

Assignment - 4

Assignment Date	7 NOVEMBER 2022
Student Name	SANTHOSH RAJ AM
TEAM ID	PNT2022TMID20737
Project	Personal Expense Tracker

Question 1:

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





Question 2:

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

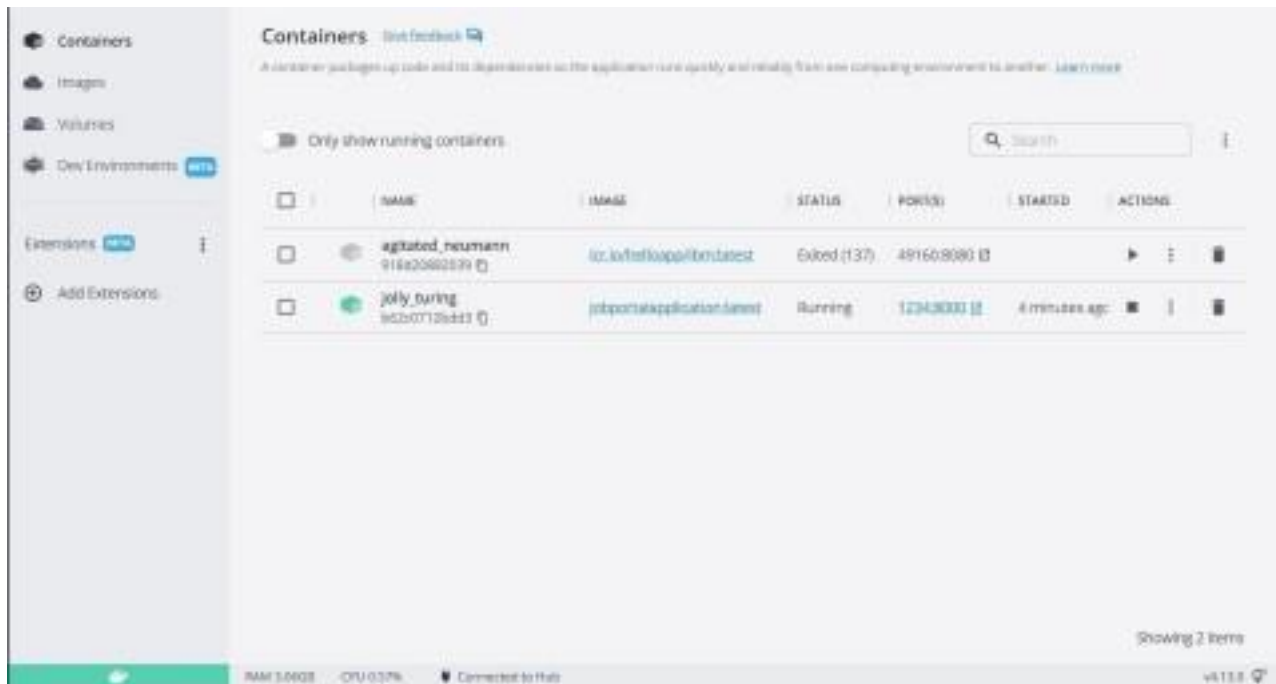
desktopapplication. DOCKER FILE:

```

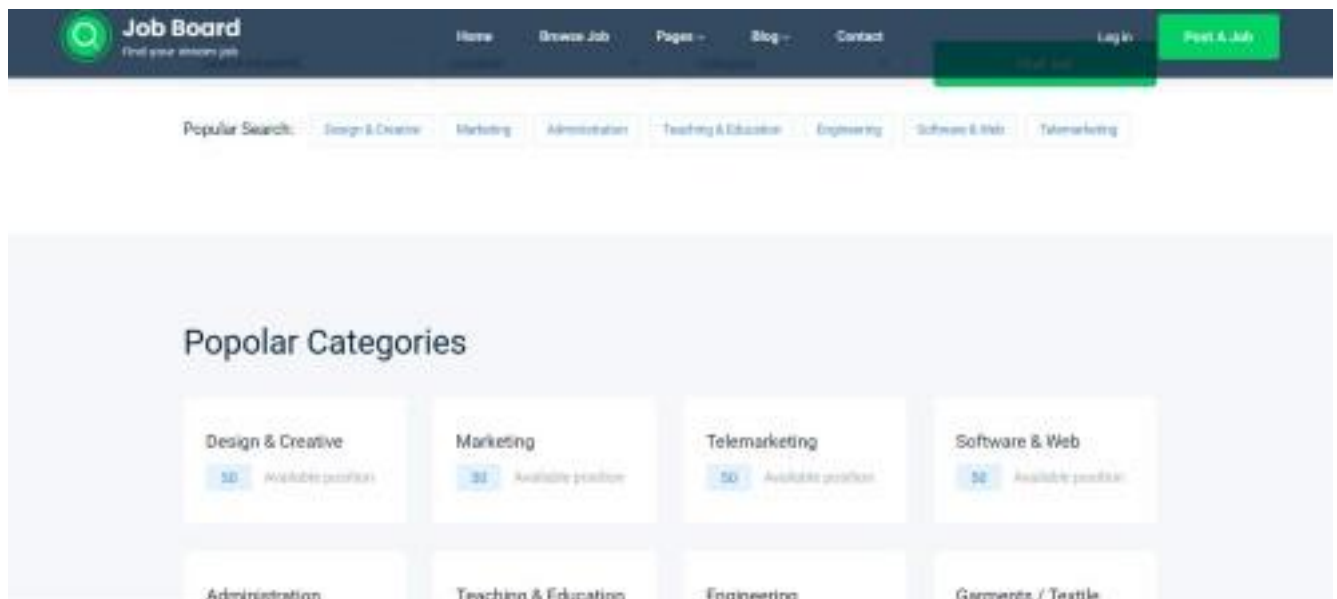
1 FROM python:3.8-buster
2
3 WORKDIR /app
4
5 COPY requirements.txt /app/
6
7 RUN pip install -r requirements.txt
8
9 COPY . /app/
10
11 RUN cp .env.dev.sample .env
12
13 EXPOSE 8000
14
15 RUN chmod +x entrypoint.sh
16
17 CMD ["sh", "entrypoint.sh"]

