

Continuous Assessment – 3

CSE -427

VIRTUALIZATION AND CLOUD COMPUTING LABORATORY

Name : Routhu Siddhartha
Reg. No. : 12010599
Section : K20CT
Group : 2
Roll No. : RK20CTB51
Course code : CSE427
Task : Continuous Assessment 3
Date : 06-04-2023

Under the guidance of
AJAY KUMAR BADHAN Sir,
(27337).



L LOVELY
P ROFESSIONAL
U NIVERSITY

Installing a Dockers in the UBUNTU Operating system and running hello-world.

Step 1:

Open the terminal in the UBUNTU and start with the command “**sudo -s**” to create the root in the terminal.

Step 2:

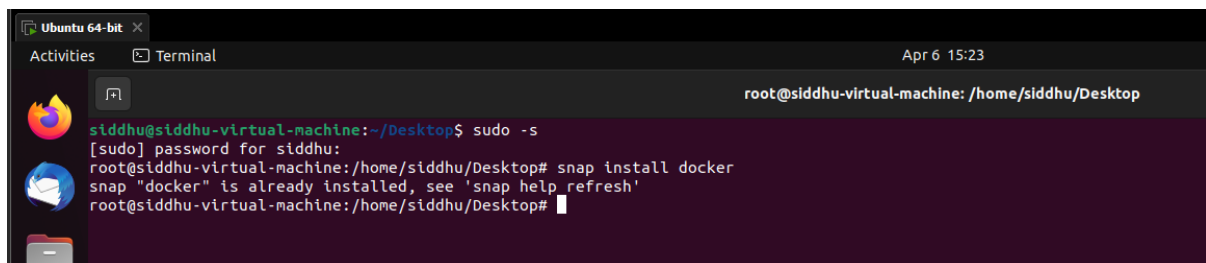
Entering the password to enter the account.

Step 3:

Now, to install the docker use the command “**snap install docker**”

It will install the environment of docker in our operating system with latest version.

If docker already exists it will trigger that docker is already installed



```
Ubuntu 64-bit x
Activities Terminal Apr 6 15:23
root@siddhu-virtual-machine: /home/siddhu/Desktop

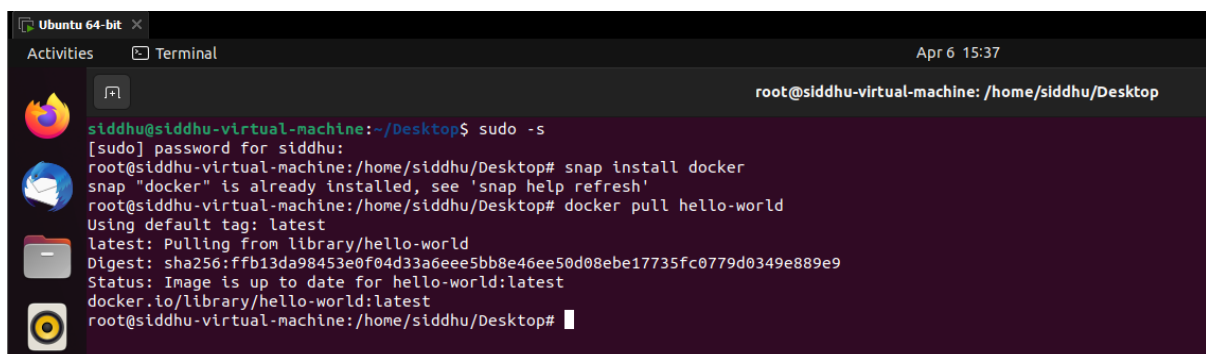
siddhu@siddhu-virtual-machine:~/Desktop$ sudo -s
[sudo] password for siddhu:
root@siddhu-virtual-machine:/home/siddhu/Desktop# snap install docker
snap "docker" is already installed, see 'snap help refresh'
root@siddhu-virtual-machine:/home/siddhu/Desktop#
```

Running Hello-World

Step 1:

We need to pull the image of the particular executing by the command
so, for pull the image of hello-world , entering the command “**docker pull hello-world**”

It will install the certain requirements of that image.



```
Ubuntu 64-bit x
Activities Terminal Apr 6 15:37
root@siddhu-virtual-machine: /home/siddhu/Desktop

siddhu@siddhu-virtual-machine:~/Desktop$ sudo -s
[sudo] password for siddhu:
root@siddhu-virtual-machine:/home/siddhu/Desktop# snap install docker
snap "docker" is already installed, see 'snap help refresh'
root@siddhu-virtual-machine:/home/siddhu/Desktop# docker pull hello-world
Using default tag: latest
latest: Pulling from library/hello-world
Digest: sha256:fffb13da98453e0f04d33a6eee5bb8e46ee50d08ebe17735fc0779d0349e889e9
Status: Image is up to date for hello-world:latest
docker.io/library/hello-world:latest
root@siddhu-virtual-machine:/home/siddhu/Desktop#
```

Step 2:

Now, Need to run the image which we pulled.

We use “**docker run hello-world**” command to run the image file and give the output.

```
Ubuntu 64-bit x
Activities Terminal Apr 6 15:47
root@siddhu-virtual-machine: /home/siddhu/Desktop

siddhu@siddhu-virtual-machine:~/Desktop$ sudo -s
[sudo] password for siddhu:
root@siddhu-virtual-machine:/home/siddhu/Desktop# snap install docker
snap "docker" is already installed, see 'snap help refresh'
root@siddhu-virtual-machine:/home/siddhu/Desktop# docker pull hello-world
Using default tag: latest
latest: Pulling from library/hello-world
Digest: sha256:fffb13da98453e0f04d33a6eee5bb8e46ee50d08ebe17735fc0779d0349e889e9
Status: Image is up to date for hello-world:latest
docker.io/library/hello-world:latest
root@siddhu-virtual-machine:/home/siddhu/Desktop# 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@siddhu-virtual-machine:/home/siddhu/Desktop#
```

Task 2:

Pulling the images of the whalesay and minesweeper game

Part 1:

- Whales-say game

To pull this game in a new container we use the command **“docker pull docker/whalesay”**

It will install all the related packages.

```
Terminal Apr 6 15:52
root@siddhu-virtual-machine: /home/siddhu/Desktop

https://docs.docker.com/get-started/

root@siddhu-virtual-machine:/home/siddhu/Desktop# docker pull docker/whalesay
Using default tag: latest
latest: Pulling from docker/whalesay
Image docker.io/docker/whalesay:latest uses outdated schema1 manifest format. Please upgrade to a schema2 image for better future compatibility. More in
formation at https://docs.docker.com/registry/spec/deprecated-schema-v1/
e190868d63f8: Already exists
909cd34c6fd7: Already exists
0b9bfabab7c1: Already exists
a3ed95caeb02: Already exists
00bf65475aba: Already exists
c57b6bcc83e3: Already exists
8978f6879e2f: Already exists
8eed3712d2cf: Already exists
Digest: sha256:178598e51a26abbc958b8a2e48825c90bc22e641de3d31e18aaf55f3258ba93b
Status: Downloaded newer image for docker/whalesay:latest
docker.io/docker/whalesay:latest
root@siddhu-virtual-machine:/home/siddhu/Desktop#
```

To run the game we need to use the following the command **“docker run docker/whalesay cowsay boo”**

This will give the following output. And accordingly we can change the command with our customized output command in place of **BOO** (example is provided in the output figure below)

[illegible]

Part 2:

