

Assignment -4

Docker and Kubernetes

Assignment Date	26 October 2022
Student Name	KEERTHANA P L
Student Roll Number	711719104043
Maximum Marks	2 Marks

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

The screenshot displays the Docker Hub page for the repository `uifd/ui-for-docker`. The page indicates that the repository is deprecated and development has moved to `portainer/portainer`. A Docker Pull Command box shows the command `docker pull uifd/ui-for-docker`.

Below the Docker Hub page, the Docker Playground interface is shown. It includes a sidebar with a clock (03:42:30), a 'CLOSE SESSION' button, and an 'Instances' section. The main area displays the IP address `192.168.0.13` and the SSH command `ssh ip172-18-0-4-cd9an2u3tccg00fgf6k0@direct.labs.play-w`. A terminal window at the bottom shows 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 FWD 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
841194d080c8: Pull complete
Digest: sha256:fe371ff5a69549269b24073a5ab1244dd4c0b834cbadf244870572150b1cb749
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
c590dd163101ae795bdcea0eb1ddd98f6fe549cb5f24dab9ff7c1931923fc0d
(node1) (local) root@192.168.0.13 ~
$
```



UI For Docker



UI For Docker

The UI for Docker container engine

[Learn more.](#)

Running Containers

- beautiful_goldwasser [Up About a minute](#)

Status



UI For Docker



Running Containers

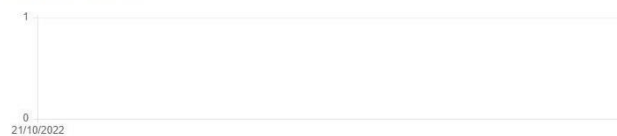
- beautiful_goldwasser [Up About a minute](#)

Status



Running Stopped Ghost

Containers created

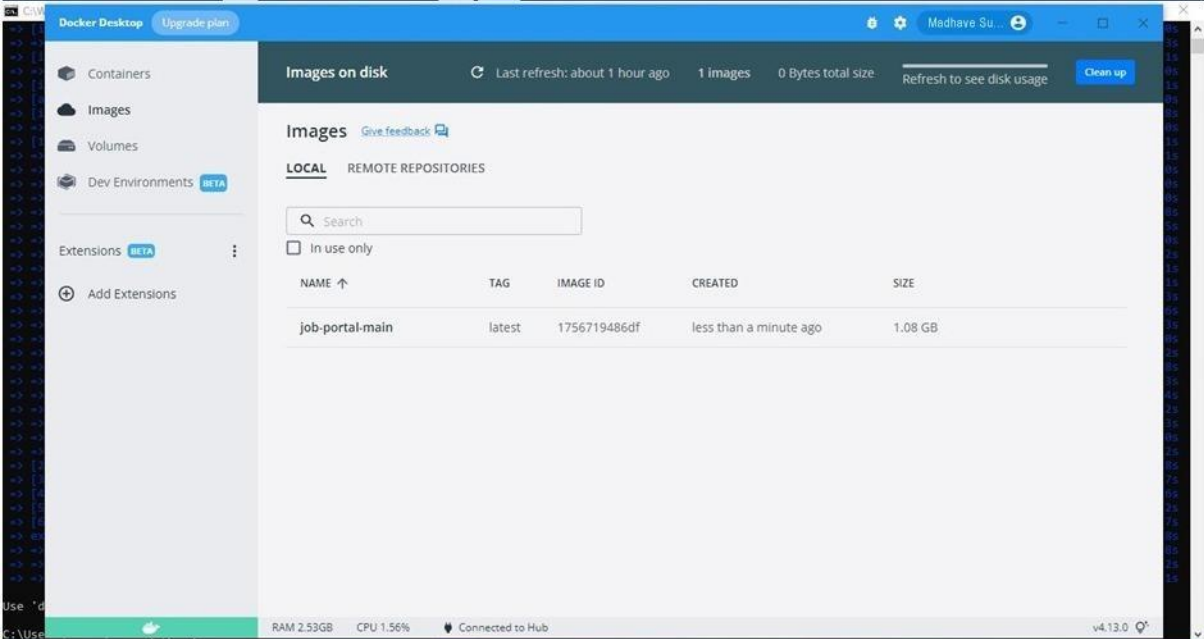


Images created



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: 32B
-> [Internal] load .dockerignore
-> => transferring context: 2B
-> [Internal] load metadata for docker.io/library/python:3.8
-> [auth] library/python:pull token for registry-1.docker.io
-> [Internal] load build context
-> => transferring context: 688B
-> [1/6] FROM docker.io/library/python:3.8@sha256:f852aef86c25f0d22354d547d892591067aa4028a7fa0ad810df9f30aaf6fc
-> resolve docker.io/library/python:3.8@sha256:f852aef86c25f0d22354d547d892591067aa4028a7fa0ad810df9f30aaf6fc
