UNIVERSITÉ
DE REIMS
CHAMPAGNE-ARDENNE

Docker

PRACTICAL WORK 6

TIZIANO NARDONE
UNIVERSITY OF REIMS | FRANCE

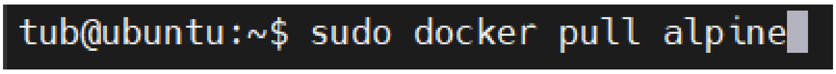
Table of Contents

I. Question 1: Docker installation for Ubuntu 20.04 (From repo)

- https://docs.docker.com/engine/install/ubuntu/

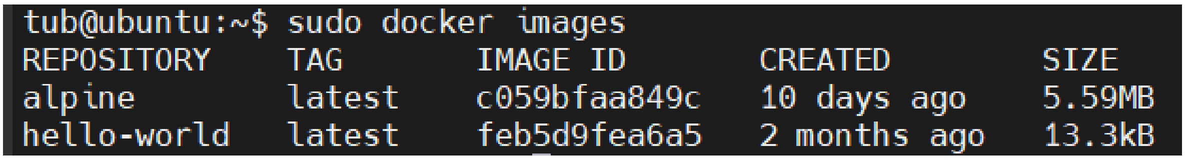
II. Question 2: Alpine with interactive mode

1. Alpine download



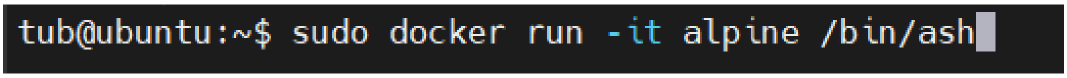
```
tub@ubuntu:~$ sudo docker pull alpine
```

2. List docker images



```
tub@ubuntu:~$ sudo docker images
REPOSITORY     TAG       IMAGE ID       CREATED         SIZE
alpine         latest    c059bfaa849c   10 days ago     5.59MB
hello-world    latest    feb5d9fea6a5   2 months ago    13.3kB
```

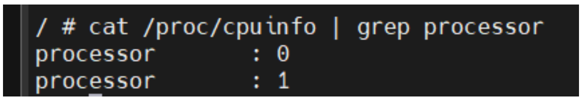
3. Launch Alpine with the interactive mode



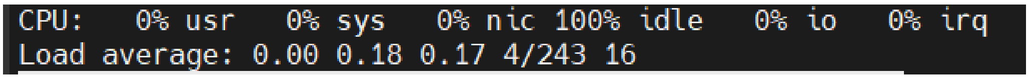
```
tub@ubuntu:~$ sudo docker run -it alpine /bin/ash
```

4. Get information of the container

- CPU

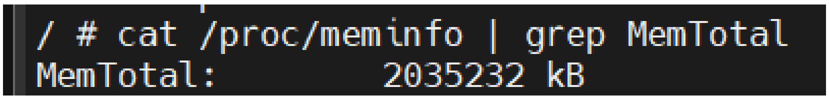


```
/ # cat /proc/cpuinfo | grep processor
processor       : 0
processor       : 1
```



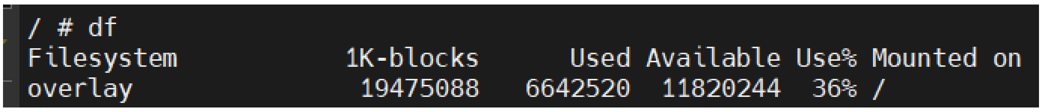
```
CPU:   0% usr   0% sys   0% nic 100% idle   0% io   0% irq
Load average: 0.00 0.18 0.17 4/243 16
```

- Memory



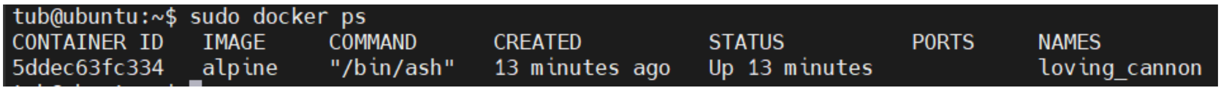
```
/ # cat /proc/meminfo | grep MemTotal
MemTotal:        2035232 kB
```

- Disk



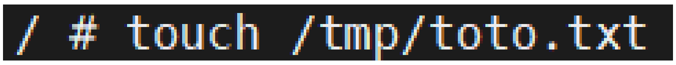
```
/ # df
Filesystem           1K-blocks      Used Available Use% Mounted on
overlay               19475088   6642520  11820244  36% /
```

