

Assignment - 4

Docker and Kubernetes

Assignment Date	01 November 2022
Student Name	Meera V
Student Roll Number	815119106021
Maximum Marks	2 Marks

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

The screenshot shows a web browser with two tabs. The first tab is Docker Hub, displaying the repository page for `uifd/ui-for-docker`. The repository is marked as deprecated, with a note that development continues at `portainer/portainer`. The Docker Pull Command is shown as `docker pull uifd/ui-for-docker`.

The second tab is Docker Playground, showing a terminal session. The terminal output is as follows:

```
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 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:fe371fff3a69549269b24073a5ab1244dd4c0b834cbadf244870572150b1cb749
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
c590dd163101ae795bdca0eb1dd498f6fe549cb5f24dab9ff7c1931923Fc0d
[node1] (local) root@192.168.0.13 ~
```

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

The image displays two screenshots of the 'UI For Docker' web application interface, accessed via a browser. The browser's address bar shows the URL: `ip172-18-0-4-cd9an2u3tccg00fgf6k0-9000.direct.labs.play-with-docker.com/#/`.

Top Screenshot:

- Navigation Bar:** Includes tabs for Dashboard, Containers, Containers Network, Images, Networks, Volumes, and Info. A 'Refresh' button is on the right.
- Header:** 'UI For Docker' with the subtitle 'The UI for Docker container engine' and a 'Learn more.' button.
- Running Containers:** A list showing one container named 'beautiful_goldwasser' with a status of 'Up About a minute'.
- Status:** A large green donut chart indicating the overall system status.

Bottom Screenshot:

- Running Containers:** Same as the top screenshot, showing 'beautiful_goldwasser' is up.
- Status:** The donut chart is shown with a legend: Running (green), Stopped (red), and Ghost (grey).
- Containers created:** A line graph showing a count of 1 container created on 21/10/2022.
- Images created:** A line graph showing a count of 1 image created.

3. Create an IBM Container registry and deploy

The image shows a Windows command prompt window and the Docker Desktop application interface.

Command Prompt Window:

```
C:\Windows\System32\cmd.exe
-> [internal] load build definition from Dockerfile
-> => transferring dockerfile: 32B
-> [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:f8652aaf88c25f8d22354d547d892591067aad026a7fa9ae819df9f308a6f6c
-> resolve docker.io/library/python:3.8@sha256:f8652aaf88c25f8d22354d547d892591067aad026a7fa9ae819df9f308a6f6c
-> sha256:f8652aaf88c25f8d22354d547d892591067aad026a7fa9ae819df9f308a6f6c: 1.86kB / 1.86kB
-> sha256:d807a4097a8ec079df5ac31872359c2de510f82214c0448e926393b376d3b6d4d: 2.22kB / 2.22kB
-> sha256:5420863b097c5e3ad24c0e21fc889abbc8486a27634c0892088ff71f3f44b104: 0.27kB / 0.27kB
-> sha256:0e29546d541cd8d309281d21a73a9d1db70665c1b05b74f32b009e0b77a6e1e3: 54.92MB / 54.92MB
-> sha256:98829c73b52b92b97d5c07a54fb0f3e921995a296c714b53a32ae67d19231fcd: 5.15MB / 5.15MB
-> sha256:cb5b7ae361722f070eca53f35823aed21baa85d01d5d95cd5a95ab53d748cdd56: 10.87MB / 10.87MB
-> sha256:6494e481182b31c097c9e322ca403073f4885f560b03ef1518aa0e710793: 54.57MB / 54.57MB
-> sha256:6f9f74890df93fe0172f594f4ba85e04e0481a0fe0112efc7e4d3c7877: 196.51MB / 196.51MB
-> sha256:5a3b1213efc56598e78bd602081945c164de2a37205e06a62dad823124dc743: 6.29MB / 6.29MB
-> extracting sha256:0e29546d541cd8d309281d21a73a9d1db70665c1b05b74f32b009e0b77a6e1e3
-> extracting sha256:9fddfd56334f2e6efad7e241bf5e7459c40ed105c5478676f41c1244bd96752: 14.21MB / 14.21MB
-> extracting sha256:98829c73b52b92b97d5c07a54fb0f3e921995a296c714b53a32ae67d19231fcd: 2.38
-> extracting sha256:cb5b7ae361722f070eca53f35823aed21baa85d01d5d95cd5a95ab53d748cdd56: 4.06
-> sha256:404f02044bac0432ca522cbb9f254b1c91fcea806bfeef80e0b243b2f31bab7: 235B / 235B
-> sha256:c4f42be2b53b00ebfffc040c16f13de538434ccc5f5d954a56048a6169a3af: 2.21MB / 2.21MB
-> extracting sha256:6494e481182b31c097c9e322ca403073f4885f560b03ef1518aa0e710793
-> extracting sha256:6f9f74890df93fe0172f594f4ba85e04e0481a0fe0112efc7e4d3c7877: 131.45
-> extracting sha256:5a3b1213efc56598e78bd602081945c164de2a37205e06a62dad823124dc743: 8.25
-> extracting sha256:9fddfd56334f2e6efad7e241bf5e7459c40ed105c5478676f41c1244bd96752: 11.35
-> extracting sha256:404f02044bac0432ca522cbb9f254b1c91fcea806bfeef80e0b243b2f31bab7: 0.06
-> extracting sha256:c4f42be2b53b00ebfffc040c16f13de538434ccc5f5d954a56048a6169a3af: 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 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\VK-PC\Desktop\job-portal-main>
```

Docker Desktop Interface:

The Docker Desktop window shows the "Images on disk" tab. It displays a table of local images:

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

The interface also shows a sidebar with navigation options: Containers, Images, Volumes, Dev Environments, and Extensions. The bottom status bar indicates the system is connected to the Hub, with RAM usage at 2.53GB and CPU usage at 1.56%.