

Assignment - 4

| | |
|---------------------|-----------------|
| Assignment Date | 27 October 2022 |
| Student Name | ASHFAQ AHMAD P |
| Student Roll Number | 962219205011 |
| Maximum Marks | 2 Marks |

Question 1:

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

The screenshot shows the Docker Playground interface. On the left, there's a sidebar with a clock showing 03:57:32, a 'CLOSE SESSION' button, and a list of instances. The main area displays the instance 'cddvksm0_cddvkvm0qau000a07j5g' with IP 192.168.0.8. Below this, there's a terminal window showing the following commands and output:

```
[root@192.168.0.8 ~]# docker pull hello-world
Using default tag: latest
latest: Pulling from library/hello-world
0a2b73121e: Pull complete
Digest: sha256:c18fba77aefab402a71ab3c4e05d3473c951ab2a6752ab68742451e7
Status: Downloaded newer image for hello-world:latest
[root@192.168.0.8 ~]# docker run hello-world
```

The terminal output shows a warning message: "WARNING!! This is a sandbox environment. Using personal credentials is strongly discouraged. Any consequences of doing so are completely the user's responsibility. The PWD team."

The screenshot shows the Docker Playground interface with the same instance 'cddvksm0_cddvkvm0qau000a07j5g'. The terminal window now shows the output of the 'docker run hello-world' command:

```
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/

[root@192.168.0.8 ~]#
```

Question 2:

