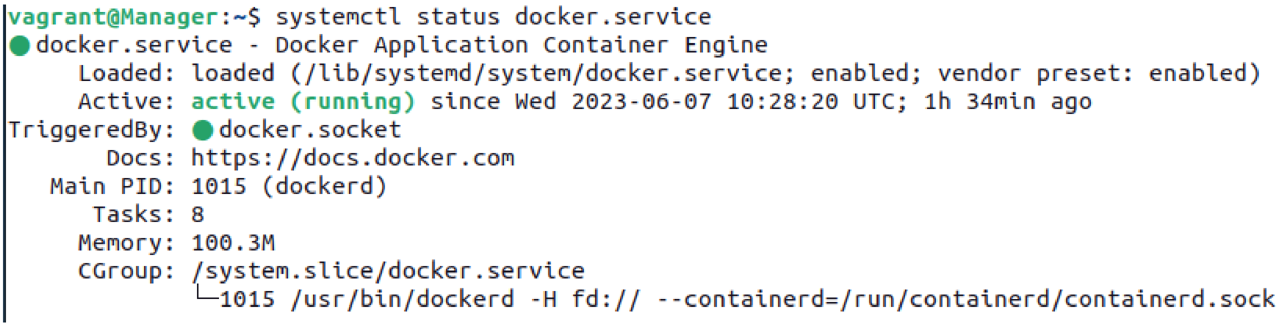
Lab 3 – Docker Engine

Manage containers, images and volumes

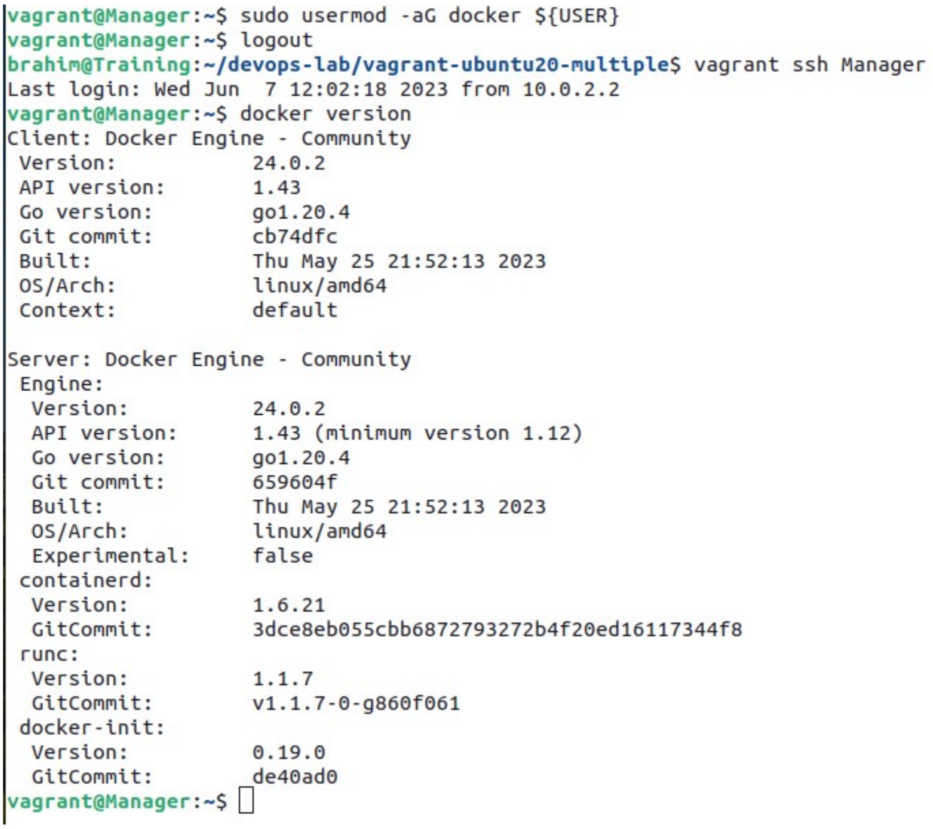
Docker status informations

1. Verify docker is installed and works.



```
vagrant@Manager:~$ systemctl status docker.service
● docker.service - Docker Application Container Engine
     Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: enabled)
     Active: active (running) since Wed 2023-06-07 10:28:20 UTC; 1h 34min ago
TriggeredBy: ● docker.socket
       Docs: https://docs.docker.com
   Main PID: 1015 (dockerd)
      Tasks: 8
     Memory: 100.3M
     CGroup: /system.slice/docker.service
             └─1015 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock
```

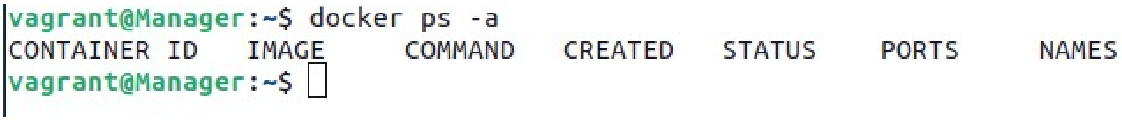
2. Display information about the docker version:
   - What is the docker client version ?
   - What is the docker server version ?



```
vagrant@Manager:~$ sudo usermod -aG docker ${USER}
vagrant@Manager:~$ logout
brahim@Training:~/devops-lab/vagrant-ubuntu20-multiple$ vagrant ssh Manager
Last login: Wed Jun  7 12:02:18 2023 from 10.0.2.2
vagrant@Manager:~$ docker version
Client: Docker Engine - Community
 Version:           24.0.2
 API version:       1.43
 Go version:        go1.20.4
 Git commit:        cb74dfc
 Built:             Thu May 25 21:52:13 2023
 OS/Arch:           linux/amd64
 Context:           default

Server: Docker Engine - Community
 Engine:
  Version:          24.0.2
  API version:      1.43 (minimum version 1.12)
  Go version:       go1.20.4
  Git commit:       659604f
  Built:            Thu May 25 21:52:13 2023
  OS/Arch:          linux/amd64
  Experimental:     false
 containerd:
  Version:          1.6.21
  GitCommit:        3dce8eb055cbb6872793272b4f20ed16117344f8
 runc:
  Version:          1.1.7
  GitCommit:        v1.1.7-0-g860f061
 docker-init:
  Version:          0.19.0
  GitCommit:        de40ad0
vagrant@Manager:~$ 
```

