

Assignment - 4 Docker and Kubernetes

Batch no	11
Student Name	VIDHYASRI S
Student Reg Number	921319104219

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

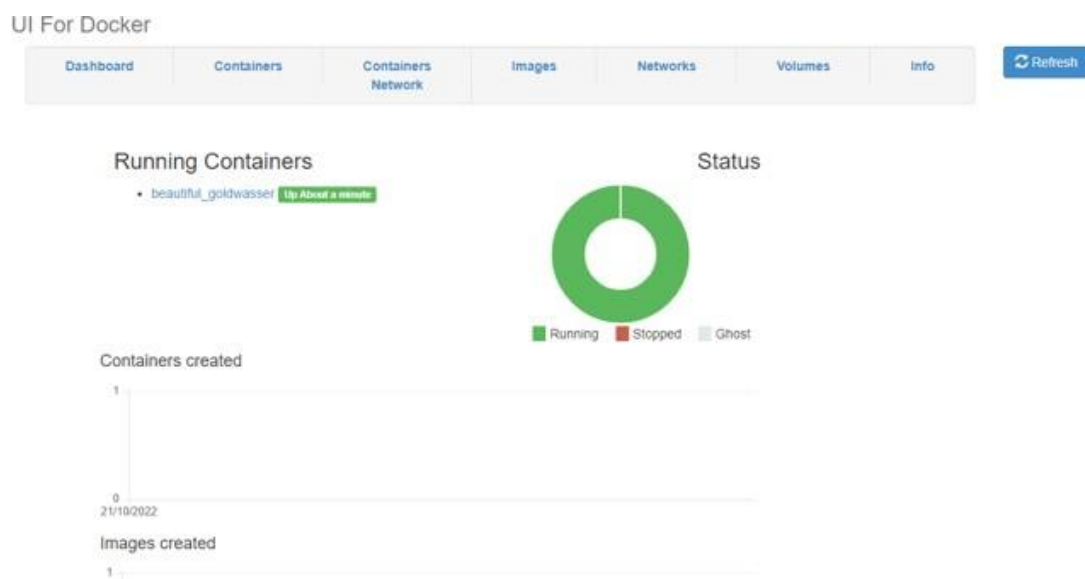
The screenshot shows the Docker Hub page for the repository 'uifd/ui-for-docker'. The page includes a blue cube icon, the repository name, and a star icon. Below the name, it says 'By uifd • Updated 6 years ago' and 'A web interface for Docker, formerly known as DockerUI. Deprecated, use Portainer for new features.' There are buttons for 'Other' and 'Image'. The 'Overview' tab is selected, showing a message that the repo is deprecated and development continues at 'portainer/portainer'. A 'chat on gitter' button is also present. On the right, the 'Tags' tab is visible. A 'Docker Pull Command' box shows the command 'docker pull uifd/ui-for-docker'.

The screenshot shows the Docker Playground interface. On the left, there's a sidebar with a clock showing '03:42:30', a 'CLOSE SESSION' button, and an 'Instances' section with a '+ ADD NEW INSTANCE' button. Below this, there's a list of instances with one instance named '192.168.0.13' and 'node1'. The main area shows the details of the selected instance, including its IP address '192.168.0.13', memory, CPU, and SSH access. A terminal window is open, showing the following commands and output:

```
cd9an2u3_cd9av060qau0008hbjs0
@
192.168.0.13 OPEN PORT
Memory CPU
SSH
ssh ip172-18-0-4-cd9an2u3tccg00fg6k0@direct.labs.play-a
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.
=====
[roob1] (local) root@192.168.0.13 ~
$ docker pull uifd/ui-for-docker
Using default tag: latest
latest: Pulling from uifd/ui-for-docker
41194d080c8: Pull complete
Digest: sha256:fe371ff5a69549269b24073a5ab1244dd4c0b834cbadf244870572150b1eb749
Status: Downloaded newer image for uifd/ui-for-docker:latest
docker.io/uifd/ui-for-docker:latest
[roob1] (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
c590dd163101ae795bdcea0eb1dd498f6fe549cb5f24dab9ff7c1931923fcd0d
[roob1] (local) root@192.168.0.13 ~
$
```

