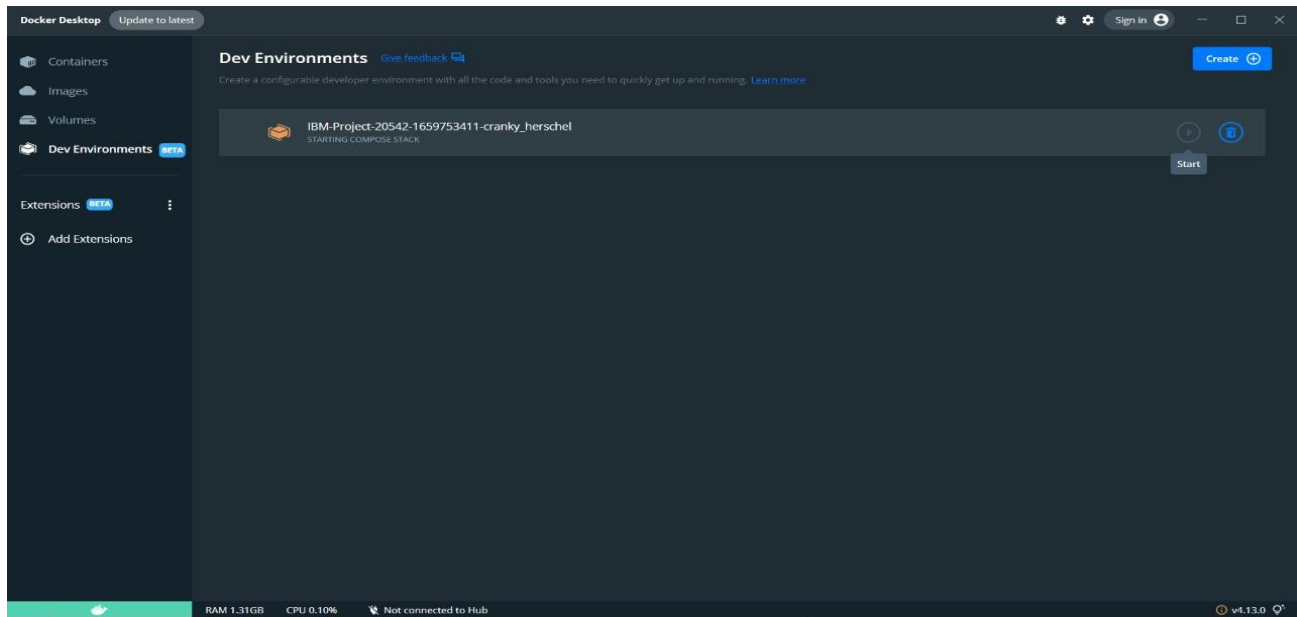


ASSIGNMENT-4

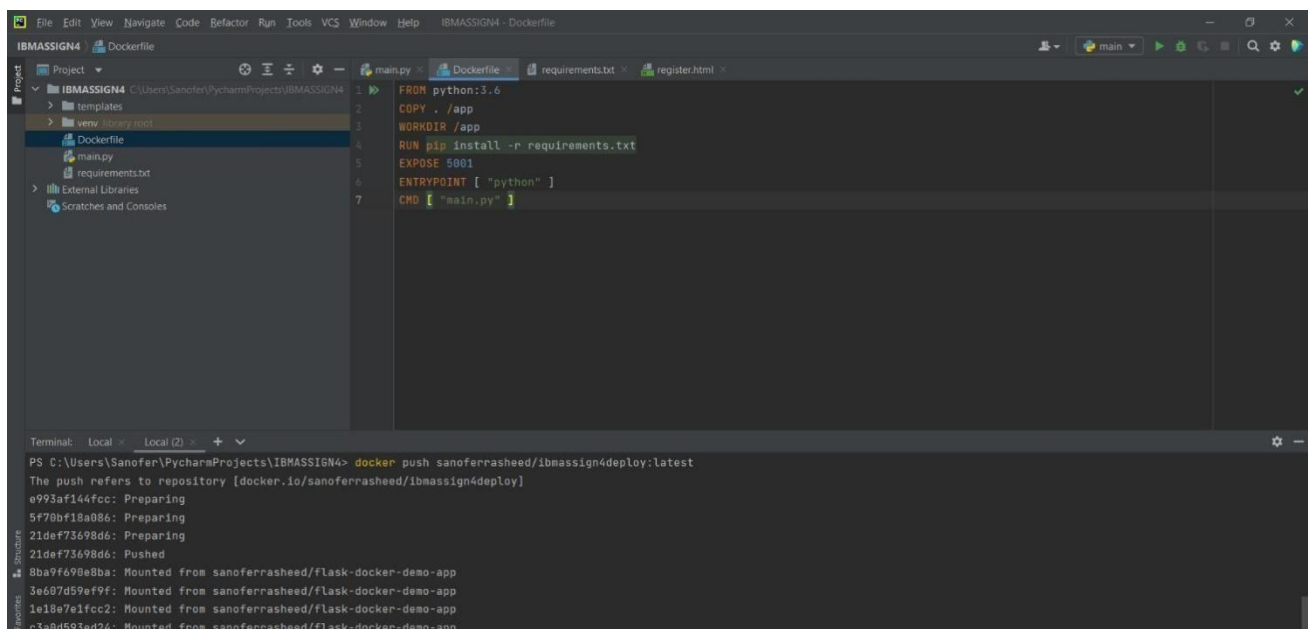
Assignment Date	05 October2022
Student Name	Jindat Baradia
Student Roll Number	311019205022
Maximum Marks	2 Marks