- Minesweeper

Pulling the image by the command by **"docker pull nadav42/minesweeper"**

It will install all the related packages accordingly.

```
terminal Apr 6 16:00
root@siddhu-virtual-machine: /home/siddhu/Desktop

root@siddhu-virtual-machine:/home/siddhu/Desktop# docker pull nadav42/minesweeper
Using default tag: latest
latest: Pulling from nadav42/minesweeper
Digest: sha256:2f6fa3c0564968ec9cfc54b1e274a246b51f39259103fbc951ffc166a7effea7
Status: Image is up to date for nadav42/minesweeper:latest
docker.io/nadav42/minesweeper:latest
root@siddhu-virtual-machine:/home/siddhu/Desktop#
```

→To run the game we need to use the following the command **"docker run -it --rm -p 5000:5000 nadav42/minesweeper"**

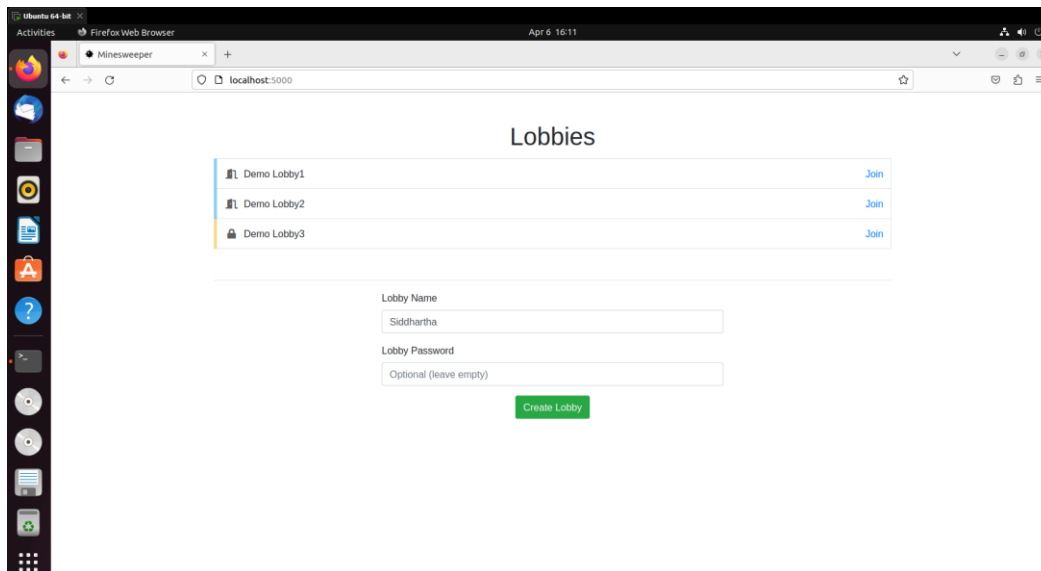
```
Apr 6 16:06
root@siddhu-virtual-machine: /home/siddhu/Desktop
root@siddhu-virtual-machine:/home/siddhu/Desktop# docker pull nadav42/minesweeper
Using default tag: latest
latest: Pulling from nadav42/minesweeper
Digest: sha256:2f6fa3c0564968ec9cfc54b1e274a246b51f39259103fbc951ffc166a7effea7
Status: Image is up to date for nadav42/minesweeper:latest
docker.io/nadav42/minesweeper:latest
root@siddhu-virtual-machine:/home/siddhu/Desktop# docker run -it --rm -p 5000:5000 nava
v42/minesweeper
SCRIPT_PATH=/app
SESSION_TYPE=filesystem
changed flask static folder to: /react/build/static

placed 10 mines on board

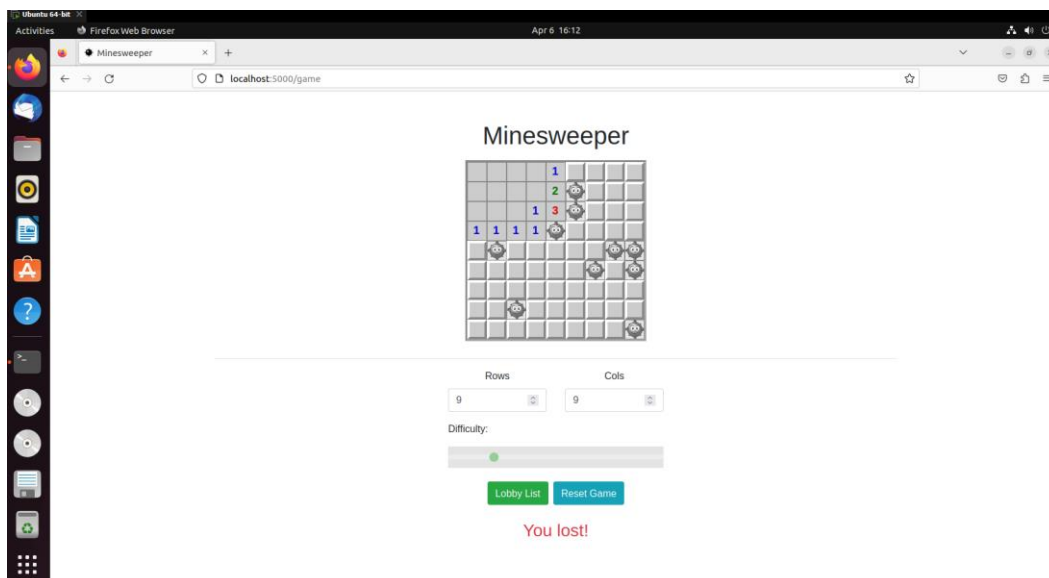
placed 10 mines on board

placed 10 mines on board
```

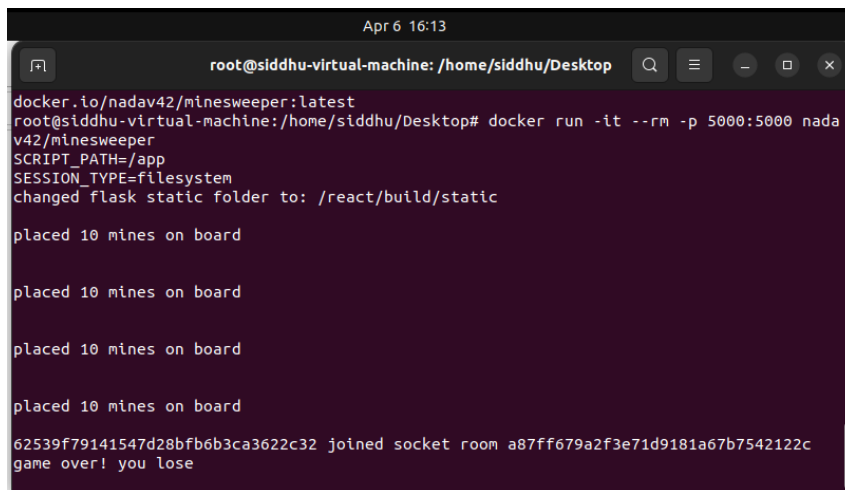
➔ Open any browser and go with the particular host number given to the game. Here, in this game the localhost number is 5000. So, enter <http://localhost:5000/> in the browser to get the game loaded in the interface of the browser.



Here is the game we lost.



