

Assignment 4

Cloud Application Development

Date	31 October 2022
Name	Harish V
Register number	211719106024
Maximum Marks	2marks

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

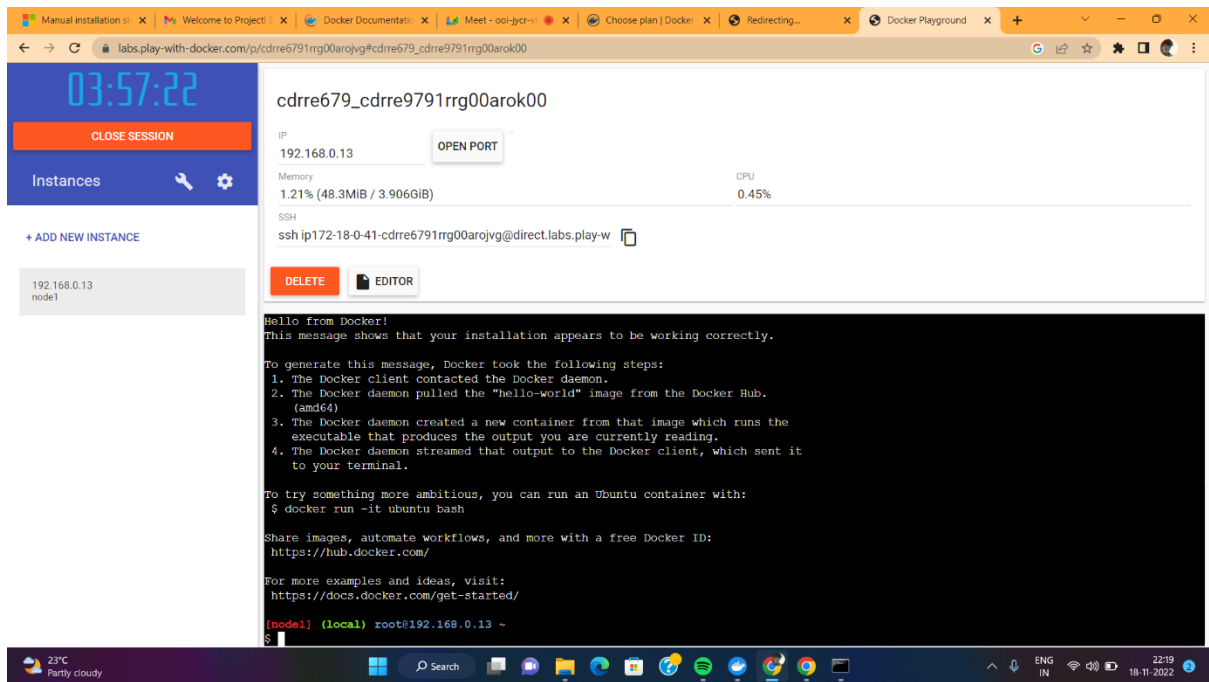
Pushed my own Image to Docker Hub and used that for this assignment.

