

## Assignment - 4

### Docker and Kubernetes

Assignment Date	19.11.2022
Student Name	Ajaymanikandan
Student Roll Number	913019104003
Maximum Marks	2 Marks

#### Question-1:

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

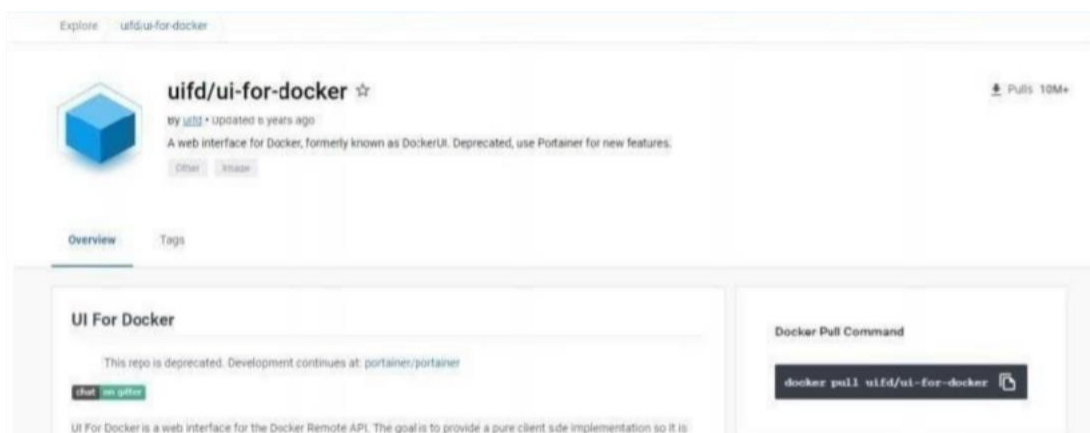
Solution:

```
docker run --rm -p 8787:8787 rocker/verse
docker pull rocker/verse
docker login --username=nishanthc --email=ssnehasri178@gmail.com
WARNING: login credentials saved in
/home/nishanthc/.docker/config.jsonLogin Succeeded
```

```
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
verse_gapminder_gsl  latest  023ab91c6291   3 minutes ago  1.975 GB
verse_gapminder     latest  bb38976d03cf   13 minutes ago 1.955 GB
rocker/verse       latest  0168d115f220   3 days ago    1.954 GB
docker tag bb38976d03cf nishanthc
/verse_gapminder:firsttry
docker push nishanthc
/verse_gapminder
```

Saving and loading images

```
docker save
verse_gapminder
docker save verse_gapminder > verse_gapminder.tar
docker load --input verse_gapminder.tar
docker load --input verse_gapminder.tar
```



03:42:30  
CLOSE SESSION

Instances

+ ADD NEW INSTANCE

192.168.0.13  
node1

cd9an2u3\_cd9av060qau0008hbjs0

IP: 192.168.0.13 OPEN PORT

Memory CPU

SSH  
ssh ip172.18.0.4 cd9an2u3tcog00fgf6k0@direct.labs.play-with-docker.com

DELETE EDITOR

```
This is a sandbox environment. Being personal credentials  
is HIGHLY discouraged. Any consequences of doing so are  
completely the user's responsibilities.  
#  
# The FWD team.  
#####  
[root@i3] (local) root@192.168.0.13 ~  
# docker pull uifd/ui-for-docker  
Using default tag: latest  
latest: Pulling from uifd/ui-for-docker  
841194d080c8: Pull complete  
Digest: sha256:fe371ff5a69549269b24073a5ab1244dd4e0b834cbadf244870572150b1cb749  
Status: Downloaded newer image for uifd/ui-for-docker:latest  
docker.io/uifd/ui-for-docker:latest  
[root@i3] (local) root@192.168.0.13 ~  
# docker run -d -p 9000:9000 --privileged --v /var/run/docker.sock:/var/run/docker.sock uifd/ui-for-docker  
c590dd163101aw795bdawa0ab1ddd98f6f6549cb5f24dcb9ef7c19319236e0d  
[root@i3] (local) root@192.168.0.13 ~
```

UI For Docker

Dashboard Containers Containers Network Images Networks Volumes Info [Feedback](#)

# UI For Docker


The UI for Docker container engine

[Learn more](#)

Running Containers

- beav000r\_g0k000000 [View Details](#)

Status




UI For Docker

Dashboard Containers Containers Network Images Networks Volumes Info [Feedback](#)

Running Containers

- beav000r\_g0k000000 [View Details](#)