5. State of the container from the host



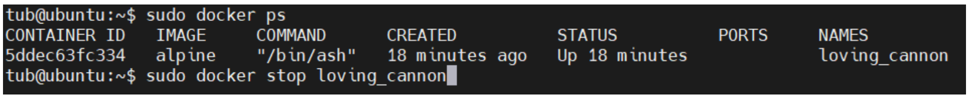
```
tub@ubuntu:~$ sudo docker ps
CONTAINER ID   IMAGE     COMMAND       CREATED          STATUS          PORTS     NAMES
5ddec63fc334   alpine    "/bin/ash"    13 minutes ago   Up 13 minutes             loving_cannon
```

6. Create a file on the '/tmp' of the container



```
/ # touch /tmp/toto.txt
```

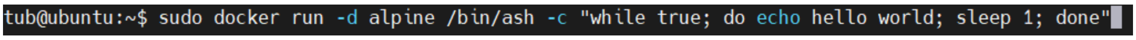
7. Stop the container or "exit" command inside the container



```
tub@ubuntu:~$ sudo docker ps
CONTAINER ID   IMAGE     COMMAND      CREATED          STATUS          PORTS     NAMES
5ddec63fc334   alpine    "/bin/ash"   18 minutes ago   Up 18 minutes             loving_cannon
tub@ubuntu:~$ sudo docker stop loving_cannon
```

III. Container with daemon mode

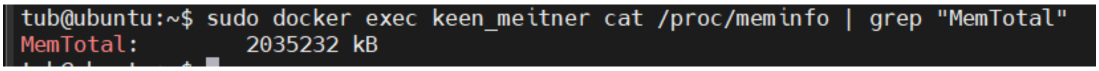
8. Launch Alpine with daemon mode



```
tub@ubuntu:~$ sudo docker run -d alpine /bin/ash -c "while true; do echo hello world; sleep 1; done"
```

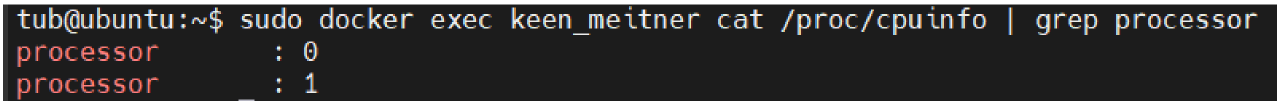
9. Get information of the container from the host

- Memory

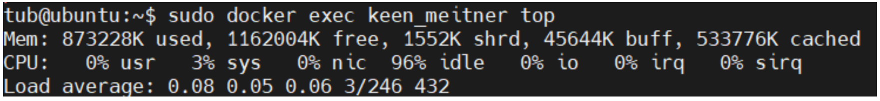


```
tub@ubuntu:~$ sudo docker exec keen_meitner cat /proc/meminfo | grep "MemTotal"
MemTotal:        2035232 kB
```

- CPU

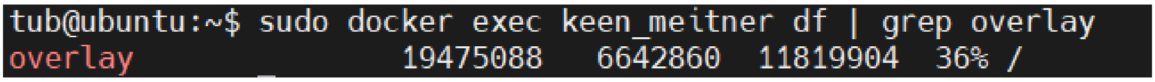


```
tub@ubuntu:~$ sudo docker exec keen_meitner cat /proc/cpuinfo | grep processor
processor       : 0
processor       : 1
```



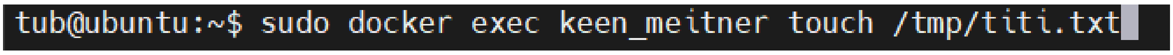
```
tub@ubuntu:~$ sudo docker exec keen_meitner top
Mem: 873228K used, 1162004K free, 1552K shrd, 45644K buff, 533776K cached
CPU:   0% usr   3% sys   0% nic  96% idle   0% io   0% irq   0% sirq
Load average: 0.08 0.05 0.06 3/246 432
```

- Disk



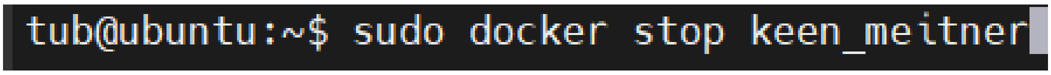
```
tub@ubuntu:~$ sudo docker exec keen_meitner df | grep overlay
overlay              19475088   6642860  11819904  36% /
```

