

**Assignment - 4**  
Kubernetes/Docker

Assignment Date	9 September 2022
Student Name	Dhanesh B
Student Roll Number	111519205006
Maximum Marks	2 Marks

## Question 1:

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

03:55:01

CLOSE SESSION

Instances

+ ADD NEW INSTANCE

192.168.0.18  
node1

cdqjc0m0\_cdqjc360qau0009ecni0

IP  
192.168.0.18

OPEN PORT

Memory

CPU

SSH  
ssh ip172-18-0-56-cdqjc0m0qau0009ecnhg@direct.labs.pla

DELETE

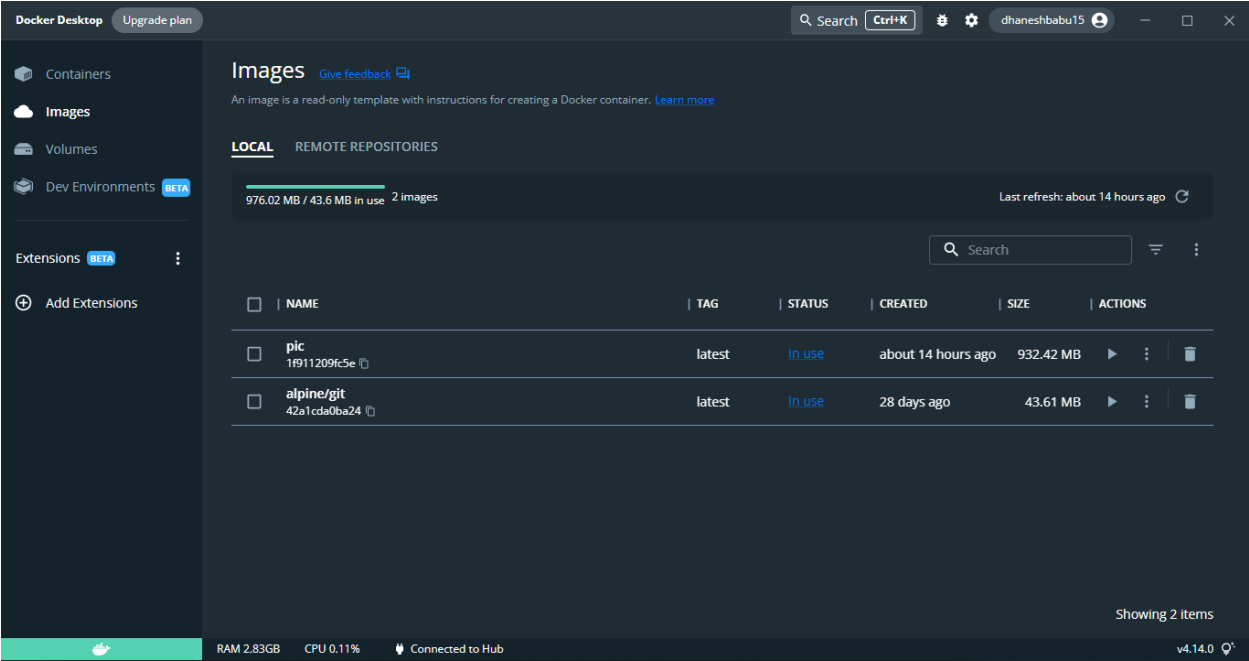
EDITOR

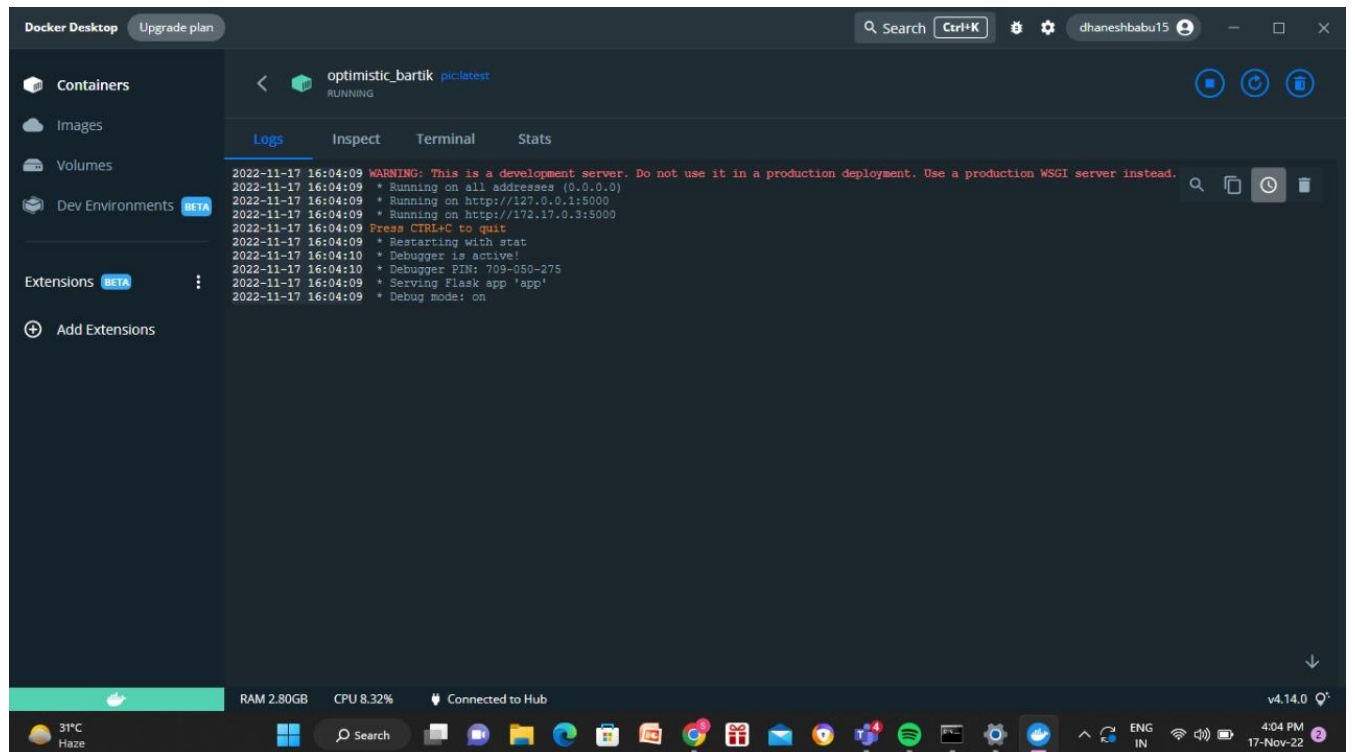
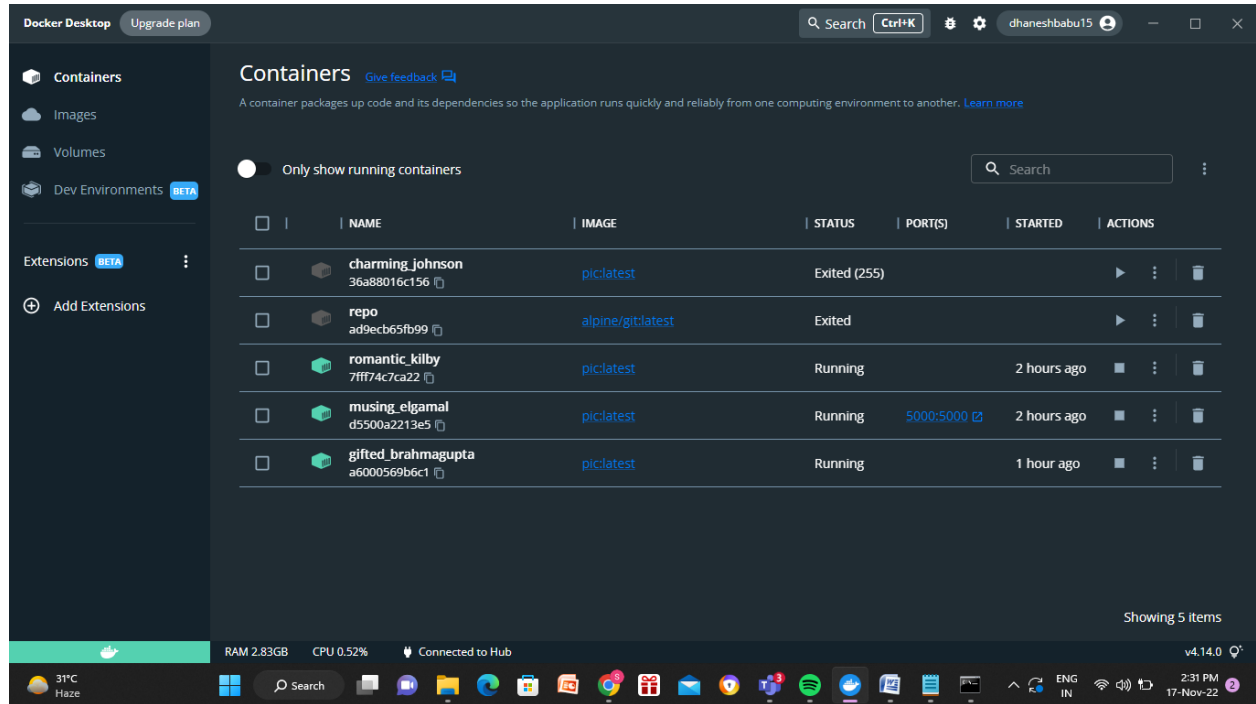
```
#####
[node1] (local) root@192.168.0.18 ~
$ docker run -d -p 80:80 docker/getting-started
Unable to find image 'docker/getting-started:latest' locally
latest: Pulling from docker/getting-started
4f9b9388f04a: Pull complete
5867c8a5fcbd: Pull complete
4b639e65cb3b: Pull complete
061ed9e2b976: Pull complete
bc19f3e8eeb1: Pull complete
4071be97c256: Pull complete
79b586f1a54b: Pull complete
0c9732f525d6: Pull complete
