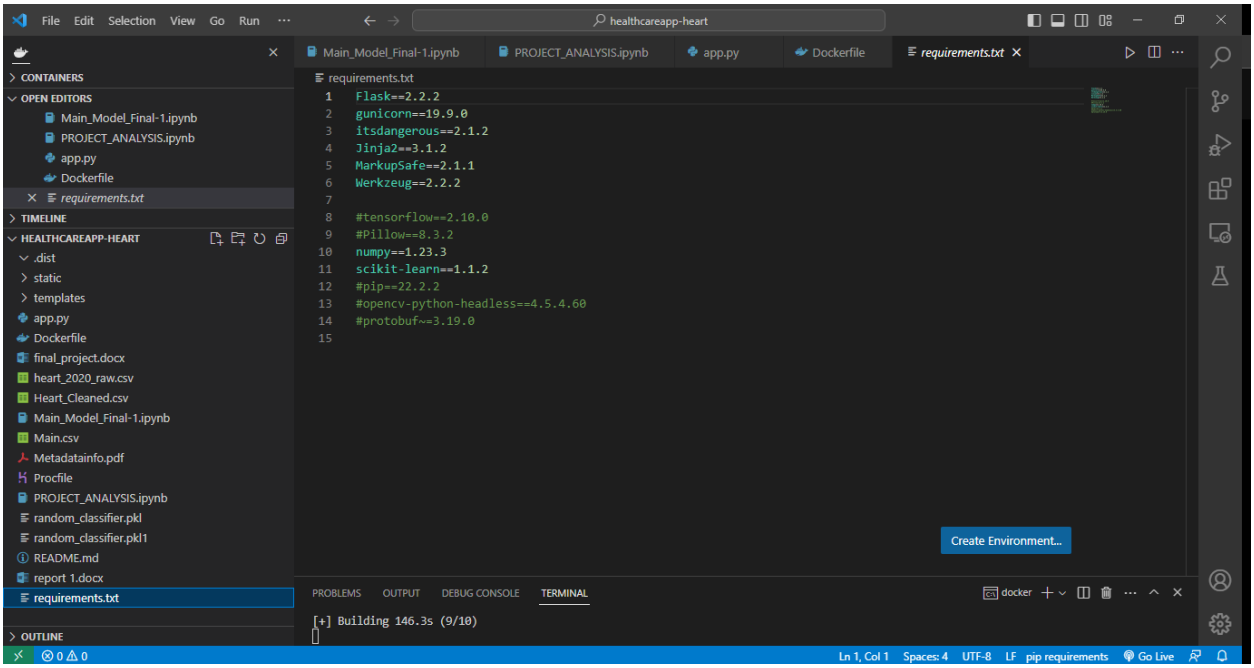
STEP 1: CRETED APP

2. Create a python project with UI and Backend with database, create Docker file and deploy it.

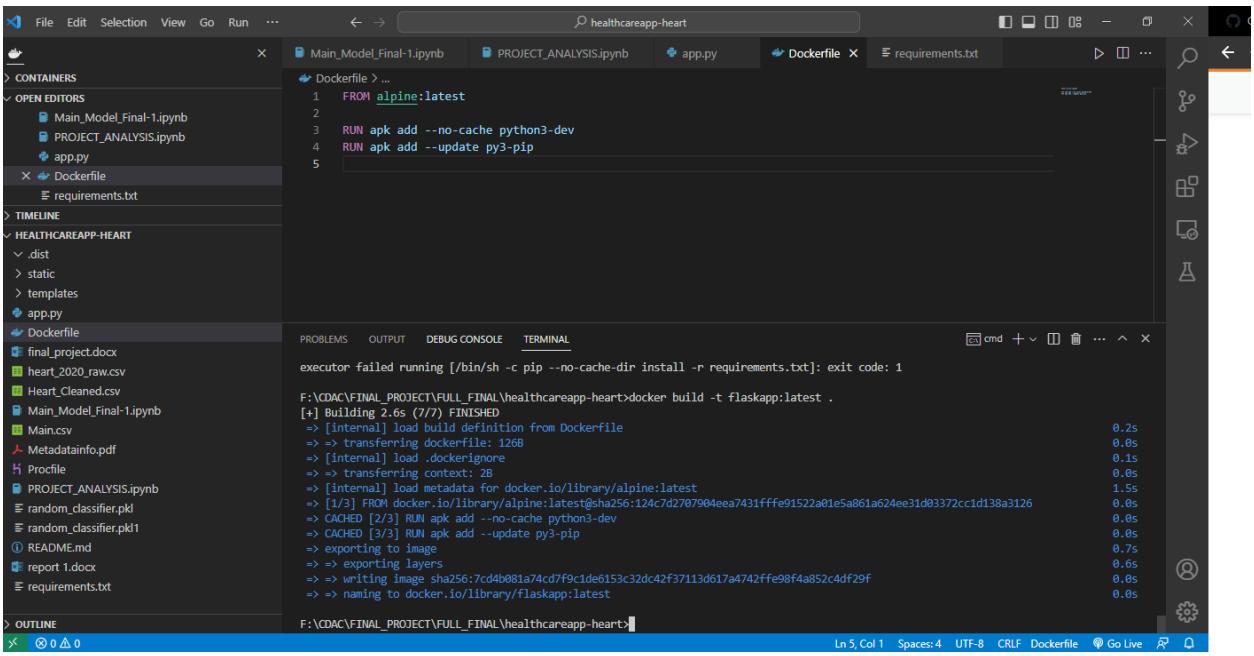
STEP 1: Creating python project using flask.



