Exercises

Part 1, Basics

1.1

Practice the commands.

Start 3 containers from image that does not automatically exit, such as nginx, detached.

Stop 2 of the containers leaving 1 up.

Prove that you have completed this part of exercise by delivering the output for docker ps -a.

$ docker run mono

```
cd@DESKTOP-VCNQ3D1:~$ docker run mono
Unable to find image 'mono:latest' locally
latest: Pulling from library/mono
a4b007099961: Downloading [==================================================>  ]  26.24MB/27.15MB
18785d1845f5: Download complete
f29cf656fe88: Downloading [============================>                        ]  37.15MB/64.76MB
8c82c867b649: Downloading [==>                                                  ]  6.466MB/141.4MB
```

$ docker run jetty

```
cd@DESKTOP-VCNQ3D1:~$ docker run jetty
2022-03-27 12:06:10.619:INFO::main: Logging initialized @106ms to org.eclipse.jetty.util.log.StdErrLog
2022-03-27 12:06:10.871:INFO:oejs.Server:main: jetty-9.4.45.v20220203; built: 2022-02-03T09:14:34.105Z; git: 4a0c91c0be53805e3fcffdcdcc9587d5301863db; jvm 17.0.2+8-86
2022-03-27 12:06:10.888:INFO:oejdp.ScanningAppProvider:main: Deployment monitor [file:///var/lib/jetty/webapps/] at interval 1
2022-03-27 12:06:10.912:INFO:oejs.AbstractConnector:main: Started ServerConnector@358ee631{HTTP/1.1, (http/1.1)}{0.0.0.0:8080}
2022-03-27 12:06:10.913:INFO:oejs.Server:main: Started @413ms
```

$ docker run nginx

```
cd@DESKTOP-VCNQ3D1:~$ docker run nginx
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
/docker-entrypoint.sh: Configuration complete; ready for start up
2022/03/27 12:07:46 [notice] 1#1: using the "epoll" event method
2022/03/27 12:07:46 [notice] 1#1: nginx/1.21.6
2022/03/27 12:07:46 [notice] 1#1: built by gcc 10.2.1 20210110 (Debian 10.2.1-6)
2022/03/27 12:07:46 [notice] 1#1: OS: Linux 5.10.16.3-microsoft-standard-WSL2
2022/03/27 12:07:46 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2022/03/27 12:07:46 [notice] 1#1: start worker processes
2022/03/27 12:07:46 [notice] 1#1: start worker process 31
2022/03/27 12:07:46 [notice] 1#1: start worker process 32
2022/03/27 12:07:46 [notice] 1#1: start worker process 33
2022/03/27 12:07:46 [notice] 1#1: start worker process 34
2022/03/27 12:07:46 [notice] 1#1: start worker process 35
2022/03/27 12:07:46 [notice] 1#1: start worker process 36
2022/03/27 12:07:46 [notice] 1#1: start worker process 37
2022/03/27 12:07:46 [notice] 1#1: start worker process 38
2022/03/27 12:07:46 [notice] 1#1: start worker process 39
2022/03/27 12:07:46 [notice] 1#1: start worker process 40
2022/03/27 12:07:46 [notice] 1#1: start worker process 41
2022/03/27 12:07:46 [notice] 1#1: start worker process 42
^C2022/03/27 12:07:53 [notice] 1#1: signal 2 (SIGINT) received, exiting
2022/03/27 12:07:53 [notice] 31#31: exiting
```

1.2

We've left containers and a image that won't be used anymore and are taking space, as docker ps -a and docker images will reveal. Clean the docker daemon from all images and containers.

Prove that you have completed this part of exercise by delivering the output for docker ps -a and docker images