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

desktopapplication. **DOCKER FILE:**

```
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:

Containers

Images

Volumes

Dev Environments

Extensions

Add Extensions

Containers

Only show running containers

| | NAME | IMAGE | STATUS | PORT(S) | STARTED | ACTIONS |
|--------------------------|----------------------------------|---------------------------------|--------------|------------|---------------|---------|
| <input type="checkbox"/> | agitated_neumann 518d25982339 | icr.io/helloapp/helloapp:latest | Exited (137) | 49160:8080 | | |
| <input type="checkbox"/> | jolly_buring b4207712edd3 | jobportalapplication:latest | Running | 1234:8000 | 4 minutes ago | |

Showing 2 items

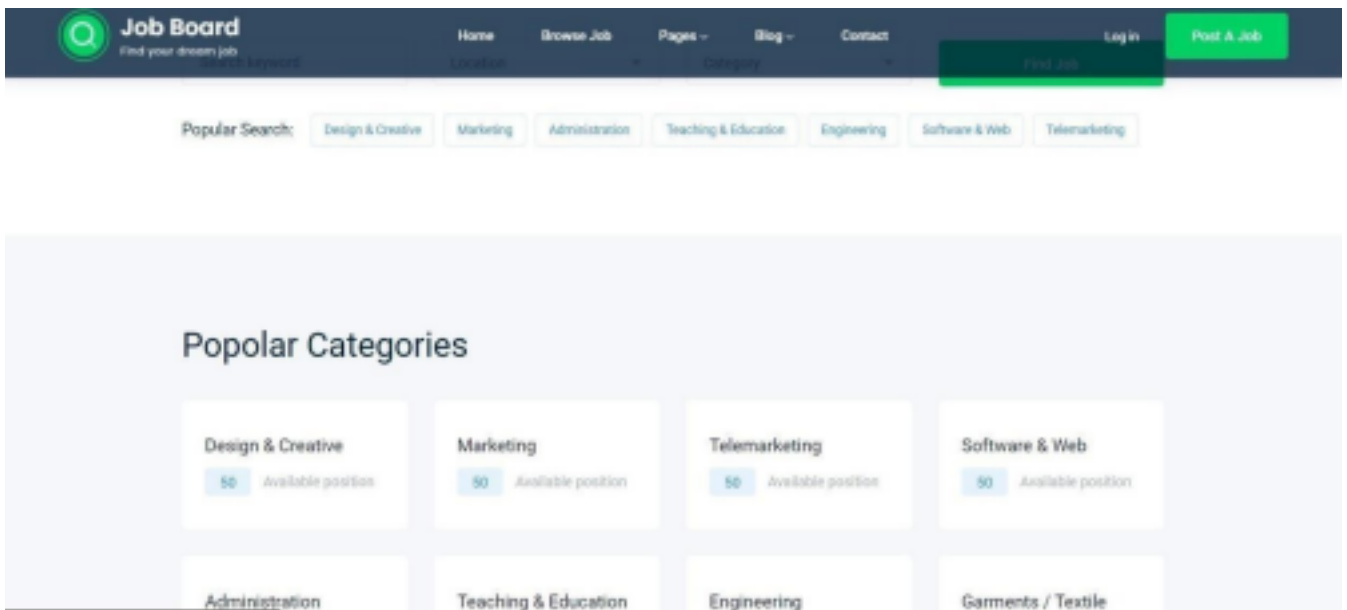
RAM 3.06GB

CPU 6.57%

Connected to Hub

v4.13.0

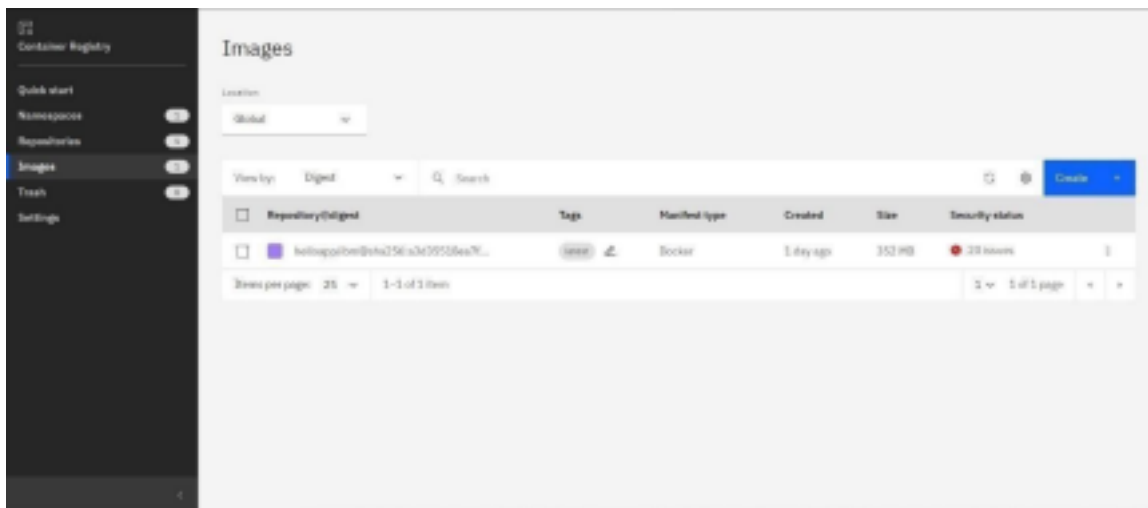
OUTPUT:



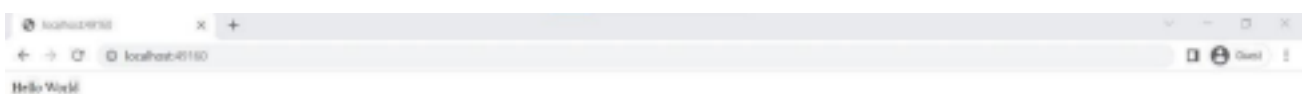
Question 3:

Create a IBM container registry and deploy hello-world app or job port

app.**IBM CONTAINER REGISTRY DEPLOYMENT:**



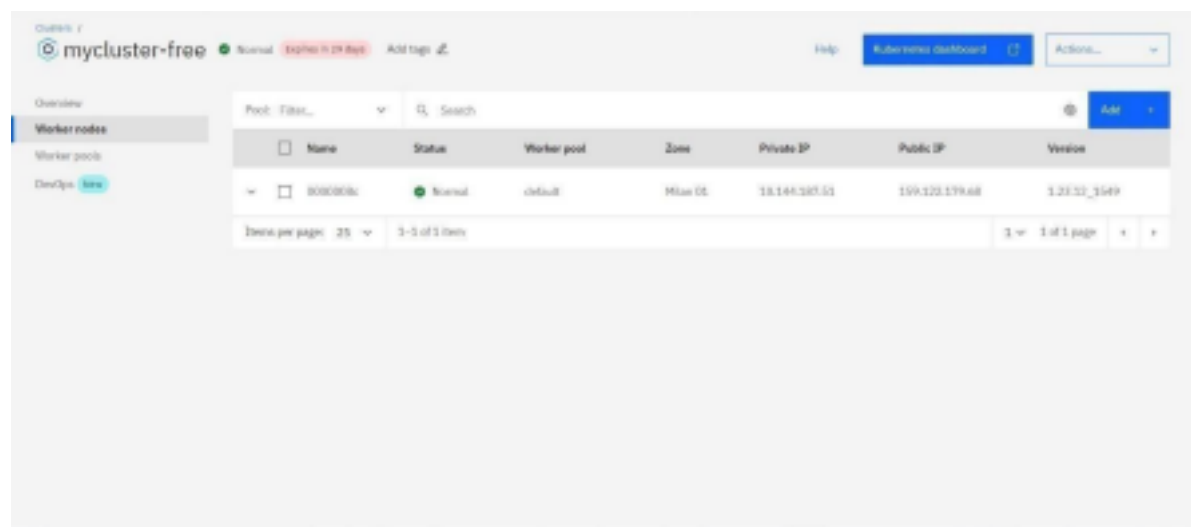
OUTPUT:



Question 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.

Creating Kubernetes cluster in IBM cloud and exposing node port:



Output:

