

Assignment 4

Assignment Date	21 September 2022
Student Name	Archana
Student Roll Number	961819104018
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 an 'Instances' section with a list of instances including '192.168.0.8 node1'. The main area displays details for a container named 'cddvkms0_cddvkvm0qau000a07j5g'. It shows the IP address '192.168.0.8', memory usage '1.24% (49.52MiB / 3.906GiB)', and CPU usage '0.31%'. Below this, there's an SSH command: 'ssh ip172-18-0-22-cddvkms0qau000a07j50@direct.labs.pla'. There are 'DELETE' and 'EDITOR' buttons. The terminal output shows a warning message, followed by the command 'docker pull hello-world', which successfully pulls the latest image from Docker Hub. The final command shown is 'docker run hello-world'.

This screenshot shows the same Docker Playground interface as the previous one, but with the terminal output of the 'hello-world' container. The output lists four steps: 1. The Docker daemon pulled the 'hello-world' image from the Docker Hub. 2. The Docker daemon created a new container from that image which runs the executable that produces the output you are currently reading. 3. The Docker daemon streamed that output to the Docker client, which sent it to your terminal. 4. The Docker daemon streamed that output to the Docker client, which sent it to your terminal. Below this, it suggests running 'docker run -it ubuntu bash' for something more ambitious. It also provides links to Docker Hub and Docker documentation. The terminal prompt is now '\$'.

Question 2

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

```
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"]
```

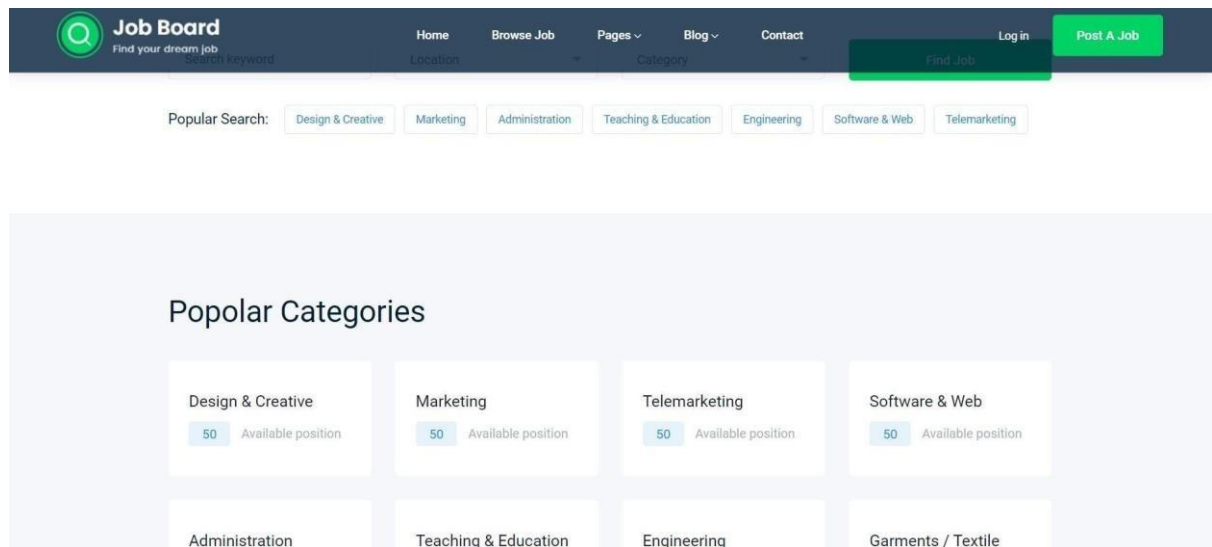
DEPLOYMENT OF JOBPORTAL APPLICATION:

The screenshot shows the Docker Desktop application interface. On the left is a sidebar with navigation options: Containers, Images, Volumes, Dev Environments (marked BETA), Extensions (marked BETA), and Add Extensions. The main area is titled 'Containers' and includes a toggle for 'Only show running containers' and a search bar. Below this is a table listing containers:

	NAME	IMAGE	STATUS	PORT(S)	STARTED	ACTIONS
<input type="checkbox"/>	agitated_neumann 918d20882039	icr.io/helloapp/ibm:latest	Exited (137)	49160:8080		▶ ⋮ 🗑
<input type="checkbox"/>	jolly_turing b62c0712bdd3	jobportalapplication:latest	Running	1234:8000	4 minutes ago	■ ⋮ 🗑

At the bottom right, it says 'Showing 2 items'. The status bar at the very bottom indicates 'RAM 3.06GB', 'CPU 0.57%', 'Connected to Hub', and 'v4.13.0'.

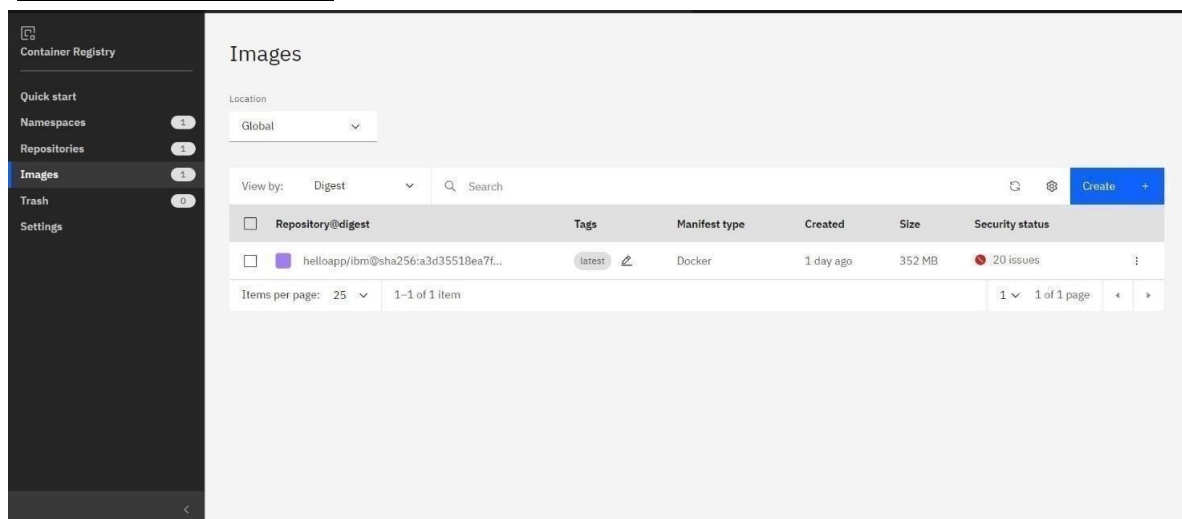
OUTPUT:



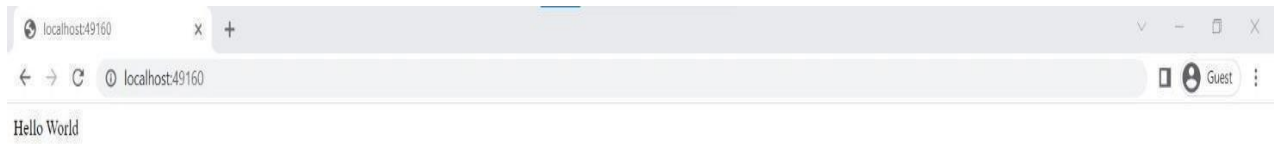
Question 3

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

DEPLOYMENT:



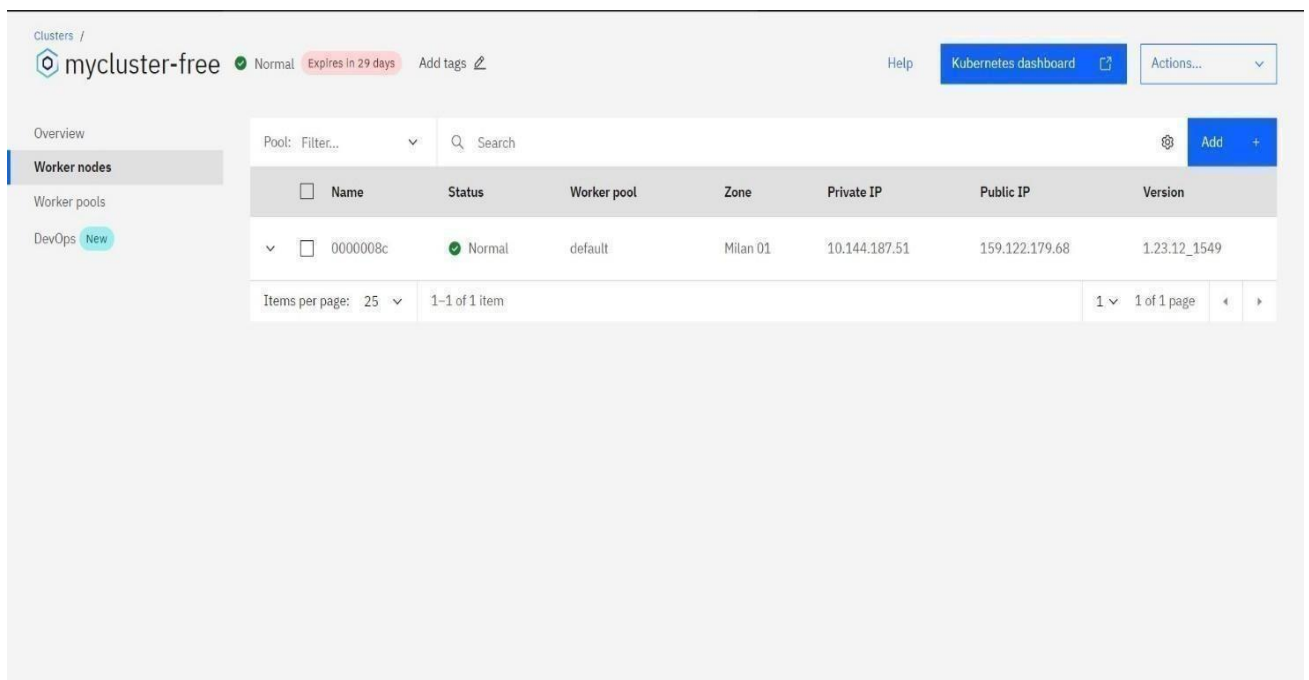
OUTPUT:



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