It show the result in the terminal as well

A terminal window titled 'root@siddhu-virtual-machine: /home/siddhu/Desktop' with a timestamp 'Apr 6 16:13'. The terminal shows the execution of a Docker command to run a Minesweeper game. The output includes environment variables, a message about the static folder, and four 'placed 10 mines on board' messages. It ends with a connection to a socket room and a 'game over! you lose' message.

```
docker.io/nadav42/minesweeper:latest
root@siddhu-virtual-machine:/home/siddhu/Desktop# docker run -it --rm -p 5000:5000 nada
v42/minesweeper
SCRIPT_PATH=/app
SESSION_TYPE=filesystem
changed flask static folder to: /react/build/static

placed 10 mines on board

placed 10 mines on board

placed 10 mines on board

placed 10 mines on board

62539f79141547d28bfb6b3ca3622c32 joined socket room a87ff679a2f3e71d9181a67b7542122c
game over! you lose
```

Task 3

Terminal Commands on Docker in Ubuntu

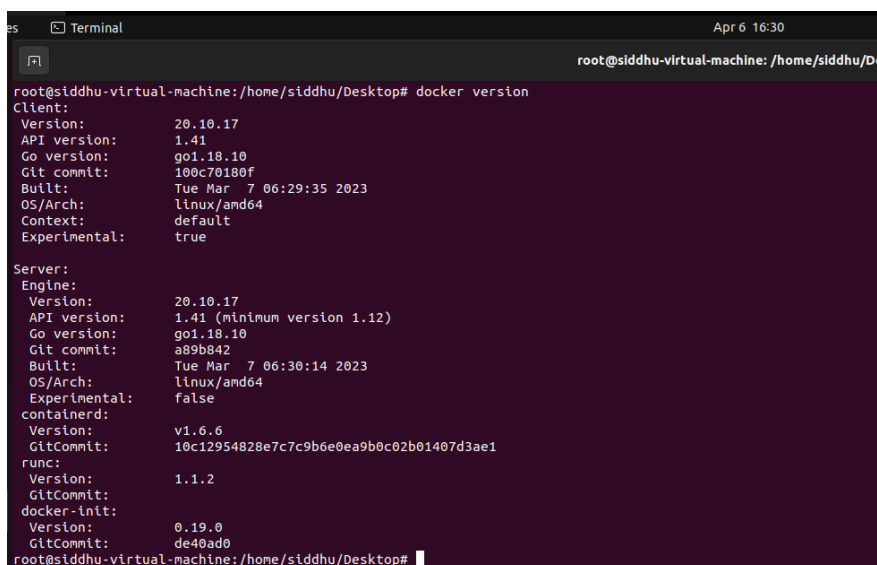
Docker is a containerization platform that simplifies the process of building, deploying, and managing applications. It provides a command-line interface that allows users to interact with the Docker platform and manage containers, images, networks, and volumes. Common Docker commands include `docker run`, `docker build`, and `docker push/pull`, which are used to create, deploy, and share Docker images. Learning Docker's command-line interface is essential for efficiently working with containerized applications. Overall, Docker is a powerful tool for improving application portability, scalability, and deployment efficiency.

Some of the most useful 50 commands in docker'

1. `docker version`: it will displays Docker version information

Syntax: `docker version [OPTIONS]`

Command: `docker version`

A terminal window titled 'Terminal' with a timestamp 'Apr 6 16:30'. The terminal shows the output of the 'docker version' command, displaying client and server information including version, API version, Go version, Git commit, built date, OS/architecture, context, and experimental status.

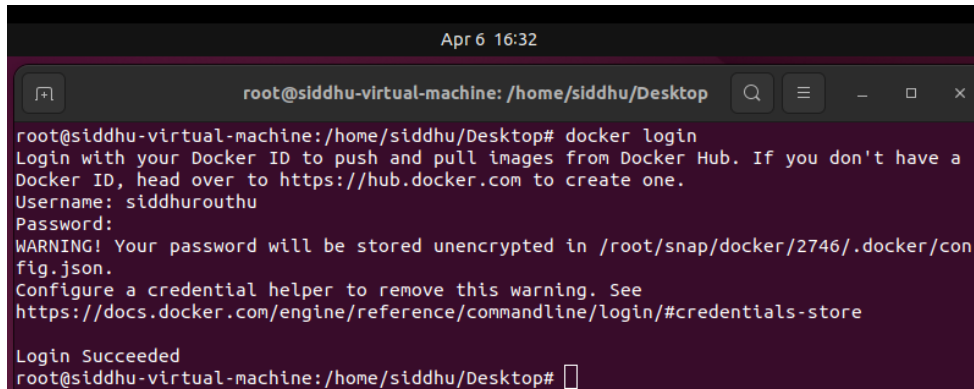
```
root@siddhu-virtual-machine:/home/siddhu/Desktop# docker version
Client:
 Version:           20.10.17
 API version:       1.41
 Go version:        go1.18.10
 Git commit:        100c70180f
 Built:             Tue Mar  7 06:29:35 2023
 OS/Arch:           linux/amd64
 Context:           default
 Experimental:      true

Server:
 Engine:
  Version:          20.10.17
  API version:      1.41 (minimum version 1.12)
  Go version:       go1.18.10
  Git commit:       a89b842
  Built:            Tue Mar  7 06:30:14 2023
  OS/Arch:          linux/amd64
  Experimental:     false
 containerd:
  Version:          v1.6.6
  GitCommit:        10c12954828e7c7c9b6e0ea9b0c02b01407d3ae1
 runc:
  Version:          1.1.2
  GitCommit:
 docker-init:
  Version:          0.19.0
  GitCommit:        de40ad0
root@siddhu-virtual-machine:/home/siddhu/Desktop#
```

2. docker login: Logs in to a Docker registry.

Syntax: docker login [OPTIONS] [SERVER]

Command: docker login

A terminal window titled 'Apr 6 16:32' showing the execution of the 'docker login' command. The prompt is 'root@siddhu-virtual-machine: /home/siddhu/Desktop'. The output shows instructions for logging in, the username 'siddhurouthu', a password prompt, a warning about unencrypted storage, and a successful login message.

