



Name: Vanshika
SAP ID: 500069061
Roll number: R171218110

Application Containerization file

Experiment 2

DOCKER VOLUME

1) Checking data in container without docker volume attached and then attaching docker volume to the container

1.1 check docker version

```
sumyak@sumyak-VirtualBox: ~  
sumyak@sumyak-VirtualBox:~$ docker --version  
Docker version 19.03.13, build 4484c46d9d
```

1.2 see any running container

```
sumyak@sumyak-VirtualBox:~$ docker ps  
CONTAINER ID    IMAGE    COMMAND    CREATED  
STATUS          PORTS    NAMES
```

1.3 see all docker container run so far

```
sumyak@sumyak-VirtualBox:~$ docker ps -a  
CONTAINER ID    IMAGE    COMMAND    CREATED  
STATUS          PORTS    NAMES  
e079f6f9cc19    ubuntu    "ps"        2 months ago  
Exited (0) 2 months ago    festive_pike  
69a9ea2dfd75    redis    "docker-entrypoint.s..." 2 months ago  
Exited (255) 2 months ago    6379/tcp    sumyakredismapped  
d5a04ec65179    redis:latest    "docker-entrypoint.s..." 2 months ago  
Exited (255) 2 months ago    0.0.0.0:32768->6379/tcp    SumyakredisDynamic  
a9b661419bb2    redis:latest    "docker-entrypoint.s..." 2 months ago  
Exited (255) 2 months ago    0.0.0.0:6379->6379/tcp    sumyakredis  
0f9e420ad2d8    redis    "docker-entrypoint.s..." 2 months ago  
Exited (255) 2 months ago    6379/tcp    romantic_wilbur  
e4f361074fc8    ubuntu    "/bin/bash" 3 months ago  
Exited (255) 2 months ago    wizardly_heisenberg
```

1.4 make a container of alpine image

```
sumyak@sumyak-VirtualBox:~$ docker run -it alpine  
Unable to find image 'alpine:latest' locally  
latest: Pulling from library/alpine  
4c0d98bf9879: Pull complete
```

1.5

- go to root folder of alpine image i.e “mnt”
- Create files a.txt, b.txt
- And view files
- The exit the container

```
sumyak@sumyak-VirtualBox:~$ docker run -it alpine
Unable to find image 'alpine:latest' locally
latest: Pulling from library/alpine
4c0d98bf9879: Pull complete
Digest: sha256:08d6ca16c60fe7490c03d10dc339d9fd8ea67c6466dea8d558526b1330a85930
Status: Downloaded newer image for alpine:latest
/ # ls
bin    etc    lib    mnt    proc  run    srv    tmp    var
dev    home  media  opt    root  sbin   sys    usr
/ # cd mnt
/mnt # touch abcd.txt
/mnt # touch a.txt
/mnt # ls
a.txt  abcd.txt
/mnt # exit
```

1.6 see the container is exited

```
sumyak@sumyak-VirtualBox:~$ docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED
STATUS        PORTS         NAMES
ed1965e19138   alpine        "/bin/sh"               About a minute
ago          Exited (0) 19 seconds ago    bold_wozniak
e079f6f9cc19   ubuntu       "ps"                    2 months ago
Exited (0) 2 months ago      festive_pike
69a9ea2dfd75   redis        "docker-entrypoint.s..." 2 months ago
Exited (255) 2 months ago    6379/tcp                 sumyakredismapped
d5a04ec65179   redis:latest  "docker-entrypoint.s..." 2 months ago
Exited (255) 2 months ago    0.0.0.0:32768->6379/tcp   SunyakredisDynamic
a9b661419bb2   redis:latest  "docker-entrypoint.s..." 2 months ago
Exited (255) 2 months ago    0.0.0.0:6379->6379/tcp   sumyakredis
0f9e420ad2d8   redis        "docker-entrypoint.s..." 2 months ago
Exited (255) 2 months ago    6379/tcp                 romantic_wilbur
e4f361074fc8   ubuntu       "/bin/bash"             3 months ago
Exited (255) 2 months ago    wizardly_heisenberg
```

1.7

- make alpine container again
- And go to its root folder and check for the files created before (they are not there)

```
sumyak@sumyak-VirtualBox:~$ docker run -it alpine
/ # cd mnt
/mnt # ls
/mnt # exit
```

1.8 now make another container but this time with “docker volume” And again

- go to root folder of alpine image i.e “mnt”
- Create files a.txt, b.txt
- And view files
- The exit the container

```
sunyak@sunyak-VirtualBox:~$ docker run -it -v myvol1:/mnt alpine
/ # ls
bin    etc    lib    mnt    proc  run    srv    tmp    var
dev    home  media  opt    root  sbin   sys    usr
/ # cd mnt
/mnt # touch a.txt
/mnt # touch b.txt
/mnt # ls
a.txt  b.txt
/mnt # exit
```