1. Pull an Image from docker hub and run it in docker playground.

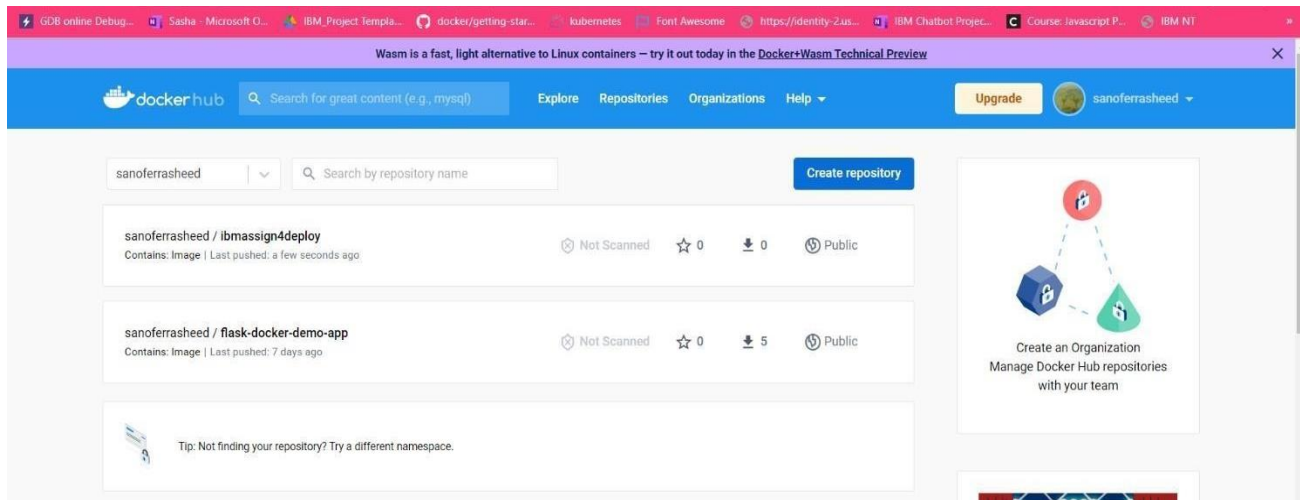
The image is built.



The same image is pushed to docker hub using the command



Here image name is ibmassign4deploy. Thus it is pushed in docker hub.



The app is running at the specified port.