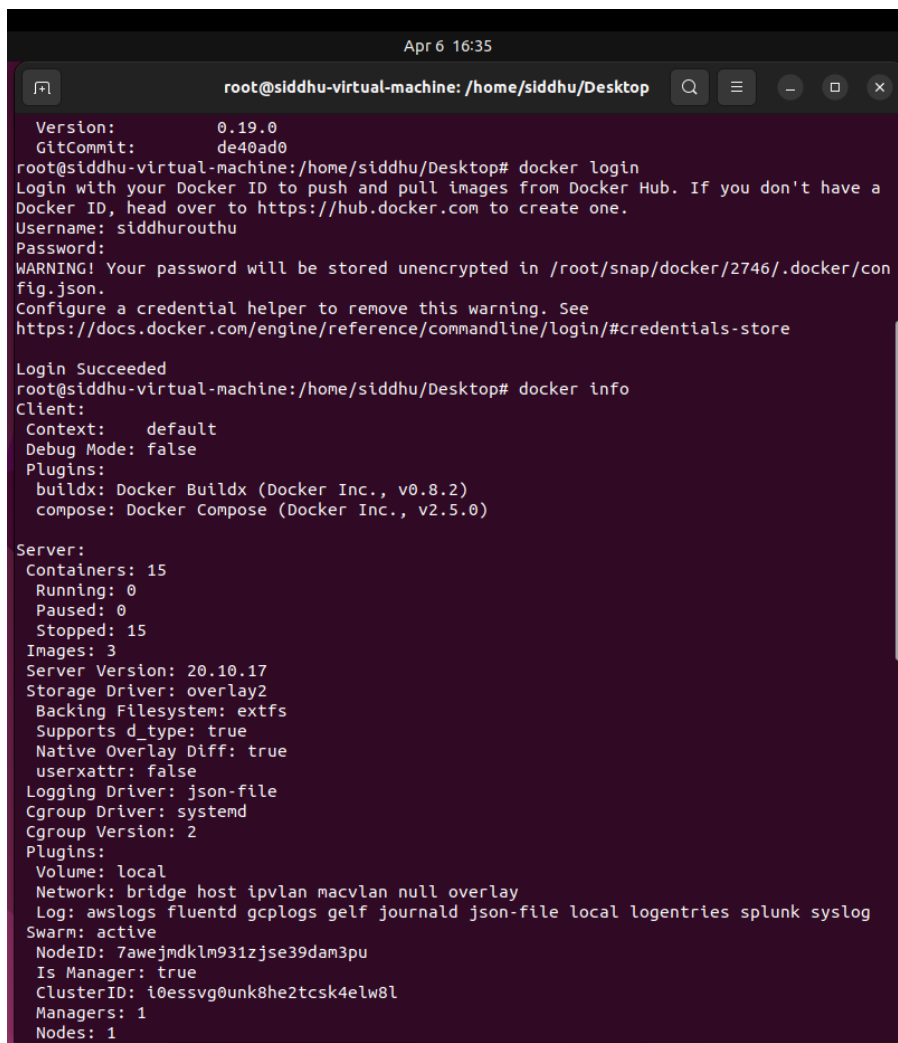
```
Apr 6 16:32
root@siddhu-virtual-machine: /home/siddhu/Desktop
root@siddhu-virtual-machine:/home/siddhu/Desktop# docker login
Login with your Docker ID to push and pull images from Docker Hub. If you don't have a
Docker ID, head over to https://hub.docker.com to create one.
Username: siddhurouthu
Password:
WARNING! Your password will be stored unencrypted in /root/.docker/credentials.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
root@siddhu-virtual-machine:/home/siddhu/Desktop#
```

3. docker info: Shows the system-wide information about the Docker installation, including the number of containers, images, and more.

Syntax: docker info [OPTIONS]

Command: docker info

A terminal window titled 'Apr 6 16:35' showing the output of the 'docker info' command. The prompt is 'root@siddhu-virtual-machine: /home/siddhu/Desktop'. The output displays detailed system information including version, client context, server status (15 containers, 3 images), storage driver (overlay2), and swarm mode details.

```
Apr 6 16:35
root@siddhu-virtual-machine: /home/siddhu/Desktop
Version: 0.19.0
GitCommit: de40ad0
root@siddhu-virtual-machine:/home/siddhu/Desktop# docker login
Login with your Docker ID to push and pull images from Docker Hub. If you don't have a
Docker ID, head over to https://hub.docker.com to create one.
Username: siddhurouthu
Password:
WARNING! Your password will be stored unencrypted in /root/.docker/credentials.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
root@siddhu-virtual-machine:/home/siddhu/Desktop# docker info
Client:
Context: default
Debug Mode: false
Plugins:
  buildx: Docker Buildx (Docker Inc., v0.8.2)
  compose: Docker Compose (Docker Inc., v2.5.0)
Server:
Containers: 15
  Running: 0
  Paused: 0
  Stopped: 15
Images: 3
Server Version: 20.10.17
Storage Driver: overlay2
  Backing Filesystem: extfs
  Supports d_type: true
  Native Overlay Diff: true
  userxattr: false
Logging Driver: json-file
Cgroup Driver: systemd
Cgroup Version: 2
Plugins:
  Volume: local
  Network: bridge host ipvlan macvlan null overlay
  Log: awslogs fluentd gcplogs gelf journald json-file local logentries splunk syslog
Swarm: active
  NodeID: 7awejmklm931zjse39dam3pu
  Is Manager: true
  ClusterID: i0essvg0unk8he2tcsk4elw8l
  Managers: 1
  Nodes: 1
```

4. docker image ls: Lists all available images on the local machine (same as docker images).

Syntax: docker image ls [OPTIONS] [REPOSITORY[:TAG]]

Command: docker image ls

```
Apr 6 16:38
root@siddhu-virtual-machine: /home/siddhu/Desktop

root@siddhu-virtual-machine:/home/siddhu/Desktop# docker image ls
REPOSITORY          TAG             IMAGE ID        CREATED         SIZE
hello-world          latest          feb5d9fea6a5    18 months ago  13.3kB
nadav42/minesweeper  latest          24648c3ff6d8    2 years ago    214MB
docker/whalesay       latest          6b362a9f73eb    7 years ago    247MB
root@siddhu-virtual-machine:/home/siddhu/Desktop#
```

5. docker run: runs a command in a new container

Command: runs a command in a new container

```
Ubuntu 64-bit x
Activities Terminal
Apr 6 15:47
root@siddhu-virtual-machine: /home/siddhu/Desktop

siddhu@siddhu-virtual-machine:~/Desktop$ sudo -s
[sudo] password for siddhu:
root@siddhu-virtual-machine:/home/siddhu/Desktop# snap install docker
snap "docker" is already installed, see 'snap help refresh'
root@siddhu-virtual-machine:/home/siddhu/Desktop# docker pull hello-world
Using default tag: latest
latest: Pulling from library/hello-world
Digest: sha256:ffb13da98453e0f04d33a6ee5bb8e46ee50d08ebe17735fc0779d0349e889e9
Status: Image is up to date for hello-world:latest
docker.io/library/hello-world:latest
root@siddhu-virtual-machine:/home/siddhu/Desktop# 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@siddhu-virtual-machine:/home/siddhu/Desktop#
```

6. docker images : Lists all available images on the local machine.

Syntax: docker images [OPTIONS] [REPOSITORY[:TAG]]

Command: docker images


```
Apr 6 16:41
root@siddhu-virtual-machine: /home/siddhu/Desktop

root@siddhu-virtual-machine:/home/siddhu/Desktop# docker images
REPOSITORY          TAG             IMAGE ID         CREATED          SIZE
hello-world          latest          feb5d9fea6a5     18 months ago   13.3kB
nadav42/minesweeper  latest         24648c3ff6d8     2 years ago     214MB
docker/whalesay       latest         6b362a9f73eb     7 years ago     247MB
root@siddhu-virtual-machine:/home/siddhu/Desktop#
```

7. docker logout: Logs out of a Docker registry.

Syntax: docker logout [SERVER]

Command: docker logout

```
Apr 6 16:42
root@siddhu-virtual-machine: /home/siddhu/Desktop

root@siddhu-virtual-machine:/home/siddhu/Desktop# docker logout
Removing login credentials for https://index.docker.io/v1/
root@siddhu-virtual-machine:/home/siddhu/Desktop#
```

8. Docker container list: Lists all running containers (same as docker ps).

Syntax: docker container ls [OPTIONS]

Command: docker container ls

```
Apr 6 16:46
root@siddhu-virtual-machine: /home/siddhu/Desktop

root@siddhu-virtual-machine:/home/siddhu/Desktop# docker container ls
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
root@siddhu-virtual-machine:/home/siddhu/Desktop#
```

9. Docker container total list: Lists all containers, including those that are not running.

Syntax: docker container ls [OPTIONS]