Digest: sha256:b558be874169471bd4e65bd6eac8c303b271a7ee8553ba47481b73b2bf597aee
Status: Downloaded newer image for docker/getting-started:latest
7dfdefc1ac7004d41ac96127b2c8cc10ff2bab808630c71387aa4de85dd59276
[node1] (local) root@192.168.0.18 ~
$
```

```
Command Prompt
C:\Users\Siva\Desktop\flask with form_and_docker-main>docker build -t firstimage .
[+] Building 79.0s (11/11) FINISHED
-> [internal] load build definition from Dockerfile
-> [internal] load .dockerignore
-> [internal] load metadata for docker.io/library/python:3.10.6
-> [auth] library/python:pull token for registry-1.docker.io
-> [internal] load build context
-> [internal] transferring context: 10.98kB
-> [1/5] FROM docker.io/library/python:3.10.6@sha256:745efdfb7e4aac9a8422bd8c62d8bc35a693e8979a240d29677cb03e6aa
-> resolve docker.io/library/python:3.10.6@sha256:745efdfb7e4aac9a8422bd8c62d8bc35a693e8979a240d29677cb03e6aa
-> sha256:025a6e380b10283603ff696d777bba5c0b10126fb0be7d118b9574946bcf84 8.53kB / 8.53kB
-> sha256:1671565cc0d4f8c365c9b6d1d3f6c164e72d01f100430c0179580420f990da2e 55.01MB / 55.01MB