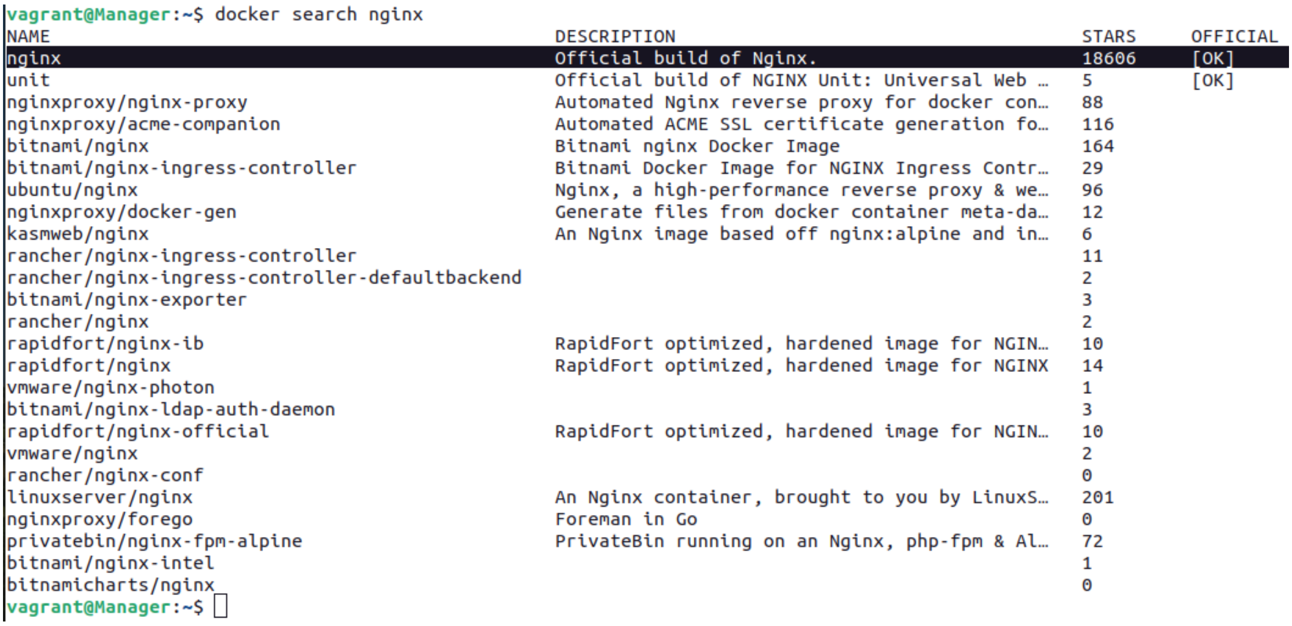
Manage containers

3. List all docker containers



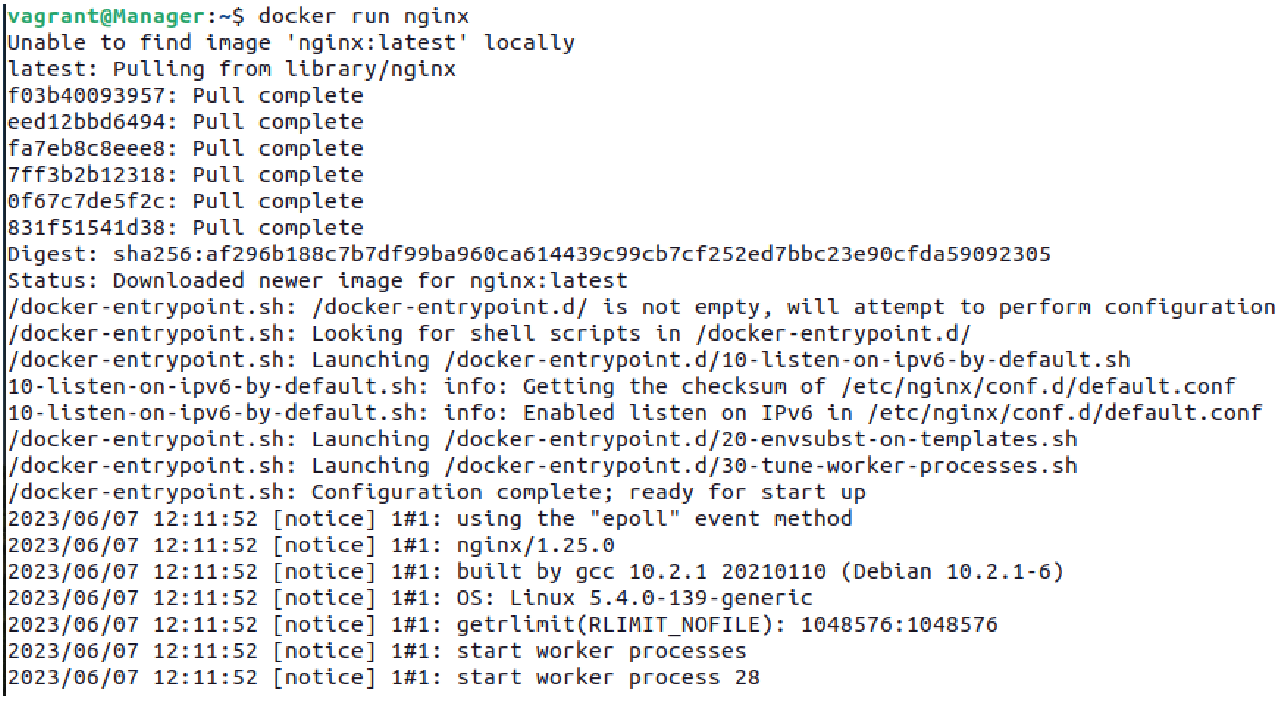
```
vagrant@Manager:~$ docker ps -a
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
vagrant@Manager:~$ 
```

4. Find existing *nginx* images at the registry hub.docker.com/ .



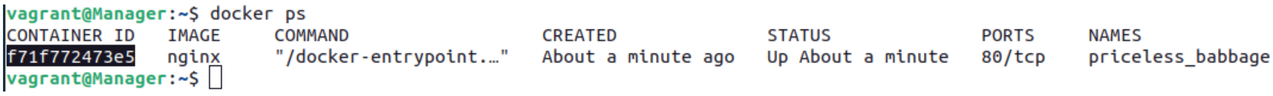
```
vagrant@Manager:~$ docker search nginx
NAME                                              DESCRIPTION                                     STARS     OFFICIAL
nginx                                             Official build of Nginx.                        18606     [OK]
unit                                              Official build of NGINX Unit: Universal Web …   5         [OK]
nginxproxy/nginx-proxy                            Automated Nginx reverse proxy for docker con…   88
nginxproxy/acme-companion                         Automated ACME SSL certificate generation fo…   116
bitnami/nginx                                     Bitnami nginx Docker Image                      164
bitnami/nginx-ingress-controller                  Bitnami Docker Image for NGINX Ingress Contr…   29
ubuntu/nginx                                      Nginx, a high-performance reverse proxy & we…   96
nginxproxy/docker-gen                             Generate files from docker container meta-da…   12
kasmweb/nginx                                     An Nginx image based off nginx:alpine and in…   6
rancher/nginx-ingress-controller                                                                  11
rancher/nginx-ingress-controller-defaultbackend                                                   2
bitnami/nginx-exporter                                                                            3
rancher/nginx                                                                                     2
rapidfort/nginx-ib                                RapidFort optimized, hardened image for NGIN…   10
rapidfort/nginx                                   RapidFort optimized, hardened image for NGINX   14
vmware/nginx-photon                                                                               1
bitnami/nginx-ldap-auth-daemon                                                                    3
rapidfort/nginx-official                          RapidFort optimized, hardened image for NGIN…   10
vmware/nginx                                                                                      2
rancher/nginx-conf                                                                                0
linuxserver/nginx                                 An Nginx container, brought to you by LinuxS…   201
nginxproxy/forego                                 Foreman in Go                                   0
privatebin/nginx-fpm-alpine                       PrivateBin running on an Nginx, php-fpm & Al…   72
bitnami/nginx-intel                                                                               1
bitnamicharts/nginx                                                                               0
vagrant@Manager:~$ 
```

5. Start a container based on the latest *nginx* Docker Image



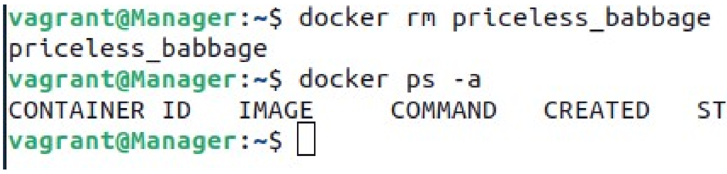
```
vagrant@Manager:~$ docker run nginx
Unable to find image 'nginx:latest' locally
latest: Pulling from library/nginx
f03b40093957: Pull complete
eed12bbd6494: Pull complete
fa7eb8c8eee8: Pull complete
7ff3b2b12318: Pull complete
0f67c7de5f2c: Pull complete
831f51541d38: Pull complete
Digest: sha256:af296b188c7b7df99ba960ca614439c99cb7cf252ed7bbc23e90cfda59092305
Status: Downloaded newer image for nginx:latest
/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
2023/06/07 12:11:52 [notice] 1#1: using the "epoll" event method
2023/06/07 12:11:52 [notice] 1#1: nginx/1.25.0
2023/06/07 12:11:52 [notice] 1#1: built by gcc 10.2.1 20210110 (Debian 10.2.1-6)
2023/06/07 12:11:52 [notice] 1#1: OS: Linux 5.4.0-139-generic
2023/06/07 12:11:52 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2023/06/07 12:11:52 [notice] 1#1: start worker processes
2023/06/07 12:11:52 [notice] 1#1: start worker process 28
```

- What is the id, name and status of the created container ?



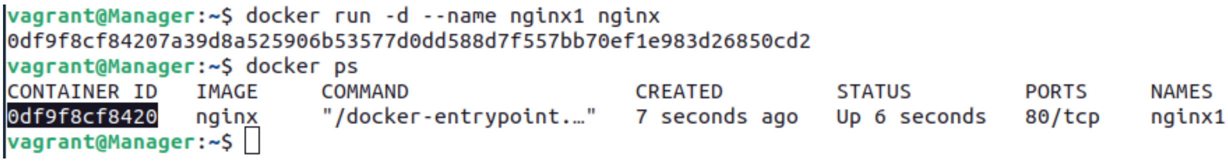
```
vagrant@Manager:~$ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED              STATUS              PORTS     NAMES
f71f772473e5   nginx     "/docker-entrypoint.…"   About a minute ago   Up About a minute   80/tcp    priceless_babbage
vagrant@Manager:~$ 
```

- Remove the container



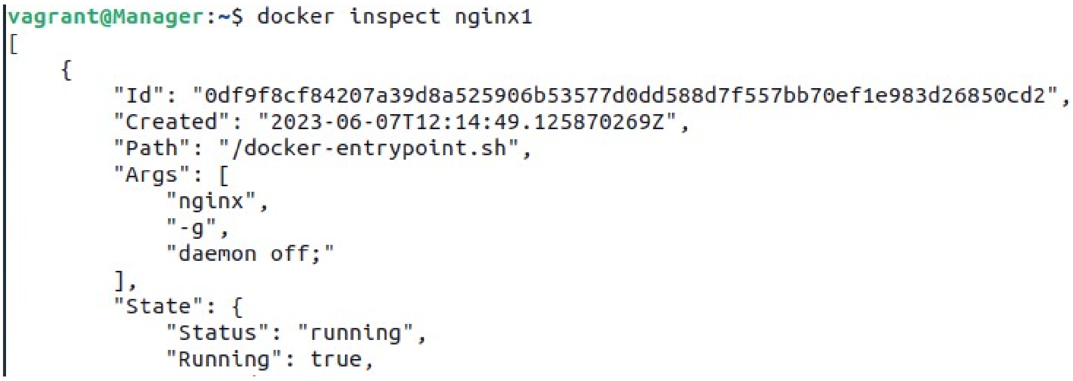
```
vagrant@Manager:~$ docker rm priceless_babbage
priceless_babbage
vagrant@Manager:~$ docker ps -a
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
vagrant@Manager:~$ 
```

6. Start a container based on the latest nginx Docker Image in the background, with *nginx1* name. What is the id, name and status of the created container ?