Command: docker container ls -a

```
Apr 6 16:46
root@siddhu-virtual-machine: /home/siddhu/Desktop
root@siddhu-virtual-machine:/home/siddhu/Desktop# docker container ls -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS
c90606cd4c15   docker/whalesay  "cowsay Routhu Siddh..." 50 minutes ago Exited (0) 5
0 minutes ago   goofy_margulis
ff825057a73c   docker/whalesay  "cowsay boo"              50 minutes ago Exited (0) 5
0 minutes ago   frosty_shamir
b6dd9bfe95d3   hello-world     "/hello"                  59 minutes ago Exited (0) 5
9 minutes ago   bold_wiles
a0938d902065   hello-world     "/hello"                  4 days ago     Created
kind_ellis
fc67d193cf29   hello-world     "/hello"                  4 days ago     Created
boring_dubinsky
fe2c11ac1639   hello-world     "/hello"                  4 days ago     Created
elated_shaw
ef0dcce6f024   hello-world     "/hello"                  4 days ago     Exited (0) 4
days ago      busy_haslett
680e51d6e0de   hello-world     "/hello"                  4 days ago     Exited (0) 4
days ago      objective_buck
41695b258cf7   hello-world     "/hello"                  4 days ago     Created
nostalgic_antonelli
74a156310fce   hello-world     "/hello"                  4 days ago     Created
vibrant_brattain
856ae876c990   hello-world     "/hello"                  4 days ago     Exited (0) 4
days ago      charming_stonebraker
e9b6b9fa68d9   docker/whalesay  "cowsay Good Morning..." 7 days ago     Exited (0) 7
days ago      loving_carson
76899d637169   docker/whalesay  "cowsay boo"              7 days ago     Exited (0) 7
days ago      kind_cray
eeb463432cf8   hello-world     "/hello"                  7 days ago     Exited (0) 7
days ago      determined_feistel
0abaefabd91f   hello-world     "/hello"                  11 days ago    Exited (0) 1
1 days ago     kind_keller
root@siddhu-virtual-machine:/home/siddhu/Desktop#
```

10. Docker container prune: Removes all stopped containers.

Syntax: docker container prune [OPTIONS]

Command: docker container prune

```
Apr 6 16:47
root@siddhu-virtual-machine: /home/siddhu/Desktop
root@siddhu-virtual-machine:/home/siddhu/Desktop# docker container prune
WARNING! This will remove all stopped containers.
Are you sure you want to continue? [y/N] n
Total reclaimed space: 0B
root@siddhu-virtual-machine:/home/siddhu/Desktop#
```

11. docker ps: Lists all running containers.

Syntax: docker ps [OPTIONS]

Command: docker ps --all

```
Apr 6 16:48
root@siddhu-virtual-machine: /home/siddhu/Desktop
root@siddhu-virtual-machine:/home/siddhu/Desktop# docker ps --all
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS
PORTS         NAMES
c90606cd4c15   docker/whalesay "cowsay Routhu Siddh..." 51 minutes ago Exited (0
) 51 minutes ago    goofy_margulis
ff825057a73c   docker/whalesay "cowsay boo"              52 minutes ago Exited (0
) 52 minutes ago    frosty_shamir
b6dd9bfe95d3   hello-world    "/hello"                About an hour ago Exited (0
) About an hour ago bold_wiles
a0938d902065   hello-world    "/hello"                4 days ago      Created
kind_ellis
fc67d193cf29   hello-world    "/hello"                4 days ago      Created
boring_dubinsky
fe2c11ac1639   hello-world    "/hello"                4 days ago      Created
elated_shaw
ef0dcce6f024   hello-world    "/hello"                4 days ago      Exited (0
) 4 days ago        busy_haslett
680e51d6e0de   hello-world    "/hello"                4 days ago      Exited (0
) 4 days ago        objective_buck
41695b258cf7   hello-world    "/hello"                4 days ago      Created
nostalgic_antonelli
74a156310fce   hello-world    "/hello"                4 days ago      Created
vibrant_brattain
856ae876c990   hello-world    "/hello"                4 days ago      Exited (0
) 4 days ago        charming_stonebraker
e9b6b9fa68d9   docker/whalesay "cowsay Good Morning..." 7 days ago      Exited (0
) 7 days ago        loving_carson
76899d637169   docker/whalesay "cowsay boo"              7 days ago      Exited (0
) 7 days ago        kind_cray
eeb463432cf8   hello-world    "/hello"                7 days ago      Exited (0
) 7 days ago        determined_feistel
0abaefabd91f   hello-world    "/hello"                11 days ago     Exited (0
) 11 days ago       kind_keller
root@siddhu-virtual-machine:/home/siddhu/Desktop#
```

12. Creating new docker network: Creates a new network.

Syntax: docker network create [OPTIONS] NETWORK

Command: docker network create --attachable NETWORK

```
root@siddhu-virtual-machine:/home/siddhu# docker network create --attachable NETWORK
d03e5d00545e94965812b2c40e81f26e53bb82786339fcff65d6e3d829d99a4f
```

13. Lists of docker network: Shows the list of the docker network

Syntax: docker network ls [OPTIONS]

Command: docker network ls

```
Apr 6 16:50
root@siddhu-virtual-machine: /home/siddhu/Desktop
root@siddhu-virtual-machine:/home/siddhu/Desktop# docker network ls
NETWORK ID          NAME                DRIVER             SCOPE
d03e5d00545e        NETWORK            bridge             local
f4f2b5a54c71        bridge             bridge             local
5177b7606a89        docker_gwbridge    bridge             local
965bb0bbfa5e        host               host               local
db41xs1gbgwk        ingress            overlay            swarm
2002a64095bf        none               null               local
root@siddhu-virtual-machine:/home/siddhu/Desktop#
```

14. Inspect the docker network: displays detailed information on one or more networks

Syntax: `docker network inspect [OPTIONS] NETWORK [NETWORK...]`

Command: `docker network inspect --verbose NETWORK d03e5d00545e`

```
Apr 6 16:51
root@siddhu-virtual-machine: /home/siddhu/Desktop
root@siddhu-virtual-machine:/home/siddhu/Desktop# docker network inspect -verbose NETWORK d03e5d00545e
[
  {
    "Name": "NETWORK",
    "Id": "d03e5d00545e94965812b2c40e81f26e53bb82786339fcff65d6e3d829d99a4f",
    "Created": "2023-04-02T13:51:47.452663232+05:30",
    "Scope": "local",
    "Driver": "bridge",
    "EnableIPv6": false,
    "IPAM": {
      "Driver": "default",
      "Options": {},
      "Config": [
        {
          "Subnet": "172.18.0.0/16",
          "Gateway": "172.18.0.1"
        }
      ]
    },
    "Internal": false,
    "Attachable": true,
    "Ingress": false,
    "ConfigFrom": {
      "Network": ""
    },
    "ConfigOnly": false,
    "Containers": {},
    "Options": {},
    "Labels": {}
  },
  {
    "Name": "NETWORK",
    "Id": "d03e5d00545e94965812b2c40e81f26e53bb82786339fcff65d6e3d829d99a4f",
    "Created": "2023-04-02T13:51:47.452663232+05:30",
    "Scope": "local",
    "Driver": "bridge",
```

