

## Assignment -4

### Docker and Kubernetes

Assignment Date	2 November 2022
Student Name	Karthikeyan S
Student Roll Number	722819205021
Maximum Marks	2 Marks

#### 1. Pull an image from docker hub and run it in docker Playground

The screenshot is divided into two main horizontal sections. The top section shows the Docker Hub repository page for `uifd/ui-for-docker`. The page includes the repository name, a description stating it is deprecated and to use Portainer instead, and a 'Pulls 10M+' badge. A 'Docker Pull Command' box displays `docker pull uifd/ui-for-docker`. The bottom section shows the Docker Playground interface. On the left, there's a sidebar with a timer at 03:42:30, a 'CLOSE SESSION' button, and an 'Instances' list showing one instance named 'node1' at IP 192.168.0.13. The main area displays the instance details for 'cd9an2u3\_cd9av060qau0008hbjs0', including its IP (192.168.0.13), memory, CPU, and an SSH command. Below this, a terminal window shows the execution of `docker pull uifd/ui-for-docker`, which successfully pulls the latest image. The terminal output includes the digest `sha256:fe371ff5a69549269b24073a5ab1244dd4c0b834cbadf244870572150b1cb749` and the status 'Downloaded newer image for uifd/ui-for-docker:latest'. The terminal prompt is `node1 (local) root@192.168.0.13 ~`.

UI For Docker

Dashboard Containers Containers Network Images Networks Volumes Info Refresh

# UI For Docker

The UI for Docker container engine

Learn more.

Running Containers

- beautiful\_goldwasser Up About a minute

Status



Containers created



Images created



Running Stopped Ghost

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

C:\Windows\System32\cmd.exe

```
>> [internal] load build definition from Dockerfile
>> transferring dockerfile: 328
>> [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: 6878
>> [1/6] FROM docker.io/library/python:3.6@sha256:f8652afaf88c25f0d22354d547d892591067aa4026a7fa9a6819df9f300af6fc
>> resolve docker.io/library/python:3.6@sha256:f8652afaf88c25f0d22354d547d892591067aa4026a7fa9a6819df9f300af6fc
>> sha256:f8652afaf88c25f0d22354d547d892591067aa4026a7fa9a6819df9f300af6fc 1.86kB / 1.86kB
>> sha256:d097a4907a8e0c099df5ae31b7235926e510f02214c0448e92639b3b76db060d 2.22kB / 2.22kB
>> sha256:54268638d07c5e3ad24c6e21fc889abbce8486a27634c8891086ff71f3f44b104 9.27kB / 9.27kB
>> sha256:0e29546d541cdd309281d21a73a9d1db78665c1b95b74f32b00e0b77a6e1e3 54.92MB / 54.92MB
>> sha256:9b829c73b52b92b97d5c07a54fb0f3e921995a296c714b53a32ae67d19231fcd 5.15MB / 5.15MB
>> sha256:cb5b7ae361722f070eca53f35823ed21baa85d61d5d95cd5a95ab53d748cdd56 10.87MB / 10.87MB
>> sha256:6494e4811622b31c027ccac322ca463937fd805f569a93e6f15c01aade718793 54.57MB / 54.57MB
>> sha256:6f9f74896dfa93fe0172f594faba85e0b4e8a0481a0fef9d9112efc7e4d3c78f7 196.51MB / 196.51MB
>> sha256:5e3b1213efc56598e78bd602983945c164de2a37205e06ae2dada823124dc743 6.29MB / 6.29MB
>> sha256:0e29546d541cdd309281d21a73a9d1db78665c1b95b74f32b00e0b77a6e1e3 54.92MB / 54.92MB
>> sha256:9fddfd56334f2eefad7e241bf5e7459c40ed105c5478676f41c1244bd96752 14.21MB / 14.21MB
>> extracting sha256:9b829c73b52b92b97d5c07a54fb0f3e921995a296c714b53a32ae67d19231fcd 2.3s
>> extracting sha256:cb5b7ae361722f070eca53f35823ed21baa85d61d5d95cd5a95ab53d748cdd56 4.0s
>> sha256:404f02044bac0432ca522cbb9f254b1c91fcea6806bfeef0be0b243b2f31bab7 235B / 235B
>> sha256:c4f42be2b53b900ebffc040c1df13de538434ccc5f5d954a56848a6169a3a3f 2.21MB / 2.21MB
>> extracting sha256:6494e4811622b31c027ccac322ca463937fd805f569a93e6f15c01aade718793 27.3s
>> extracting sha256:6f9f74896dfa93fe0172f594faba85e0b4e8a0481a0fef9d9112efc7e4d3c78f7 131.4s
>> extracting sha256:5e3b1213efc56598e78bd602983945c164de2a37205e06ae2dada823124dc743 8.2s
>> sha256:9fddfd56334f2eefad7e241bf5e7459c40ed105c5478676f41c1244bd96752 11.3s
>> extracting sha256:404f02044bac0432ca522cbb9f254b1c91fcea6806bfeef0be0b243b2f31bab7 0.0s
>> extracting sha256:c4f42be2b53b900ebffc040c1df13de538434ccc5f5d954a56848a6169a3a3f 2.2s
>> [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 ibm_db
>> exporting to image
>> exporting layers
>> writing image sha256:1756719486df002fad5dae305c5221513f2ff2d1b49a8d242b22a28af0379f19
>> naming to docker.io/library/job-portal-main
```

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

C:\Users\WK-PC\Desktop>job-portal-main>

Docker Desktop Upgrade plan

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

Search

☐ In use only

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

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

### 3. Create a IBM container registry and deploy helloworld app