Status



Running Stopped Ghost

Containers created

1

0

20130303

Images created

1

### Question-2:

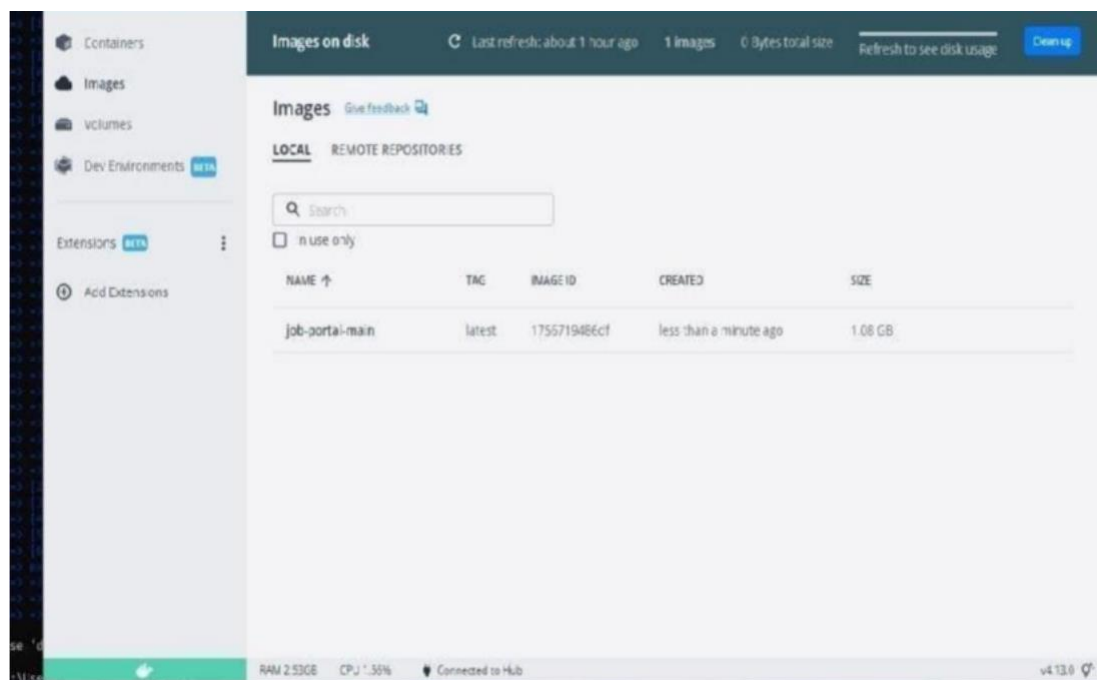
**2. Create a docker file for the jobportal application and deploy it in Docker desktop application.**

**SOLUTION:**

```

[1/6] docker scan debian:bookworm-20240814
[1/6] transferring dockerfile: 22B
[1/6] [internal] Load dockerignore
[1/6] transferring context: 2B
[1/6] [internal] Load metadata for docker.io/library/python:3.9
[1/6] [library/python:3.9] pull token for registry-1.docker.io
[1/6] [internal] Load image list
[1/6] transferring context: 487B
[1/6] FROM docker.io/library/python:3.9@sha256:f8054e4681f9622554067389925510074a4676677a468e09f730646f9
[1/6] register docker.io/library/python:3.9@sha256:f8054e4681f9622554067389925510074a4676677a468e09f730646f9
[1/6] sha256:f8054e4681f9622554067389925510074a4676677a468e09f730646f9 1.80KB / 1.80KB
[1/6] sha256:409740827a0c0745c3c8f7259c2de10f921214e04609353036376d3040d 2.22KB / 2.22KB
[1/6] sha256:5436883007c3e3a2d4c6e21f3893b8c8400427634c8892081f71f3146d108 9.27KB / 9.27KB
[1/6] sha256:56f546d31c8d30019321e73d9d1d7805433b55574732d408e0b77461e5 54.93KB / 54.93KB
[1/6] sha256:0a123c738534320b735d48f34f88f3e021095c290c714b5313a6371d62337cd 4.33KB / 4.33KB
[1/6] sha256:c7979a03817229708c433738823ed13a405463456c0c9e5a5651d7463d56 18.87KB / 18.87KB
[1/6] sha256:6ad457770110b37eac7370630c754809c64ebd6f4948a0d797075 48.53KB / 54.97KB
[1/6] sha256:17774500700770d71739740eb088400481e8f709212d734052c707f 390.53KB / 390.53KB
[1/6] sha256:5d5121314f16536978908000105413034624572854046324a8d31340d783 4.20KB / 4.20KB
[1/6] extracting sha256:66295408543c0d3891832c473d931c0730912091574732d90e0677a61e5
[1/6] sha256:17774500700770d71739740eb088400481e8f709212d734052c707f 390.53KB / 390.53KB
[1/6] extracting sha256:5d5121314f16536978908000105413034624572854046324a8d31340d783
[1/6] extracting sha256:0a123c738534320b735d48f34f88f3e021095c290c714b5313a6371d62337cd
[1/6] sha256:0a123c738534320b735d48f34f88f3e021095c290c714b5313a6371d62337cd 2.93B / 2.93B
[1/6] sha256:c4f43c7205309808f7c046c1d712d518434cc51d5754d54044c6165a3a3f 2.23KB / 2.23KB
[1/6] extracting sha256:6ad457770110b37eac7370630c754809c64ebd6f4948a0d797075
[1/6] extracting sha256:c7979a03817229708c433738823ed13a405463456c0c9e5a5651d7463d56
[1/6] sha256:5d5121314f16536978908000105413034624572854046324a8d31340d783
[1/6] sha256:0a123c738534320b735d48f34f88f3e021095c290c714b5313a6371d62337cd
[1/6] sha256:409740827a0c0745c3c8f7259c2de10f921214e04609353036376d3040d
[1/6] sha256:c4f43c7205309808f7c046c1d712d518434cc51d5754d54044c6165a3a3f
[1/6] MONDOX App
[1/6] add - /app
[1/6] COPY requirements.txt /app
[1/6] RUN python -m pip install -r requirements.txt
[1/6] RUN python -m pip install flask
Supporting to make
exporting layers
[1/6] writing image sha256:11567108400f8072ad0a385c521251327612d1484ad0242632a3d8370f16
[1/6] sending to docker.io/library/pub-mal-wsl

