

## Assignment-4

Assignment Date	28 October 2022
Student Name	T.K.Shalini Rani
Student Roll Number	913319104041
Maximum Marks	2 Marks

Question 1:

Pull an image from docker hub and run it on docker playground.

**Solution:**

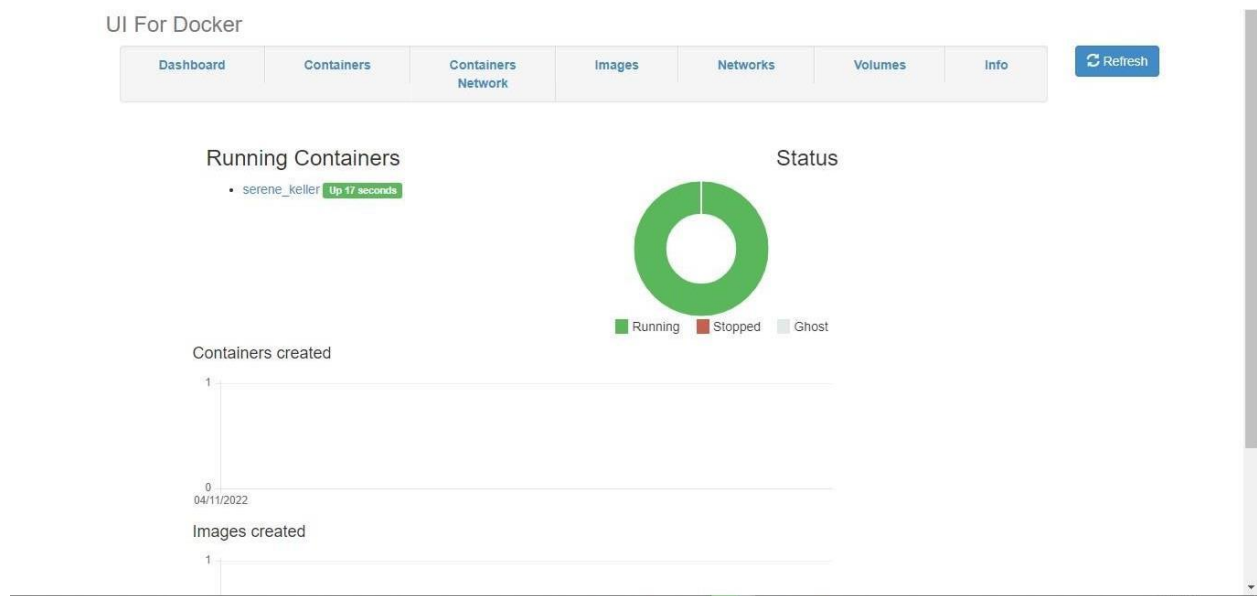
docker pull registry

docker run -d -p 9000:9000 --privileged -v

/var/run/docker.sock:/var/run/docker.sock registry

The screenshot displays the Docker Playground interface. On the left, a sidebar shows a timer at 03:38:03, a 'CLOSE SESSION' button, and a list of instances including '192.168.0.28 model'. The main panel shows details for the instance 'cdiuhke3\_cdiuofm0qau000fq8s20', including its IP (192.168.0.28), memory usage (2.29%), CPU usage (0.68%), and an SSH command. Below this, a terminal window shows the following commands and output:

```
# The FWD team.#####
[model] (local) root@192.168.0.28 ~
$ docker pull registry
Using default tag: latest
latest: Pulling from library/registry
213ec9aee27d: Pull complete
4583459ba037: Pull complete
6f6a6c5733af: Pull complete
b136d5c19b1d: Pull complete
fd4a5435f342: Pull complete
Digest: sha256:2e830e8b682d73a1b70cac4343a6a541a87d5271617841d87eeb67a824a5b3f2
Status: Downloaded newer image for registry:latest
docker.io/library/registry:latest
[model] (local) root@192.168.0.28 ~
$ docker run -d -p 9000:9000 --privileged -v /var/run/docker.sock:/var/run/docker.sock registry
7a5d897ccb6fbac91b8c46b3bb8e45510584a8ea2a26388cd65e9d5e295d2001
[model] (local) root@192.168.0.28 ~
```



## Question 2:

Create a docker file for the job portal / flask application and deploy it in docker desktop application.

## Solution:

docker - Notepad

File Edit Format View Help

```
FROM python:3.8
```

```
WORKDIR /app
```

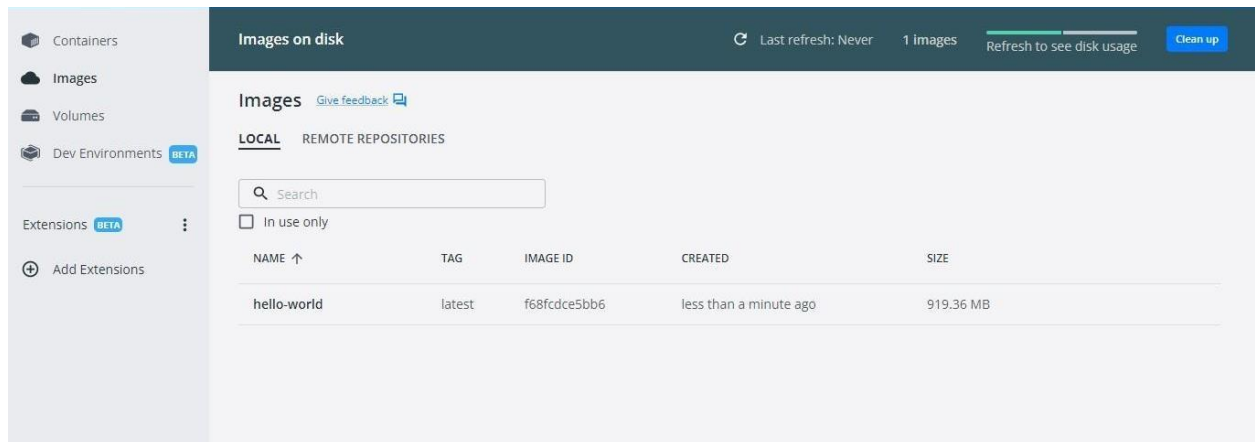
```
ADD . /app
```

```
COPY requirements.txt /app
```

```
RUN python3 -m pip install -r requirements.txt
```

```
EXPOSE 5000
```