```

DEPLOYMENT OF JOBPORTAL APPLICATION:



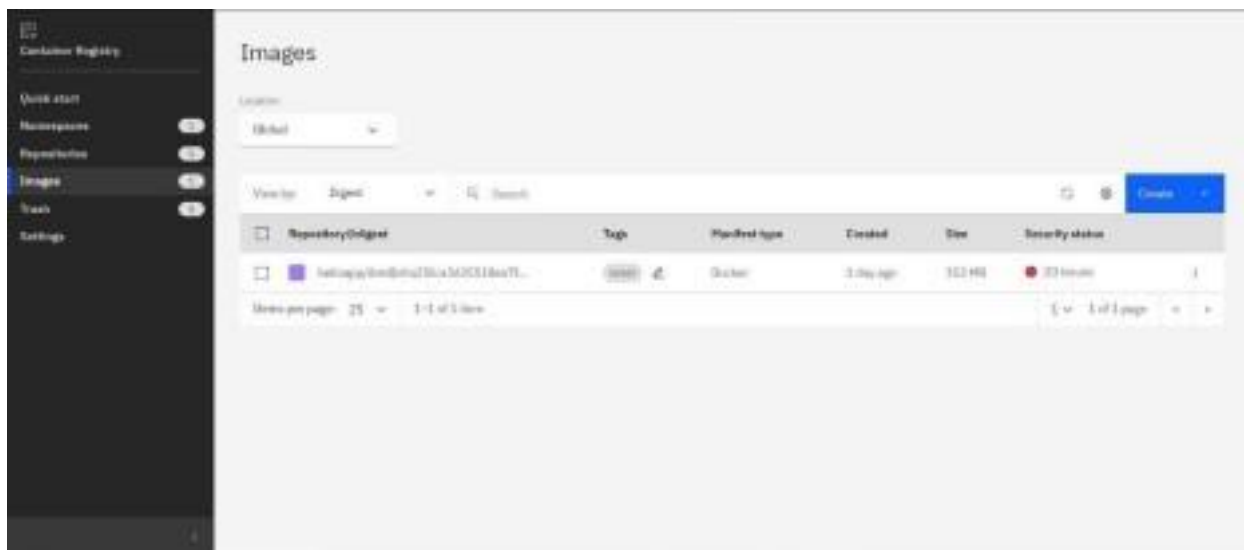
OUTPUT:



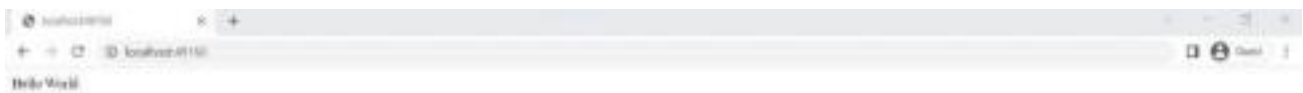
Question 3:

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:

mycluster-free

Help [Kubernetes Dashboard](#)

Overview **Worker nodes** Worker pods Test jobs

From: Filter... Search Add

Name	Status	Worker pool	Zone	Private IP	Public IP	Version
node000000	Running	default	Asia 01	10.100.187.11	194.102.196.68	1.28.12_1408

Items per page: 25 1-1 of 1 item

2 1 of 1 page

Output:

```

1  #!/bin/bash
2  # Create a Kubernetes cluster using k3s
3  # This script will create a single node cluster
4  # and install the necessary components
5  # for running a simple web application
6  # on the cluster.
7
8  # Create the cluster
9  k3s --write-kubeconfig --write-kubeconfig-path=/etc/rancher/k3s/k3s.yaml
10
11 # Install the necessary components
12 # for running a simple web application
13 # on the cluster.
14
15 # Create a namespace for the application
16 kubectl create namespace myapp
17
18 # Create a deployment for the application
19 kubectl create deployment myapp --image=nginx --namespace=myapp
20
21 # Create a service for the application
22 kubectl create service myapp --type=ClusterIP --namespace=myapp
23
24 # Create an ingress for the application
25 kubectl create ingress myapp --rule=/ --namespace=myapp
26
27 # Create a test job for the application
28 kubectl create job test --image=nginx --namespace=myapp
29
30 # Wait for the test job to complete
31 kubectl wait --for=condition=complete job/test --namespace=myapp
32
33 # Print the results of the test
34 kubectl get pods --namespace=myapp
35
36 # End of script

```

mycluster-free

Help [Kubernetes Dashboard](#)

Overview **Worker nodes** Worker pods Test jobs

From: Filter... Search Add

Name	Status	Worker pool	Zone	Private IP	Public IP	Version
node000000	Running	default	Asia 01	10.100.187.11	194.102.196.68	1.28.12_1408

Items per page: 25 1-1 of 1 item

2 1 of 1 page

