Mohanad Mahmoud Sayed

1. Task1:

- Run the container hello-world

→Here I used command docker run hello-world to run the container hello-world

```
[root@localhost ~]# docker run hello-world

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
 $ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
 https://hub.docker.com/

For more examples and ideas, visit:
 https://docs.docker.com/get-started/

[root@localhost ~]#
```

- Check the container status

```
[root@localhost ~]# docker ps -a
CONTAINER ID   IMAGE         COMMAND       CREATED           STATUS                       PORTS     NAMES
c0190e442eed   hello-world   "/hello"      2 minutes ago     Exited (0) 2 minutes ago               busy_poincare
74e57b4297c1   ubuntu        "/bin/bash"   55 minutes ago    Exited (0) 55 minutes ago              quizzical_goldwasser
f270ceca67c3   ubuntu        "/bin/bash"   56 minutes ago    Exited (0) 56 minutes ago              goofy_blackburn
8c4a7a172427   ubuntu        "bash"        56 minutes ago    Exited (0) 56 minutes ago              agitated_volhard
6c350e6681d5   ubuntu        "bash"        About an hour ago Exited (127) About an hour ago         practical_kepler
7047b0cc04b4   hello-world   "/hello"      About an hour ago Exited (0) About an hour ago           eloquent_wright
[root@localhost ~]#
```

→Here appears the status of the hello-world container which I ran 2 mins ago

- Start the stopped container

→Using the ID of the hello world container which is the last container I ran, I started this stopped container and then checked using docker logs (container ID)

```
[root@localhost ~]# docker start c0190e442eed
c0190e442eed
[root@localhost ~]# docker logs c0190e442eed

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
 $ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
 https://hub.docker.com/

For more examples and ideas, visit:
 https://docs.docker.com/get-started/


Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
 $ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
 https://hub.docker.com/

For more examples and ideas, visit:
 https://docs.docker.com/get-started/

[root@localhost ~]#
```

- Remove the container

→Using the container ID I removed the container and checked using docker ps -a

```
[root@localhost ~]# docker rm c0190e442eed
c0190e442eed
[root@localhost ~]# docker ps -a
CONTAINER ID   IMAGE         COMMAND       CREATED             STATUS                          PORTS     NAMES
74e57b4297c1   ubuntu        "/bin/bash"   About an hour ago   Exited (0) About an hour ago              quizzical_goldwasser
f270ceca67c3   ubuntu        "/bin/bash"   About an hour ago   Exited (0) About an hour ago              goofy_blackburn
8c4a7a172427   ubuntu        "bash"        About an hour ago   Exited (0) About an hour ago              agitated_volhard
6c350e6681d5   ubuntu        "bash"        About an hour ago   Exited (127) About an hour ago            practical_kepler
7047b0cc04b4   hello-world   "/hello"      About an hour ago   Exited (0) About an hour ago              eloquent_wright
[root@localhost ~]#
mouse pointer inside or press Ctrl+G.
```

▪ Remove the image

```
[root@localhost ~]# docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
ubuntu        latest    602eb6fb314b   5 days ago     78.1MB
hello-world   latest    74cc54e27dc4   2 months ago   10.1kB
[root@localhost ~]# docker rmi 74cc54e27dc4
Untagged: hello-world:latest
Untagged: hello-world@sha256:424f1f86cdf501deb591ace8d14d2f40272617b51b374915a87a2886b2025ece
Deleted: sha256:74cc54e27dc41bb10dc4b2226072d469509f2f22f1a3ce74f4a59661a1d44602
Deleted: sha256:63a41026379f4391a306242eb0b9f26dc3550d863b7fdbb97d899f6eb89efe72
[root@localhost ~]# docker images
REPOSITORY   TAG       IMAGE ID       CREATED      SIZE
ubuntu       latest    602eb6fb314b   5 days ago   78.1MB
[root@localhost ~]#
```

---

2. Task2:

▪ Run container centos or ubuntu in an interactive mode

```
[root@localhost ~]# docker run -it ubuntu
root@3ff5a8f2b1b3:/#
```

▪ Run the following command in the container “echo docker ”

```
root@3ff5a8f2b1b3:/# echo docker
docker
root@3ff5a8f2b1b3:/#
```

▪ Open a bash shell in the container and touch a file named hello-docker

```
root@3ff5a8f2b1b3:/# touch hello-docker
root@3ff5a8f2b1b3:/# ls
bin  boot  dev  etc  hello-docker  home  lib  lib64  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
```

▪ Stop the container and remove it. Write your comment about the file hello-docker

```
root@3ff5a8f2b1b3:/# exit
exit
[root@localhost ~]# docker ps -a
CONTAINER ID   IMAGE     COMMAND       CREATED         STATUS                       PORTS     NAMES
3ff5a8f2b1b3   ubuntu    "/bin/bash"   2 minutes ago   Exited (0) 7 seconds ago               gallant_roentgen
e6c5a2e4f557   ubuntu    "/bin/bash"   2 minutes ago   Exited (0) 2 minutes ago               vibrant_chebyshev
1cbbab4aad7b   ubuntu    "/bin/bash"   5 hours ago     Exited (255) 3 minutes ago             dazzling_banach
74e57b4297c1   ubuntu    "/bin/bash"   6 hours ago     Exited (0) 6 hours ago                 quizzical_goldwasser
f270ceca67c3   ubuntu    "/bin/bash"   6 hours ago     Exited (0) 6 hours ago                 goofy_blackburn
8c4a7a172427   ubuntu    "bash"        6 hours ago     Exited (0) 6 hours ago                 agitated_volhard
6c350e6681d5   ubuntu    "bash"        6 hours ago     Exited (127) 6 hours ago               practical_kepler
```

```
[root@localhost ~]# docker stop 3ff5a8f2b1b3
3ff5a8f2b1b3
[root@localhost ~]# do
do                          dockerd-rootless-setuptool.sh  domainname                  dosfsck
docker                      dockerd-rootless.sh            done                        dosfslabel
dockerd                     docker-proxy                   dos2unix
[root@localhost ~]# docker rm 3ff5a8f2b1b3
3ff5a8f2b1b3
[root@localhost ~]#
```

ouse pointer inside or press Ctrl+G.