UI For Docker



```

-> [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: 607B
-> [1/8] FROM docker.io/library/python:3.6@sha256:f8052a7f80c25f0d22354d547082591867aa4026a7faba81bd9f180aaf0fc
-> resolve docker.io/library/python:3.6@sha256:f8052a7f80c25f0d22354d547082591867aa4026a7faba81bd9f180aaf0fc
-> sha256:f8052a7f80c25f0d22354d547082591867aa4026a7faba81bd9f180aaf0fc 1.05kB / 1.05kB
-> sha256:d097a4907a8ec070d5ac11022359c2de510f82114c0440e926303b376d3d60d 2.22kB / 2.22kB
-> sha256:54260030073e3ac74c8e21fc688abbc8486a27634c002000ff71f3f44b104 9.27kB / 9.27kB
-> sha256:0e29546d541c0d409281d21a73a0d1db70665c1b05b74f32b00000077ade1e3 54.92MB / 54.92MB
-> sha256:90828c73052b02b07d5c07a54f10f3e921095a290c714053a32ae67d182211fd 5.15MB / 5.15MB
-> sha256:c0503ae103722f070eac53f50821ed71ba88581d5095cd5a05ab532748cd56 10.87MB / 10.87MB
-> sha256:6454a4211022031c027cc322c443137f080f540a3a6f15c01aade718791 54.37MB / 54.53MB
-> sha256:0f9f74800d053fe01727594fand5e0044a0401a9fef001120fc764d3c70f7 100.51MB / 100.51MB
-> sha256:5e301215efc50598c78b0007081045c1644e2a7705e06a02dad021124dc743 6.20MB / 6.20MB
-> extracting sha256:0e29546d541c0d409281d21a73a0d1db70665c1b05b74f32b00000077ade1e3
-> sha256:0f0f0f58134f5a6d7e241b5a745c40a0105c5070070f41c1344b00752 14.11MB / 14.21MB
-> extracting sha256:90828c73052b02b07d5c07a54f10f3e921095a290c714053a32ae67d182211fd
-> extracting sha256:c0503ae103722f070eac53f50821ed71ba88581d5095cd5a05ab532748cd56
-> sha256:404f02040a0412c52c0b0754b109f6e0000f0e00005430f31b07 3.95B / 3.95B
-> sha256:c442be2be5109000bffc040c10f13de530434ccc5f5d0504a5608a0100aba3f
-> extracting sha256:04040e11022031c027cc32c443137f080f540a3a6f15c01aade718791
-> extracting sha256:0f9f74800d053fe01727594fand5e0044a0401a9fef001120fc764d3c70f7
-> extracting sha256:5e301215efc50598c78b0007081045c1644e2a7705e06a02dad021124dc743
-> extracting sha256:0f0f0f58134f5a6d7e241b5a745c40a0105c5070070f41c1344b00752
-> extracting sha256:04040e11022031c027cc32c443137f080f540a3a6f15c01aade718791
-> extracting sha256:c442be2be5109000bffc040c10f13de530434ccc5f5d0504a5608a0100aba3f
-> [2/8] WORKDIR /app
-> [3/8] ADD . /app
-> [4/8] COPY requirements.txt /app
-> [5/8] RUN python3 -m pip install -r requirements.txt
-> [6/8] RUN python3 -m pip install lib_d0
-> exporting to image
-> exporting layers
-> writing image sha256:17507104000f002fa05da0305c322333f2f2a1b00001d2b22a20f0370f10
-> 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\WK-PC\Desktop\job-portal-main>

create a docker file for the job portal application and deploy it in Docker desktop application