15. Docker-compose up: Starts all services defined in a docker-compose.yml file.

Syntax: docker [OPTIONS] COMMAND

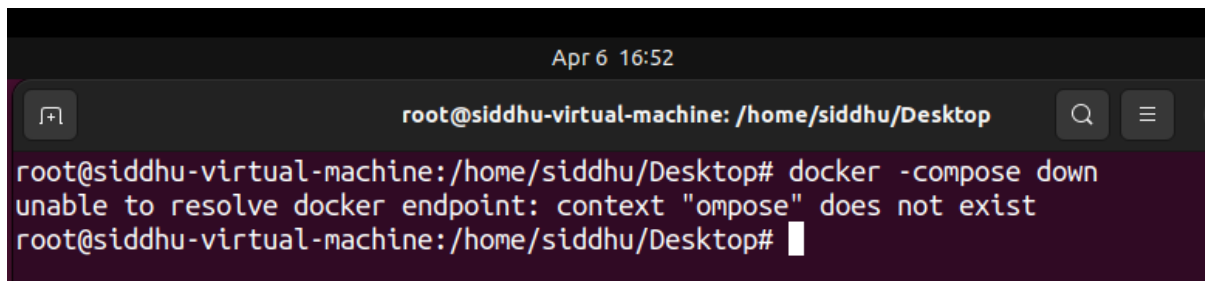
Command: docker -compose up

```
Apr 6 16:52
root@siddhu-virtual-machine: /home/siddhu/Desktop
root@siddhu-virtual-machine:/home/siddhu/Desktop# docker -compose up
unable to resolve docker endpoint: context "ompose" does not exist
root@siddhu-virtual-machine:/home/siddhu/Desktop#
```

16. Docker-compose down: Stops all services defined in a docker-compose.yml file

Syntax: docker [OPTIONS] COMMAND

Command: docker -compose down



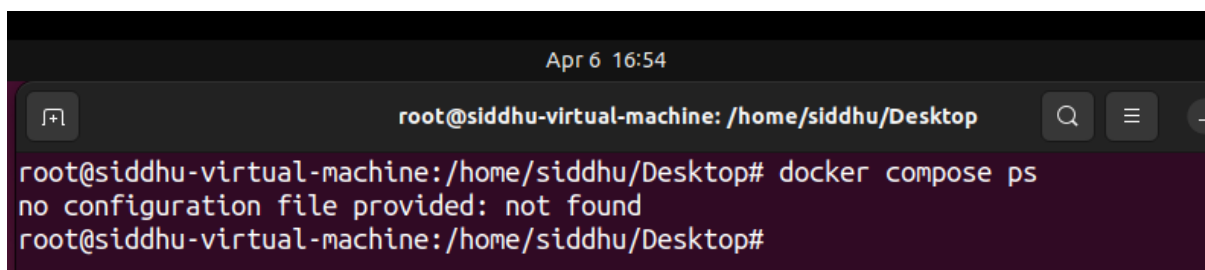
A terminal window titled 'Apr 6 16:52' with the prompt 'root@siddhu-virtual-machine: /home/siddhu/Desktop'. The user enters the command 'docker -compose down'. The output is 'unable to resolve docker endpoint: context "ompose" does not exist'. The prompt returns to 'root@siddhu-virtual-machine: /home/siddhu/Desktop# '.

```
Apr 6 16:52
root@siddhu-virtual-machine: /home/siddhu/Desktop
root@siddhu-virtual-machine:/home/siddhu/Desktop# docker -compose down
unable to resolve docker endpoint: context "ompose" does not exist
root@siddhu-virtual-machine:/home/siddhu/Desktop#
```

17. Docker-compose ps: lists the running containers in a docker-compose.yml file

Syntax: docker compose ps [SERVICE...]

Command: docker compose ps



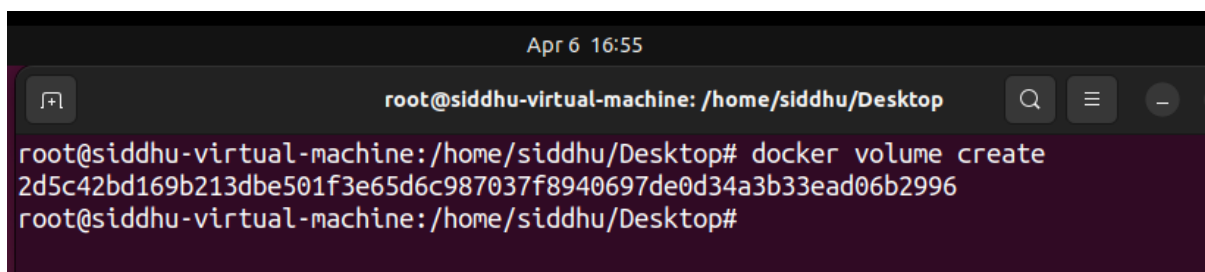
A terminal window titled 'Apr 6 16:54' with the prompt 'root@siddhu-virtual-machine: /home/siddhu/Desktop'. The user enters the command 'docker compose ps'. The output is 'no configuration file provided: not found'. The prompt returns to 'root@siddhu-virtual-machine: /home/siddhu/Desktop# '.

```
Apr 6 16:54
root@siddhu-virtual-machine: /home/siddhu/Desktop
root@siddhu-virtual-machine:/home/siddhu/Desktop# docker compose ps
no configuration file provided: not found
root@siddhu-virtual-machine:/home/siddhu/Desktop#
```

18. Docker volume create: creates a new volume

Syntax: docker volume create [OPTIONS] [VOLUME]

Command: docker volume create



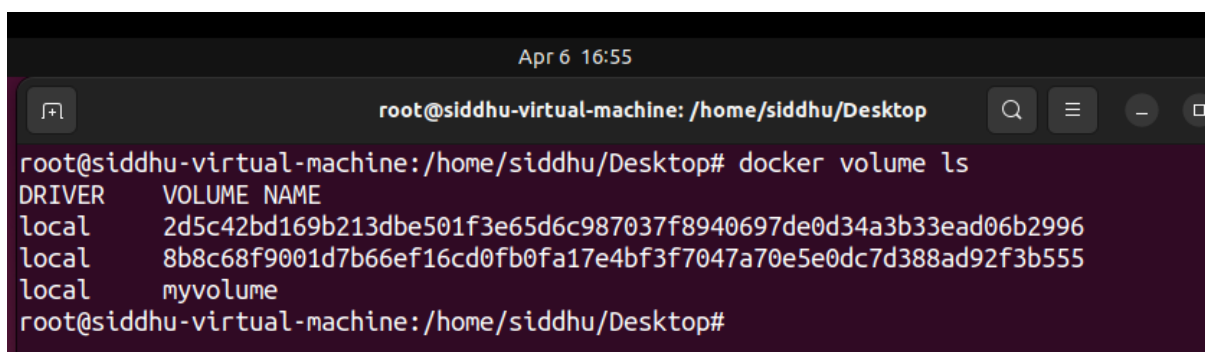
A terminal window titled 'Apr 6 16:55' with the prompt 'root@siddhu-virtual-machine: /home/siddhu/Desktop'. The user enters the command 'docker volume create'. The output is a long alphanumeric string: '2d5c42bd169b213dbe501f3e65d6c987037f8940697de0d34a3b33ead06b2996'. The prompt returns to 'root@siddhu-virtual-machine: /home/siddhu/Desktop# '.

```
Apr 6 16:55
root@siddhu-virtual-machine: /home/siddhu/Desktop
root@siddhu-virtual-machine:/home/siddhu/Desktop# docker volume create
2d5c42bd169b213dbe501f3e65d6c987037f8940697de0d34a3b33ead06b2996
root@siddhu-virtual-machine:/home/siddhu/Desktop#
```

19. Docker volume ls: To show list of volumes

Syntax: docker volume ls [OPTIONS]

