

Assignment -4

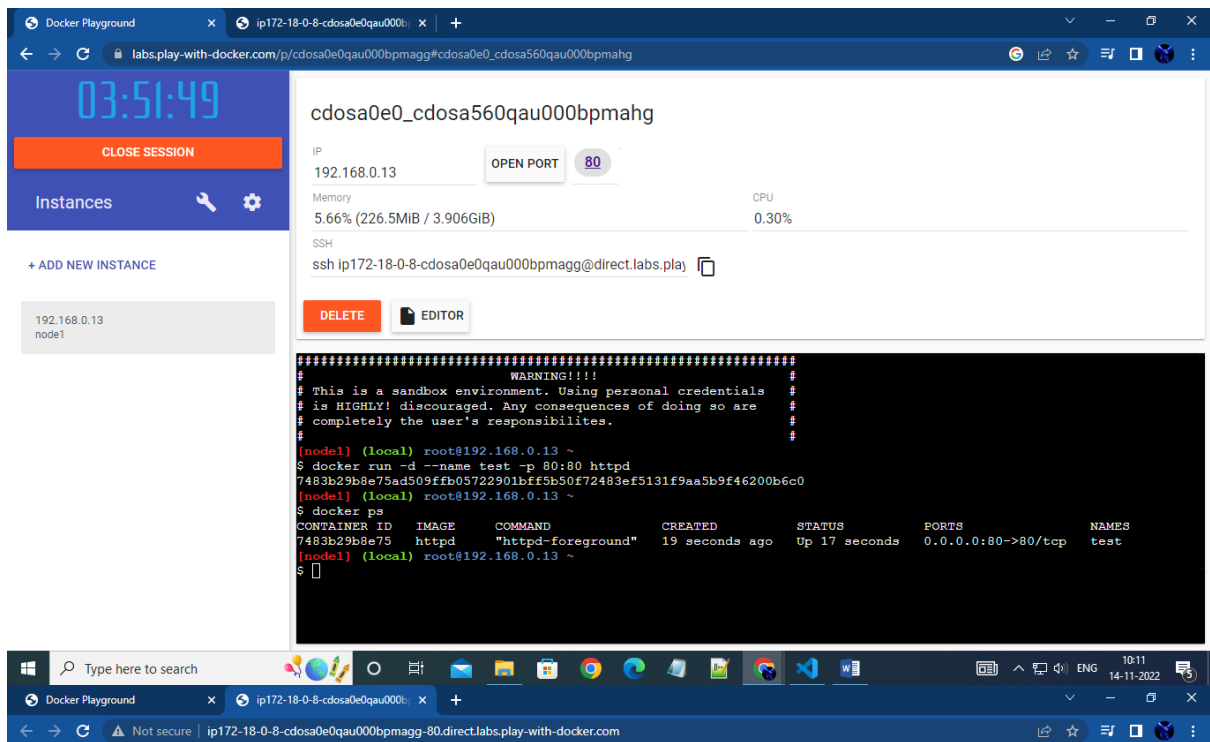
Assignment Date	15 November 2022
Student Name	PRADHYUN P
Student Roll Number	212219040108
Maximum Marks	2 Marks

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

The image shows two screenshots of the Docker Playground interface, illustrating the process of pulling a Docker image from Docker Hub.

Top Screenshot: The interface shows the instance name "cdosa0e0_cdosa560qau000bpmahg" with IP "192.168.0.13". The memory usage is 5.44% (217.6MiB / 3.906GiB) and CPU usage is 0.43%. The SSH command is "ssh ip172-18-0-8-cdosa0e0qau000bpmagg@direct.labs.play". The terminal output shows the command "\$ docker pull httpd:latest" being executed, resulting in the image being pulled successfully. The output includes the image ID "e9995326b091" and the size "145MB".

Bottom Screenshot: The interface shows the same instance name and IP. The memory usage is 5.32% (212.8MiB / 3.906GiB) and CPU usage is 0.23%. The SSH command is "ssh ip172-18-0-8-cdosa0e0qau000bpmagg@direct.labs.play". The terminal output shows the command "\$ docker images" being executed, displaying the list of images. The output shows the "httpd" image with the tag "latest" and the image ID "fe8735c23ec5", which was created "2 weeks ago" and has a size of "145MB".



It works!