10. Create a file inside the container



```
tub@ubuntu:~$ sudo docker exec keen_meitner touch /tmp/titi.txt
```

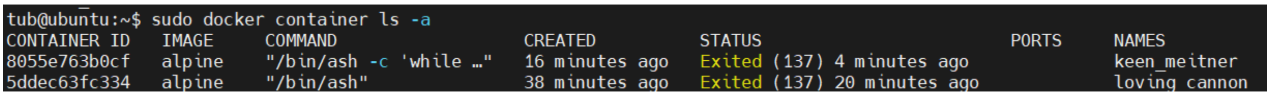
11. Stop the container



```
tub@ubuntu:~$ sudo docker stop keen_meitner
```

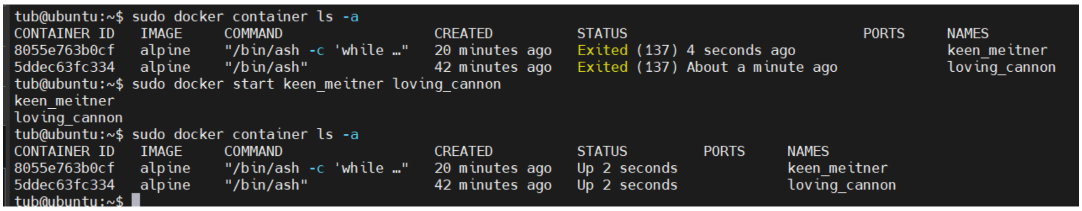
IV. Two containers

12. List containers available



```
tub@ubuntu:~$ sudo docker container ls -a
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS                       PORTS     NAMES
8055e763b0cf   alpine    "/bin/ash -c 'while …"   16 minutes ago   Exited (137) 4 minutes ago             keen_meitner
5ddec63fc334   alpine    "/bin/ash"               38 minutes ago   Exited (137) 20 minutes ago            loving cannon
```

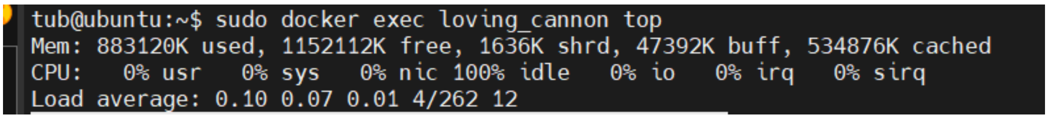
13. Launch two container with daemon mode



```
tub@ubuntu:~$ sudo docker container ls -a
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS                          PORTS     NAMES
8055e763b0cf   alpine    "/bin/ash -c 'while …"   20 minutes ago   Exited (137) 4 seconds ago                keen_meitner
5ddec63fc334   alpine    "/bin/ash"               42 minutes ago   Exited (137) About a minute ago           loving_cannon
tub@ubuntu:~$ sudo docker start keen_meitner loving_cannon
keen_meitner
loving_cannon
tub@ubuntu:~$ sudo docker container ls -a
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS         PORTS     NAMES
8055e763b0cf   alpine    "/bin/ash -c 'while …"   20 minutes ago   Up 2 seconds             keen_meitner
5ddec63fc334   alpine    "/bin/ash"               42 minutes ago   Up 2 seconds             loving_cannon
tub@ubuntu:~$
```

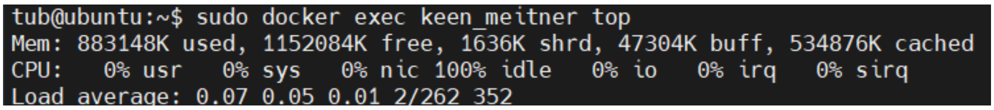
14. Get information of the two containers

- CTN 1



```
tub@ubuntu:~$ sudo docker exec loving_cannon top
Mem: 883120K used, 1152112K free, 1636K shrd, 47392K buff, 534876K cached
CPU:   0% usr   0% sys   0% nic 100% idle   0% io   0% irq   0% sirq
Load average: 0.10 0.07 0.01 4/262 12
```