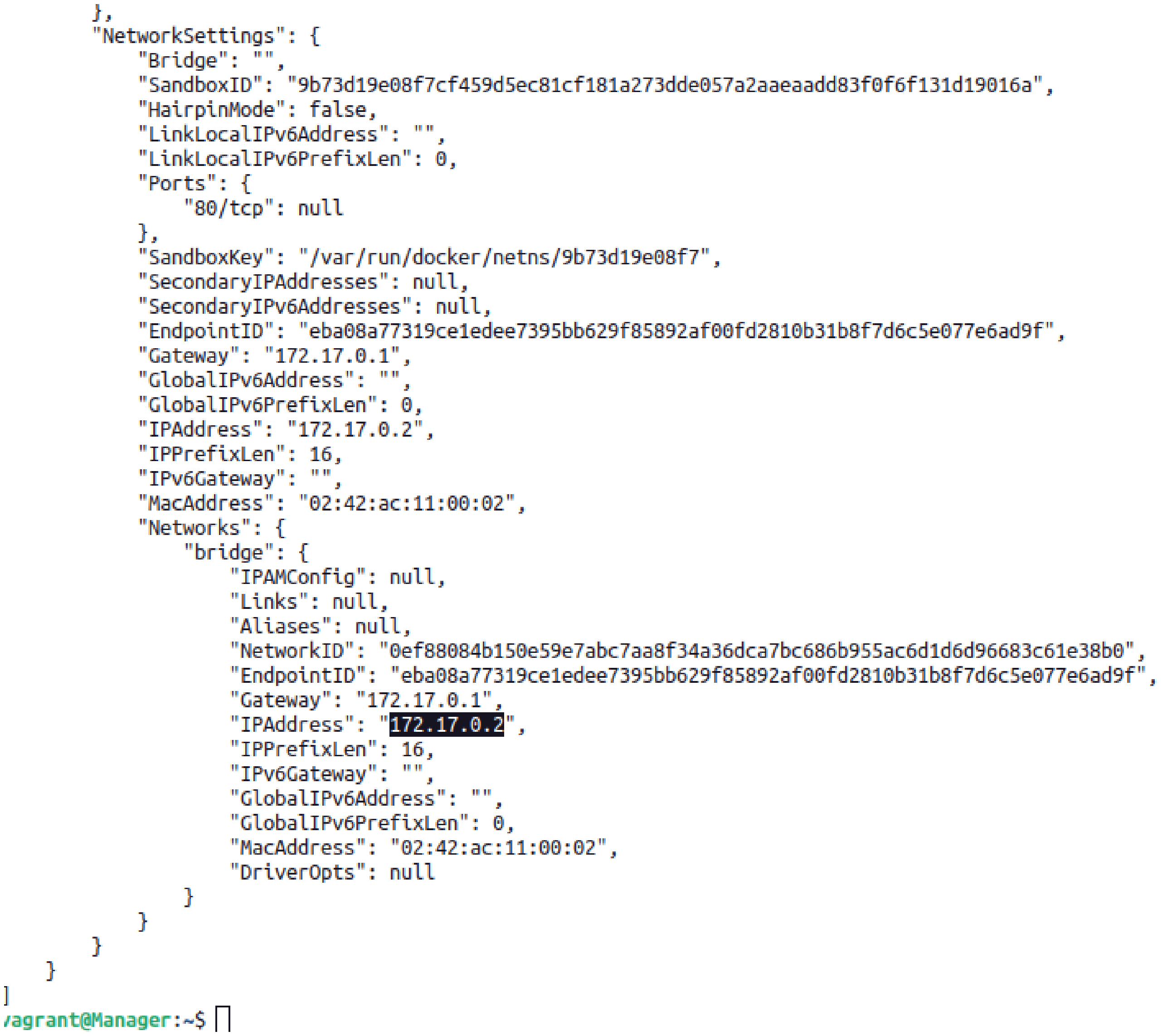
```
vagrant@Manager:~$ docker run -d --name nginx1 nginx
0df9f8cf84207a39d8a525906b53577d0dd588d7f557bb70ef1e983d26850cd2
vagrant@Manager:~$ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED         STATUS         PORTS     NAMES
0df9f8cf8420   nginx     "/docker-entrypoint.…"   7 seconds ago   Up 6 seconds   80/tcp    nginx1
vagrant@Manager:~$ 
```

7. Display more informations about nginx1 container.



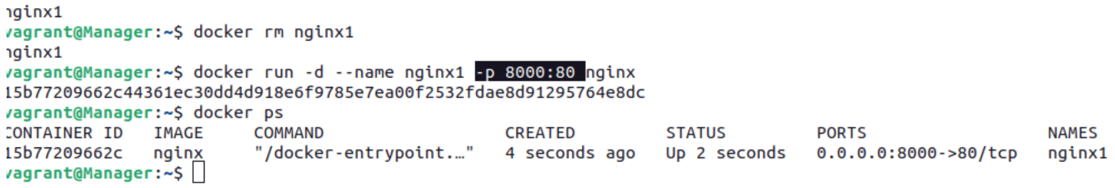
```
vagrant@Manager:~$ docker inspect nginx1
[
    {
        "Id": "0df9f8cf84207a39d8a525906b53577d0dd588d7f557bb70ef1e983d26850cd2",
        "Created": "2023-06-07T12:14:49.125870269Z",
        "Path": "/docker-entrypoint.sh",
        "Args": [
            "nginx",
            "-g",
            "daemon off;"
        ],
        "State": {
            "Status": "running",
            "Running": true,
```

- What is the ip address, netmask, gateway, MAC address and open TCP/ports of nginx1 container ?

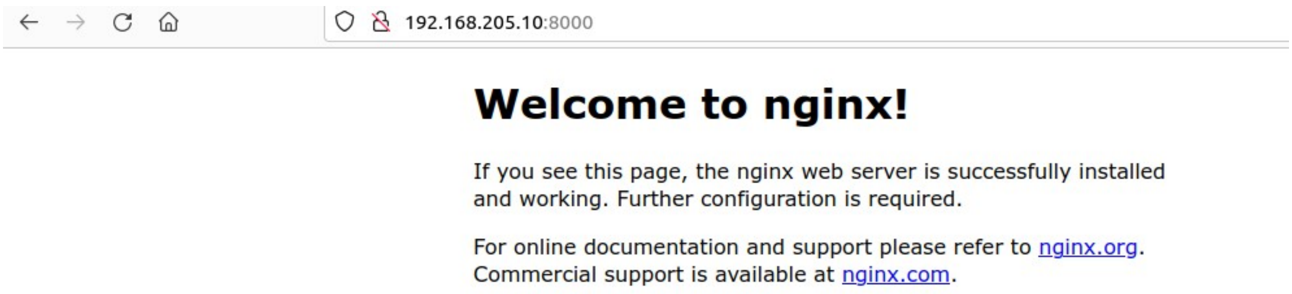


```
        },
        "NetworkSettings": {
            "Bridge": "",
            "SandboxID": "9b73d19e08f7cf459d5ec81cf181a273dde057a2aaeaadd83f0f6f131d19016a",
            "HairpinMode": false,
            "LinkLocalIPv6Address": "",
            "LinkLocalIPv6PrefixLen": 0,
            "Ports": {
                "80/tcp": null
            },
            "SandboxKey": "/var/run/docker/netns/9b73d19e08f7",
            "SecondaryIPAddresses": null,
            "SecondaryIPv6Addresses": null,
            "EndpointID": "eba08a77319ce1edee7395bb629f85892af00fd2810b31b8f7d6c5e077e6ad9f",
            "Gateway": "172.17.0.1",
            "GlobalIPv6Address": "",
            "GlobalIPv6PrefixLen": 0,
            "IPAddress": "172.17.0.2",
            "IPPrefixLen": 16,
            "IPv6Gateway": "",
            "MacAddress": "02:42:ac:11:00:02",
            "Networks": {
                "bridge": {
                    "IPAMConfig": null,
                    "Links": null,
                    "Aliases": null,
                    "NetworkID": "0ef88084b150e59e7abc7aa8f34a36dca7bc686b955ac6d1d6d96683c61e38b0",
                    "EndpointID": "eba08a77319ce1edee7395bb629f85892af00fd2810b31b8f7d6c5e077e6ad9f",
                    "Gateway": "172.17.0.1",
                    "IPAddress": "172.17.0.2",
                    "IPPrefixLen": 16,
                    "IPv6Gateway": "",
                    "GlobalIPv6Address": "",
                    "GlobalIPv6PrefixLen": 0,
                    "MacAddress": "02:42:ac:11:00:02",
                    "DriverOpts": null
                }
            }
        }
    }
]
/agrant@Manager:~$ 
```

