

Assignment - 4 Docker and Kubernetes

Assignment Date	November 3
Student Name	YUVANESH R
Student Roll Number	2116190801202
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=abuthahir --email=ssnehasri178@gmail.com WARNING: login credentials
saved in
/home/abuthahir/.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
0168d115f220	3 days ago		1.954 GB	rocker/verse latest

```
docker tag bb38976d03cf abuthahir
/verse_gapminder:firsttry docker push abuthahir
/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

[Explore](#) [uifd/ui-for-docker](#)



uifd/ui-for-docker

By [uifd](#) • Updated 6 years ago

A web interface for Docker, formerly known as DockerUI. Deprecated, use Portainer for new features.

[Other](#) [Image](#)

[Overview](#) [Tags](#)

UI For Docker

This repo is deprecated. Development continues at: [portainer/portainer](#)

[chat](#) [on github](#)

UI For Docker is a web interface for the Docker Remote API. The goal is to provide a pure client side implementation so it is

Docker Pull Command

```
docker pull uifd/ui-for-docker
```

The screenshot shows the Play with Docker interface. On the left, there's a sidebar with a clock showing 03:42:30, a 'CLOSE SESSION' button, and a list of instances including '192.168.0.13 node1'. The main area displays details for a container named 'cd9an2u3_cd9av060qau0008hbjs0', including its IP (192.168.0.13) and an 'OPEN PORT' button. Below this is a terminal window showing the following commands and output:

```

# This is a sandbox environment. Using personal credentials #
# is HIGHLY discouraged. Any consequences of doing so are #
# completely the user's responsibilities. #
# #
# The PWD team. #
#####
[node1] (local) root@192.168.0.13 ~
$ docker pull uifd/ui-for-docker
Using default tag: latest
latest: Pulling from uifd/ui-for-docker
841194d000c8: Pull complete
Digest: sha256:fe371ff3a69549269b24073a5ab1244dd4c0b834cbadf244870572150b1cb749
Status: Downloaded newer image for uifd/ui-for-docker:latest
docker.io/uifd/ui-for-docker:latest
[node1] (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
c390dd163101ae795bdcea0eb1ddd98f6fe549cb5f24dadb9ff7c1931923fc0d
[node1] (local) root@192.168.0.13 ~

```

The screenshot shows the 'UI For Docker' dashboard. At the top, there's a navigation bar with tabs for Dashboard, Containers, Containers Network, Images, Networks, Volumes, and Info. A 'Refresh' button is on the right. The main content area is divided into two sections: 'Running Containers' and 'Status'. The 'Running Containers' section lists one container: 'beautiful_goldwasser' with a status of 'Up About a minute'. The 'Status' section features a donut chart showing 100% Running (green), 0% Stopped (red), and 0% Ghost (grey). Below these sections are two line graphs: 'Containers created' and 'Images created', both showing a count of 1 over time.

Question-2:

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

SOLUTION:

```

[internal] load build definition from Dockerfile
-> transferring dockerfile: 32B
[internal] load .dockerignore
-> transferring context: 2B
[internal] load metadata for docker.io/library/python:3.6
[auth] library/python:pull token for registry-1.docker.io
[internal] load build context
-> transferring context: 887B
[1/6] FROM docker.io/library/python:3.6@sha256:f8652afef88c15f6d22354d547d892501067aa4026a7fa9a0819df9f300af6fc
-> resolve docker.io/library/python:3.6@sha256:f8652afef88c15f6d22354d547d892501067aa4026a7fa9a0819df9f300af6fc
-> sha256:f8652afef88c15f6d22354d547d892501067aa4026a7fa9a0819df9f300af6fc 1.86kB / 1.86kB
-> sha256:d007a407a80c879df5ac31872359c2de510f92214c0448e928393b376d3b00d 2.22kB / 2.22kB
-> sha256:54200638d07c5e3ad24c6e21fc889abbc8406a27634c8002006ff71f3f44b104 9.27kB / 9.27kB
-> sha256:8e29546d541cddb309281d21a73a9d1db78665c1b95b74f32b009e0b77a6e1e3 54.92MB / 54.92MB
-> sha256:9b820c73b52b02b07d5c07a54fb0f3e921995a296c714b53a32ae67d19231fcd 5.15MB / 5.15MB
-> sha256:cb5b7ae361722f070eca53f35823ed21baa85d61d5d95cd5a9ab53d740dd56 10.87MB / 10.87MB
-> sha256:6494e4811622b31c027ccac322ca63937fd005f569a930ef15c01aade718793 54.57MB / 54.57MB
-> sha256:6f9f74896dfa93fe8172f594fab85e0b4e8a0481a0fe9d9112efc7e4d3c76f7 196.51MB / 196.51MB
-> sha256:5e3b1213efc56598e78bd082983045c164de2a37285e06a52dada823124dc743 6.20MB / 6.20MB
-> extracting sha256:8e29546d541cddb309281d21a73a9d1db78665c1b95b74f32b009e0b77a6e1e3
-> sha256:9fddfd56334f2e0efad7e241bf5e7459c40ed105c5478676f41c1244bd06752 14.21MB / 14.21MB
-> extracting sha256:9b820c73b52b02b07d5c07a54fb0f3e921995a296c714b53a32ae67d19231fcd
-> sha256:cb5b7ae361722f070eca53f35823ed21baa85d61d5d95cd5a9ab53d740dd56
-> sha256:404f02044buc8432ca522cb09f254b1c91fcea0806bfeef0be0b243b2f31bab7 235B / 235B
-> sha256:c4f42be2be53b900ebffc040c1df13de538434ccc5f5d0954a5684a0109a3a3f 2.21MB / 2.21MB
-> extracting sha256:6494e4811622b31c027ccac322ca63937fd005f569a930ef15c01aade718793
-> extracting sha256:6f9f74896dfa93fe8172f594fab85e0b4e8a0481a0fe9d9112efc7e4d3c76f7
-> extracting sha256:5e3b1213efc56598e78bd082983045c164de2a37285e06a52dada823124dc743
-> extracting sha256:9fddfd56334f2e0efad7e241bf5e7459c40ed105c5478676f41c1244bd06752
-> extracting sha256:404f02044buc8432ca522cb09f254b1c91fcea0806bfeef0be0b243b2f31bab7
-> extracting sha256:c4f42be2be53b900ebffc040c1df13de538434ccc5f5d0954a5684a0109a3a3f
[2/6] WORKDIR /app
[3/6] ADD . /app
[4/6] COPY requirements.txt /app
[5/6] RUN python3 -m pip install -r requirements.txt
[6/6] RUN python3 -m pip install lbm_db
exporting to image
-> exporting layers
-> writing image sha256:1756719486df002fad5dee305c5221513f2ff2d3b49e0d242b22a28af0379ff19
-> naming to docker.io/library/job-portal-main

# 'docker scan' to run Snky tests against images to find vulnerabilities and learn how to fix them

```

Containers

Images

Volumes

Dev Environments BETA

Extensions BETA

Add Extensions

Images on disk

Last refresh: about 1 hour ago 1 Images 0 Bytes total size Refresh to see disk usage Clean up

Images Give feedback

LOCAL REMOTE REPOSITORIES

In use only

NAME ↑	TAG	IMAGE ID	CREATED	SIZE
job-portal-main	latest	1756719486df	less than a minute ago	1.08 GB

RAM 2.53 GB CPU 1.56% Connected to Hub v4.13.0

QUESTION-3:

1. 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 IBM Cloud Deploy console. At the top, there's a 'DEPLOY' header with a 'DELETE' button. Below it are tabs for 'INPUT', 'JOBS', and 'ENVIRONMENT PROPERTIES'. The 'JOBS' tab is active, showing a 'Rolling Deploy' section with a 'ROLLING DEPLOY' button and an 'ADD JOB' button. The 'Rolling Deploy' section contains a 'Deploy configuration' table with the following settings:

Deploy configuration	
Deployer type	Cloud Foundry
IBM Cloud region	US South - https://api.ng.bluemix.net
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 }
```

QUESTION-4:

1. 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 abuthahir-add  
<your_abuthahir>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>/icdpkg" # 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>/icdpkg .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>
```

Hello World!

Thanks for creating an [IBM Cloud Databases for PostgreSQL](#) database.

Add a word to the database

The word is defined as

Database output

```
The word bye is defined as a goodbye
The word bye is defined as a farewell
The word hello is defined as a greeting
The word hello is defined as a greeting
The word hello bob is defined as a greeting
The word hello bob is defined as a greeting
```