

ASSIGNMENT-4

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

03:58:22

CLOSE SESSION

Instances

+ ADD NEW INSTANCE

192.168.0.8
node1

cdrmc160_cdrmc3tccg009i26g0

IP
192.168.0.8

Memory
1.13% (45.27MiB / 3.906GiB)

CPU
18.45%

SSH
ssh ip172-18-0-18-cdrmc160qau000ahcqqg@direct.labs.pla

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 FWD team.
#####
(node1) (local) root@192.168.0.8 ~
$ docker pull hello-world
Using default tag: latest
latest: Pulling from library/hello-world
2db29710123e: Pull complete
Digest: sha256:faa03e786c97f07ef34423fccceec2398ec8a5759259f94d99078f264e9d7af
Status: Downloaded newer image for hello-world:latest
docker.io/library/hello-world:latest
(node1) (local) root@192.168.0.8 ~
$
```

03:57:54

CLOSE SESSION

Instances

+ ADD NEW INSTANCE

192.168.0.8
node1

cdrmc160_cdrmc3tccg009i26g0

IP
192.168.0.8

Memory
1.35% (53.93MiB / 3.906GiB)

CPU
4.23%

SSH
ssh ip172-18-0-18-cdrmc160qau000ahcqqg@direct.labs.pla

DELETE EDITOR

```
Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
1. The Docker client contacted the Docker daemon.
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/
(node1) (local) root@192.168.0.8 ~
$
```

Inventory Management System for Retailer

PNT2022TMID20628
Lakshmanan G-49621915061

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

```
FROM python:3.7
```

```
COPY. /app
```

```
WORKDIR /app
```

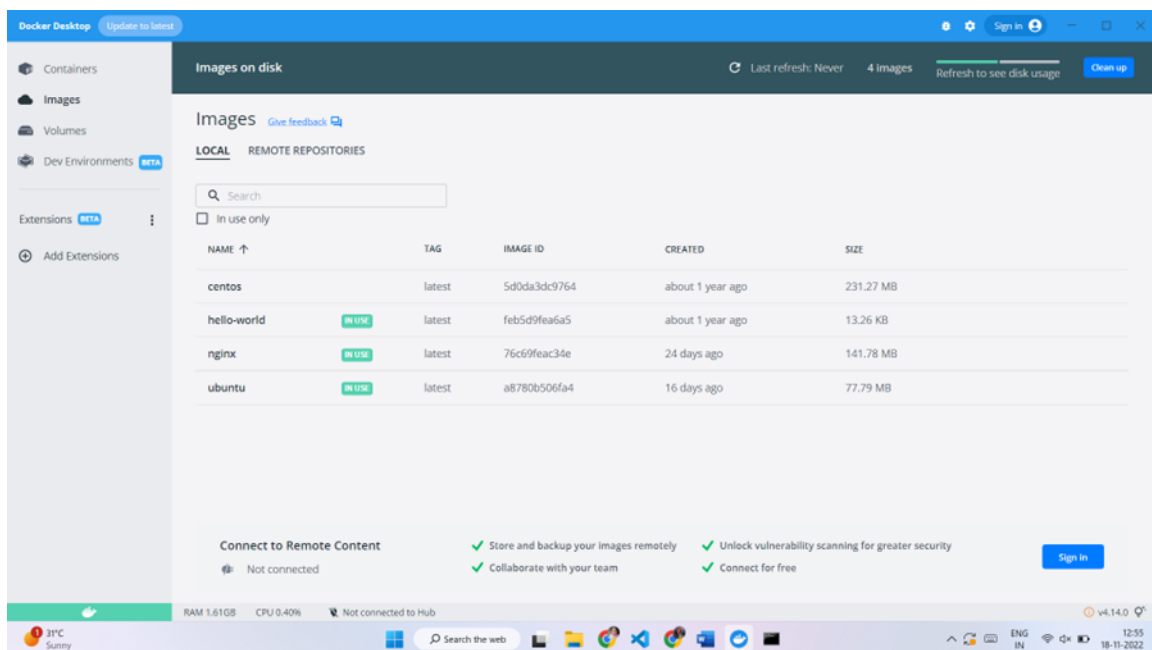
```
COPY requirements.txt /app
```

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

```
EXPOSE 5001
```

```
ENTRYPOINT [ "python" ]
```

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



Inventory Management System for Retailer

PNT2022TMID20628
Lakshmanan G-49621915061

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

The screenshot displays the IBM Cloud Container Registry console. The left sidebar shows the 'Namespaces' section with a count of 1. The main area shows a table of namespaces:

Name	Resource group	Repository count	Image count	Retention policy
lakshmanan	Default	0	0	

Below the console, a terminal window shows the output 'Hello World'.

Hello World

The Windows taskbar shows the system clock at 16:26 on 13-11-2022. The taskbar also includes the Start button, Search bar, and various application icons.

Inventory Management System for Retailer

PNT2022TMID20628
Lakshmanan G-49621915061

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.