8. Test the nginx web page on the browser.



```
nginx1
/agrant@Manager:~$ docker rm nginx1
nginx1
/agrant@Manager:~$ docker run -d --name nginx1 -p 8000:80 nginx
15b77209662c44361ec30dd4d918e6f9785e7ea00f2532fdae8d91295764e8dc
/agrant@Manager:~$ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED         STATUS         PORTS                  NAMES
15b77209662c   nginx     "/docker-entrypoint.…"   4 seconds ago   Up 2 seconds   0.0.0.0:8000->80/tcp   nginx1
/agrant@Manager:~$ 
```



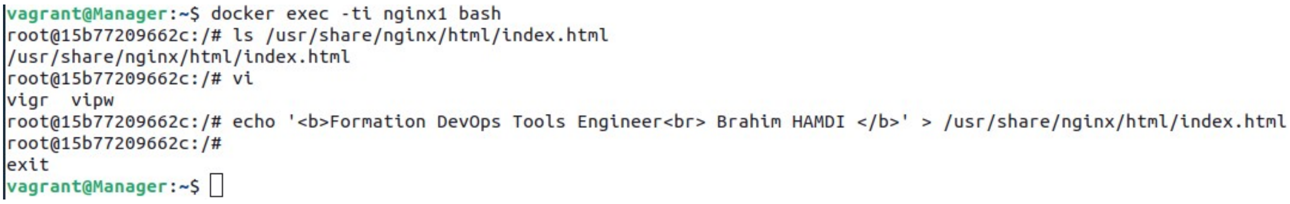
192.168.205.10:8000

Welcome to nginx!

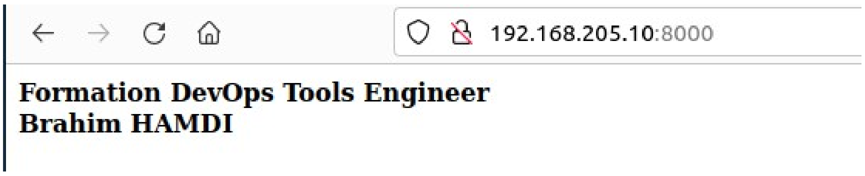
If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Connect to nginx1 container and modify the « index.html » to display this :
Formation DevOps Tools Engineer
[Your Name]



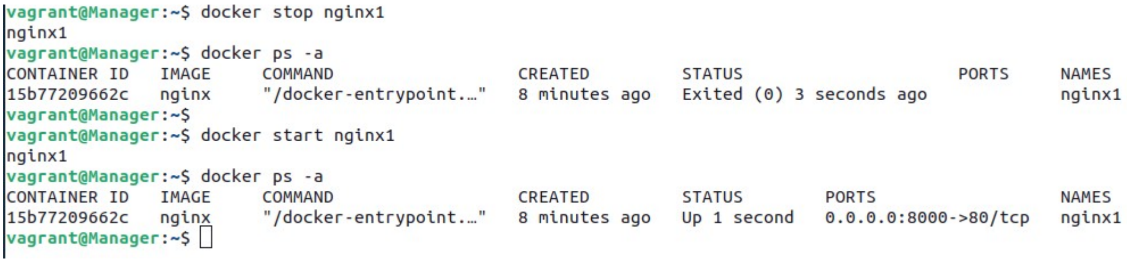
```
vagrant@Manager:~$ docker exec -ti nginx1 bash
root@15b77209662c:/# ls /usr/share/nginx/html/index.html
/usr/share/nginx/html/index.html
root@15b77209662c:/# vi
vigr  vipw
root@15b77209662c:/# echo '<b>Formation DevOps Tools Engineer<br> Brahim HAMDI </b>' > /usr/share/nginx/html/index.html
root@15b77209662c:/#
exit
vagrant@Manager:~$ 
```



192.168.205.10:8000

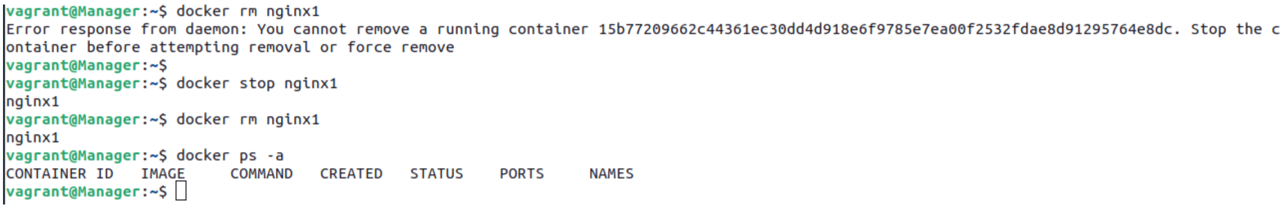
Formation DevOps Tools Engineer
Brahim HAMDI

9. Stop nginx1 container and then start it.



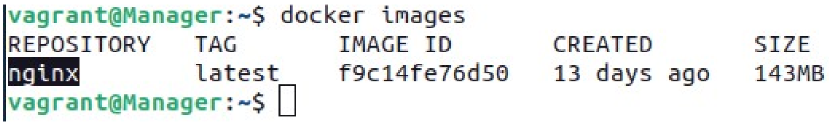
```
vagrant@Manager:~$ docker stop nginx1
nginx1
vagrant@Manager:~$ docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED         STATUS                     PORTS                  NAMES
15b77209662c   nginx     "/docker-entrypoint.…"   8 minutes ago   Exited (0) 3 seconds ago                          nginx1
vagrant@Manager:~$
vagrant@Manager:~$ docker start nginx1
nginx1
vagrant@Manager:~$ docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED         STATUS         PORTS                  NAMES
15b77209662c   nginx     "/docker-entrypoint.…"   8 minutes ago   Up 1 second    0.0.0.0:8000->80/tcp   nginx1
vagrant@Manager:~$ 
```

10. Remove nginx1 container.



```
vagrant@Manager:~$ docker rm nginx1
Error response from daemon: You cannot remove a running container 15b77209662c44361ec30dd4d918e6f9785e7ea00f2532fdae8d91295764e8dc. Stop the c
ontainer before attempting removal or force remove
vagrant@Manager:~$
vagrant@Manager:~$ docker stop nginx1
nginx1
vagrant@Manager:~$ docker rm nginx1
nginx1
vagrant@Manager:~$ docker ps -a
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
vagrant@Manager:~$ 
```

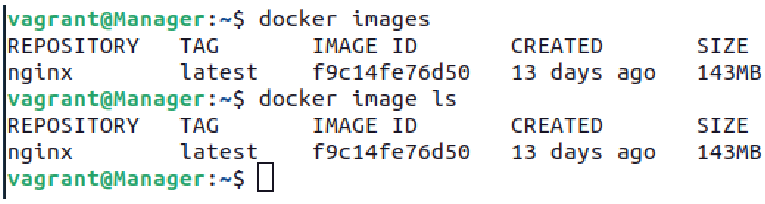
- Is nginx image removed too ?



```
vagrant@Manager:~$ docker images
REPOSITORY   TAG       IMAGE ID       CREATED        SIZE
nginx        latest    f9c14fe76d50   13 days ago    143MB
vagrant@Manager:~$ 
```

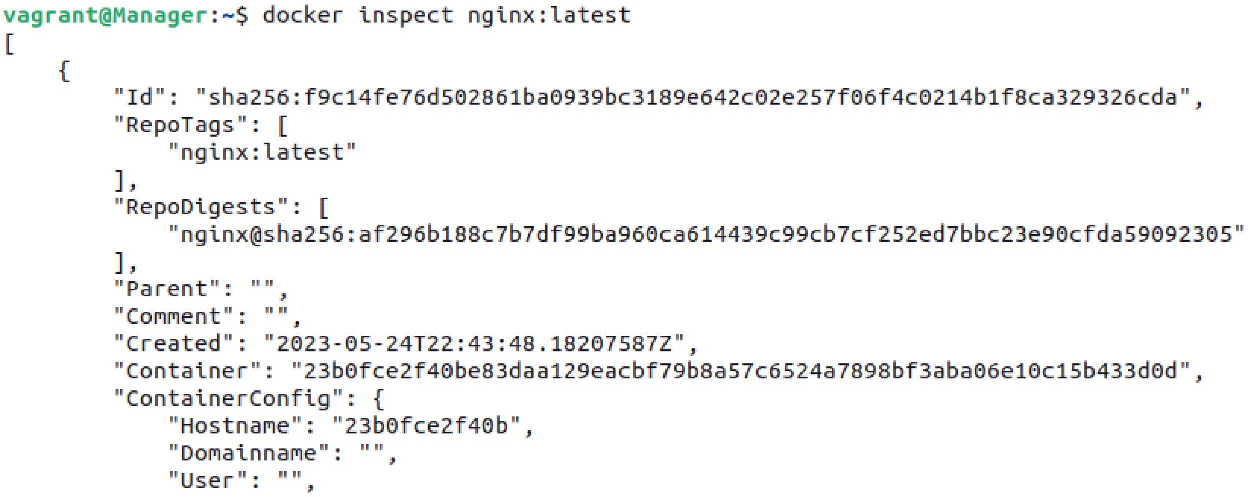
Manage images

11. List all local docker images



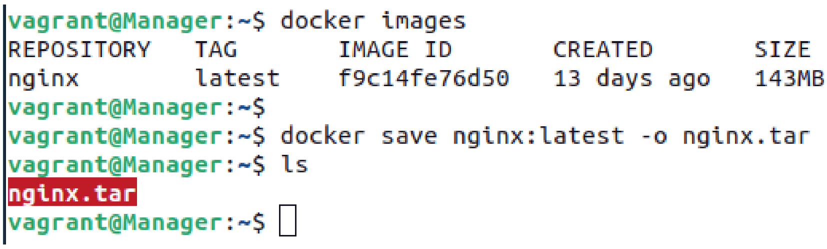
```
vagrant@Manager:~$ docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
nginx         latest    f9c14fe76d50   13 days ago    143MB
vagrant@Manager:~$ docker image ls
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
nginx         latest    f9c14fe76d50   13 days ago    143MB
vagrant@Manager:~$ 
```