1.9 make another container and check for the files created before now this time files are present Even after we have exited the container

```
sunyak@sunyak-VirtualBox:~$ docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED
STATUS        PORTS     NAMES
sunyak@sunyak-VirtualBox:~$ docker run -it -v myvol1:/mnt alpine
/ # cd mnt
/mnt # ls
a.txt  b.txt
/mnt # exit
sunyak@sunyak-VirtualBox:~$
```

2) Docker volume inspection

2.1 you can check all the volumes created

```
sumyak@sumyak-VirtualBox:~$ docker volume ls
DRIVER      VOLUME NAME
local       b2b4e45313908d64b9b494597c0da5d0e086f5fa1d806cc5b7532c389fc9
5970
local       ca0ec7acdb2cb99160161f84fff787d438448bbc84161ff07938c3e9bcd8
18d8
local       f21a7e167458c5b2d0590769e99ff19aa540c3c65fa9ce0be1e61cec0317
e8e4
local       myvol1
```

2.2 you can inspect your volume and know its location

```
sumyak@sumyak-VirtualBox:~$ docker volume inspect myvol1
[
  {
    "CreatedAt": "2021-02-01T16:19:02+05:30",
    "Driver": "local",
    "Labels": null,
    "Mountpoint": "/var/lib/docker/volumes/myvol1/_data",
    "Name": "myvol1",
    "Options": null,
    "Scope": "local"
  }
]
sumyak@sumyak-VirtualBox:~$
```

2.3 you can go to your volume location and check for the content in it

```
sumyak@sumyak-VirtualBox:~$ sudo ls /var/lib/docker/volumes/myvol1/_data
[sudo] password for sumyak:
a.txt  b.txt
sumyak@sumyak-VirtualBox:~$
```

2.4 removing docker volume

```
sumyak@sumyak-VirtualBox: ~
sumyak@sumyak-VirtualBox:~$ docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED
STATUS        PORTS      NAMES
sumyak@sumyak-VirtualBox:~$ docker volume

Usage:  docker volume COMMAND

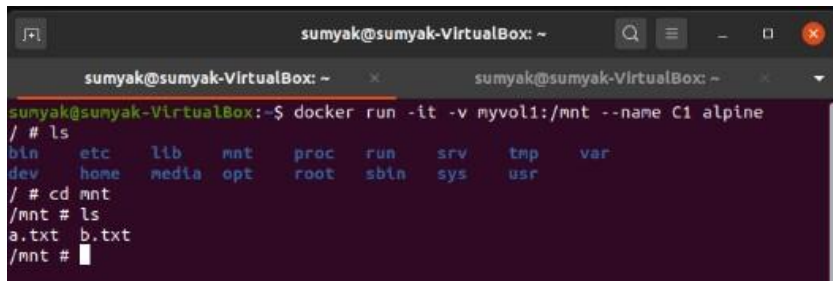
Manage volumes

Commands:
  create      Create a volume
  inspect     Display detailed information on one or more volumes
  ls          List volumes
  prune       Remove all unused local volumes
  rm          Remove one or more volumes

Run 'docker volume COMMAND --help' for more information on a command.
sumyak@sumyak-VirtualBox:~$ docker volume rm myvol1
Error response from daemon: remove myvol1: volume is in use - [26a6487c93c581801
6382b7f6c0430bd28944683e09d5464ce549a5e28491106, 8e4952f96c0ae82ae630dfc75cd303f
348e5148daf2db4bab06428e3e5952aad, 7fid8ae33a8a73a4076734a3a08725975e1b8341a1cb3
d988d33d734bcb013c, ba9bf09ccc1641205cc12213a9a312467c6e9d54a8de2e001916ff2c139
faa3e, c74cf9d93599a52510baec1082e9f28c40b26df655d3f6407dfb92b786f9293f]
sumyak@sumyak-VirtualBox:~$
```

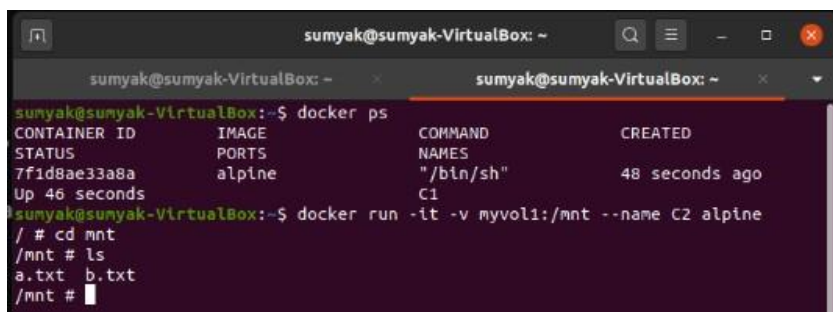
3) making container of same image with different names and sharing files between them

3.1 create a container naming c1 with same docker volume attached and see the files present in it