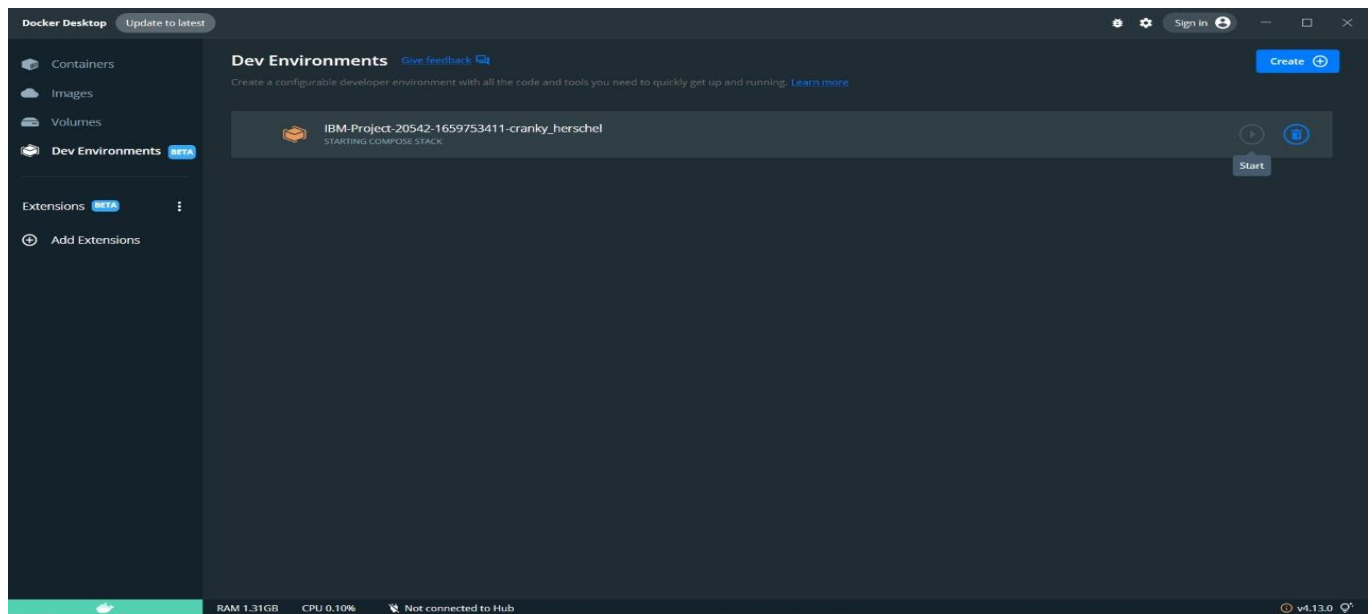
A screenshot of a web page titled 'Registration Form'. It features a series of input fields for 'Enter name', 'Enter Email', 'Enter Mobile', 'Enter City', 'Enter State', and 'Enter Country'. A 'Submit' button is located at the bottom right of the form. The page has a pink header with various browser tabs open.

2. Create a docker file for the application and deploy it in Docker desktop application.

Dockerfile:

```
FROM python:3.6
COPY . /app
WORKDIR /app
RUN pip install -r requirements.txt
EXPOSE 5001
ENTRYPOINT [ "python" ]
CMD [ "main.py" ]
```

Thus docker file created and deployed in docker desktop.