-> sha256:3e94d13e5e7a4ef17ff21376f57fb95c7a1706031f8704aa99260968d81f6e4 5.16MB / 5.16MB
-> sha256:fab5c7528c685216129e8e67bf362a7702e7b1daa585ab85546a41508836657d6 10.88MB / 10.88MB
-> sha256:745efdfb7e4aac9a8422bd8c62d8bc35a693e8979a240d29677cb03e6aa01052 2.35kB / 2.35kB
-> sha256:8d1f943ceaaf3b3c0e05d5c0926e7958836b048b700176bf9c56d8f37ac13fca 2.22kB / 2.22kB
-> sha256:53ad072f9cd16fc8eb93b182b20e758e11acc0ef60babe0bf1043c08de1901a 54.58MB / 54.58MB
-> sha256:d6b983117533b718374f1701ef593dd2afa6613c7908c6553be8e2a150e6448a 196.79MB / 196.79MB
-> sha256:d8092d56ded5476fe7c302256eb4dc6ff495ae8fb4dd28aa18dbcb7581e24a6c 6.29MB / 6.29MB
-> extracting sha256:1671565cc0d4f8c365c9b6d1d3f6c164e72d01f100430c0179580420f990da2e 3.3%
-> sha256:c71afc637d59adc44c5fd3c348504df82b35bb204f0057ea22c6ac8a1d285a5 20.02MB / 20.02MB
-> extracting sha256:3e94d13e5e7a4ef17ff21376f57fb95c7a1706031f8704aa99260968d81f6e4 0.4%
-> sha256:864a10b3c704553e08cb5fcd12fb8ee1c07048f6365f0fa35e84a285413da40b 234B / 234B