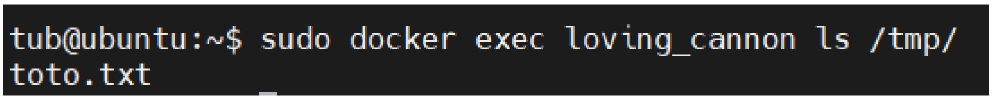
- CTN 2



```
tub@ubuntu:~$ sudo docker exec keen_meitner top
Mem: 883148K used, 1152084K free, 1636K shrd, 47304K buff, 534876K cached
CPU:   0% usr   0% sys   0% nic 100% idle   0% io   0% irq   0% sirq
Load average: 0.07 0.05 0.01 2/262 352
```

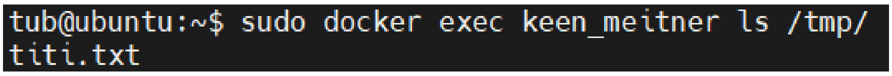
15. Check if both files created are still in the containers

- CTN 1



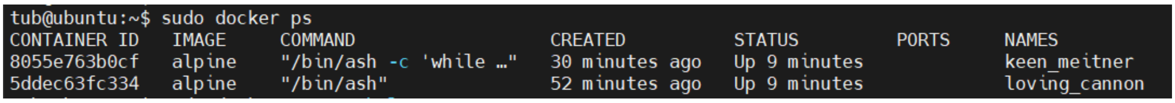
```
tub@ubuntu:~$ sudo docker exec loving_cannon ls /tmp/
toto.txt
```

- CTN 2

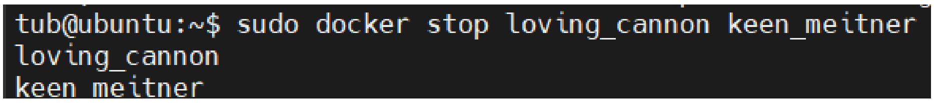


```
tub@ubuntu:~$ sudo docker exec keen_meitner ls /tmp/
titi.txt
```

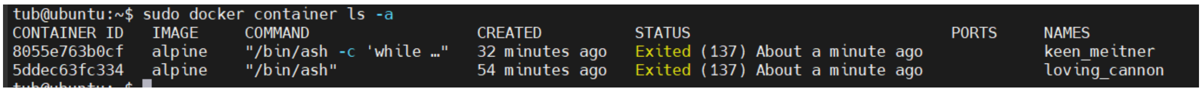
16. Stop both containers



```
tub@ubuntu:~$ sudo docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS         PORTS     NAMES
8055e763b0cf   alpine    "/bin/ash -c 'while …"   30 minutes ago   Up 9 minutes             keen_meitner
5ddec63fc334   alpine    "/bin/ash"               52 minutes ago   Up 9 minutes             loving_cannon
```

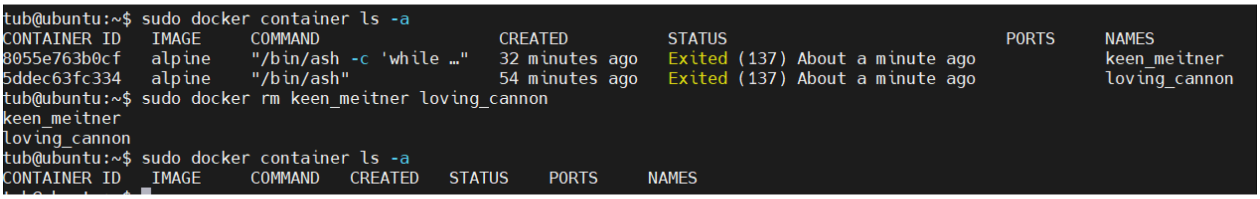


```
tub@ubuntu:~$ sudo docker stop loving_cannon keen_meitner
loving_cannon
keen_meitner
```



```
tub@ubuntu:~$ sudo docker container ls -a
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS                            PORTS     NAMES
8055e763b0cf   alpine    "/bin/ash -c 'while …"   32 minutes ago   Exited (137) About a minute ago             keen_meitner
5ddec63fc334   alpine    "/bin/ash"               54 minutes ago   Exited (137) About a minute ago             loving_cannon
```