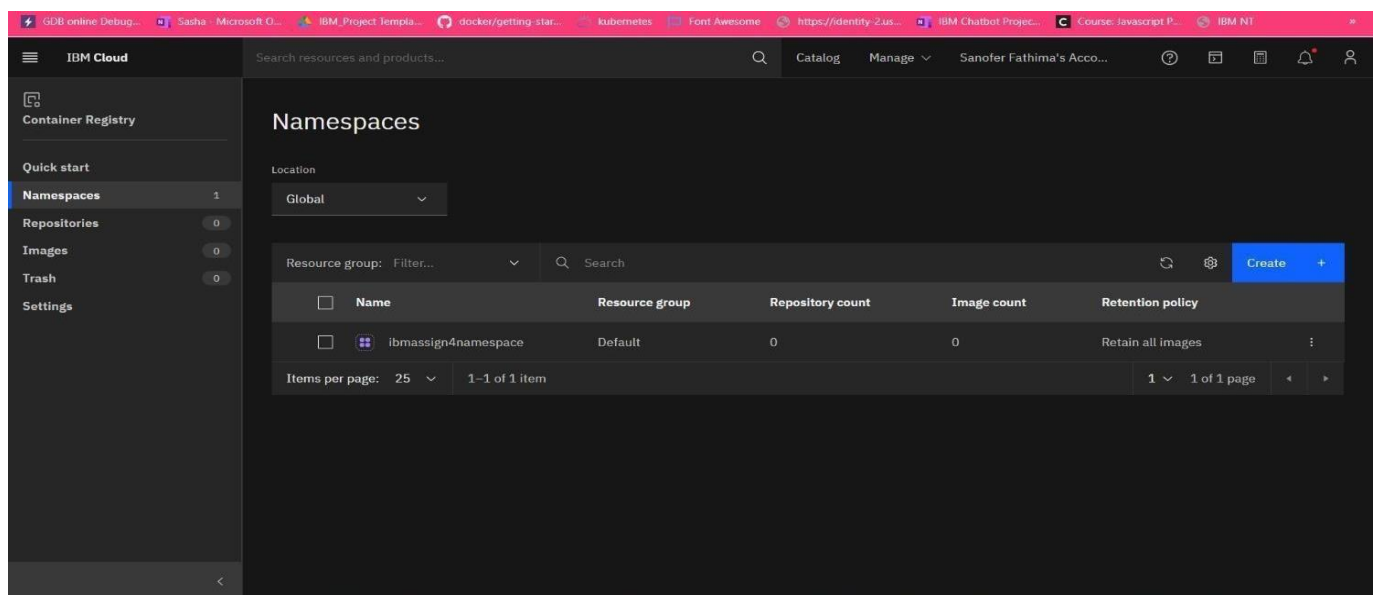


3. Create a IBM container registry and deploy hello world app.

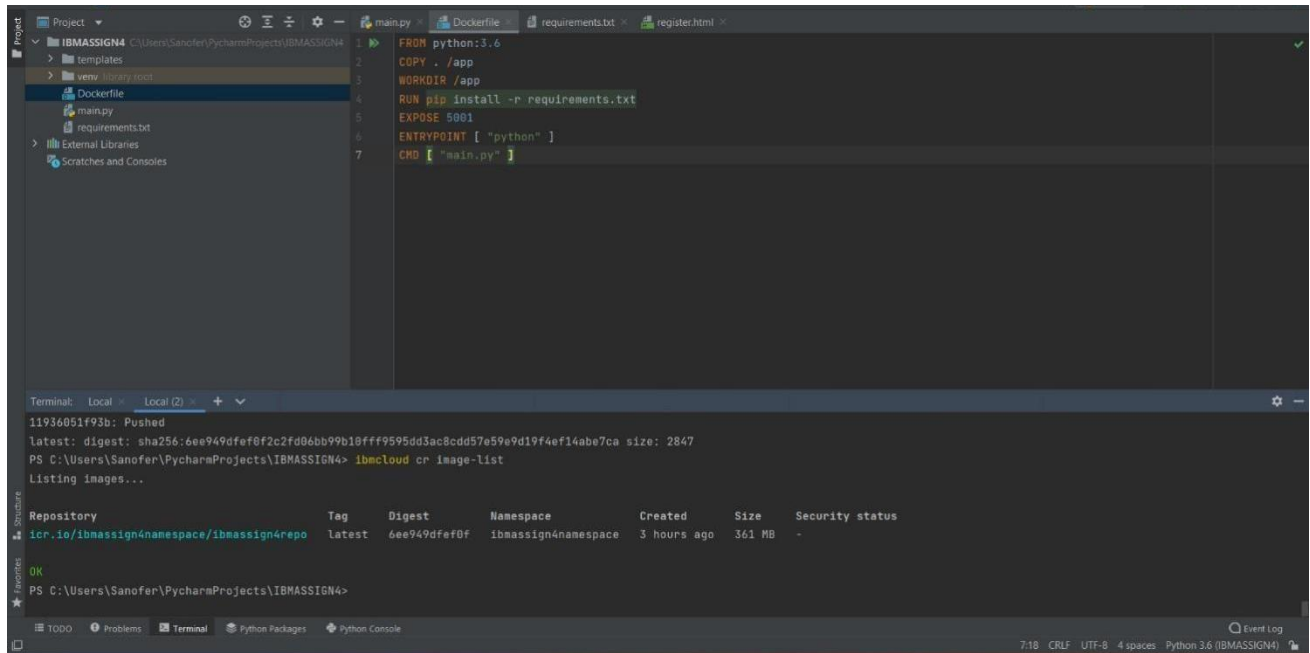
Container registry created using

```
> docker tag sanoferasheed/ibmassign4deploy:latest  
icr.io/ibmassign4namespace/ibmassign4repo:latest
```

```
> docker push icr.io/ibmassign4namespace/ibmassign4repo:latest
```