```



### QUESTION-3:

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

Solution:

```
<html>
<body>
  Hello, IBM Cloud World!
</body>
</html>---
```

applications:

- buildpack: <https://github.com/cloudfoundry/staticfile-buildpack.git>
- host: simple-website- $\{random\}$
- name: simple-website- $\{random\}$
- memory: 64M
- stack: cflinuxfs2

The screenshot shows the 'DEPLOY' tab in the IBM Cloud console. It includes a 'DELETE' button and tabs for 'INPUT', 'JOBS', and 'ENVIRONMENT PROPERTIES'. Below these are icons for 'Rolling Deploy' and 'ADD JOB'. The main section is titled 'Rolling Deploy' and contains a 'Deploy configuration' table with the following details:

Deploy configuration	
Deployer type	Cloud Foundry
IBM Cloud region	US South - <a href="https://api.ng.bluemix.net">https://api.ng.bluemix.net</a>
Organization	bluemix_devops@ibm.com
Space	demo
Application name	simple-website-ae7f5ff6

```
1  {
2    "ServiceId": "com.ibm.cloudoe.orion.client.deploy",
3    "Params": {
4      "Target": {
5        "Url": "https://api.ng.bluemix.net",
6        "Org": "bluemix_devops@ibm.com",
7        "Space": "demo"
8      },
9      "Name": "simple-website-ae7f5ff6",
10     "Instrumentation": {}
11   },
12   "Path": "manifest.yml",
13   "Type": "Cloud Foundry"
14 }
```

Hello, IBM Cloud World!

#### QUESTION-4:

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.

Solution:

```
ibmcloud target -g <resource_group_name>ibmcloud cr nishanthc-add  
<your_nishanthc>ibmcloudresource service-instance-create example-postgresql databases-for-  
postgresql standard us- southibmcloud ks cluster-service-bind mycluster default example-  
postgresqlgit clone -b node git@github.com:IBM-Cloud/cloudatabases-helloworld-kubernetes-  
examples.gitspec:
```

```
replicas: 3name: cloudpostgres-nodejs-app
```

```
image: "registry.<region>.bluemix.net/<namespace>/icdpg" # Edit me
```

```
imagePullPolicy: Alwaysibmcloud cr regionYou are targeting region 'us-south', the registry is  
'registry.ng.bluemix.net'.ibmcloud cr build -t registry.ng.bluemix.net/<namespace>/icdpg .ibmcloud  
cr images
```

env:

```
- name: BINDING
```

```
valueFrom:
```

```
secretKeyRef:
```

```
name: <postgres-secret-name> # Edit me
```

```
key: binding
```

```
apiVersion: v1
```

```
kind: Service
```

```
metadata:
```

```
name: cloudpostgres-service
```

```
labels:
```

```
run: clouddb-demo
```

```
spec:
```

```
type: NodePort
```

```
selector:
```

```
run: clouddb-demo
```

```
ports:
```

```
- protocol: TCP
```

```
port: 8080
```

```
nodePort: 30081
```

```
kubectl apply -f clouddb-deployment.yml
```

```
deployment.apps/icdpostgres-app created
```

```
service/cloudpostgres-service created
```

```
kubectl get pods -o wideibmcloud ks workers <your_cluster_name>
```