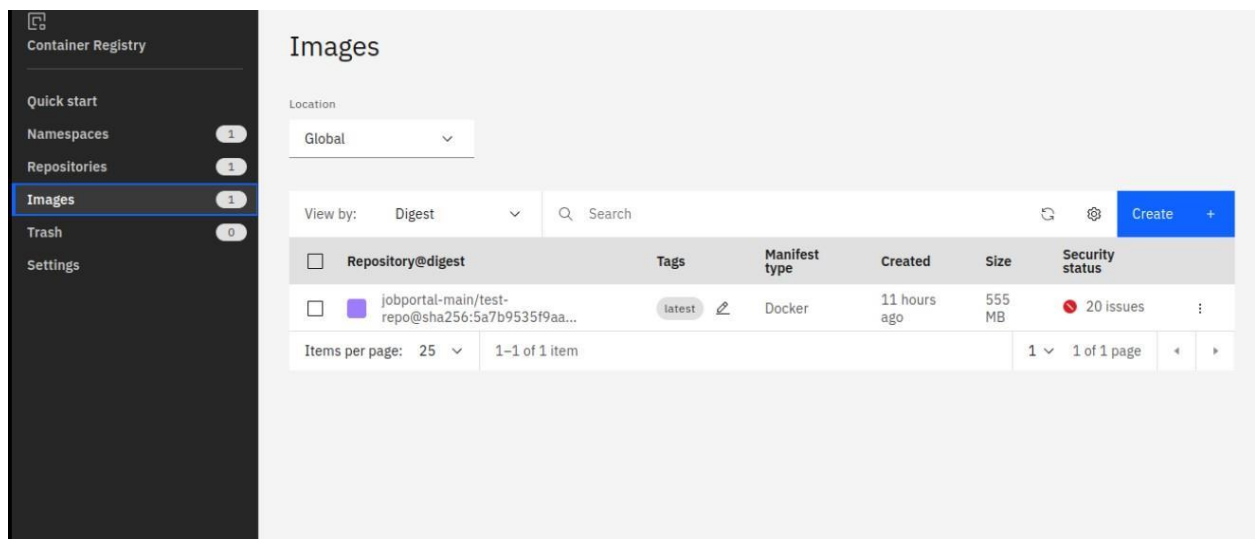
```
CMD ["python","app.py"]
```



### Question 3:

Create an IBM container registry and push docker image of flask application or job portal app.

**Solution:**



### Question 4:

Create a kubernetes cluster in IBM cloud and deploy flask application image or job portal image and also expose the same app to run in nodeport.

### Solution:

```
apiVersion:
v1kind:
Service
metadata:
name: hello-world
deploymentspec:
ports:
- port: 5000
targetPort:
5000selector: app:
hello-world
---
apiVersion:
apps/v1kind:
Deployment
metadata:
name: hello-world
deploymentspec:
replic
as: 1
select
or:
matchLabels: app:
hello
worldtemplate:
meta
da
ta
:
la
be
ls
:
app: hello
worldspec:
containers:
- name: hello-world
image: au.icr.io/hello-world-app/hello
worldimagePullPolicy: Always
ports:
- containerPort: 5000
```

Clusters / mycluster-free Normal Expires in 29 days [Add tags](#) [Help](#) [Kubernetes dashboard](#) [Actions...](#)

**Overview**

Worker nodes  
Worker pools  
DevOps New

**Expires in 29 days:**  
Be sure to back up your data, your cluster will be deleted in 29 days. To access the full capabilities of the service, try out a [standard cluster](#).

<b>Node status</b> <b>1 of 1</b> <span>Normal</span> <a href="#">Details ↓</a>	<b>Add-on status</b> <b>0 of 0</b> <span>Normal</span> <a href="#">Details ↓</a>	<b>Master status</b> <b>Normal</b> <span>✓</span> <a href="#">Docs ↗</a>	<b>Ingress status</b> <b>Unknown</b> <a href="#">Docs ↗</a>
---	---	---	---

**Details**

<b>Cluster ID</b> cd11j33f0a6mchav5k1g <a href="#">Copy</a>	<b>Version</b> 1.24.7_1542	<b>Infrastructure</b> Classic	<b>Zones</b> Milan 01
<b>Created</b> 04/11/2022, 01:12	<b>Resource group</b> Default	<b>Image security enforcement</b> <a href="#">Enable</a>	

**kubernetes** default

**Workloads > Pods > hello-world-deployment-6c75b9c898-p4ntv > Logs**

**Workloads** N

- Cron Jobs
- Daemon Sets
- Deployments
- Jobs
- Pods
- Replica Sets
- Replication Controllers
- Stateful Sets

**Service**

- Ingresses N
- Ingress Classes
- Services N

Logs from hello-world in hello-world-dep... ▼

```
* Serving Flask app 'app'
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:5000
* Running on http://172.30.82.142:5000
Press CTRL+C to quit
```