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

```
PS C:\Users\RIT\Desktop\job-portal-master> docker run -d -it --rm --name jp-mongodb -p 27017:27017 -e MONGO_INITDB_ROOT_USERNAME="root" -e MONGO_I
NITDB_ROOT_PASSWORD="root-rusteez" mongo
Unable to find image 'mongo:latest' locally
latest: Pulling from library/mongo
eaead16dc43b: Pull complete
8a00eb9f68a0: Pull complete
f683956749c5: Pull complete
b33b2f05ea20: Pull complete
3a342bea915a: Pull complete
fa956ab1c2f0: Pull complete
138a8542a624: Pull complete
acab179af07: Pull complete
f88335710e84: Pull complete
Digest: sha256:71a63fc2438e45714f6c8a2505968ee0beeb94ec77a88ef12190f7cee9b95f32
Status: Downloaded newer image for mongo:latest
01adeb5492af8323e5d62299c7a1372f34e2583891bdefae268f95f74a76a80c
```

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19043.2006]
(c) Microsoft Corporation. All rights reserved.

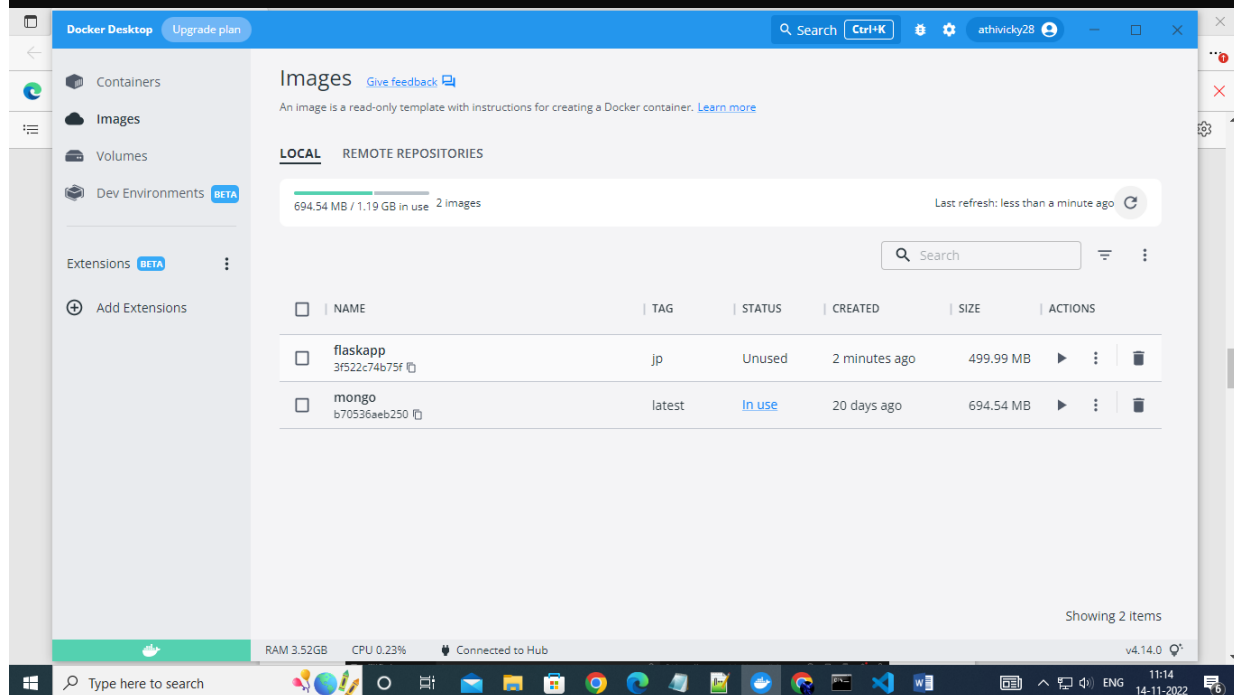
C:\Users\RIT\Desktop\job-portal-master>code .