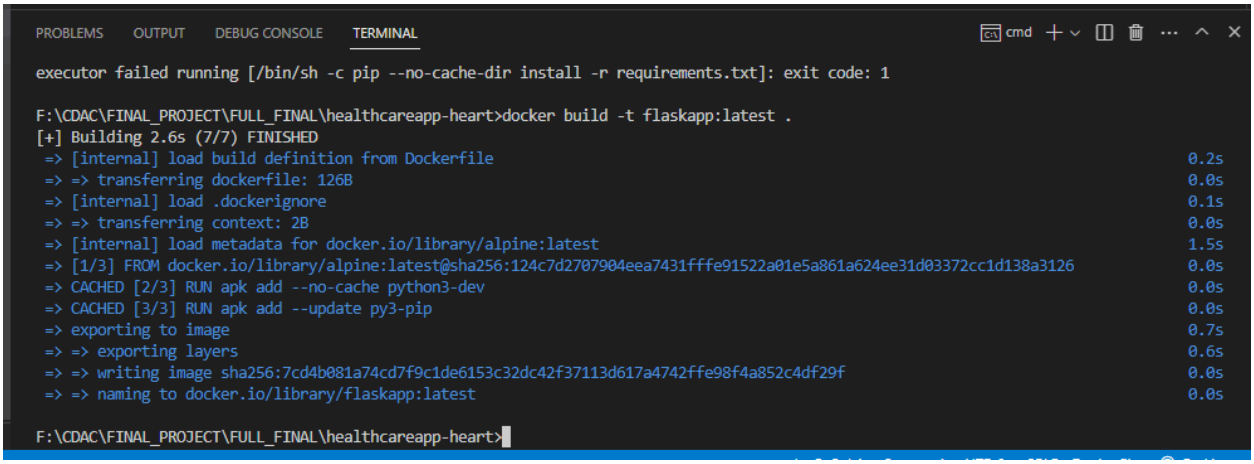
STEP 2: Adding requirement.txt i.e environment setup requirement



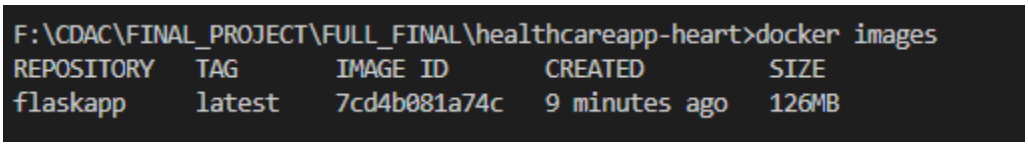
STEP 3: Creating Docker file to install all dependencies