12. Display detailed informations about the nginx image. What is the hash, version and creation date of the nginx image ?



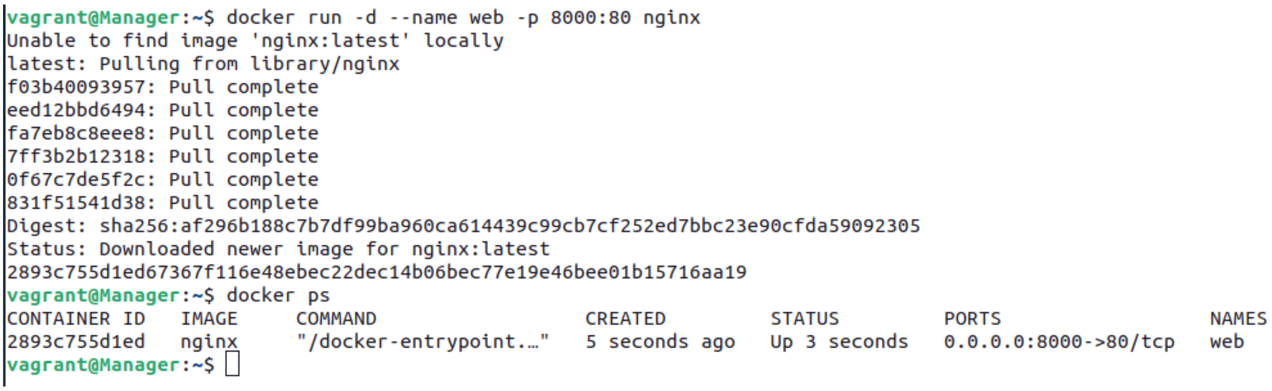
```
vagrant@Manager:~$ docker inspect nginx:latest
[
    {
        "Id": "sha256:f9c14fe76d502861ba0939bc3189e642c02e257f06f4c0214b1f8ca329326cda",
        "RepoTags": [
            "nginx:latest"
        ],
        "RepoDigests": [
            "nginx@sha256:af296b188c7b7df99ba960ca614439c99cb7cf252ed7bbc23e90cfda59092305"
        ],
        "Parent": "",
        "Comment": "",
        "Created": "2023-05-24T22:43:48.18207587Z",
        "Container": "23b0fce2f40be83daa129eacbf79b8a57c6524a7898bf3aba06e10c15b433d0d",
        "ContainerConfig": {
            "Hostname": "23b0fce2f40b",
            "Domainname": "",
            "User": "",
```

13. Save nginx image to *nginx.tar* file



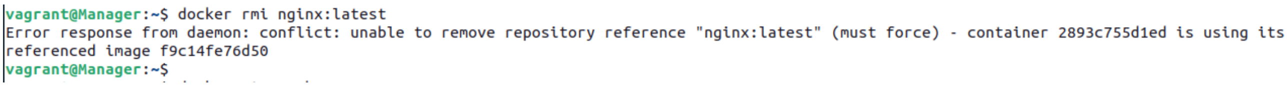
```
vagrant@Manager:~$ docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
nginx         latest    f9c14fe76d50   13 days ago    143MB
vagrant@Manager:~$
vagrant@Manager:~$ docker save nginx:latest -o nginx.tar
vagrant@Manager:~$ ls
nginx.tar
vagrant@Manager:~$ 
```

14. Run a web container with nginx image



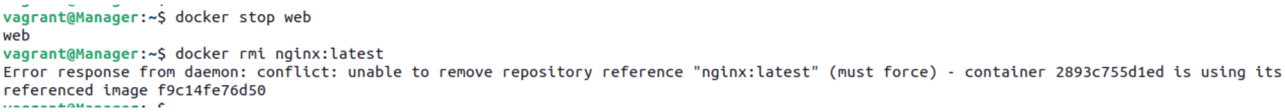
```
vagrant@Manager:~$ docker run -d --name web -p 8000:80 nginx
Unable to find image 'nginx:latest' locally
latest: Pulling from library/nginx
f03b40093957: Pull complete
eed12bbd6494: Pull complete
fa7eb8c8eee8: Pull complete
7ff3b2b12318: Pull complete
0f67c7de5f2c: Pull complete
831f51541d38: Pull complete
Digest: sha256:af296b188c7b7df99ba960ca614439c99cb7cf252ed7bbc23e90cfda59092305
Status: Downloaded newer image for nginx:latest
2893c755d1ed67367f116e48ebec22dec14b06bec77e19e46bee01b15716aa19
vagrant@Manager:~$ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED         STATUS         PORTS                  NAMES
2893c755d1ed   nginx     "/docker-entrypoint.…"   5 seconds ago   Up 3 seconds   0.0.0.0:8000->80/tcp   web
vagrant@Manager:~$ 
```

15. Remove the nginx image. Is the nginx image removed ? Why ?



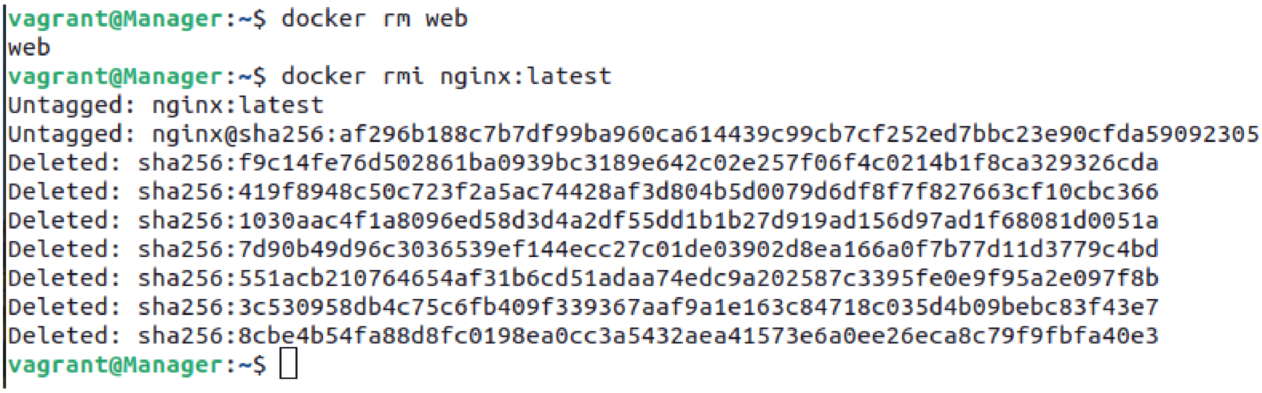
```
vagrant@Manager:~$ docker rmi nginx:latest
Error response from daemon: conflict: unable to remove repository reference "nginx:latest" (must force) - container 2893c755d1ed is using its
referenced image f9c14fe76d50
vagrant@Manager:~$
```

16. Stop the container, then remove the image.



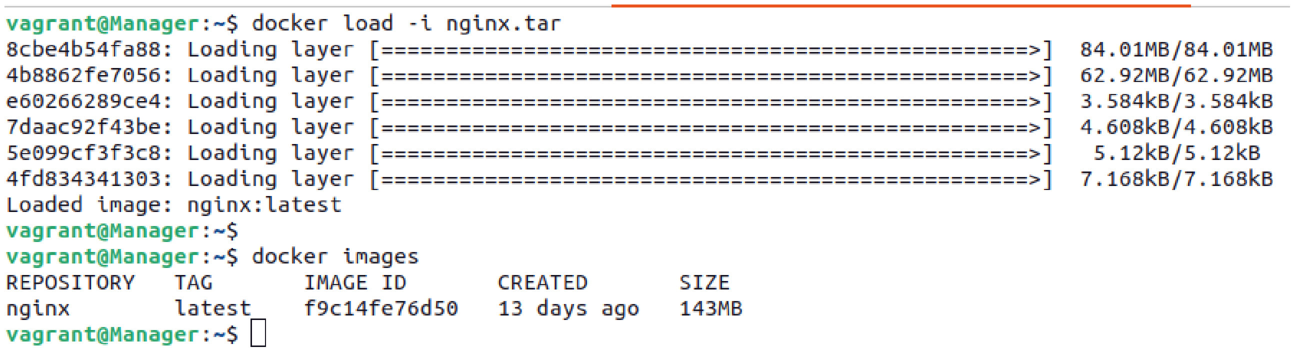
```
vagrant@Manager:~$ docker stop web
web
vagrant@Manager:~$ docker rmi nginx:latest
Error response from daemon: conflict: unable to remove repository reference "nginx:latest" (must force) - container 2893c755d1ed is using its
referenced image f9c14fe76d50
```