\$docker pull hello-world

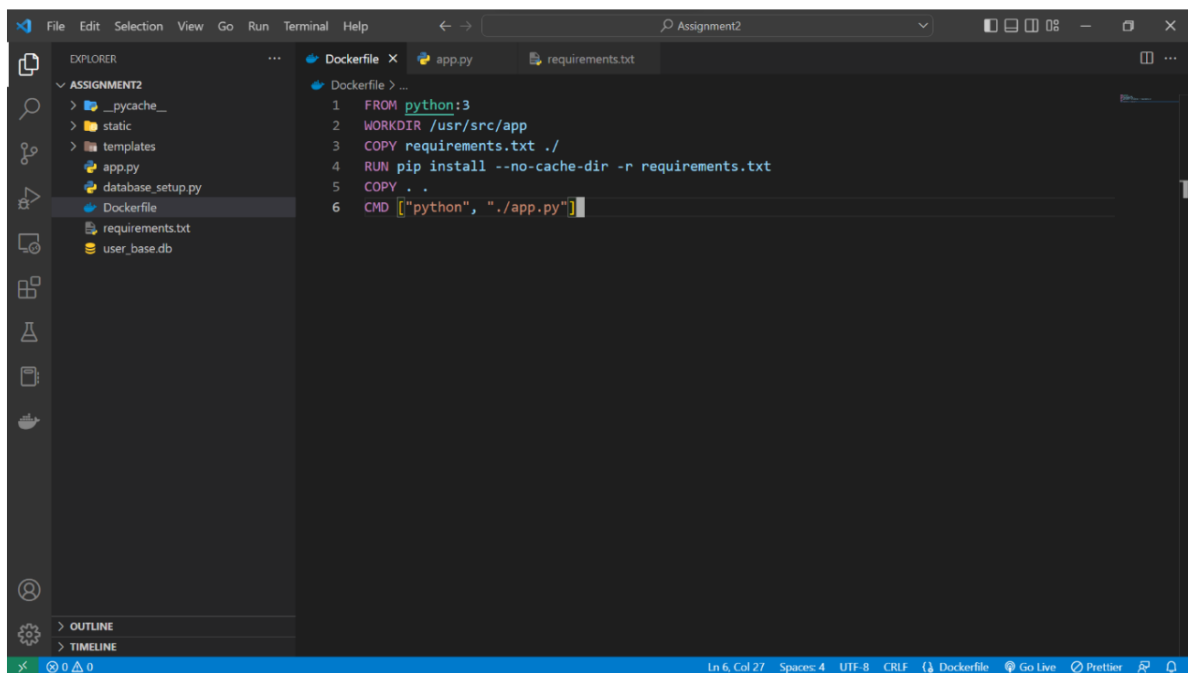
\$docker image list

The screenshot shows the Docker Playground interface in a web browser. The top navigation bar includes links for Manual Installation, Welcome to Project, Docker Documentation, Meet - ooi-jyot, Choose plan | Docker, Redirecting..., and Docker Playground. The main content area displays the instance details for 'cdrre679_cdrre9791rrg00arok00', including IP (192.168.0.13), Memory (1.12% / 3.906GiB), CPU (0.49%), and an SSH command. Below this, there are buttons for DELETE and EDITOR. The terminal window shows the following commands and output:

```
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@192.168.0.13 ~
$ 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@192.168.0.13 ~
$ docker image list
REPOSITORY TAG IMAGE ID CREATED SIZE
hello-world latest feb5d9fea6a5 14 months ago 13.3kB
(node1) (local) root@192.168.0.13 ~
$
```



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



\$docker build -t hello-world

\$docker image list

```
C:\windows\system32\cmd.exe
Microsoft Windows [Version 10.0.22000.1219]
(c) Microsoft Corporation. All rights reserved.

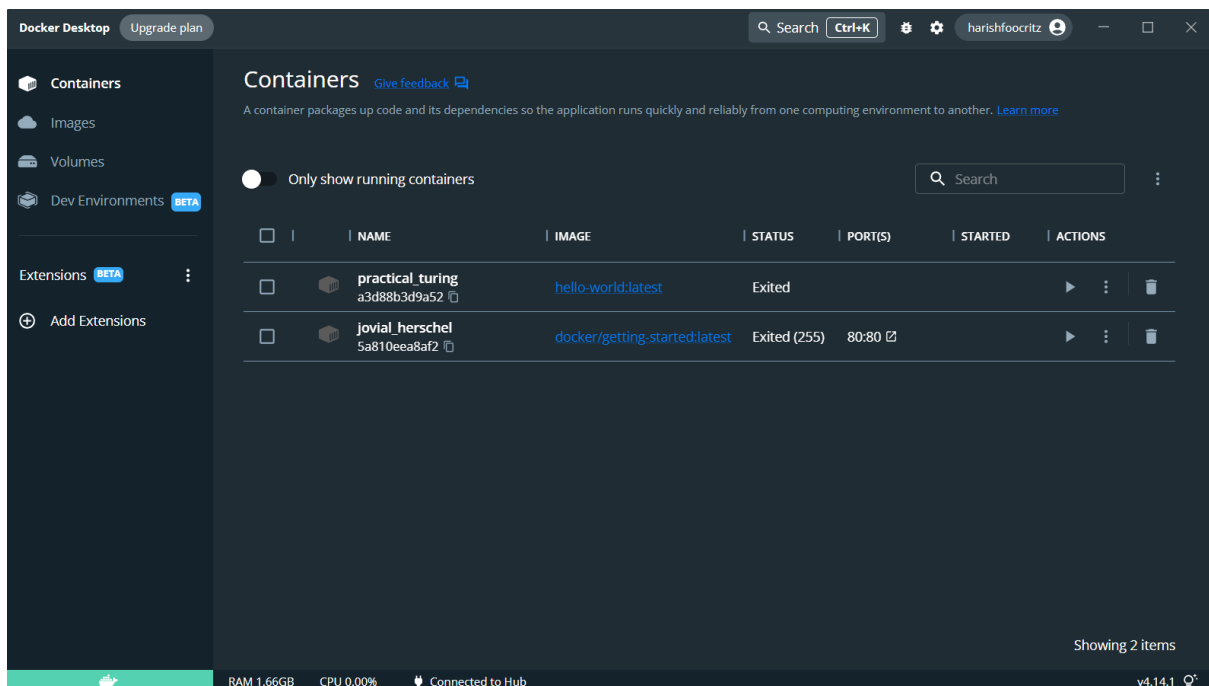
C:\Users\itsme>docker build -t hello-world
["docker build" requires exactly 1 argument.
See 'docker build --help'.

Usage:  docker build [OPTIONS] PATH | URL | -

Build an image from a Dockerfile

C:\Users\itsme>docker image list
REPOSITORY          TAG         IMAGE ID      CREATED        SIZE
docker/getting-started latest      cb90f98fd791  7 months ago  28.8MB
hello-world         latest     feb5d9fea6a5  14 months ago  13.3kB

C:\Users\itsme>
```



