

Seif El-Din Mohamed Aboelhassan Hendawy

Lab 3

Problem 1:

Create a reactjs simple app

Create a dockerfile to containerize the reactapp

Build the image and test it

(Bonus) create a dockerfile for the same app in smaller size using multi staging

```
Dockerfile 1 X
my-app > Dockerfile > ...
1 FROM node:14 as builder
2
3 WORKDIR /app
4
5 COPY package*.json ./
6
7 RUN npm install
8
9 COPY . .
10
11 RUN npm run build
12
13 FROM nginx:alpine
14
15 COPY --from=builder /app/build /usr/share/nginx/html
16
17 EXPOSE 80
18
19 CMD ["nginx", "-g", "daemon off;"]
20
```

```
seif@alienware-17-r5:~/Downloads/my-app$ docker build -t my-react-app .
[+] Building 207.3s (14/14) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 266B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load metadata for docker.io/library/nginx:alpine
=> [internal] load metadata for docker.io/library/node:14
=> CACHED [stage-1 1/2] FROM docker.io/library/nginx:alpine@sha256:647c5c83418c19eef0cddc647b9899326e30815763
=> [builder 1/6] FROM docker.io/library/node:14@sha256:a158d3b9b4e3fa813fa6c8c590b8f0a860e015ad4e59bbce5744d2
=> [internal] load build context
=> => transferring context: 176.33MB
=> CACHED [builder 2/6] WORKDIR /app
=> [builder 3/6] COPY package*.json ./
=> [builder 4/6] RUN npm install
=> [builder 5/6] COPY . .
=> [builder 6/6] RUN npm run build
=> [stage-1 2/2] COPY --from=builder /app/build /usr/share/nginx/html
=> exporting to image
=> => exporting layers
=> writing image sha256:873f3a4a91d1bf8993e812b828327912324da28ce6bf3f811b6a345ed4109f5f
=> naming to docker.io/library/my-react-app
seif@alienware-17-r5:~/Downloads/my-app$ docker run -d -p 8080:80 --name react-app-container my-react-app
1378a3d8d10cde961218edda879a9c392cc1665e918e97b4669c729d6b482
```



Problem 2:

Create a dockerfile for ubuntu image which sleeps by default for 5 sec or sleeps according to the given number in the docker command

```
Dockerfile U X
Lab3 > Dockerfile > ...
1 FROM ubuntu:latest
2
3 ENTRYPOINT ["sleep"]
4
5 CMD ["5"]

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL GITLENS
bash - Lab3 + - - - - -

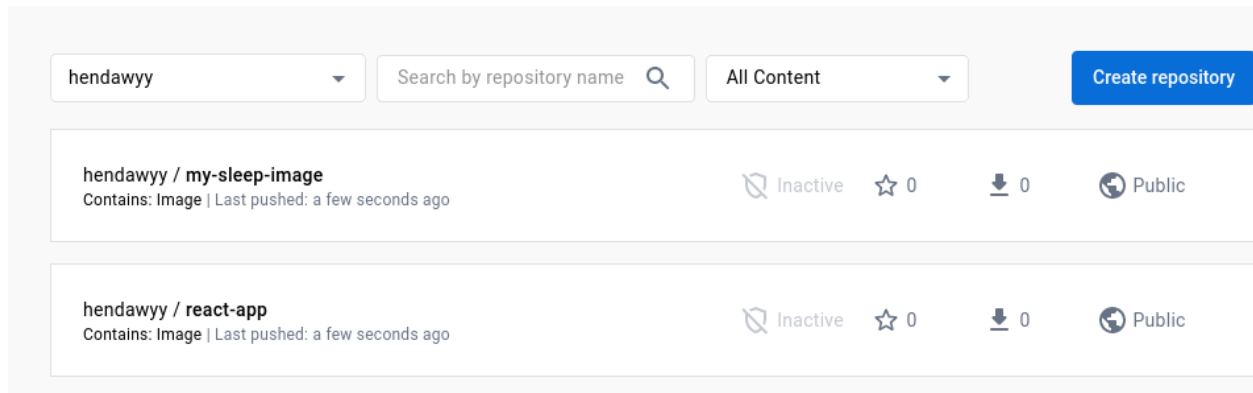
seif@alienware-17-r5:~/Downloads/Lab3$ docker build -t my-sleep-image .
[+] Building 2.6s (5/5) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 88B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load metadata for docker.io/library/ubuntu:latest
=> CACHED [1/1] FROM docker.io/library/ubuntu:latest@sha256:0bced47fffa3361afa981854fcabcd4577cd43cebbb808cea2b1f33a3dd7f508
=> => resolve docker.io/library/ubuntu:latest@sha256:0bced47fffa3361afa981854fcabcd4577cd43cebbb808cea2b1f33a3dd7f508
=> => exporting to image
=> => exporting layers
=> => writing image sha256:22efad91322ba60e1b0c0f71cedf26884e3006bdb27de8ea2893d6d920c0f28a
=> => naming to docker.io/library/my-sleep-image
seif@alienware-17-r5:~/Downloads/Lab3$ docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
react-app latest 873f3a4a91d1 27 hours ago 42MB
nginx latest 5cb37b3417a1 2 days ago 187MB
nginx <none> 89da1fb6dcb9 2 weeks ago 187MB
my-sleep-image latest 22efad91322b 6 weeks ago 77.8MB
seif@alienware-17-r5:~/Downloads/Lab3$ docker run my-sleep-image
seif@alienware-17-r5:~/Downloads/Lab3$ docker run my-sleep-image 10
seif@alienware-17-r5:~/Downloads/Lab3$ docker run my-sleep-image 30
seif@alienware-17-r5:~/Downloads/Lab3$ docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
react-app latest 873f3a4a91d1 27 hours ago 42MB
nginx latest 5cb37b3417a1 2 days ago 187MB
nginx <none> 89da1fb6dcb9 2 weeks ago 187MB
my-sleep-image latest 22efad91322b 6 weeks ago 77.8MB
```