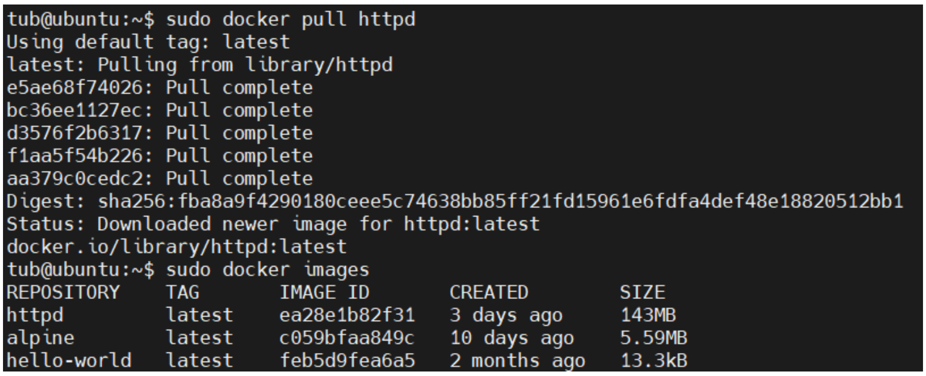
17. Destroy both containers



```
tub@ubuntu:~$ sudo docker container ls -a
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS                            PORTS     NAMES
8055e763b0cf   alpine    "/bin/ash -c 'while …"   32 minutes ago   Exited (137) About a minute ago             keen_meitner
5ddec63fc334   alpine    "/bin/ash"               54 minutes ago   Exited (137) About a minute ago             loving_cannon
tub@ubuntu:~$ sudo docker rm keen_meitner loving_cannon
keen_meitner
loving_cannon
tub@ubuntu:~$ sudo docker container ls -a
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
```

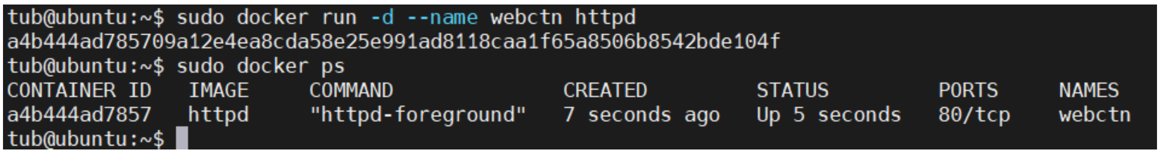
V. Simple web container

18. Download httpd image



```
tub@ubuntu:~$ sudo docker pull httpd
Using default tag: latest
latest: Pulling from library/httpd
e5ae68f74026: Pull complete
bc36ee1127ec: Pull complete
d3576f2b6317: Pull complete
f1aa5f54b226: Pull complete
aa379c0cedc2: Pull complete
Digest: sha256:fba8a9f4290180ceee5c74638bb85ff21fd15961e6fdfa4def48e18820512bb1
Status: Downloaded newer image for httpd:latest
docker.io/library/httpd:latest
tub@ubuntu:~$ sudo docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
httpd         latest    ea28e1b82f31   3 days ago     143MB
alpine        latest    c059bfaa849c   10 days ago    5.59MB
hello-world   latest    feb5d9fea6a5   2 months ago   13.3kB
```

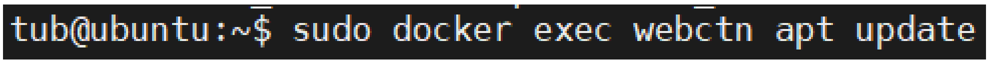
19. Start the Web container with daemon mode



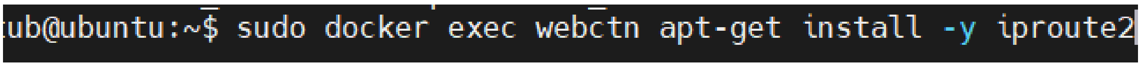
```
tub@ubuntu:~$ sudo docker run -d --name webctn httpd
a4b444ad785709a12e4ea8cda58e25e991ad8118caa1f65a8506b8542bde104f
tub@ubuntu:~$ sudo docker ps
CONTAINER ID   IMAGE     COMMAND              CREATED         STATUS         PORTS     NAMES
a4b444ad7857   httpd     "httpd-foreground"   7 seconds ago   Up 5 seconds   80/tcp    webctn
tub@ubuntu:~$
```

