

Assignment - 4

Docker and Kubernetes

Assignment Date	01 November 2022
Student Name	MENATI VASUDEVA REDDY
Student Roll Number	727819TUCS124
Maximum Marks	2 Marks

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

The screenshot displays two browser windows. The top window shows the Docker Hub page for the repository `uifd/ui-for-docker`. The page indicates that the repository is deprecated and development continues at `portainer/portainer`. It provides a Docker Pull Command: `docker pull uifd/ui-for-docker`.

The bottom window shows the Docker Playground interface. It displays the instance ID `cd9an2u3_cd9av060qau0008hbjs0` and the IP address `192.168.0.13`. The terminal output shows the following commands and results:

```
# 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. #
#####
[nsdml] (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:fe371fff5a69549269b24073a5ab1244dd4c0b834cbadf244870572150b1cb749
Status: Downloaded newer image for uifd/ui-for-docker:latest
docker.io/uifd/ui-for-docker:latest
[nsdml] (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
c590d4ii63101ae795bacea0b1ddd98ffefe549cb5f24dadb9ff7c1931923Fc0d
[nsdml] (local) root@192.168.0.13 ~
```

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

The image displays two screenshots of the 'UI For Docker' web interface, which is a dashboard for managing Docker containers. The interface is accessed via a web browser at the URL `ip172-18-0-4-cd9an2u3tccg00fgf6k0-9000.direct.labs.play-with-docker.com/#/`.

Top Screenshot: The dashboard shows the 'Running Containers' section with one container named `beautiful_goldwasser` listed as 'Up About a minute'. The 'Status' section features a large green donut chart indicating that all containers are running. The top navigation bar includes links for Dashboard, Containers, Containers Network, Images, Networks, Volumes, and Info, along with a Refresh button.

Bottom Screenshot: This screenshot provides a more detailed view of the dashboard. It shows the 'Running Containers' section with the same container `beautiful_goldwasser`. Below this, there are two line graphs: 'Containers created' and 'Images created', both showing a count of 1 over time. The 'Status' section includes a legend for the donut chart: Running (green), Stopped (red), and Ghost (grey).

3. Create an IBM Container registry and deploy

```
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: 687B
-> [1/6] FROM docker.io/library/python:3.8@sha256:f8652aef88c25f8d2354d547d892591067aa076a7fa0810df9f308a6f6c
-> resolve docker.io/library/python:3.8@sha256:f8652aef88c25f8d2354d547d892591067aa076a7fa0810df9f308a6f6c
-> sha256:f8652aef88c25f8d2354d547d892591067aa076a7fa0810df9f308a6f6c: 1.86kB / 1.86kB
-> sha256:d007a4997a8e679df5ac31872359c2de510f82214c0448e928393b376d3b0d0d: 2.22kB / 2.22kB
-> sha256:54206038007c5e3ad24c0e21f889abbcb486a27634c0892886ff71f3f44b104: 9.27kB / 9.27kB
-> sha256:0e2954dd541cd8d309281d21a73a9d1db70665c1b95b74f32b009e0b77a6e1e3: 54.92MB / 54.92MB
-> sha256:90829c73b52b92b97d5c07e54fb0f3e921995a296c714b53a32ae67d19231fcd: 5.15MB / 5.15MB
-> sha256:cb5b7ae361722f070eca53f35823ed21baa85d01d5d95cd5a95ab53d740cdd56: 10.87MB / 10.87MB
-> sha256:6a9a4a811622b31c027c0c322ca463937f4085f560a93a6f15c01a0d710795: 54.57MB / 54.57MB
-> sha256:6f9f74809df293fe0727f394fabad5e0b4ed9a021a0f0f0112efc7e4d3c76f7: 196.51MB / 196.51MB
-> sha256:5a3b1213efc56598e78bd002083945c164de2a37205e06a62dad823124dc743: 6.29MB / 6.29MB
-> extracting sha256:0e2954dd541cd8d309281d21a73a9d1db70665c1b95b74f32b009e0b77a6e1e3
-> sha256:0fdddfc56334f2e6efad7e241bf5e7459c40ed105c5470676f41c1244bd90752: 14.21MB / 14.21MB
-> extracting sha256:90829c73b52b92b97d5c07e54fb0f3e921995a296c714b53a32ae67d19231fcd
-> extracting sha256:cb5b7ae361722f070eca53f35823ed21baa85d01d5d95cd5a95ab53d740cdd56
-> sha256:404f02044bac0432ca522cbb9f254b1c91fcea806bfeef8be0b243b2f31bab7: 235B / 235B
-> sha256:c4f42be2be53b00ebffcc040c1d13de538434cc5f5d954a56048a169a3af: 2.21MB / 2.21MB
-> extracting sha256:6a9a4a811622b31c027c0c322ca463937f4085f560a93a6f15c01a0d710795
-> extracting sha256:6f9f74809df293fe0727f394fabad5e0b4ed9a021a0f0f0112efc7e4d3c76f7
-> extracting sha256:5a3b1213efc56598e78bd002083945c164de2a37205e06a62dad823124dc743
-> extracting sha256:0fdddfc56334f2e6efad7e241bf5e7459c40ed105c5470676f41c1244bd90752
-> extracting sha256:404f02044bac0432ca522cbb9f254b1c91fcea806bfeef8be0b243b2f31bab7
-> extracting sha256:c4f42be2be53b00ebffcc040c1d13de538434cc5f5d954a56048a169a3af
-> [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 sha256:1756719486df002fad5dae305c5221513f2ff2d1b49add242b22a28af0379f19
-> 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>
```

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

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