- Is the nginx image removed ? How to remove the image ?



```
vagrant@Manager:~$ docker rm web
web
vagrant@Manager:~$ docker rmi nginx:latest
Untagged: nginx:latest
Untagged: nginx@sha256:af296b188c7b7df99ba960ca614439c99cb7cf252ed7bbc23e90cfda59092305
Deleted: sha256:f9c14fe76d502861ba0939bc3189e642c02e257f06f4c0214b1f8ca329326cda
Deleted: sha256:419f8948c50c723f2a5ac74428af3d804b5d0079d6df8f7f827663cf10cbc366
Deleted: sha256:1030aac4f1a8096ed58d3d4a2df55dd1b1b27d919ad156d97ad1f68081d0051a
Deleted: sha256:7d90b49d96c3036539ef144ecc27c01de03902d8ea166a0f7b77d11d3779c4bd
Deleted: sha256:551acb210764654af31b6cd51adaa74edc9a202587c3395fe0e9f95a2e097f8b
Deleted: sha256:3c530958db4c75c6fb409f339367aaf9a1e163c84718c035d4b09bebc83f43e7
Deleted: sha256:8cbe4b54fa88d8fc0198ea0cc3a5432aea41573e6a0ee26eca8c79f9fbfa40e3
vagrant@Manager:~$ 
```

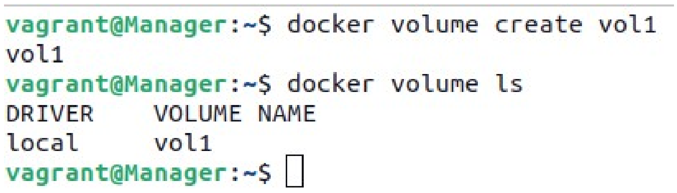
17. Load nginx image from nginx.tar archive



```
vagrant@Manager:~$ docker load -i nginx.tar
8cbe4b54fa88: Loading layer [==================================================>]  84.01MB/84.01MB
4b8862fe7056: Loading layer [==================================================>]  62.92MB/62.92MB
e60266289ce4: Loading layer [==================================================>]  3.584kB/3.584kB
7daac92f43be: Loading layer [==================================================>]  4.608kB/4.608kB
5e099cf3f3c8: Loading layer [==================================================>]   5.12kB/5.12kB
4fd834341303: Loading layer [==================================================>]  7.168kB/7.168kB
Loaded image: nginx:latest
vagrant@Manager:~$
vagrant@Manager:~$ docker images
REPOSITORY   TAG       IMAGE ID       CREATED       SIZE
nginx        latest    f9c14fe76d50   13 days ago   143MB
vagrant@Manager:~$
```

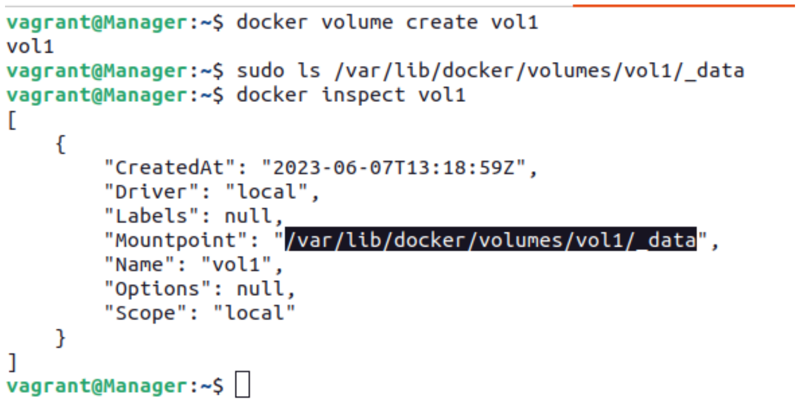
Manage volumes

18. Create volume vol1.



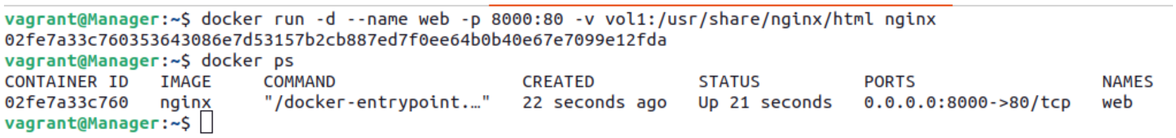
```
vagrant@Manager:~$ docker volume create vol1
vol1
vagrant@Manager:~$ docker volume ls
DRIVER    VOLUME NAME
local     vol1
vagrant@Manager:~$
```

What is the path of vol1 mount point ? Is it empty ?



```
vagrant@Manager:~$ docker volume create vol1
vol1
vagrant@Manager:~$ sudo ls /var/lib/docker/volumes/vol1/_data
vagrant@Manager:~$ docker inspect vol1
[
    {
        "CreatedAt": "2023-06-07T13:18:59Z",
        "Driver": "local",
        "Labels": null,
        "Mountpoint": "/var/lib/docker/volumes/vol1/_data",
        "Name": "vol1",
        "Options": null,
        "Scope": "local"
    }
]
vagrant@Manager:~$
```

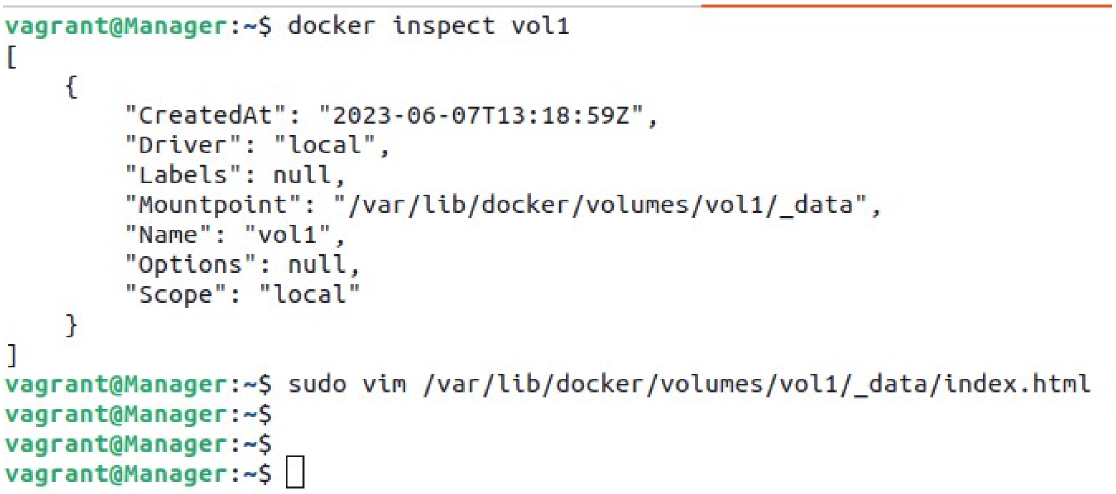
19. Run a container with nginx image, that attach volume vol1 to */usr/share/nginx/html* folder.



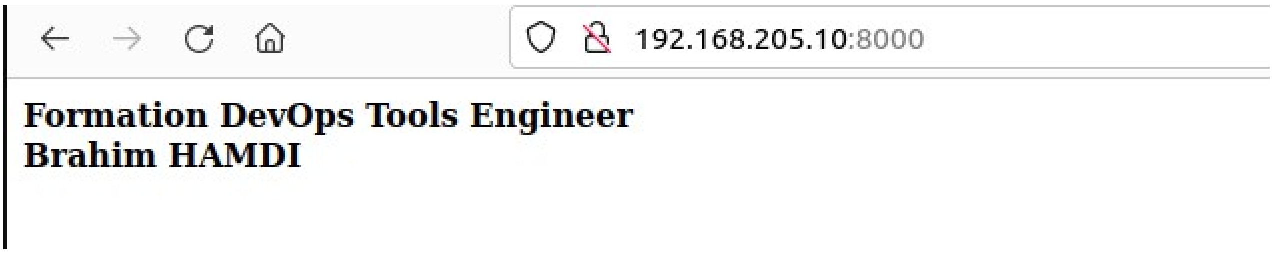
```
vagrant@Manager:~$ docker run -d --name web -p 8000:80 -v vol1:/usr/share/nginx/html nginx
02fe7a33c760353643086e7d53157b2cb887ed7f0ee64b0b40e67e7099e12fda
vagrant@Manager:~$ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS                  NAMES
02fe7a33c760   nginx     "/docker-entrypoint.…"   22 seconds ago   Up 21 seconds   0.0.0.0:8000->80/tcp   web
vagrant@Manager:~$
```

20. Using vol1, modify « index.html » file to display this :

Formation DevOps Tools Engineer
[Your Name]

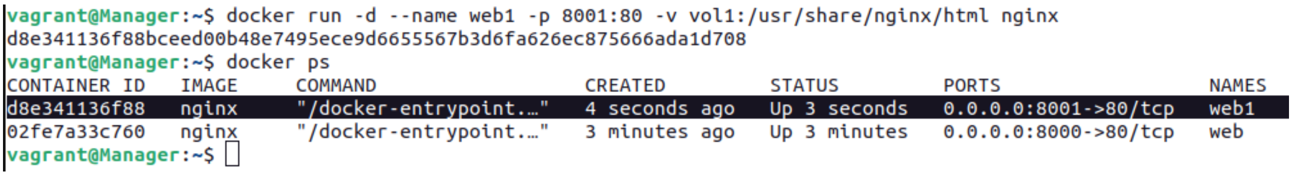


```
vagrant@Manager:~$ docker inspect vol1
[
    {
        "CreatedAt": "2023-06-07T13:18:59Z",
        "Driver": "local",
        "Labels": null,
        "Mountpoint": "/var/lib/docker/volumes/vol1/_data",
        "Name": "vol1",
        "Options": null,
        "Scope": "local"
    }
]
vagrant@Manager:~$ sudo vim /var/lib/docker/volumes/vol1/_data/index.html
vagrant@Manager:~$
vagrant@Manager:~$
vagrant@Manager:~$ 
```



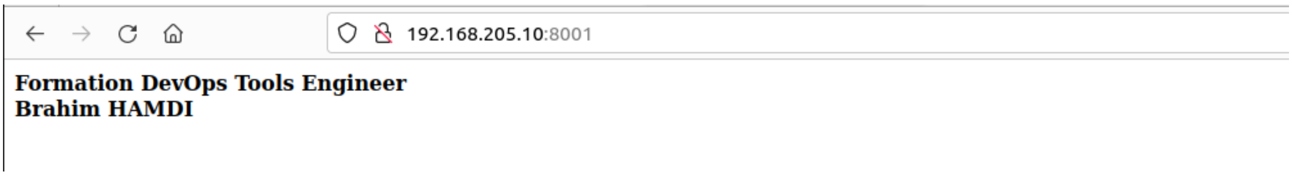
Formation DevOps Tools Engineer
Brahim HAMDI

21. Run another container with nginx image, that attach vol1.



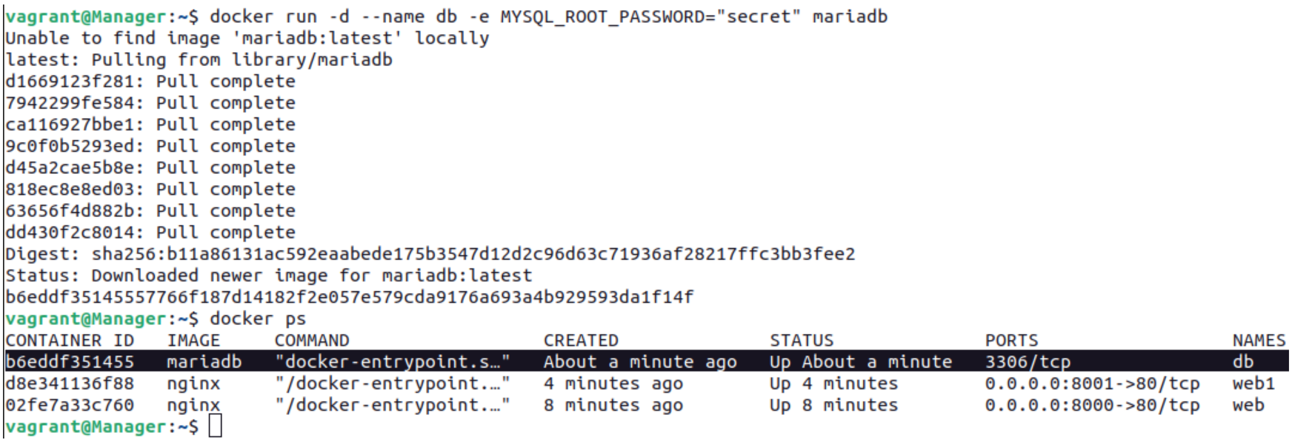
```
vagrant@Manager:~$ docker run -d --name web1 -p 8001:80 -v vol1:/usr/share/nginx/html nginx
d8e341136f88bceed00b48e7495ece9d6655567b3d6fa626ec875666ada1d708
vagrant@Manager:~$ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS                  NAMES
d8e341136f88   nginx     "/docker-entrypoint.…"   4 seconds ago    Up 3 seconds    0.0.0.0:8001->80/tcp   web1
02fe7a33c760   nginx     "/docker-entrypoint.…"   3 minutes ago    Up 3 minutes    0.0.0.0:8000->80/tcp   web
vagrant@Manager:~$ 
```

What it display on the web page ?



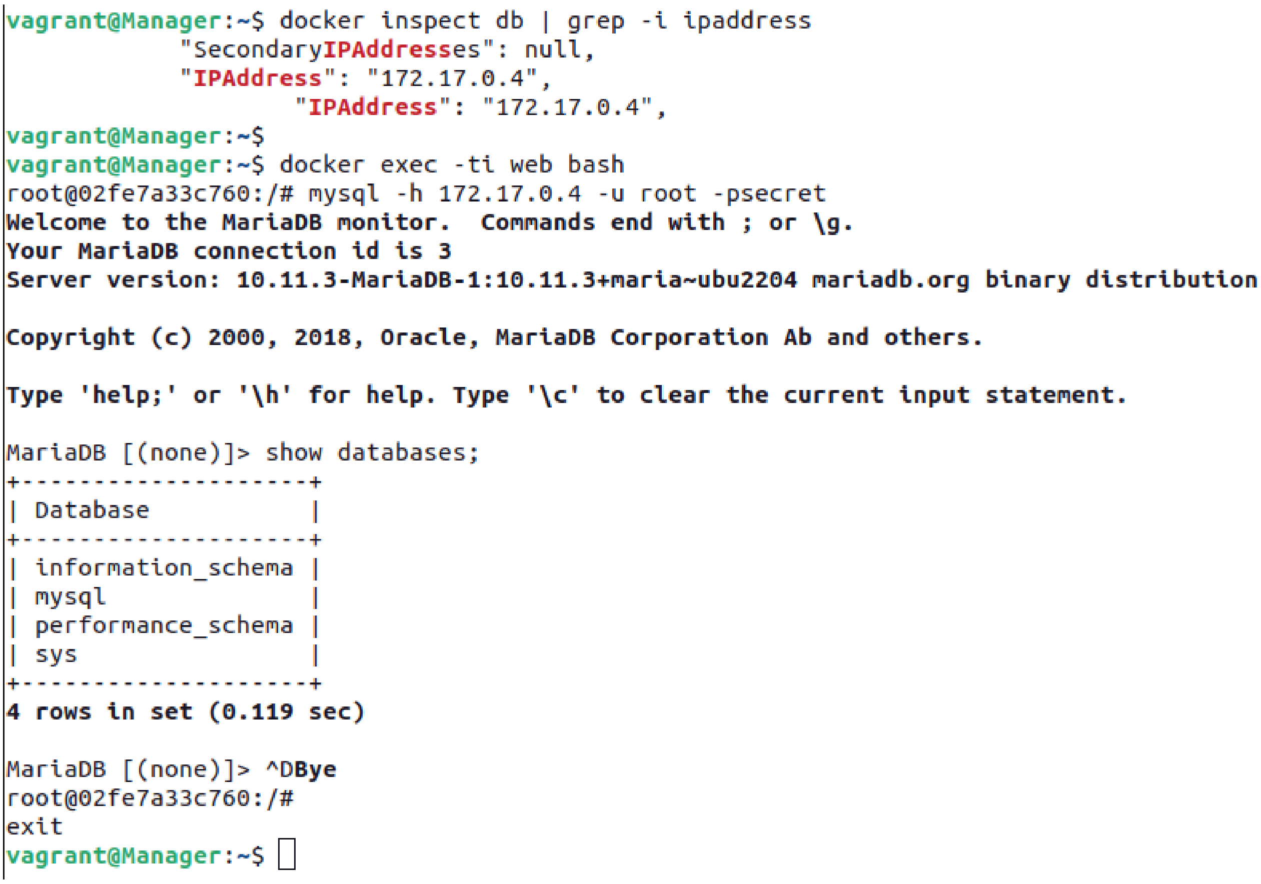
Formation DevOps Tools Engineer
Brahim HAMDI