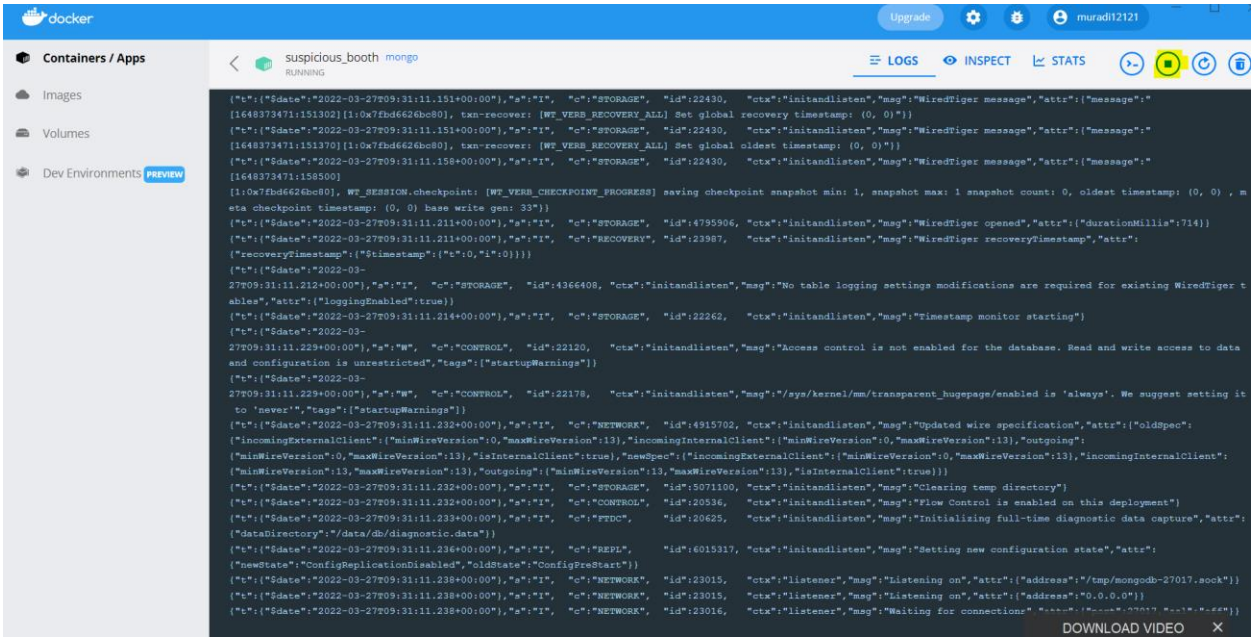
$ docker ps –a

```
cd@DESKTOP-VCNQ3D1:~$ docker ps -a
CONTAINER ID   IMAGE                                COMMAND                  CREATED       STATUS                     PORTS     NAMES
3034d9a60417   devopsdockeruh/exec_bash_exercise    "docker-entrypoint.s…"   2 hours ago   Exited (0) 16 minutes ago            elated_mcnulty
acd6aafa5692   devopsdockeruh/pull_exercise         "node index.js"          2 hours ago   Exited (0) 2 hours ago               nostalgic_golick
d5ded45a2605   devopsdockeruh/pull_exercise:latest  "node index.js"          2 hours ago   Up 2 hours                           ecstatic_blackwell
8b32a929e221   devopsdockeruh/pull_exercise:latest  "node index.js"          2 hours ago   Up 2 hours                           charming_kalam
43d0c7cc97de   devopsdockeruh/pull_exercise         "node index.js"          2 hours ago   Up 2 hours                           focused_easley
ad2dbecd70ff   devopsdockeruh/pull_exercise         "node index.js"          2 hours ago   Up 2 hours                           affectionate_hugle
1c92a95d7815   nginx:latest                         "/docker-entrypoint.…"   2 hours ago   Exited (0) 2 hours ago               hopeful_liskov
4c00b0bb886e   jetty:latest                         "/docker-entrypoint.…"   2 hours ago   Exited (143) 2 hours ago             infallible_bose
da80b98e6bfd   jetty                                "/docker-entrypoint.…"   2 hours ago   Exited (130) 2 hours ago             wizardly_darwin
87eb9a727b90   mongo                                "docker-entrypoint.s…"   3 hours ago   Exited (0) 2 hours ago               suspicious_booth
eb718f31c726   ubuntu                               "bash"                   3 hours ago   Exited (0) 3 hours ago               great_euclid
8af43183017b   nginx                                "/docker-entrypoint.…"   3 hours ago   Exited (0) 2 hours ago               amazing_lederberg
5e02c66db1df   ubuntu:16.04                         "/bin/bash"              28 hours ago  Exited (0) 27 hours ago              vibrant_panini
fc23f50aa20e   ubuntu:16.04                         "/bin/bash"              28 hours ago  Exited (0) 28 hours ago              jovial_antonelli
f8e8526b0ef4   ubuntu:16.04                         "/bin/bash"              2 days ago    Exited (0) 2 days ago                serene_burnell
674db01e3227   ubuntu:16.10                         "/bin/echo 'Hello wo…"   2 days ago    Exited (0) 2 days ago                heuristic_meninsky
cd@DESKTOP-VCNQ3D1:~$
```