Command: docker volume ls



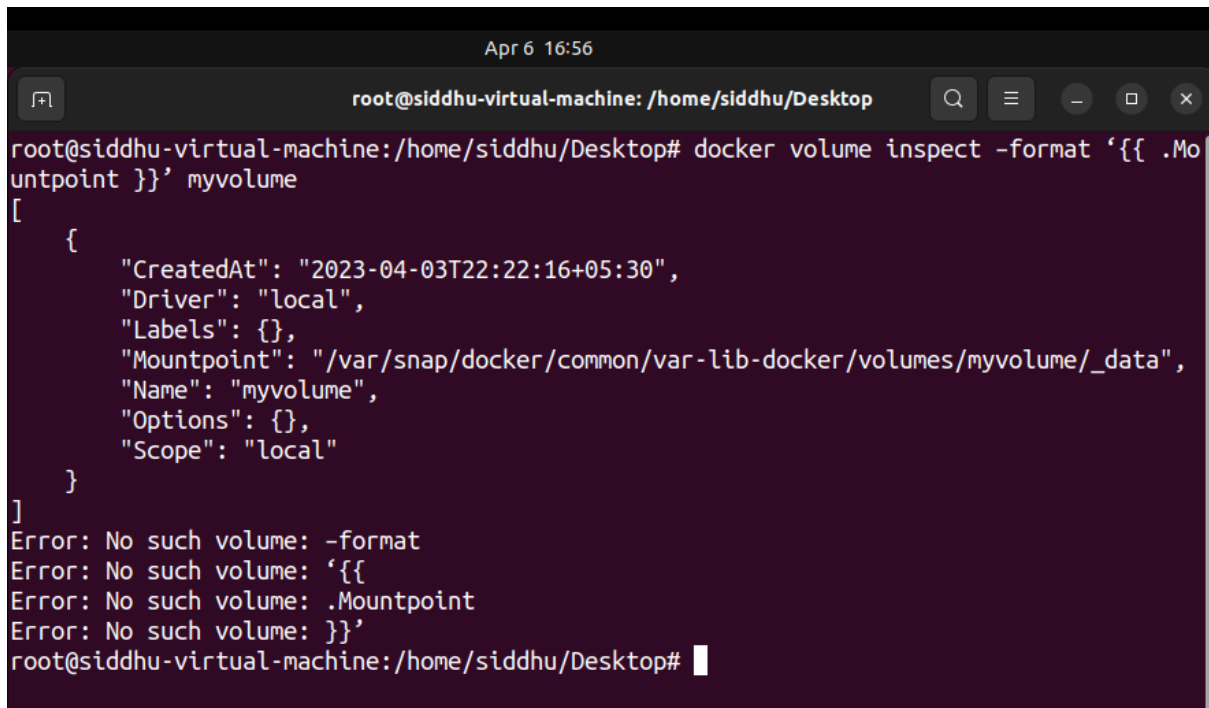
A terminal window titled 'Apr 6 16:55' with the prompt 'root@siddhu-virtual-machine: /home/siddhu/Desktop'. The user enters the command 'docker volume ls'. The output is a table with two columns: 'DRIVER' and 'VOLUME NAME'. The first column has three entries: 'local', 'local', and 'local'. The second column has three entries: '2d5c42bd169b213dbe501f3e65d6c987037f8940697de0d34a3b33ead06b2996', '8b8c68f9001d7b66ef16cd0fb0fa17e4bf3f7047a70e5e0dc7d388ad92f3b555', and 'myvolume'. The prompt returns to 'root@siddhu-virtual-machine: /home/siddhu/Desktop# '.

```
Apr 6 16:55
root@siddhu-virtual-machine: /home/siddhu/Desktop
root@siddhu-virtual-machine:/home/siddhu/Desktop# docker volume ls
DRIVER      VOLUME NAME
local       2d5c42bd169b213dbe501f3e65d6c987037f8940697de0d34a3b33ead06b2996
local       8b8c68f9001d7b66ef16cd0fb0fa17e4bf3f7047a70e5e0dc7d388ad92f3b555
local       myvolume
root@siddhu-virtual-machine:/home/siddhu/Desktop#
```

20. Docker volume inspect: displays detailed information on one or more volumes

Syntax: docker volume inspect [OPTIONS] VOLUME [VOLUME...]

Command: docker volume inspect --format '{{ .Mountpoint }}' myvolume

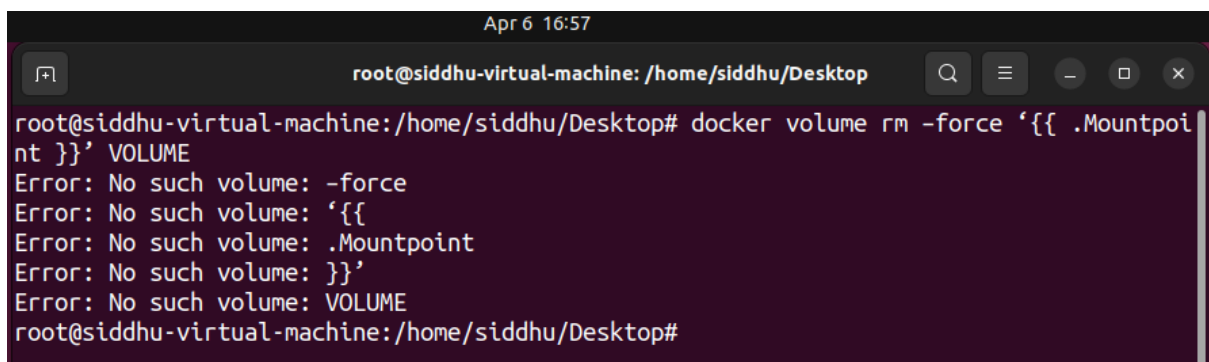


```
Apr 6 16:56
root@siddhu-virtual-machine: /home/siddhu/Desktop
root@siddhu-virtual-machine:/home/siddhu/Desktop# docker volume inspect --format '{{ .Mountpoint }}' myvolume
[
  {
    "CreatedAt": "2023-04-03T22:22:16+05:30",
    "Driver": "local",
    "Labels": {},
    "Mountpoint": "/var/snap/docker/common/var-lib-docker/volumes/myvolume/_data",
    "Name": "myvolume",
    "Options": {},
    "Scope": "local"
  }
]
Error: No such volume: --format
Error: No such volume: '{{
Error: No such volume: .Mountpoint
Error: No such volume: }}'
root@siddhu-virtual-machine:/home/siddhu/Desktop#
```

21. Docker volume rm: removes one or more volumes

Syntax: docker volume rm [OPTIONS] VOLUME [VOLUME...]

Command: docker volume rm --force '{{ .Mountpoint }}' VOLUME



```
Apr 6 16:57
root@siddhu-virtual-machine: /home/siddhu/Desktop
root@siddhu-virtual-machine:/home/siddhu/Desktop# docker volume rm --force '{{ .Mountpoint }}' VOLUME
Error: No such volume: --force
Error: No such volume: '{{
Error: No such volume: .Mountpoint
Error: No such volume: }}'
Error: No such volume: VOLUME
root@siddhu-virtual-machine:/home/siddhu/Desktop#
```

22. Docker load: loads an image from a tar archive

Syntax: docker load [OPTIONS]

Command: docker load


```
Apr 6 16:58
root@siddhu-virtual-machine: /home/siddhu/Desktop
root@siddhu-virtual-machine:/home/siddhu/Desktop# docker load
requested load from stdin, but stdin is empty
root@siddhu-virtual-machine:/home/siddhu/Desktop#
```

23. Docker stats: displays a live stream of container(s) resource usage statistics

Syntax: docker stats [OPTIONS] [CONTAINER...]

