

Assignment -4


Docker and Kubernetes

Assignment Date	21 October 2022
Student Name	Mr. Prabu K
Student Roll Number	711319CS111
Maximum Marks	2 Marks

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

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](#)

Pulls 10M+

Overview

Tags

UI For Docker

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

[chat on gitter](#)

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 effortless to connect and manage docker.

Docker Pull Command

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

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-cd9an2u3tccg00fgf6k0@direct.labs.play-w

DELETE EDITOR

```
# 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
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
c590dd163101ae795bdcea0eb1dd498f6fe549eb5f24dadb9ff7c1931923fc0d
(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.

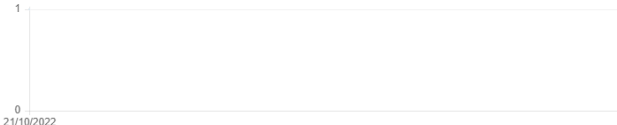
Running Containers

- beautiful_goldwasser Up About a minute


Status



Containers created



Images created



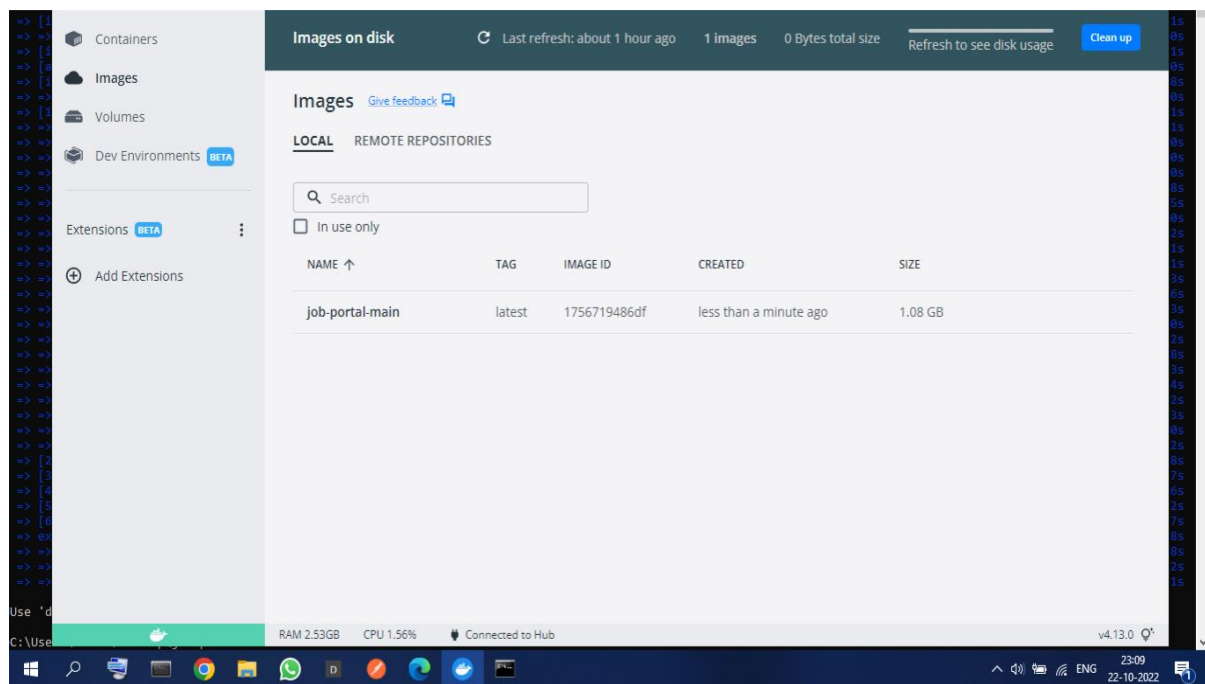
Running Stopped Ghost

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
[Internal] load .dockerignore
[Internal] load metadata for docker.io/library/python:3.6
[auth] library/python:pull token for registry-1.docker.io
[Internal] load build context
[Internal] load build context: 687B
[1/6] FROM docker.io/library/python:3.6@sha256:f8652afaf88c25f0d22354d547d892591067aa4026a7fa9a6819df9f300af6fc
[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 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\VK-PC\Desktop\job-portal-main>
```



3. Create a IBM container registry and deploy helloworld app