22. Run a container with mariadb image.



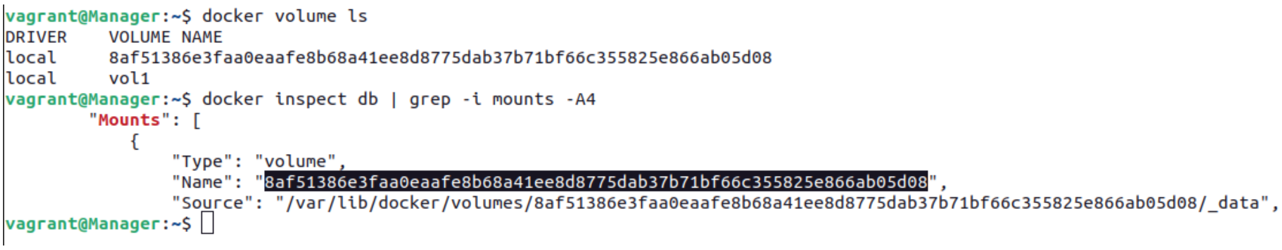
```
vagrant@Manager:~$ docker run -d --name db -e MYSQL_ROOT_PASSWORD="secret" mariadb
Unable to find image 'mariadb:latest' locally
latest: Pulling from library/mariadb
d1669123f281: Pull complete
7942299fe584: Pull complete
ca116927bbe1: Pull complete
9c0f0b5293ed: Pull complete
d45a2cae5b8e: Pull complete
818ec8e8ed03: Pull complete
63656f4d882b: Pull complete
dd430f2c8014: Pull complete
Digest: sha256:b11a86131ac592eaabede175b3547d12d2c96d63c71936af28217ffc3bb3fee2
Status: Downloaded newer image for mariadb:latest
b6eddf35145557766f187d14182f2e057e579cda9176a693a4b929593da1f14f
vagrant@Manager:~$ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED              STATUS              PORTS                  NAMES
b6eddf351455   mariadb   "docker-entrypoint.s…"   About a minute ago   Up About a minute   3306/tcp               db
d8e341136f88   nginx     "/docker-entrypoint.…"   4 minutes ago        Up 4 minutes        0.0.0.0:8001->80/tcp   web1
02fe7a33c760   nginx     "/docker-entrypoint.…"   8 minutes ago        Up 8 minutes        0.0.0.0:8000->80/tcp   web
vagrant@Manager:~$ 
```

Test sql connexion to the container.



```
vagrant@Manager:~$ docker inspect db | grep -i ipaddress
            "SecondaryIPAddresses": null,
            "IPAddress": "172.17.0.4",
                    "IPAddress": "172.17.0.4",
vagrant@Manager:~$
vagrant@Manager:~$ docker exec -ti web bash
root@02fe7a33c760:/# mysql -h 172.17.0.4 -u root -psecret
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 3
Server version: 10.11.3-MariaDB-1:10.11.3+maria~ubu2204 mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
4 rows in set (0.119 sec)

MariaDB [(none)]> ^DBye
root@02fe7a33c760:/#
exit
vagrant@Manager:~$ 
```

23. Is there any attached volume to the DB container ? If yes, where does it come from ?

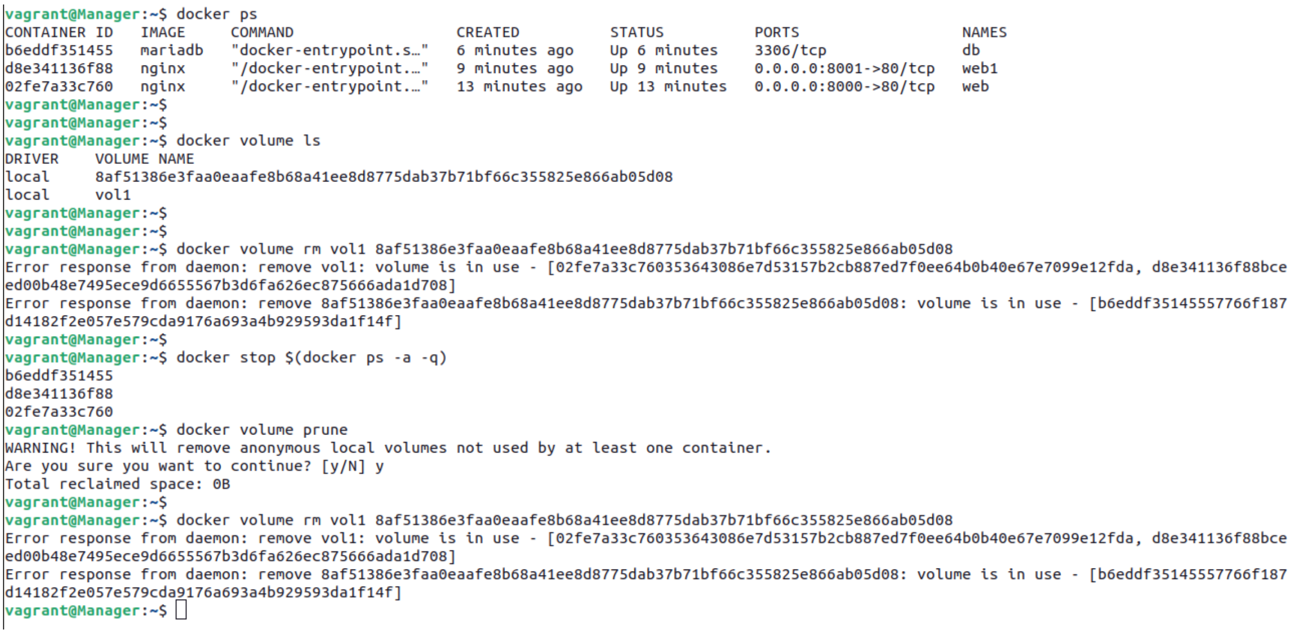


```
vagrant@Manager:~$ docker volume ls
DRIVER    VOLUME NAME
local     8af51386e3faa0eaafe8b68a41ee8d8775dab37b71bf66c355825e866ab05d08
local     vol1
vagrant@Manager:~$ docker inspect db | grep -i mounts -A4
        "Mounts": [
            {
                "Type": "volume",
                "Name": "8af51386e3faa0eaafe8b68a41ee8d8775dab37b71bf66c355825e866ab05d08",
                "Source": "/var/lib/docker/volumes/8af51386e3faa0eaafe8b68a41ee8d8775dab37b71bf66c355825e866ab05d08/_data",
vagrant@Manager:~$ 
```

24. Remove all volumes. Is there any problem ? Why ?

Stop all containers and then remove all volumes. Is there any problem ? Why ?

Remove all volumes



```
vagrant@Manager:~$ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS                  NAMES
b6eddf351455   mariadb   "docker-entrypoint.s…"   6 minutes ago    Up 6 minutes    3306/tcp               db
d8e341136f88   nginx     "/docker-entrypoint.…"   9 minutes ago    Up 9 minutes    0.0.0.0:8001->80/tcp   web1
02fe7a33c760   nginx     "/docker-entrypoint.…"   13 minutes ago   Up 13 minutes   0.0.0.0:8000->80/tcp   web
vagrant@Manager:~$
vagrant@Manager:~$
vagrant@Manager:~$ docker volume ls
DRIVER    VOLUME NAME
local     8af51386e3faa0eaafe8b68a41ee8d8775dab37b71bf66c355825e866ab05d08
local     vol1
vagrant@Manager:~$
vagrant@Manager:~$
vagrant@Manager:~$ docker volume rm vol1 8af51386e3faa0eaafe8b68a41ee8d8775dab37b71bf66c355825e866ab05d08
Error response from daemon: remove vol1: volume is in use - [02fe7a33c760353643086e7d53157b2cb887ed7f0ee64b0b40e67e7099e12fda, d8e341136f88bce
ed00b48e7495ece9d6655567b3d6fa626ec875666ada1d708]
Error response from daemon: remove 8af51386e3faa0eaafe8b68a41ee8d8775dab37b71bf66c355825e866ab05d08: volume is in use - [b6eddf35145557766f187
d14182f2e057e579cda9176a693a4b929593da1f14f]
vagrant@Manager:~$
vagrant@Manager:~$ docker stop $(docker ps -a -q)
b6eddf351455
d8e341136f88
02fe7a33c760
vagrant@Manager:~$ docker volume prune
WARNING! This will remove anonymous local volumes not used by at least one container.
Are you sure you want to continue? [y/N] y
Total reclaimed space: 0B
vagrant@Manager:~$
vagrant@Manager:~$ docker volume rm vol1 8af51386e3faa0eaafe8b68a41ee8d8775dab37b71bf66c355825e866ab05d08
Error response from daemon: remove vol1: volume is in use - [02fe7a33c760353643086e7d53157b2cb887ed7f0ee64b0b40e67e7099e12fda, d8e341136f88bce
ed00b48e7495ece9d6655567b3d6fa626ec875666ada1d708]
Error response from daemon: remove 8af51386e3faa0eaafe8b68a41ee8d8775dab37b71bf66c355825e866ab05d08: volume is in use - [b6eddf35145557766f187
d14182f2e057e579cda9176a693a4b929593da1f14f]
vagrant@Manager:~$ 
```