20. Get IP address & running processes

- IP command isn't installed

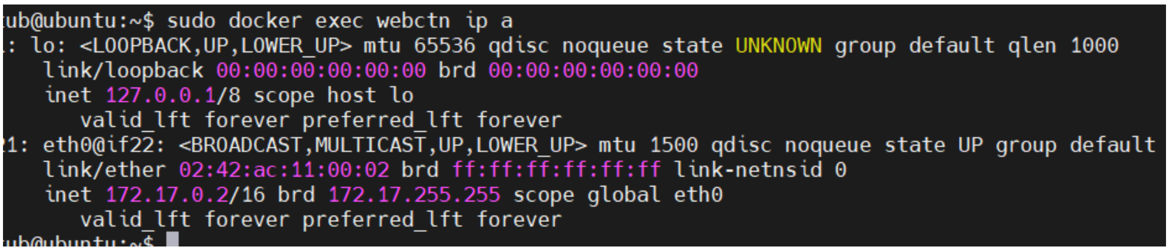


```
tub@ubuntu:~$ sudo docker exec webctn apt update
```



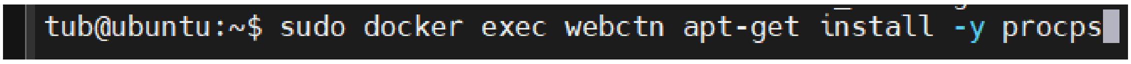
```
ub@ubuntu:~$ sudo docker exec webctn apt-get install -y iproute2
```

- IP addresses



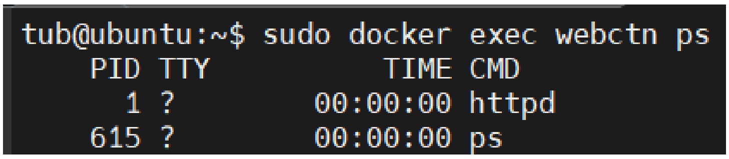
```
ub@ubuntu:~$ sudo docker exec webctn ip a
: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
   link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
   inet 127.0.0.1/8 scope host lo
      valid_lft forever preferred_lft forever
1: eth0@if22: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
   link/ether 02:42:ac:11:00:02 brd ff:ff:ff:ff:ff:ff link-netnsid 0
   inet 172.17.0.2/16 brd 172.17.255.255 scope global eth0
      valid_lft forever preferred_lft forever
ub@ubuntu:~$
```

- PS command isn't installed



```
tub@ubuntu:~$ sudo docker exec webctn apt-get install -y procps
```

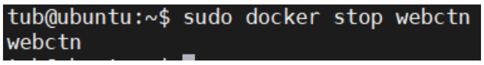
- Running processes



```
tub@ubuntu:~$ sudo docker exec webctn ps
    PID TTY          TIME CMD
      1 ?        00:00:00 httpd
    615 ?        00:00:00 ps
```

- No processes are running

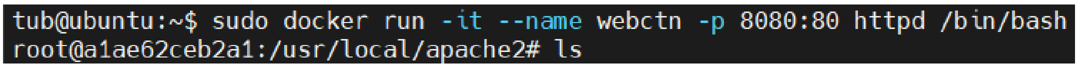
21. Stop web container



```
tub@ubuntu:~$ sudo docker stop webctn
webctn
```

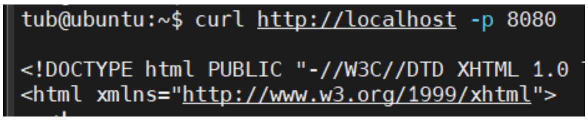
VI. Web container & port forwarding

22. Starting up web container with port forwarding



```
tub@ubuntu:~$ sudo docker run -it --name webctn -p 8080:80 httpd /bin/bash
root@a1ae62ceb2a1:/usr/local/apache2# ls
```

23. Browse the Apache web page using port forwarding

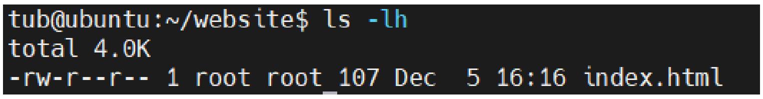


```
tub@ubuntu:~$ curl http://localhost -p 8080

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
<html xmlns="http://www.w3.org/1999/xhtml">
```

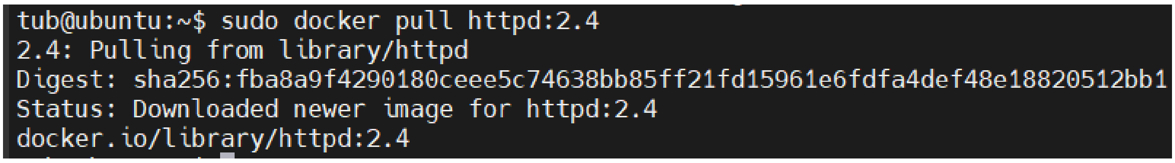
VII. Web container – Port forwarding – shared folder

