

ASSIGNMENT 4

TEAM ID : PNT2022TMID17544

Question 1:

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



03:57:32

CLOSE SESSION

Instances

+ ADD NEW INSTANCE

192.168.0.8
root

cddvksm0_cddvkvm0qau000a07j5g

IP: 192.168.0.8 OPEN PORT

Memory: 1.24% (49.52MiB / 3.906GiB)

CPU: 0.31%

SSH: ssh ip172-18-0-22-cddvksm0qau000a07j5g@direct.labs.plr

DELETE EDITOR

```
##### WARNING!!!! #####
# This is a sandbox environment. Using personal credentials
# is HIGHLY discouraged. Any consequences of doing so are
# completely the user's responsibilities.
#
# The PwD team.
#####
[redhat] (local) root@192.168.0.8 -
$ docker pull hello-world
Using default tag: latest
latest: Pulling from library/hello-world
2db29718123e: Pull complete
Digest: sha256:c181fa77aefae0d7a671ab3ec3ee05414477c951ab1a6f352a06974245fe7
Status: Downloaded newer image for hello-world:latest
docker.io/library/hello-world:latest
[redhat] (local) root@192.168.0.8 -
$ docker run hello-world
```

Activate Windows
Go to Settings to activate Windows.



03:57:05

CLOSE SESSION

Instances

+ ADD NEW INSTANCE

192.168.0.8
root

cddvksm0_cddvkvm0qau000a07j5g

IP: 192.168.0.8 OPEN PORT

Memory: 1.26% (50.45MiB / 3.906GiB)

CPU: 0.39%

SSH: ssh ip172-18-0-22-cddvkvm0qau000a07j5g@direct.labs.plr

DELETE EDITOR

```
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
   (amd64)
3. The Docker daemon created a new container from that image which runs the
   executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it
   to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/

[redhat] (local) root@192.168.0.8 -
$
```

Activate Windows
Go to Settings to activate Windows.

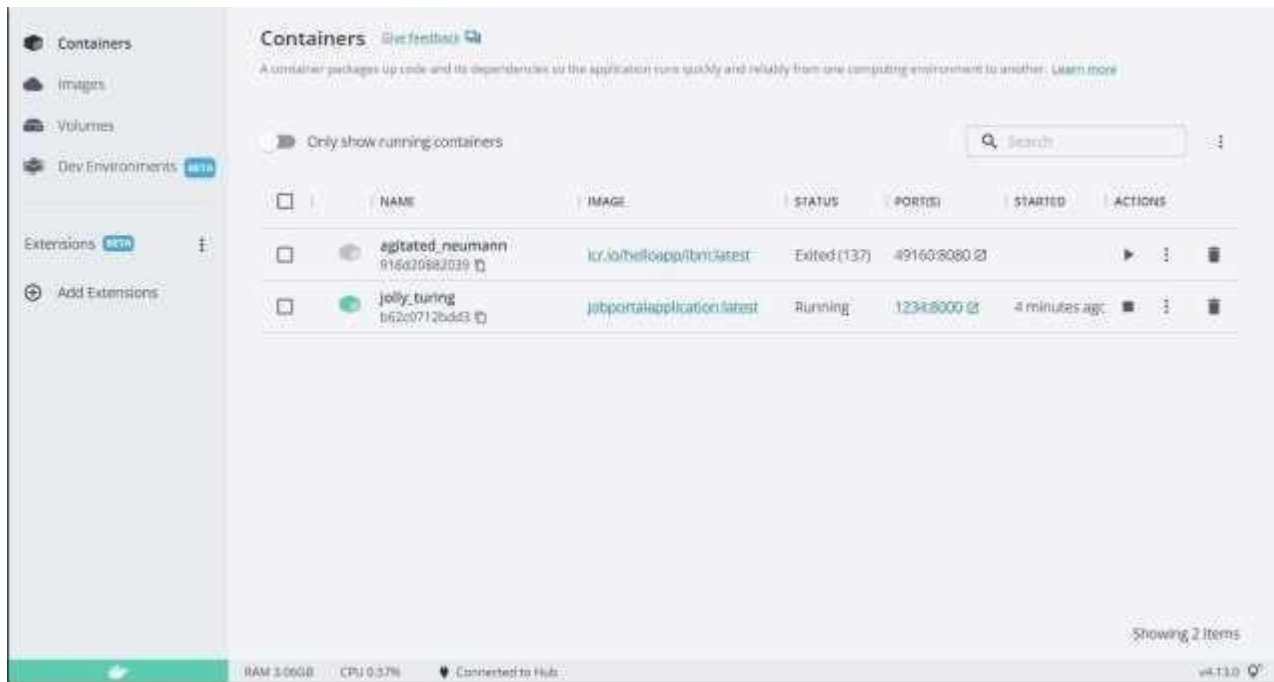
Question 2:

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

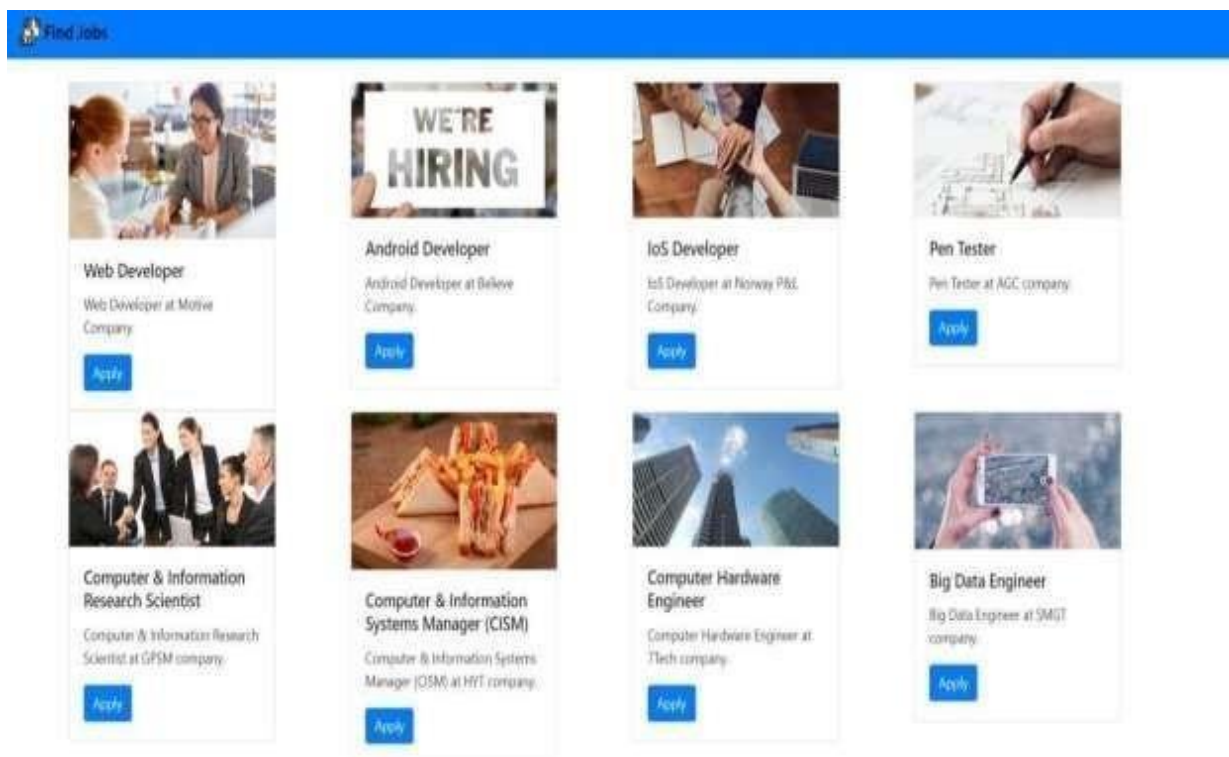
DOCKERFILE:

```
1 FROM python:3.8-buster
2
3 WORKDIR /app
4
5 COPY requirements.txt /app/
6
7 RUN pip install -r requirements.txt
8
9 COPY . /app/
10
11 RUN cp .env.dev.sample .env
12
13 EXPOSE 8000
14
15 RUN chmod +x entrypoint.sh
16
17 CMD ["sh", "entrypoint.sh"]
```

DEPLOYMENT OF JOBPORTAL APPLICATION:



OUTPUT:

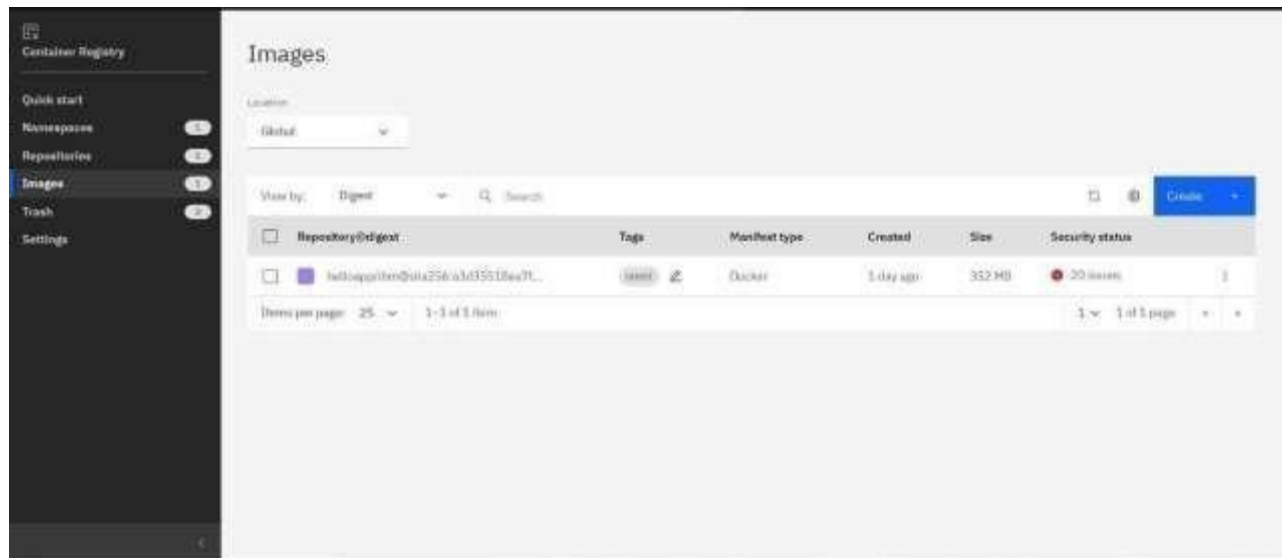


Question 3:

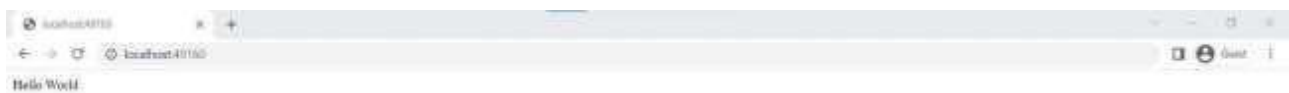
Create a IBM container registry and deploy hello world

appor jobportapp.IBM CONTAINER REGISTRY

DEPLOYMENT:

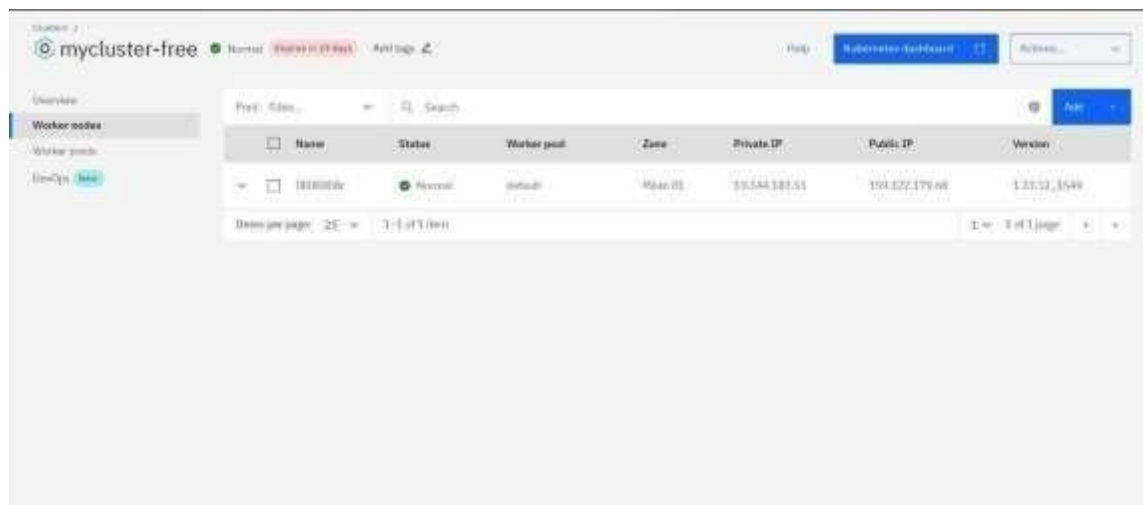


OUTPUT:



Question 4:

Create a Kubernetes cluster in IBM cloud and deploy helloworld image or jobportal image and also expose the same app to run in nodeport.
CREATING KUBERNETES CLUSTER IN IBM CLOUD AND EXPOSING NODEPORT:



OUTPUT:

