

Containerize the App

Date	19 November 2022
Team ID	PNT2022TMID12716
Project Name	Inventory Management System For Retailers

Dockerfile:

```
Dockerfile
1 FROM python:3.10.4
2 WORKDIR /app
3 ADD . /app
4 COPY requirements.txt /app
5 RUN python3 -m pip install -r requirements.txt
6 RUN python3 -m pip install ibm_db
7 EXPOSE 5000
8 ENTRYPOINT ["python"]
9 CMD ["app.py"]
```

Building Image:

docker build -t <image_name> .

Execute the above command in the working directory where Dockerfile is present

```
(myApp) D:\Inventory_Management_System_for_Retailers>docker build -t inventory-mgmt .
[+] Building 217.2s (12/12) FINISHED
=> [internal] load build definition from Dockerfile                                0.1s
=> => transferring dockerfile: 252B                                              0.0s
=> [internal] load .dockerignore                                                 0.0s
=> => transferring context: 2B                                                  0.0s
=> [internal] load metadata for docker.io/library/python:3.10.4                2.6s
=> [3/6] ADD . /app                                                             0.5s
=> [4/6] COPY requirements.txt /app                                             0.1s
=> [5/6] RUN python3 -m pip install -r requirements.txt                        208.1s
=> [6/6] RUN python3 -m pip install ibm_db                                     1.0s
=> exporting to image                                                           1.1s
=> => exporting layers                                                         1.0s
=> => writing image sha256:82d0d6d55d7e358b0b69a8f2a38ea24804dcc39a81c5d3a6554bd1a1dcf8dc78 0.0s
=> => naming to docker.io/library/inventory-mgmt                               0.0s

Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them

(myApp) D:\Inventory_Management_System_for_Retailers>docker images
REPOSITORY          TAG         IMAGE ID      CREATED        SIZE
inventory-mgmt      latest     82d0d6d55d7e  About a minute ago  1.21GB
```

Run the container:

Docker run -d -p <port>:<port> <image_name>

Execute the command to run the container in the mentioned port number

```
(myApp) D:\Inventory_Management_System_for_Retailers>docker run -d -p 5000:5000 inventory-mgmt
25eb65cdf5cc7db4171174c02a8a73024a9228d72d440fce1c0556eb764d686f

(myApp) D:\Inventory_Management_System_for_Retailers>docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                               NAMES
25eb65cdf5cc   inventory-mgmt "python app.py"         7 seconds ago Up 6 seconds  0.0.0.0:5000->5000/tcp             affectionate_chatelet

(myApp) D:\Inventory_Management_System_for_Retailers>
```

Open localhost:5000 or 127.0.0.1:5000 to view the running container



Inventory Management System for Retailers