- Create a folder on host and create a HTML file inside the folder



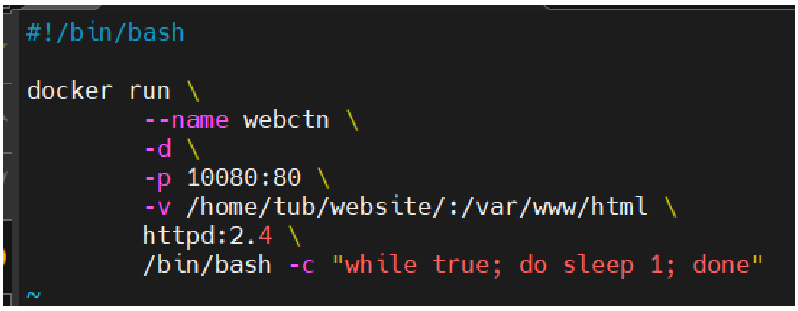
```
tub@ubuntu:~/website$ ls -lh
total 4.0K
-rw-r--r-- 1 root root 107 Dec  5 16:16 index.html
```

- Download the version 2.4 of the httpd container



```
tub@ubuntu:~$ sudo docker pull httpd:2.4
2.4: Pulling from library/httpd
Digest: sha256:fba8a9f4290180ceee5c74638bb85ff21fd15961e6fdfa4def48e18820512bb1
Status: Downloaded newer image for httpd:2.4
docker.io/library/httpd:2.4
```

24. Starting up the container

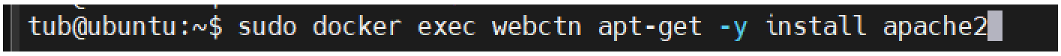


```
#!/bin/bash

docker run \
        --name webctn \
        -d \
        -p 10080:80 \
        -v /home/tub/website/:/var/www/html \
        httpd:2.4 \
        /bin/bash -c "while true; do sleep 1; done"
~
```

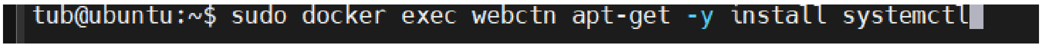
Installation of Apache2 & Systemctl packages

- Apache2



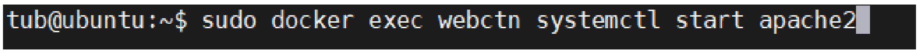
```
tub@ubuntu:~$ sudo docker exec webctn apt-get -y install apache2
```

- Systemctl

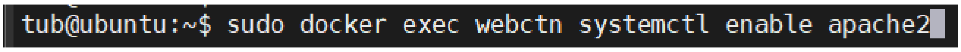


```
tub@ubuntu:~$ sudo docker exec webctn apt-get -y install systemctl
```

- Starting & enabling Apache

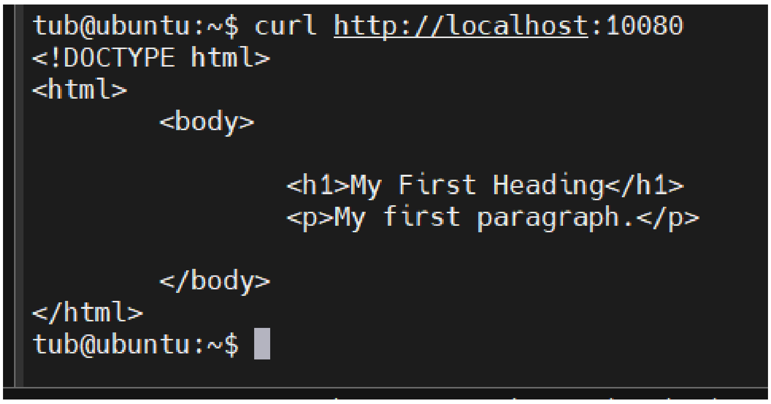


```
tub@ubuntu:~$ sudo docker exec webctn systemctl start apache2
```



```
tub@ubuntu:~$ sudo docker exec webctn systemctl enable apache2
```