-> sha256:4334b2fe8293d19ddc1c3550093aae88f21601a7c05a31c6da6c0dc48fb6ed3c 3.04MB / 3.04MB
-> extracting sha256:fab5c7528c685216129e8e67bf362a7702e7b1daa585ab85546a41508836657d6 0.4%
-> extracting sha256:53ad072f9cd16fc8eb93b182b20e758e11acc0ef60babe0bf1043c08de1901a 3.3%
-> extracting sha256:d6b983117533b718374f1701ef593dd2afa6613c7908c6553be8e2a150e6448a 7.5%
-> extracting sha256:d8092d56ded5476fe7c302256eb4dc6ff495ae8fb4dd28aa18dbcb7581e24a6c 0.7%
-> extracting sha256:c71afc637d59adc44c5fd3c348504df82b35bb204f0057ea22c6ac8a1d285a5 1.2%
-> extracting sha256:864a10b3c704553e08cb5fcd12fb8ee1c07048f6365f0fa35e84a285413da40b 0.0%
-> extracting sha256:4334b2fe8293d19ddc1c3550093aae88f21601a7c05a31c6da6c0dc48fb6ed3c 0.5%
-> [2/5] WORKDIR /app
-> [3/5] COPY requirements.txt ./
-> [4/5] RUN pip install -r requirements.txt
-> [5/5] COPY . .
-> exporting to image
-> exporting layers
-> writing image sha256:1f911200fc5ebdbfa750060822403b74978629379178f5f0239a3006f322314f
-> naming to docker.io/library/firstimage
-> 0.0%
```