Command: docker stats

```
Apr 6 16:59
root@siddhu-virtual-machine: /home/siddhu/Desktop
CONTAINER ID   NAME      CPU %       MEM USAGE / LIMIT   MEM %      NET I/O     BLOCK I/O
PIDS
```

24. Docker events: displays a live stream of real-time events from the server

Syntax: docker events [OPTIONS]

Command: docker events

```
root@siddhu-virtual-machine:/home/siddhu# docker events
abc
xyz
pqr
```

25. Docker inspect: displays detailed information on one or more containers

Syntax: docker inspect [OPTIONS] NAME|ID [NAME|ID...]

Command: docker inspect --size hello-world


```
Apr 6 16:44
root@siddhu-virtual-machine: /home/siddhu/Desktop
root@siddhu-virtual-machine:/home/siddhu/Desktop# docker inspect --size hello-world
[
  {
    "Id": "sha256:feb5d9fea6a5e9606aa995e879d862b825965ba48de054caab5ef356dc6b3412",
    "RepoTags": [
      "hello-world:latest"
    ],
    "RepoDigests": [
      "hello-world@sha256:ffb13da98453e0f04d33a6eee5bb8e46ee50d08ebe17735fc0779d0349e889e9"
    ],
    "Parent": "",
    "Comment": "",
    "Created": "2021-09-23T23:47:57.442225064Z",
    "Container": "8746661ca3c2f215da94e6d3f7dfdcafaaff5ec0b21c9aff6af3dc379a82fbc72",
    "ContainerConfig": {
      "Hostname": "8746661ca3c2",
      "Domainname": "",
      "User": "",
      "AttachStdin": false,
      "AttachStdout": false,
      "AttachStderr": false,
      "Tty": false,
      "OpenStdin": false,
      "StdinOnce": false,
      "Env": [
        "PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"
      ],
      "Cmd": [
        "/bin/sh",
        "-c",
        "#(nop) ",
        "CMD [\"/hello\"]"
      ],
      "Image": "sha256:b9935d4e8431fb1a7f0989304ec86b3329a99a25f5efdc7f09f3f8c41434ca6d",
```

26. Docker history: displays the history of an image

Syntax: docker history [OPTIONS] IMAGE

Command: docker history --human hello-world

```
root@siddhu-virtual-machine:/home/siddhu# docker history --human hello-world
IMAGE          CREATED          CREATED BY          SIZE      COMMENT
feb5d9fea6a5   18 months ago   /bin/sh -c #(nop)  CMD ["/hello"]     0B
<missing>      18 months ago   /bin/sh -c #(nop)  COPY file:50563a97010fd7ce... 13.3kB
```

27. Docker service: To manage Docker services in a swarm use "docker service" command.

Syntax: docker service COMMAND

Command: docker service

29. `docker system prune --all`: Removes all unused images, containers, networks, and volumes, including dangling ones.

Syntax: `docker system prune [OPTIONS]`

Command: `docker system prune --all`

```
Apr 6 16:48
root@siddhu-virtual-machine: /home/siddhu/Desktop
root@siddhu-virtual-machine:/home/siddhu/Desktop# docker system prune --all
WARNING! This will remove:
- all stopped containers
- all networks not used by at least one container
- all images without at least one container associated to them
- all build cache

Are you sure you want to continue? [y/N] n
root@siddhu-virtual-machine:/home/siddhu/Desktop#
```

30. `docker system df -v`: Shows detailed information about the disk usage of Docker on the host machine, including volumes.

Syntax: `docker system df [OPTIONS]`

Command: `docker system df -v`

```
Apr 6 16:49
root@siddhu-virtual-machine: /home/siddhu/Desktop
root@siddhu-virtual-machine:/home/siddhu/Desktop# docker system df -v
Images space usage:

REPOSITORY          TAG          IMAGE ID      CREATED      SIZE      SHARED SIZE  UNIQUE SIZE  CO
CONTAINERS
hello-world         latest      feb5d9fea6a5  18 months ago  13.26kB    0B           13.26kB      11
nada4v42/minesweeper latest      24648c3ff6d8  2 years ago   213.9MB    0B           213.9MB      0
docker/whalesay     latest      6b362a9f73eb  7 years ago   247MB      0B           247MB        4

Containers space usage:

CONTAINER ID  IMAGE          COMMAND                  LOCAL VOLUMES  SIZE      CREATED
STATUS
c90606cd4c15  docker/whalesay "cowsay Routhu Siddh..." 0              0B        52 minutes ago
Exited (0) 52 minutes ago
ff825057a73c  docker/whalesay "cowsay boo"                0              0B        52 minutes ago
Exited (0) 52 minutes ago
b6dd9bfe95d3  hello-world    "/hello"                 0              0B        About an hour ago
Exited (0) About an hour ago
a0938d902065  hello-world    "/hello"                 0              0B        4 days ago
Created
fc67d193cf29  hello-world    "/hello"                 0              0B        4 days ago
Created
fe2c11ac1639  hello-world    "/hello"                 0              0B        4 days ago
Created
ef0dcce6f024  hello-world    "/hello"                 0              0B        4 days ago
Exited (0) 4 days ago
680e51d6e0de  hello-world    "/hello"                 0              0B        4 days ago
Exited (0) 4 days ago
41695b258cf7  hello-world    "/hello"                 0              0B        4 days ago
Created
74a156310fce  hello-world    "/hello"                 0              0B        4 days ago
Created
856ae876c990  hello-world    "/hello"                 0              0B        4 days ago
Exited (0) 4 days ago
e9b6b9fa68d9  docker/whalesay "cowsay Good Morning..." 0              0B        7 days ago
Exited (0) 7 days ago
76899d637169  docker/whalesay "cowsay boo"              0              0B        7 days ago
```

31. docker system info: Shows detailed system-wide information about Docker.

Syntax: docker system info [OPTIONS]

Command: docker system info

```
Apr 6 16:49
root@siddhu-virtual-machine: /home/siddhu/Desktop

root@siddhu-virtual-machine:/home/siddhu/Desktop# docker system info
Client:
 Context:      default
 Debug Mode: false
 Plugins:
  buildx: Docker Buildx (Docker Inc., v0.8.2)
  compose: Docker Compose (Docker Inc., v2.5.0)

Server:
 Containers: 15
  Running: 0
  Paused: 0
  Stopped: 15
 Images: 3
 Server Version: 20.10.17
 Storage Driver: overlay2
  Backing Filesystem: extfs
  Supports d_type: true
  Native Overlay Diff: true
  userxattr: false
 Logging Driver: json-file
 Cgroup Driver: systemd
 Cgroup Version: 2
 Plugins:
  Volume: local
  Network: bridge host ipvlan macvlan null overlay
  Log: awslogs fluentd gcplogs gelf journald json-file local logentries splunk syslog
 Swarm: active
  NodeID: 7awejmklm931zjse39dam3pu
  Is Manager: true
  ClusterID: i0essvg0unk8he2tcsk4elw8l
  Managers: 1
  Nodes: 1
  Default Address Pool: 10.0.0.0/8
  SubnetSize: 24
  Data Path Port: 4789
  Orchestration:
    Task History Retention Limit: 5
```

32. docker system prune --volumes: Removes all unused volumes.

Syntax: docker system COMMAND

Command: docker system prune --volumes

```
Apr 6 16:51
root@siddhu-virtual-machine: /home/siddhu/Desktop

root@siddhu-virtual-machine:/home/siddhu/Desktop# docker system prune --volumes
WARNING! This will remove:
 - all stopped containers
 - all networks not used by at least one container
 - all volumes not used by at least one container
 - all dangling images
 - all dangling build cache

Are you sure you want to continue? [y/N] n
root@siddhu-virtual-machine:/home/siddhu/Desktop#
```