C:\Users\RIT\Desktop\job-portal-master>docker build -t flaskapp:jp --build-arg requirements="requirements.txt" --build-arg workspace="jobPortal" -f Dockerfile .
[+] Building 750.9s (13/13) FINISHED
=> [internal] load build definition from Dockerfile                                0.3s
=> => transferring dockerfile: 32B                                                0.0s
=> [internal] load .dockerignore                                                  0.3s
=> => transferring context: 2B                                                    0.0s
=> [internal] load metadata for docker.io/library/ubuntu:latest                 3.2s
=> [auth] library/ubuntu:pull token for registry-1.docker.io                   0.0s
=> CACHED [1/7] FROM docker.io/library/ubuntu:latest@sha256:4b1d0c4a2d2aaf63b3711f34eb9fa89fa1bf53dd6e4ca954d47 0.0s
=> [internal] load build context                                                0.3s
=> => transferring context: 325B                                                  0.0s
=> [2/7] RUN apt-get update                                                      177.0s
=> [3/7] RUN apt-get install -y python3 python3-pip                            541.5s
=> [4/7] RUN mkdir jobPortal                                                    1.5s
=> [5/7] COPY . /jobPortal                                                       0.7s
=> [6/7] RUN pip3 install -r /jobPortal/requirements.txt                       20.6s
=> [7/7] WORKDIR jobPortal                                                       0.9s
=> exporting to image                                                           4.4s
=> => exporting layers                                                           4.3s
=> => writing image sha256:3f522c74b75f46d1b99c47ffe3f7fe4fbf05a1cc357e341b242bf25aa4df6257 0.0s
=> => naming to docker.io/library/flaskapp:jp                                   0.0s

Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them
```

```
C:\Users\RIT\Desktop\job-portal-master>docker run -d -it --rm --name jobportal -e FLASK_APP='jobportal' -e FLASK_ENV=development -e FLASK_RUN_HOST='0.0.0.0' -p 5000:5000 flaskapp:jp
329c939f4b7ebc1cb2c753e0318332bc96d86df1697856c62710907fdca3d86b

C:\Users\RIT\Desktop\job-portal-master>
```



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

My IBM x IBM Cloud Container Registry - 1 x Reset your IBMid password - 95 x +

cloud.ibm.com/registry/namespaces

IBM Cloud

Search resources and products...

Container Registry

Quick start

Namespaces 1

Repositories 1

Images 1

Trash 0

Settings

Namespaces

Location

Tokyo

Resource group: Filter... Search Create +

<input type="checkbox"/>	Name	Resource group	Repository count	Image count	Retention policy	
<input checked="" type="checkbox"/>	test-app-002	Default	1	1	Retain all images	

Items per page: 25 1-1 of 1 item 1 1 of 1 page

Type here to search

My IBM x IBM Cloud Container Registry - 1 x Reset your IBMid password - 95 x +

cloud.ibm.com/registry/repos

IBM Cloud

Search resources and products...

Container Registry

Quick start

Namespaces 1

Repositories 1

Images 1

Trash 0

Settings

Repositories

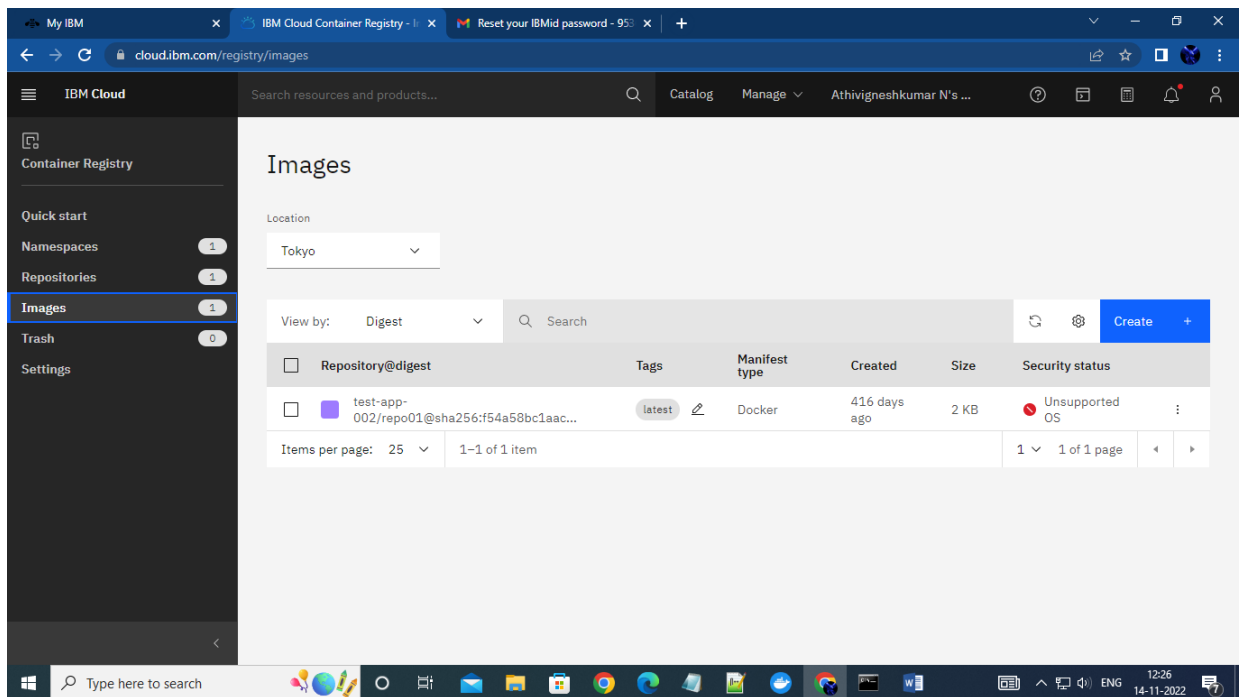
Location

Tokyo

Search Create +

<input type="checkbox"/>	Name	Image count	Namespace	Last updated	
<input checked="" type="checkbox"/>	repo01 jp.icr.io/test-app-002/repo01	1	test-app-002	416 days ago	

Items per page: 25 1-1 of 1 item 1 1 of 1 page



