

Assignment -4

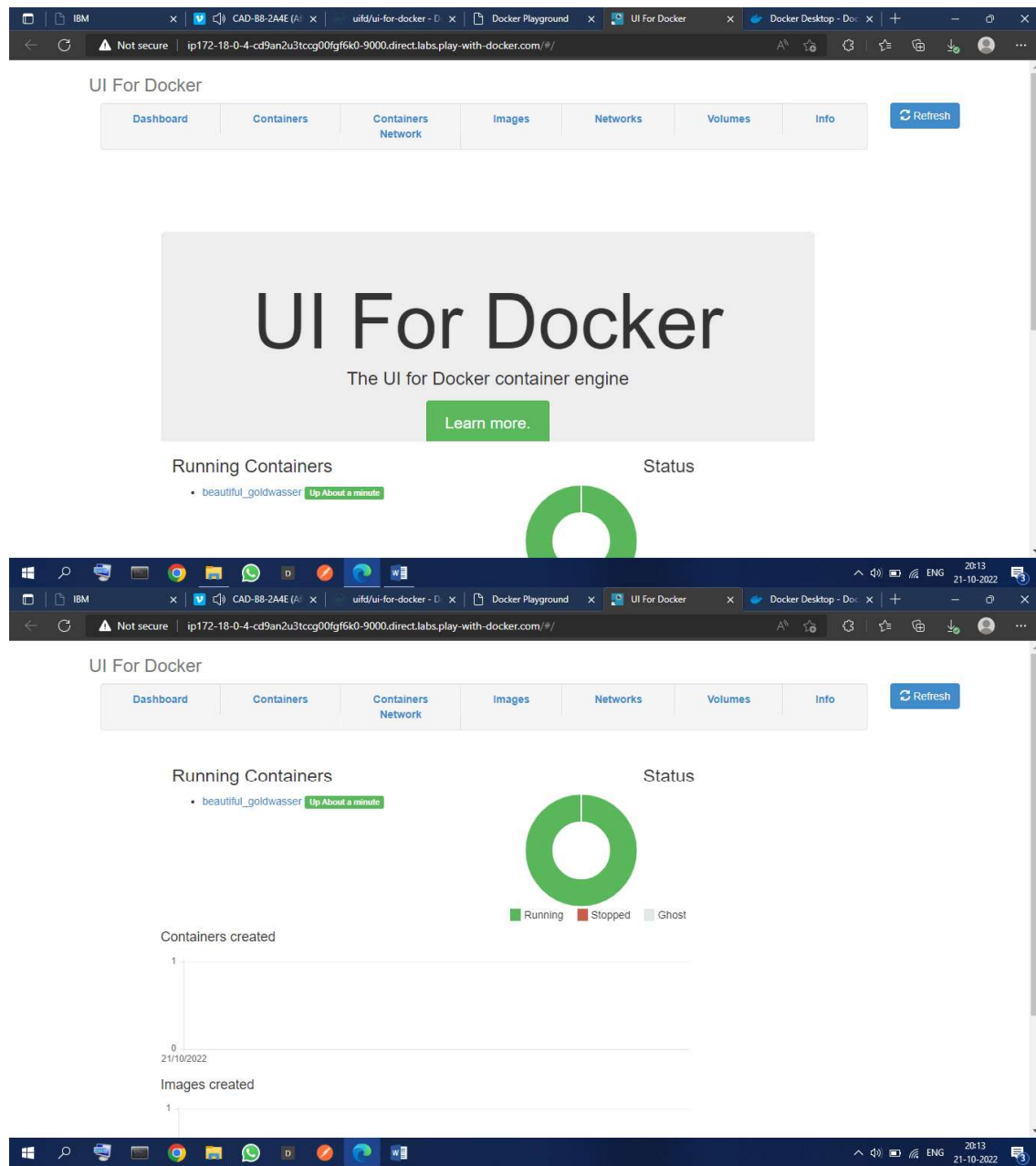
Docker and Kubernetes

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 repository is marked as deprecated, with a note stating "A web interface for Docker, formerly known as DockerUI. Deprecated, use Portainer for new features." The page includes a Docker Pull Command box with the command `docker pull uifd/ui-for-docker`.