3.Create a IBM container registry and deploy helloworld app or jobportalapp.

Pushed the image to ibm container registry.

ibmcloud login

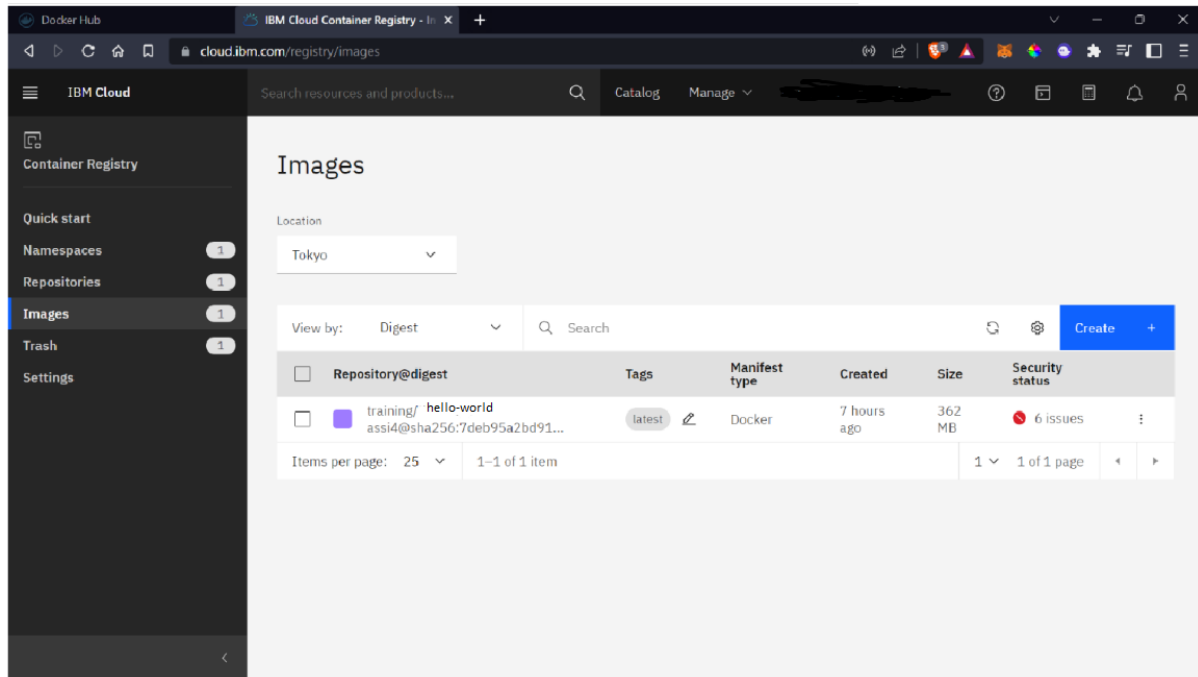
ibmcloud plugin install container-registry -r "IBM Cloud"