- Remove all stopped containers

-By running this command: (docker container prune -f) which removes **all containers with status Exited**

```
[root@localhost ~]# docker container prune -f
Deleted Containers:
e6c5a2e4f557984e58e38b6b27b31ba19c1e7d761765369870df42ea407afa2e
1cbbab4aad7b1fce6c83c5c78b9ccc1fbcea2fe67b191605eab0351962a7a502
74e57b4297c17a9507d0ccf842d7e3d17f310d3d435578f79aa705311f18dbb8
f270ceca67c347ff95d9b474fd667868a4876b06fd574309845d0eda2f7ddd6d
8c4a7a172427c4df659a4b093e1cf21ca6c1b60115c26373ecde7ba1da9f4be2
6c350e6681d523d65829db232e5f7ca1a65e5e689a9d3838eae9b03c29191ae7

Total reclaimed space: 40B
[root@localhost ~]#
```

ouse pointer inside or press Ctrl+G.

3. Task3:

▪ Run a container httpd with name apache and

attach a volume 2 volumes to the container

```
[root@localhost home]# mkdir Task3
[root@localhost home]# cd Task3/
[root@localhost Task3]# mkdir html
[root@localhost Task3]# mv /home/index.html /home/Task3/html
[root@localhost Task3]# cd html/
[root@localhost html]# ls
index.html
```

- Volume1 for containing static html file

Using nano command to edit the html file:

```
  GNU nano 5.6.1                                                  index.html
<h1>Hello from Docker Apache!</h1>
```

- Volume2 for containing httpd configuration

And now after after creating the html volume, let's run the apatche container using the default Apache config:

```
[root@localhost ~]# docker run -d --name apache-1 -v /home/Task3/html:/usr/local/apache2/htdocs  httpd
4580b8a7c4bba89aef46314b5c95147fb214a58f2888562814b1f4c6b31929d3
[root@localhost ~]# docker ps -a
CONTAINER ID   IMAGE     COMMAND              CREATED          STATUS         PORTS     NAMES
4580b8a7c4bb   httpd     "httpd-foreground"   10 seconds ago   Up 9 seconds   80/tcp    apache-1
[root@localhost ~]#
```

▪ Remove the container

```
[root@localhost html]# docker stop apache-1
apache-1
[root@localhost html]# docker rm apache-1
apache-1
[root@localhost html]# docker ps -a
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
[root@localhost html]#
```

▪ Run a new container with the following:

- Attach the 2 volumes that was attached to the previous container
- Map port 80 to port 9898 on you host machine

```
[root@localhost ~]# docker run -d --name apache-2 -v /home/Task3/html:/usr/local/apache2/htdocs -p 9898:80  httpd
c274a3069c5855fa438727a0b39da7186fad6619c9095247ea6045bb34e2f33d
[root@localhost ~]# docker ps
CONTAINER ID   IMAGE     COMMAND              CREATED         STATUS         PORTS                                    NAMES
c274a3069c58   httpd     "httpd-foreground"   4 seconds ago   Up 3 seconds   0.0.0.0:9898->80/tcp, [::]:9898->80/tcp   apache-2
[root@localhost ~]#
```

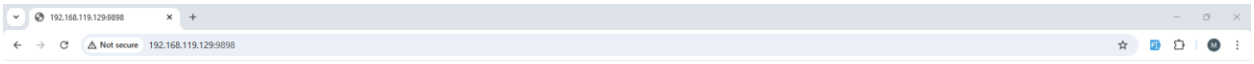
- Access the html files from your browser

```
[root@localhost config]# ip a
1: 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
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: ens160: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether 00:0c:29:6a:8d:10 brd ff:ff:ff:ff:ff:ff
    altname enp3s0
    inet 192.168.119.129/24 brd 192.168.119.255 scope global dynamic noprefixroute ens160
       valid_lft 1305sec preferred_lft 1305sec
    inet6 fe80::20c:29ff:fe6a:8d10/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
3: docker0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether e2:4d:7a:05:a8:c8 brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
       valid_lft forever preferred_lft forever
    inet6 fe80::e04d:7aff:fe05:a8c8/64 scope link
       valid_lft forever preferred_lft forever
8: vethca569eb@if2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master docker0 state UP group default
    link/ether 1e:89:00:2f:7a:1f brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet6 fe80::1c89:ff:fe2f:7a1f/64 scope link
       valid_lft forever preferred_lft forever
[root@localhost config]#
```

nouse pointer inside or press Ctrl+G.

The IP address of the virtual machine I'm using is 192.168.119.129,

And now, let's access my Apache container from my **host machine browser:**



Hello from Docker Apache!