The bottom section shows the Docker Playground interface. On the left, there is a sidebar with a timer showing 03:42:30, a "CLOSE SESSION" button, and a list of instances. The main area displays the details of a selected instance named `cd9an2u3_cd9av060qau0008hbjs0`. The instance's IP is `192.168.0.13`, and it has a memory limit of 9000:9000. The SSH command is `ssh ip172-18-0-4-cd9an2u3tccg00fgf6k0@direct.labs.play-with-docker.com`. The terminal output shows the following commands and their 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. #
#####
[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
c590dd163101ae795bdcea0eb1ddd98f6fe549cb5f24dacb9ff7c1931923fc0d
[node1] (local) root@192.168.0.13 ~
$
```



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:3.6
[auth] library/python:pull token for registry-1.docker.io
-> [internal] load build context
-> transferring context: 687B
-> [1/6] FROM docker.io/library/python:3.6@sha256:f8652afaf88c25f0d22354d547d892591067aa4026a7fa9a6819df9f300af6fc
-> resolve docker.io/library/python:3.6@sha256:f8652afaf88c25f0d22354d547d892591067aa4026a7fa9a6819df9f300af6fc
-> sha256:f8652afaf88c25f0d22354d547d892591067aa4026a7fa9a6819df9f300af6fc 1.86kB / 1.86kB
-> sha256:d97a4907a8ec07d5ac31872359c2de510f82214c0448e926393b376d3b6ad 2.22kB / 2.22kB
-> sha256:54260638d07c5e3ad24ce21fc889abbcb486a27634c0897086ff71f3f44b104 9.27kB / 9.27kB
-> sha256:0e29546d541cddb309281d21a73a9d1db78665c1b95b74f32b009e0b77a6e13 54.92MB / 54.92MB
-> sha256:9b829c73b52b92b97d5c07a54fbef3e921995a296c714b53a32ae67d19231fcd 5.15MB / 5.15MB
-> sha256:cb5b7ae361722f078ec53f35823ed21baa85d61d5d95cd5a95ab53d740cd56 10.87MB / 10.87MB
-> sha256:6494e481102b31c027ccac322ca463937fd805f569a93e6f15c01aade718793 54.57MB / 54.57MB
-> sha256:6f9f74896dfa93fe0172f594faba85e0b4e8a0481a0fed9112efc7e4d3c78f7 190.51MB / 190.51MB
-> sha256:5e3b1213efc56598e78bd607983945c164de2a37205e06a62dada023124dc743 6.29MB / 6.29MB
-> extracting sha256:0e29546d541cddb309281d21a73a9d1db78665c1b95b74f32b009e0b77a6e13
-> sha256:9fdddc56334f2eefad7e241bf5e7459c40ed105c5478076f41c1244bd96752 14.21MB / 14.21MB
-> extracting sha256:0b292c73b52b92b97d5c07a54fbef3e921995a296c714b53a32ae67d19231fcd 2.36
-> sha256:408f02044bac0432ca522cb0f254b1c91fcea6806bfeef0b0eb243b2f31bab7 235B / 235B
-> sha256:c4f42be2be53b90ebff040c1df13de538434ccc5f5d954a56848a6169a3a3f 2.21MB / 2.21MB
-> extracting sha256:6494e481102b31c027ccac322ca463937fd805f569a93e6f15c01aade718793
-> extracting sha256:6f9f74896dfa93fe0172f594faba85e0b4e8a0481a0fed9112efc7e4d3c78f7
-> extracting sha256:5e3b1213efc56598e78bd607983945c164de2a37205e06a62dada023124dc743
-> extracting sha256:9fdddc56334f2eefad7e241bf5e7459c40ed105c5478076f41c1244bd96752
-> extracting sha256:408f02044bac0432ca522cb0f254b1c91fcea6806bfeef0b0eb243b2f31bab7
-> extracting sha256:c4f42be2be53b90ebff040c1df13de538434ccc5f5d954a56848a6169a3a3f
-> [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 image
-> exporting layers
-> writing image sha256:1756719486df082fad5dae385c5221513f2f2d1b49a8d042b22a28af0379f19
-> 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>
```