Thus, images in container registry are listed



The screenshot shows the PyCharm IDE interface. The left sidebar displays the project structure for 'IBMASSIGN4', including files like 'main.py', 'requirements.txt', and 'Dockerfile'. The main editor window shows the content of 'Dockerfile' with the following code:

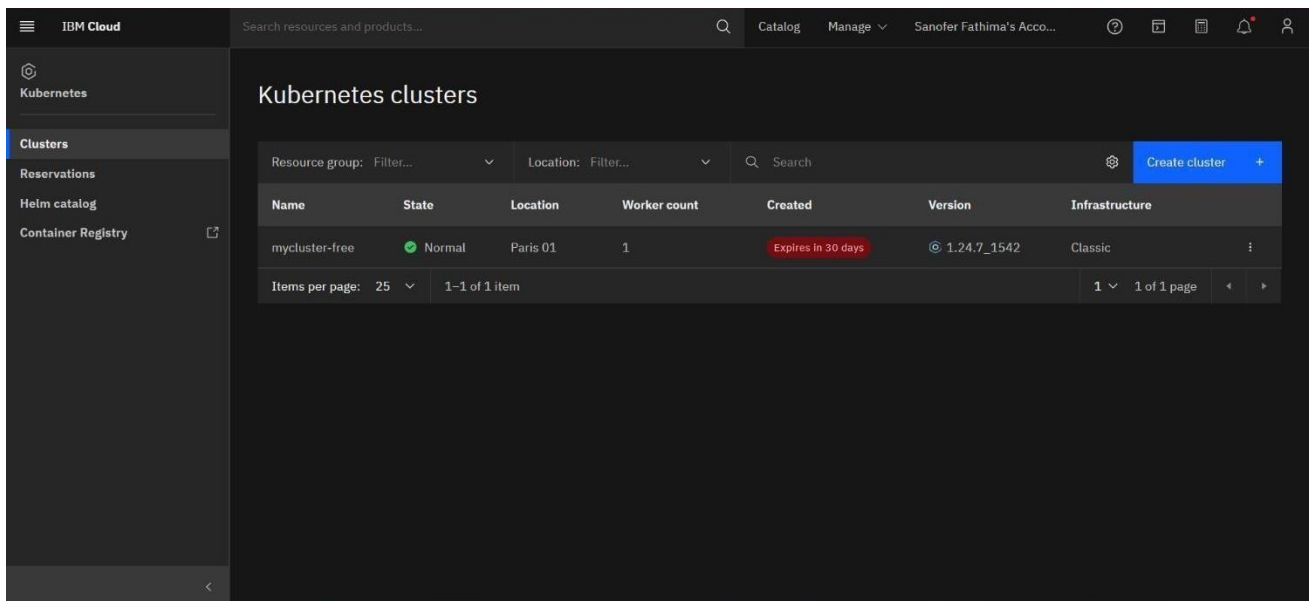
```
1 FROM python:3.6
2 COPY . /app
3 WORKDIR /app
4 RUN pip install -r requirements.txt
5 EXPOSE 5001
6 ENTRYPOINT [ "python" ]
7 CMD [ "main.py" ]
```


Below the editor, the 'Terminal' tab is active, showing the output of the 'docker push' command and the 'docker images' command. The 'docker images' command output is as follows:

Repository	Tag	Digest	Namespace	Created	Size	Security status
icr.io/ibmassign4namespace/ibmassign4repo	latest	6ee949dfe0f	ibmassign4namespace	3 hours ago	361 MB	-


4. Create a Kubernetes cluster in IBM cloud and deploy hello world image or job portal image and also expose the same app to run in node port.




Thus, cluster is created.



kubernetes

default

 Search



Service > Services

Cron Jobs

Daemon Sets

Deployments

Jobs

Pods

Replica Sets

Replication Controllers

Stateful Sets

Service

Ingresses

Ingress Classes

Services

Config and Storage

Config Maps

Persistent Volume Claims

Secrets

Services

Name	Labels	Type	Cluster IP	Internal Endpoints	External Endpoints	Created
ibmassign4appln	Show all	LoadBalancer	172.21.216.77	ibmassign4appln:5001 TCP ibmassign4appln:30878 TCP	-	7 minutes ago
kubernetes	Show all	ClusterIP	172.21.0.1	kubernetes:443 TCP kubernetes:0 TCP	-	28 minutes ago

APP IS LIVE AT <http://159.122.174.152:30089/>