$ docker images

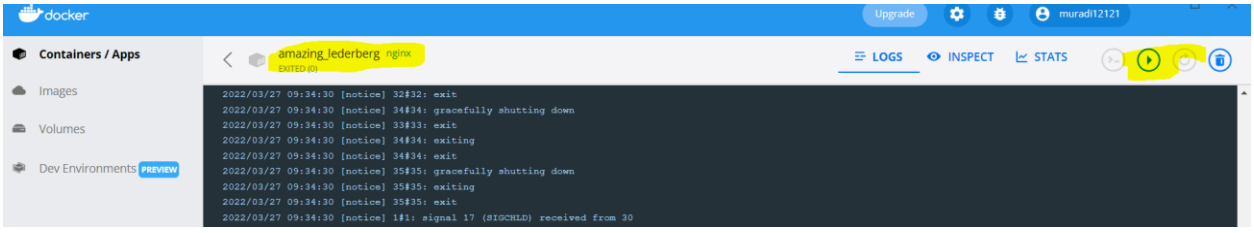
```
cd@DESKTOP-VCNQ3D1:~$ docker images
REPOSITORY                          TAG       IMAGE ID       CREATED         SIZE
delner/ping                         latest    44e13e2492ef   21 hours ago    135MB
<none>                              <none>    0c15e46ab04b   21 hours ago    170MB
jetty                               latest    08135dd496c4   2 days ago      483MB
mongo                               latest    798d1656acba   7 days ago      698MB
ubuntu                              latest    ff0fea8310f3   9 days ago      72.8MB
nginx                               latest    f2f70adc5d89   9 days ago      142MB
httpd                               2.4       b9bd7e513e0f   9 days ago      144MB
postgres                            latest    5cd1494671e9   9 days ago      376MB
ubuntu                              16.04     b6f507652425   6 months ago    135MB
devopsdockeruh/exec_bash_exercise   latest    c52ece77379e   22 months ago   942MB
devopsdockeruh/pull_exercise        latest    d9854bc0e13a   3 years ago     75.3MB
ubuntu                              16.10     7d3f705d307c   4 years ago     107MB
```

$ docker stop



Docker start container

Docker start container



1.3

Start image devopsdockeruh/pull_exercise with flags -it like so: docker run -it devopsdockeruh/pull_exercise. It will wait for your input. Navigate through docker hub to find the docs and Dockerfile that was used to create the image.

Read the Dockerfile and/or docs to learn what input will get the application to answer a “secret message”.

Submit the secret message and command(s) given to get it as your answer.

$ docker run –it devopsdockeruh/pull_exercise

```
cd@DESKTOP-VCNQ3D1:~$ docker run -it devopsdockeruh/pull_exercise
Give me the password: basics
You found the correct password. Secret message is:
"This is the secret message"
```

1.4

Now that we’ve warmed up it’s time to get inside a container while it’s running!

Start image devopsdockeruh/exec_bash_exercise, it will start a container with clock-like features and create a log. Go inside the container and use tail -f ./logs.txt to follow the logs. Every 15 seconds the clock will send you a “secret message”.

Submit the secret message and command(s) given as your answer.

$ docker run devopsdockeruh/exec_bash_exercise

```
cd@DESKTOP-VCNQ3D1:~$ docker run devopsdockeruh/exec_bash_exercise
Wrote to file /usr/app/logs.txt
Wrote to file /usr/app/logs.txt
Wrote to file /usr/app/logs.txt
Wrote to file /usr/app/logs.txt
Wrote to file /usr/app/logs.txt
Wrote to file /usr/app/logs.txt
Wrote to file /usr/app/logs.txt
Wrote to file /usr/app/logs.txt
Wrote to file /usr/app/logs.txt
```

```
cd@DESKTOP-VCNQ3D1:~$ docker tail -f ./logs.txt
```

1.5

Start a ubuntu image with the process tree

docker run -d -it --name <nimi> ubuntu Will create you the container.

docker exec -it <nimi> tree Will run the tree program and display the output.

OCI runtime exec failed: exec failed: container_linux.go:346: starting container process caused "exec: \"tree\": executable file not found in $PATH": unknown Something is obviously broken. Maybe Tree is missing?

docker exec -it <nimi> bash In to the container and fix the problem. Exit the container, leaving it running.

Now run the tree command again from the command-line.

For the following exercises, return both Dockerfile(s) and the command you used to run the container(s)

$ docker run –d –it –name jetty ubuntu

```
cd@DESKTOP-VCNQ3D1:~$ docker run -d -it --name jetty ubuntu
343328cef797b5ac1878597f0cec97a8df64c2ecabeed8288a3a6509397ea45b
cd@DESKTOP-VCNQ3D1:~$
```

$ docker exec –t jetty mono

```
d@DESKTOP-VCNQ3D1:~$ docker exec -it jetty mono
CI runtime exec failed: exec failed: container_linux.go:380: starting container process caused: exec: "mono": executable file not found in $PATH: unknown
d@DESKTOP-VCNQ3D1:~$ docker exec -it jetty bash
oot@343328cef797:/# exit
xit
d@DESKTOP-VCNQ3D1:~$ docker exec -it jetty tree
CI runtime exec failed: exec failed: container_linux.go:380: starting container process caused: exec: "tree": executable file not found in $PATH: unknown
d@DESKTOP-VCNQ3D1:~$ docker exec -it jetty ubuntu
CI runtime exec failed: exec failed: container_linux.go:380: starting container process caused: exec: "ubuntu": executable file not found in $PATH: unknown
d@DESKTOP-VCNQ3D1:~$ docker exec -it jetty ubuntu
```