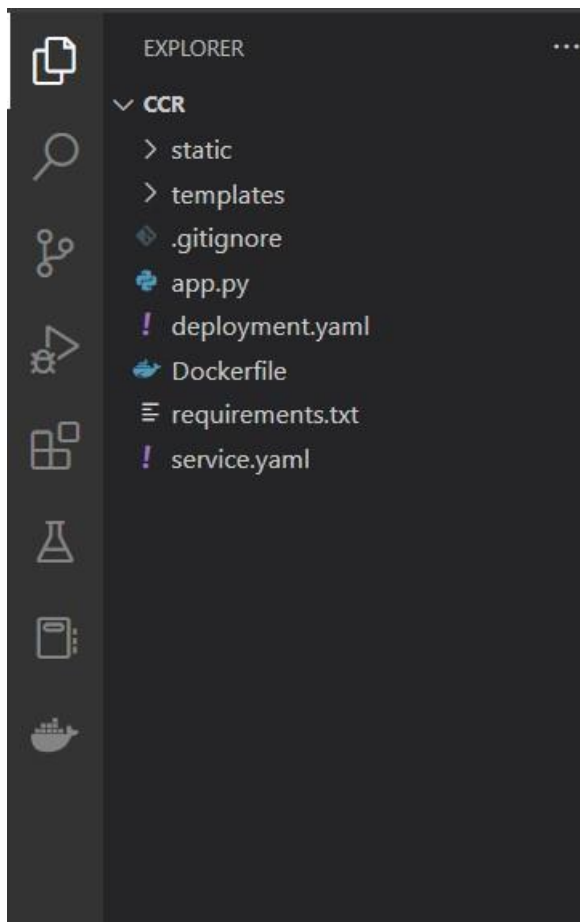


Deployment of app in IBM Cloud

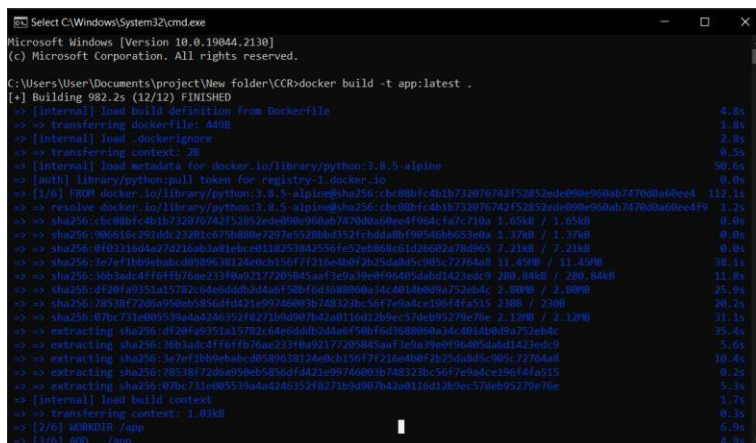
Containerize the app

Date	14 November 2022
Team ID	PNT2022TMID47409
Project Name	CUSTOMER CARE REGISTRY