## Question 2:

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





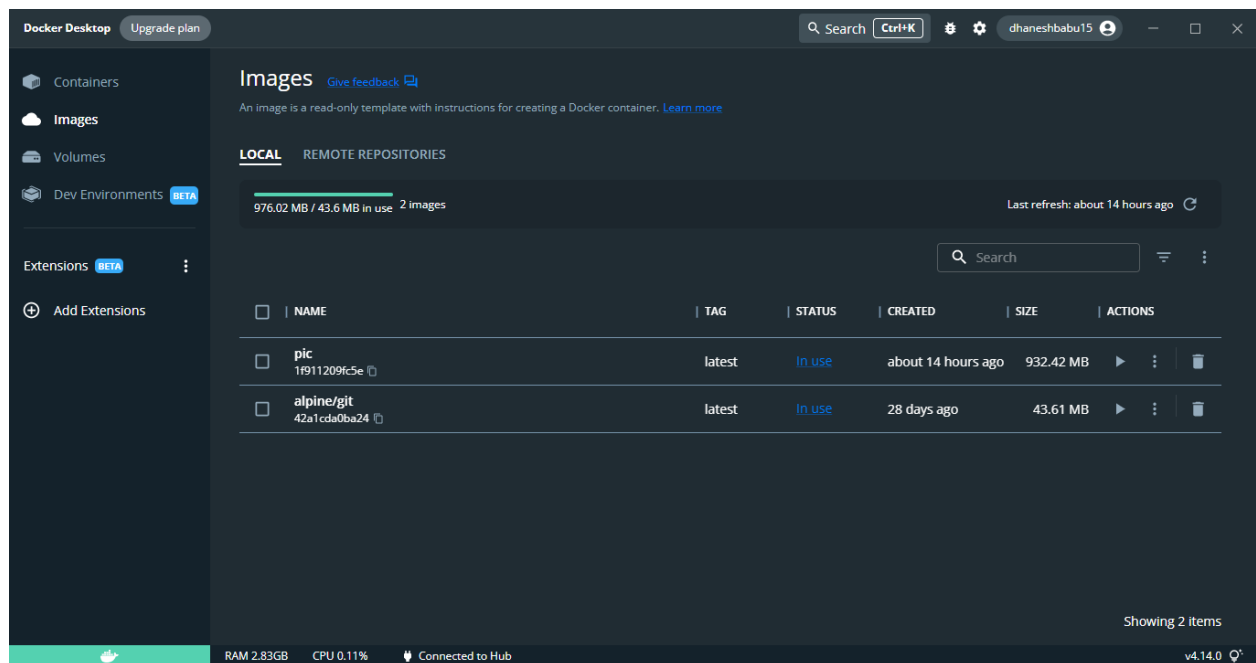
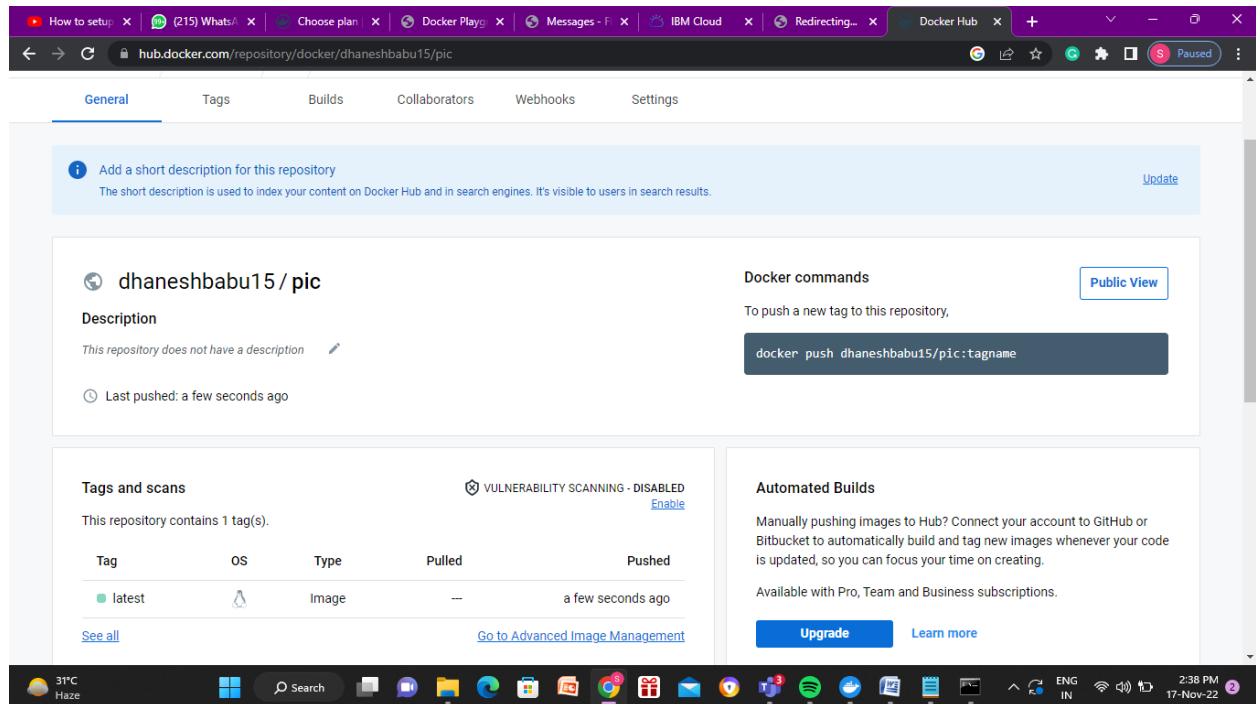
# Blog Page