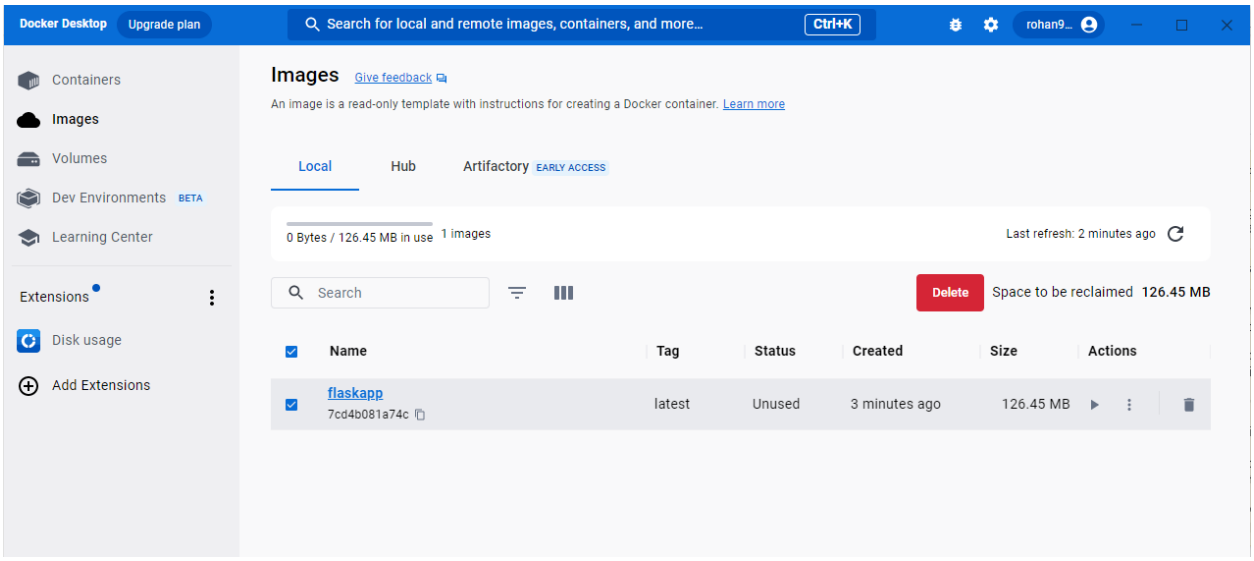


STEP 4: Running build command to create image file



STEP 5: Using Docker desktop to check wheatear image is created or not





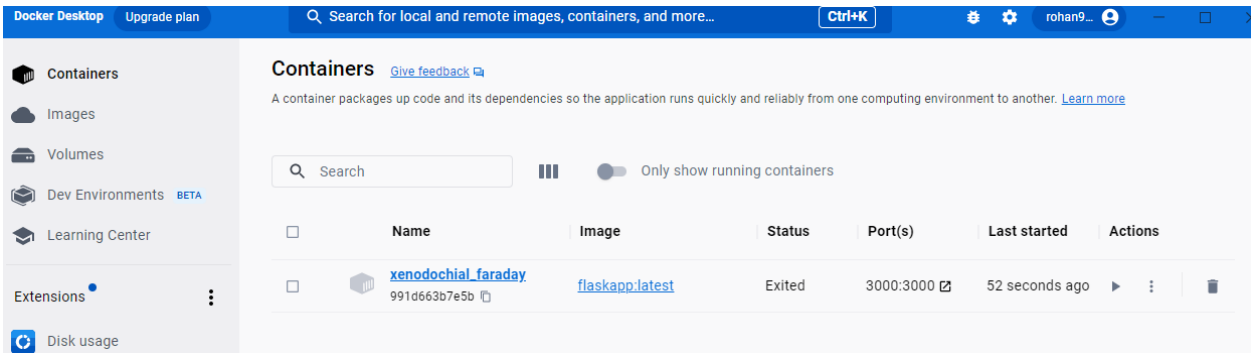
STEP 5: Adding more dependencies to Docker file

```
Dockerfile > ...
1   FROM alpine:latest
2
3   RUN apk add --no-cache python3-dev
4   RUN apk add --update py3-pip
5   WORKDIR /app
6   COPY . /app
7   RUN pip install -r requirements.txt
8   EXPOSE 3000
9   CMD python ./app.py
10
```

STEP 6: Again building a images and creating container using command

```
F:\CDAC\FINAL_PROJECT\FULL_FINAL\healthcareapp-heart>docker container run -d -p 3000:3000 flaskapp:latest
991d663b7e5b559238e46e8d95e41ece47619c56e3efdc01f10e0c0318a686fc
```

STEP 7: CHECKING CONTAINER




```
F:\CDAC\FINAL_PROJECT\FULL_FINAL\healthcareapp-heart>docker ps -a
CONTAINER ID   IMAGE             COMMAND     CREATED         STATUS                     PORTS     NAMES
991d663b7e5b   flaskapp:latest   "/bin/sh"   2 minutes ago   Exited (0) 2 minutes ago             xenodochial_faraday
```