Step 1. Create the Docker file and paste the code



Step 2. Build an image from the docker file.



```
Select C:\Windows\System32\cmd.exe
=> [internal] load metadata for docker.io/library/python:3.8.5-alpine 50.6s
=> [auth] library/python:pull token for registry-1.docker.io 0.0s
=> [1/6] FROM docker.io/library/python:3.8.5-alpine@sha256:cbc08bfc4b1b732076742f52852ede090e960ab7470d0a60ee4f9 112.1s
=> => resolve docker.io/library/python:3.8.5-alpine@sha256:cbc08bfc4b1b732076742f52852ede090e960ab7470d0a60ee4f9 1.2s
=> => sha256:cbc08bfc4b1b732076742f52852ede090e960ab7470d0a60ee4f964cfa7c710a 1.65kB / 1.65kB 0.0s
=> => sha256:906616c291ddc23201c675b880e7297e5528bbd352fcbdda8bf98546bb653e0a 1.37kB / 1.37kB 0.0s
=> => sha256:0f03316d4e27d216ab3a81ebce0118253842556fe52eb068c61d26602a78d965 7.21kB / 7.21kB 0.0s
=> => sha256:3e7ef1bb9ebabc0589638124e0cb156f7f216e4b0f2b25da8d5c905c72764a8 11.45MB / 11.45MB 38.1s
=> => sha256:36b3adc4ff6fffb76ae233f0a92177205845aaf3e9a39e0f96405dabd1423edc9 280.84kB / 280.84kB 11.8s
=> => sha256:df20fa9351a15782c64e6dddb2d4a6f50bf6d368806a34c4014b0d9a752eb4c 2.80MB / 2.80MB 25.9s
=> => sha256:78538f72d6a950eb5856dfd421e99746003b748323bc56f7e9a4ce196f4fa515 230B / 230B 20.2s
=> => sha256:07bc731e005539a4a4246352f8271b9d907b42a0116d12b9ec57deb95279e70e 2.12MB / 2.12MB 31.1s
=> => extracting sha256:df20fa9351a15782c64e6dddb2d4a6f50bf6d368806a34c4014b0d9a752eb4c 35.4s
=> => extracting sha256:36b3adc4ff6fffb76ae233f0a92177205845aaf3e9a39e0f96405dabd1423edc9 5.6s
=> => extracting sha256:3e7ef1bb9ebabc0589638124e0cb156f7f216e4b0f2b25da8d5c905c72764a8 10.4s
=> => extracting sha256:78538f72d6a950eb5856dfd421e99746003b748323bc56f7e9a4ce196f4fa515 0.2s
=> => extracting sha256:07bc731e005539a4a4246352f8271b9d907b42a0116d12b9ec57deb95279e70e 5.3s
=> [internal] load build context 1.7s
=> => transferring context: 1.03kB 0.3s
=> [2/6] WORKDIR /app 6.9s
=> [3/6] ADD . /app 4.9s
=> [4/6] RUN set -e; apk add --no-cache --virtual .build-deps gcc libc 314.0s
=> [5/6] COPY requirements.txt /app 7.9s
=> [6/6] RUN pip install -r requirements.txt 178.9s
=> exporting to image 291.3s
=> => exporting layers 288.1s
=> => writing image sha256:ab9e0817fd383e885b6b713db2a953e7b5a47eac4f9f06ca711d4280c7b9b886 0.8s
=> => naming to docker.io/library/app:latest 0.5s
C:\Users\User\Documents\project\New folder\CCR>
```

Step 3.Run your container locally.

```
C:\Windows\System32\cmd.exe
=> => transferring context: 1.03kB 0.3s
=> [2/6] WORKDIR /app 6.9s
=> [3/6] ADD . /app 4.9s
=> [4/6] RUN set -e; apk add --no-cache --virtual .build-deps gcc libc 314.0s
=> [5/6] COPY requirements.txt /app 7.9s
=> [6/6] RUN pip install -r requirements.txt 178.9s
=> exporting to image 291.3s
=> => exporting layers 288.1s
=> => writing image sha256:ab9e0817fd383e885b6b713db2a953e7b5a47eac4f9f06ca711d4280c7b9b886 0.8s
=> => naming to docker.io/library/app:latest 0.5s
C:\Users\User\Documents\project\New folder\CCR>docker run -d -p 5000:5000 app
45a85421c862d7a4a6351f6587e305a4d3371f65a325cfa6d8f29761a87315de

C:\Users\User\Documents\project\New folder\CCR>docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                               NAMES
45a85421c862   app       "python app.py"         2 minutes ago Up 19 seconds 0.0.0.0:5000->5000/tcp             youthful_nightingale
C:\Users\User\Documents\project\New folder\CCR>
```