-> sha256:f852aef86c25f0d22354d547d892591067aa4028a7fa0ad810df9f30aaf6fc 1.0kB / 1.0kB
-> sha256:d09744907a6c07f0f5a11872359c2de510f02214c8448e26393b376dbb6d 2.22kB / 2.22kB
-> sha256:5420663807c3e3ad24c6e21fc889abbcb486a27634c0091086ff71f3f44b104 0.27kB / 0.27kB
-> sha256:0e29546d541cdd309201d21a73a9d1db78665c1095b74f32b090e0b77a6e1e3 54.92MB / 54.92MB
-> sha256:00029c73b52b92b97d5c07a54fb0f3e921995a296c714b53a32ae67d10231fcd 5.15MB / 5.15MB
-> sha256:cb5b7ae361722f070eca53f35823ed21baa85d61d5d95cd5a95ab53d748cdd5e 10.07MB / 10.07MB
-> sha256:04044481162b31c0277c3c322ca463927f0805f569a93e6f16c01aada710793 54.57MB / 54.57MB
-> sha256:6f9f7489ed923f0122f504fabd5e004e04812a0f0112efc704d0c7997 196.51MB / 196.51MB
-> sha256:5e3b12130fc56598e78b0602981045c164de2a37285e06a620ada823124dc743 6.29MB / 6.29MB
-> extracting sha256:0e29546d541cdd309201d21a73a9d1db78665c1095b74f32b090e0b77a6e1e3
-> sha256:9fd0dfc56334f2e6efad7a241bf5e7459c40ed105c5478676f41c1244bd90752 14.21MB / 14.21MB
-> extracting sha256:90829c73b52b92b97d5c07a54fb0f3e921995a296c714b53a32ae67d10231fcd 2.35
-> extracting sha256:cb5b7ae361722f070eca53f35823ed21baa85d61d5d95cd5a95ab53d748cdd5e 4.08
-> sha256:404f02044bac0432ca522cbb9f254b1c91fcea680bfeef0be0b24302f31ba07 235B / 235B
-> sha256:c4f42be2be53b900ebffcc040c1df13de530434ccc5f5d954a56848a6169a3a3f 2.23MB / 2.23MB
-> extracting sha256:64044481162b31c0277c3c322ca463927f0805f569a93e6f16c01aada710793
-> extracting sha256:6f9f7489ed923f0122f504fabd5e004e04812a0f0112efc704d0c7997 77.35
-> extracting sha256:5e3b12130fc56598e78b0602981045c164de2a37285e06a620ada823124dc743 5.29
-> extracting sha256:9fd0dfc56334f2e6efad7a241bf5e7459c40ed105c5478676f41c1244bd90752 11.35
-> extracting sha256:404f02044bac0432ca522cbb9f254b1c91fcea680bfeef0be0b24302f31ba07 0.05
-> extracting sha256:c4f42be2be53b900ebffcc040c1df13de530434ccc5f5d954a56848a6169a3a3f 2.25
-> [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 sha256:1756710486df002fad5dae305c5221513f2ff2d1b40a0d242b22a28af0379f19
-> 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\VK-PC\Desktop\job-portal-main>
```



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

3. Create a IBM container registry and deploy helloworld app