25. Browse the web page located in the container using port forwarding

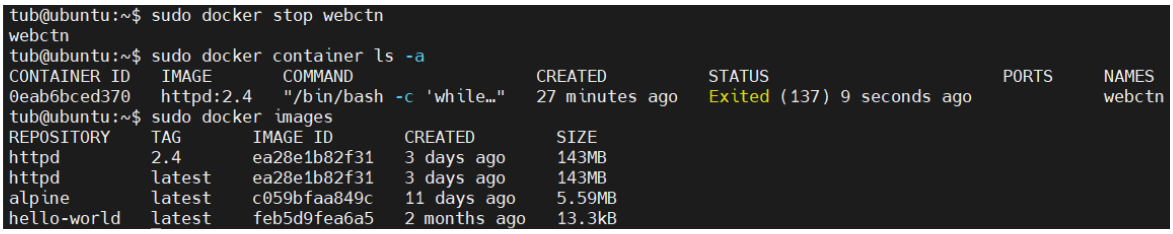


```
tub@ubuntu:~$ curl http://localhost:10080
<!DOCTYPE html>
<html>
        <body>

                <h1>My First Heading</h1>
                <p>My first paragraph.</p>

        </body>
</html>
tub@ubuntu:~$ 
```

26. Stop container – List containers & images

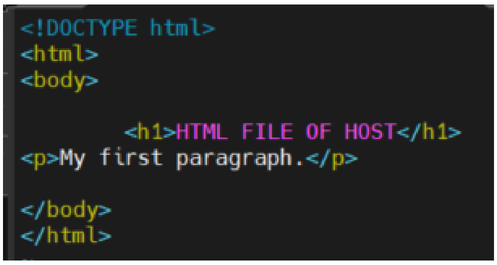


```
tub@ubuntu:~$ sudo docker stop webctn
webctn
tub@ubuntu:~$ sudo docker container ls -a
CONTAINER ID   IMAGE       COMMAND                  CREATED          STATUS                       PORTS     NAMES
0eab6bced370   httpd:2.4   "/bin/bash -c 'while…"   27 minutes ago   Exited (137) 9 seconds ago             webctn
tub@ubuntu:~$ sudo docker images
REPOSITORY    TAG       IMAGE ID       CREATED         SIZE
httpd         2.4       ea28e1b82f31   3 days ago      143MB
httpd         latest    ea28e1b82f31   3 days ago      143MB
alpine        latest    c059bfaa849c   11 days ago     5.59MB
hello-world   latest    feb5d9fea6a5   2 months ago    13.3kB
```

VIII. Question 8: Dockerfile

- Web server installation
- Create the user toto
- Copy the index.html file from host to guest
- Start container

- Update the host HMTL file

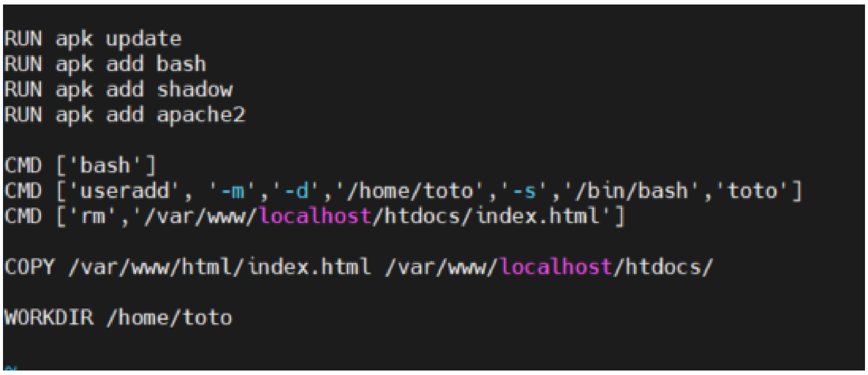


```
<!DOCTYPE html>
<html>
<body>

        <h1>HTML FILE OF HOST</h1>
<p>My first paragraph.</p>

</body>
</html>
```

- Dockerfile



```
RUN apk update
RUN apk add bash
RUN apk add shadow
RUN apk add apache2

CMD ['bash']
CMD ['useradd', '-m','-d','/home/toto','-s','/bin/bash','toto']
CMD ['rm','/var/www/localhost/htdocs/index.html']

COPY /var/www/html/index.html /var/www/localhost/htdocs/

WORKDIR /home/toto
```