Problem 3:

Push the images created in Problem 1&2 into your docker hub repo

```
seif@alienware-17-r5:~/Downloads/Lab3$ docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
react-app latest 873f3a4a91d1 27 hours ago 42MB
nginx latest 5cb37b3417a1 2 days ago 187MB
nginx <none> 89da1fb6dcb9 2 weeks ago 187MB
my-sleep-image latest 22efad91322b 6 weeks ago 77.8MB
seif@alienware-17-r5:~/Downloads/Lab3$ docker login
Login with your Docker ID to push and pull images from Docker Hub. If you don't have a Docker ID, head over to
Username: hendawyy
Password:
WARNING! Your password will be stored unencrypted in /home/seif/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
seif@alienware-17-r5:~/Downloads/Lab3$ docker tag react-app:latest hendawyy/react-app:latest
seif@alienware-17-r5:~/Downloads/Lab3$ docker tag my-sleep-image:latest hendawyy/my-sleep-image:latest
seif@alienware-17-r5:~/Downloads/Lab3$ docker push hendawyy/react-app:latest
The push refers to repository [docker.io/hendawyy/react-app]
75a65dc19327: Pushed
abaf9665d5d7: Mounted from library/nginx
6ec20fc79771: Mounted from library/nginx
6f2a9eaf22e9: Mounted from library/nginx
b938c7b9f1b5: Mounted from library/nginx
de91f65bcb4: Mounted from library/nginx
ba960ae476af: Mounted from library/nginx
604cbcdf1215: Mounted from library/nginx
36b50b131297: Mounted from library/nginx
latest: digest: sha256:4b8ec18f2dad3bf658cc869bd317183887e751a1d3fa2b7c8649d0b2efef5b1a size: 2199
seif@alienware-17-r5:~/Downloads/Lab3$ docker push hendawyy/my-sleep-image:latest
The push refers to repository [docker.io/hendawyy/my-sleep-image]
59c56ae1fb4: Mounted from library/ubuntu
latest: digest: sha256:d49678d07fade82058c8047899b49215e725bfde3cab5435a4066528dc99a942 size: 529
```



Problem 4:

Create 2 nginx containers with network type bridge, enter to one of them and use curl command to view the content of the other container.

```
seif@alienware-17-r5:~$ docker network create my-network
seif@alienware-17-r5:~$ docker run -d --name nginx-c1 --network my-network nginx
346a6142387ac45a395c8d4f21e75a76b54914bca1e47cc1b494213af1ca5097
seif@alienware-17-r5:~$ docker run -d --name nginx-c2 --network my-network nginx
d96cfd4f3aa94cb92b87830f4d53c3de033a51336ea90134c350a643b4e7b313
seif@alienware-17-r5:~$ docker exec -it nginx-c1 /bin/bash
root@346a6142387a:/# curl nginx-c2
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
body { width: 35em; margin: 0 auto;
font-family: Tahoma, Verdana, Arial, sans-serif; }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
root@346a6142387a:/#
```