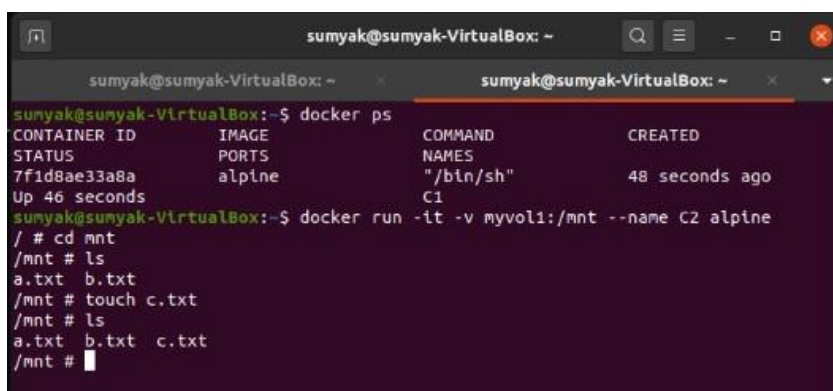
```
sumyak@sumyak-VirtualBox: ~  
sumyak@sumyak-VirtualBox: ~  
sumyak@sumyak-VirtualBox:~$ docker run -it -v myvol1:/mnt --name C1 alpine  
/ # ls  
bin      etc      lib      mnt      proc     run      srv      tmp      var  
dev      home    media    opt      root    /sbin    sys      usr  
/ # cd /mnt  
/mnt # ls  
a.txt    b.txt  
/mnt #
```

3.2 create another container with name c2 and same docker volume attached



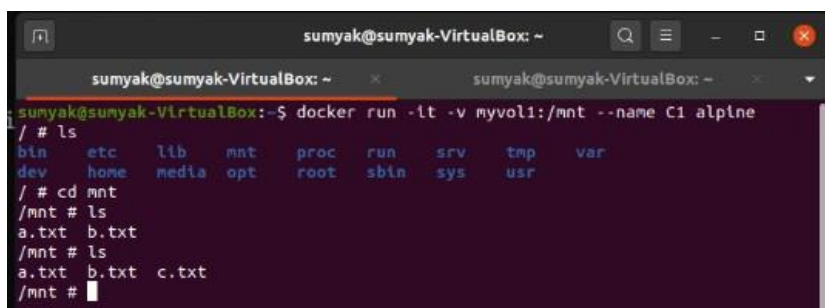
```
sumyak@sumyak-VirtualBox: ~  
sumyak@sumyak-VirtualBox: ~  
sumyak@sumyak-VirtualBox:~$ docker ps  
CONTAINER ID   IMAGE     COMMAND   CREATED  
STATUS        PORTS  
7f1d8ae33a8a   alpine    "/bin/sh" 48 seconds ago  
Up 46 seconds  
sumyak@sumyak-VirtualBox:~$ docker run -it -v myvol1:/mnt --name C2 alpine  
/ # cd /mnt  
/mnt # ls  
a.txt    b.txt  
/mnt #
```

3.3 create a file c.txt in container c2



```
sumyak@sumyak-VirtualBox: ~  
sumyak@sumyak-VirtualBox: ~  
sumyak@sumyak-VirtualBox:~$ docker ps  
CONTAINER ID   IMAGE     COMMAND   CREATED  
STATUS        PORTS  
7f1d8ae33a8a   alpine    "/bin/sh" 48 seconds ago  
Up 46 seconds  
sumyak@sumyak-VirtualBox:~$ docker run -it -v myvol1:/mnt --name C2 alpine  
/ # cd /mnt  
/mnt # ls  
a.txt    b.txt  
/mnt # touch c.txt  
/mnt # ls  
a.txt    b.txt    c.txt  
/mnt #
```

3.4 you see the same file c.txt in c1 container also



```
sumyak@sumyak-VirtualBox: ~  
sumyak@sumyak-VirtualBox: ~  
sumyak@sumyak-VirtualBox:~$ docker run -it -v myvol1:/mnt --name C1 alpine  
/ # ls  
bin      etc      lib      mnt      proc     run      srv      tmp      var  
dev      home    media    opt      root    /sbin    sys      usr  
/ # cd /mnt  
/mnt # ls  
a.txt    b.txt  
/mnt # ls  
a.txt    b.txt    c.txt  
/mnt #
```

4) making container of different image and sharing files between them

4.1

- Create another container but this time with different image “ubuntu” with same docker volume
- Go to its root folder “var”
- Create a file d.txt
- And exit the container

```
sumyak@sumyak-VirtualBox:~$ docker run -it -v myvol1:/var ubuntu
root@c74cf9d93599:/# cd var
root@c74cf9d93599:/var# ls
a.txt b.txt c.txt
root@c74cf9d93599:/var# touch d.txt
root@c74cf9d93599:/var# ls
a.txt b.txt c.txt d.txt
root@c74cf9d93599:/var#
```

4.2 check another container of different image have the same files present made on the another container with different image

```
sumyak@sumyak-VirtualBox:~$ docker run -it -v myvol1:/mnt --name C1 alpine
/ # ls
bin  etc  lib  mnt  proc  run  srv  tmp  var
dev  home media opt  root  sbin sys  usr
/ # cd mnt
/mnt # ls
a.txt b.txt
/mnt # ls
a.txt b.txt c.txt
/mnt # ls
a.txt b.txt c.txt d.txt
/mnt #
```