ibmcloud cr namespace-add training

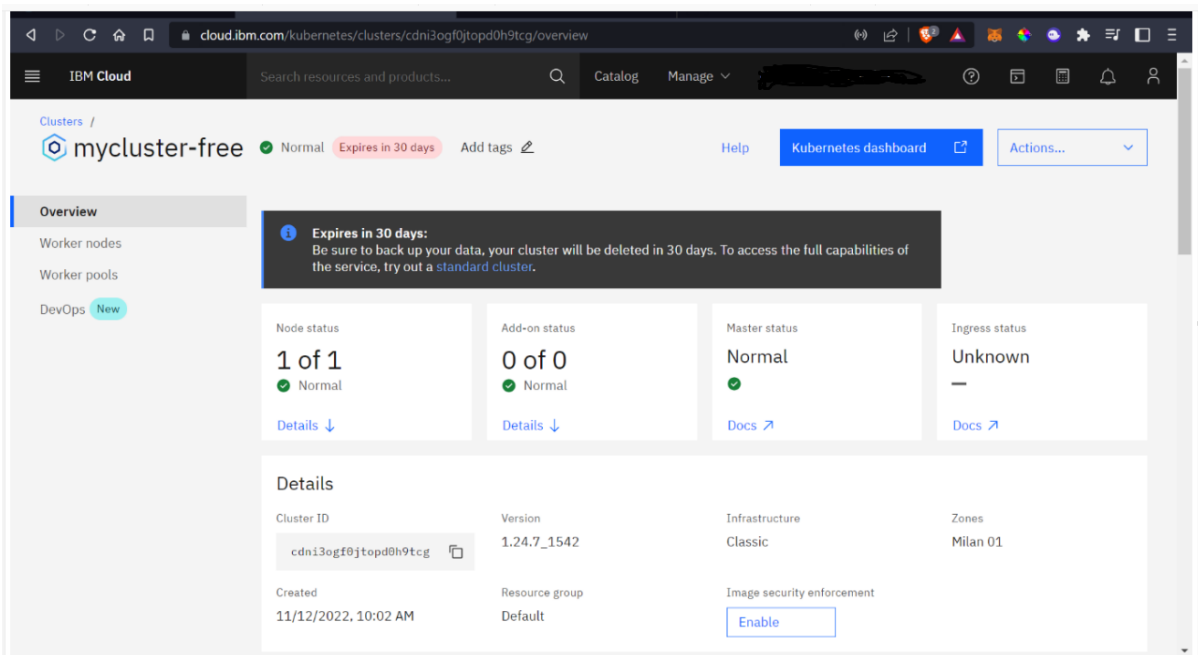
ibmcloud cr login

docker tag hello-world jp.icr.io/training/hello-world:latest

docker push jp.icr.io/training/hello-world:latest



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.



ibmcloud plugin install container-service

ibmcloud ks cluster config --cluster cdni3ogf0jtopd0h9tcg

kubectl config current-context

ibm_deployment.yaml

apiVersion: apps/v1

kind: Deployment

metadata:

name: hello-world

spec:

replicas: 5

selector:

matchLabels:

app: hello-world

template:

metadata:

labels:

app: hello-world

spec:

containers:

- name: hello-world-container

image: jp.icr.io/training/hello-world

imagePullPolicy: Always

ports:

- containerPort: 5000

protocol: TCP

hello-world_service.yaml

```
apiVersion: v1
kind: Service
metadata:
  name: hello-world-service
spec:
  type: ClusterIP
  ports:
    - port: 5000
  selector:
    app: hello-world
```

hello-world_ingress.yaml

```
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: hello-world-ingress
  annotations:
    kubernetes.io/ingress.class: nginx
    nginx.ingress.kubernetes.io/ssl-redirect: "false"
spec:
  # ingressClassName: nginx
  rules:
    - http:
        paths:
          - backend:

```

service:

name: hello-world-service

port:

number: 5000

path: /

pathType: Prefix

kubectrl apply -f kubernetes/ibm_deployment.yaml

kubectrl apply -f kubernetes/hello-world_service.yaml

kubectrl apply -f kubernetes/hello-world_ingress.yaml

kubectrl expose deployment hello-world --type=NodePort --name=hello-world