## Messages

<div>Message One</div> <div>Message One Content</div>
<div>Message Two</div> <div>Message Two Content</div>

## Question 3:

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



IBM Cloud

Search resources and products...

CatalogManageDhanesh B's Account

Container Registry

Quick start

Namespaces1

Repositories0

Images0

Trash0

Settings

Namespaces

LocationTokyo

Resource group: Filter...SearchCreate

Name	Resource group	Repository count	Image count	Retention policy
☐ dhanesh	Default	0	0	Retain all images

Items per page: 251-1 of 1 item11 of 1 page

IBM Cloud

Search resources and products...

CatalogManageDhanesh B's Account

Container Registry

Quick start

Namespaces1

Repositories1

Images1

Trash0

Settings

Repositories

LocationTokyo

SearchCreate

Name	Image count	Namespace	Last updated
☑ repo1 jp.icr.io/dhanesh/repo1	1	dhanesh	1 day ago

Items per page: 251-1 of 1 item11 of 1 page

## 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 nodeport.

The screenshot shows the Kubernetes dashboard interface. The breadcrumb navigation indicates the path: Workloads > Pods > sample-app-d9bfd84d9-fp74z. The left sidebar contains a menu with categories: Workloads (Cron Jobs, Daemon Sets, Deployments, Jobs, Pods, Replica Sets, Replication Controllers, Stateful Sets), Service (Ingresses, Ingress Classes, Services), and Config and Storage (Config Maps, Persistent Volume Claims, Secrets, Storage Classes). The main content area displays the details for the selected Pod:

- Metadata:** Name: sample-app-d9bfd84d9-fp74z, Namespace: default, Created: Oct 27, 2022, Age: a day ago, UID: 3f3b4ff6-4fa6-4f07-9454-35acb2c91631. Labels: app: sample-app, pod-template-hash: d9bfd84d9.
- Resource information:** Node: docker-desktop, Status: ImagePullBackOff, IP: 10.1.0.48, QoS Class: BestEffort, Restarts: 0, Service Account: default.
- Conditions:** A table showing the pod's status history.

Type	Status	Last probe time	Last transition time	Reason	Message
Initialized	True	-	a day ago	-	-
Ready	False	-	a day ago	ContainersNotReady	containers with unready status: [sample-app-container]

The screenshot shows the Kubernetes dashboard interface. The breadcrumb navigation indicates the path: Workloads > Deployments > sample-app. The left sidebar is identical to the previous screenshot. The main content area displays the details for the selected Deployment:

- Metadata:** Name: sample-app, Namespace: default, Created: Oct 27, 2022, Age: a day ago, UID: 9699564b-f097-4168-be80-31f40116a0fc. Annotations: deployment.kubernetes.io/revision: 1, kubectrl.kubernetes.io/last-applied-configuration.
- Resource information:** Strategy: RollingUpdate, Min ready seconds: 0, Revision history limit: 10. Selector: app: sample-app.
- Rolling update strategy:** Max surge: 25%, Max unavailable: 25%.