```
C:\Users\RIT>ibmcloud login
API endpoint: https://cloud.ibm.com
Region: jp-tok

Email> 953619104006

Password>
C:\Users\RIT>ibmcloud login
API endpoint: https://cloud.ibm.com
Region: jp-tok

Email> 953619104006@ritrjpm.ac.in

Password>
Authenticating...
OK

Targeted account Athivigneshkumar N's Account (331b915ecbfc4e7290278776aa39995b)

API endpoint: https://cloud.ibm.com
Region: jp-tok
User: 953619104006@ritrjpm.ac.in
Account: Athivigneshkumar N's Account (331b915ecbfc4e7290278776aa39995b)
Resource group: No resource group targeted, use 'ibmcloud target -g RESOURCE_GROUP'
CF API endpoint:
Org:
Space:

C:\Users\RIT>ibmcloud target -g Default
C:\Users\RIT>ibmcloud cr login --client docker
Logging 'docker' in to 'jp.icr.io'...
Logged in to 'jp.icr.io'.

OK

C:\Users\RIT>docker push jp.icr.io/test-app-002/repo01
Using default tag: latest
The push refers to repository [jp.icr.io/test-app-002/repo01]
e07ee1baac5f: Pushed
latest: digest: sha256:f54a58bc1aac5ea1a25d796ae155dc228b3f0e11d046ae276b39c4bf2f13d8c4 size: 525

C:\Users\RIT>ibmcloud cr image-list
Listing images...

Repository      Tag    Digest      Namespace    Created    Size    Security status
jp.icr.io/test-app-002/repo01  latest  f54a58bc1aac  test-app-002  1 year ago  2.5 kB  -

OK

C:\Users\RIT>
```

4. Create a Kubernetes cluster in IBM cloud and deploy helloworld image or jobportal image and also expose the same app to run in nodepor

The screenshot shows the IBM Cloud Kubernetes console for a cluster named 'mycluster-01'. The cluster is in a 'Normal' state and expires in 30 days. The overview page displays the following details:

- Node status:** 1 of 1, Normal
- Add-on status:** 0 of 0, Normal
- Master status:** Normal
- Ingress status:** Unknown
- Cluster ID:** cdouf2ef09uaonn50h60
- Version:** 1.24.7_1542
- Infrastructure:** Classic
- Zones:** Milan 01
- Created:** 11/14/2022, 12:30 PM
- Resource group:** Default
- Image security enforcement:** Disable

The screenshot shows the Kubernetes Deployments page for a deployment named 'app: sample-app'. The deployment is in a 'Normal' state and has a revision history limit of 10. The configuration details are as follows:

- Strategy:** RollingUpdate
- Min ready seconds:** 0
- Revision history limit:** 10
- Selector:** app: sample-app
- Rolling update strategy:**
 - Max surge: 25%
 - Max unavailable: 25%

The screenshot shows the Kubernetes Pods page for a pod named 'app: sample-app'. The pod is in a 'Running' state and has a pod-template-hash of d9bdf84d9. The configuration details are as follows:

- Node:** docker-desktop
- Status:** ImagePullBackOff
- IP:** 10.1.0.48
- QoS Class:** BestEffort
- Restarts:** 0
- Service Account:** default

