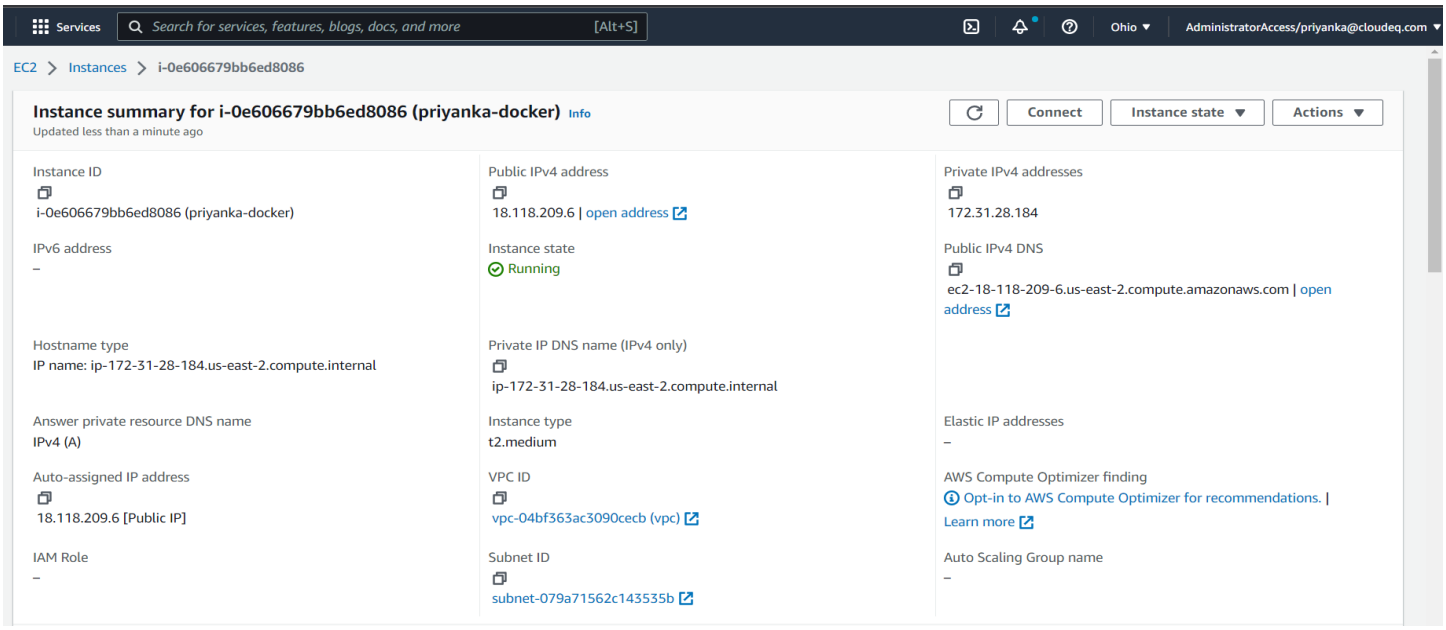
INDEX

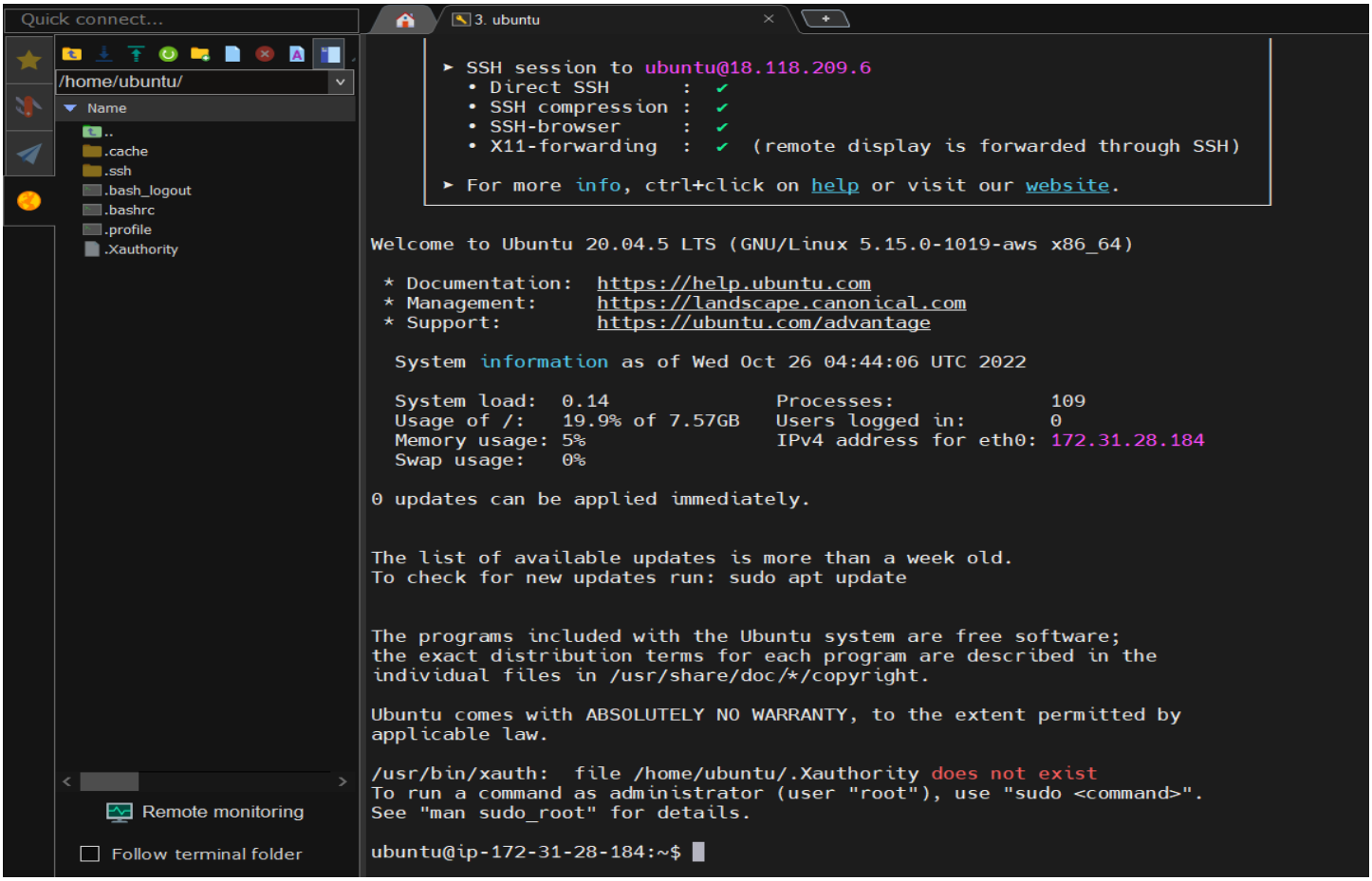
INSTALL DOCKER IN UBUNTU USING MOBAXTERM

STEPS:

1) Create an EC2 instance in AWS using Ubuntu AMI and download the key pair.



2) Create a session in MobaXterm using public IP of EC2 created in step-1:



3) Go to root to create root user (Command: $ sudo -i):

```
ubuntu@ip-172-31-28-184:~$ sudo -i
root@ip-172-31-28-184:~#
```

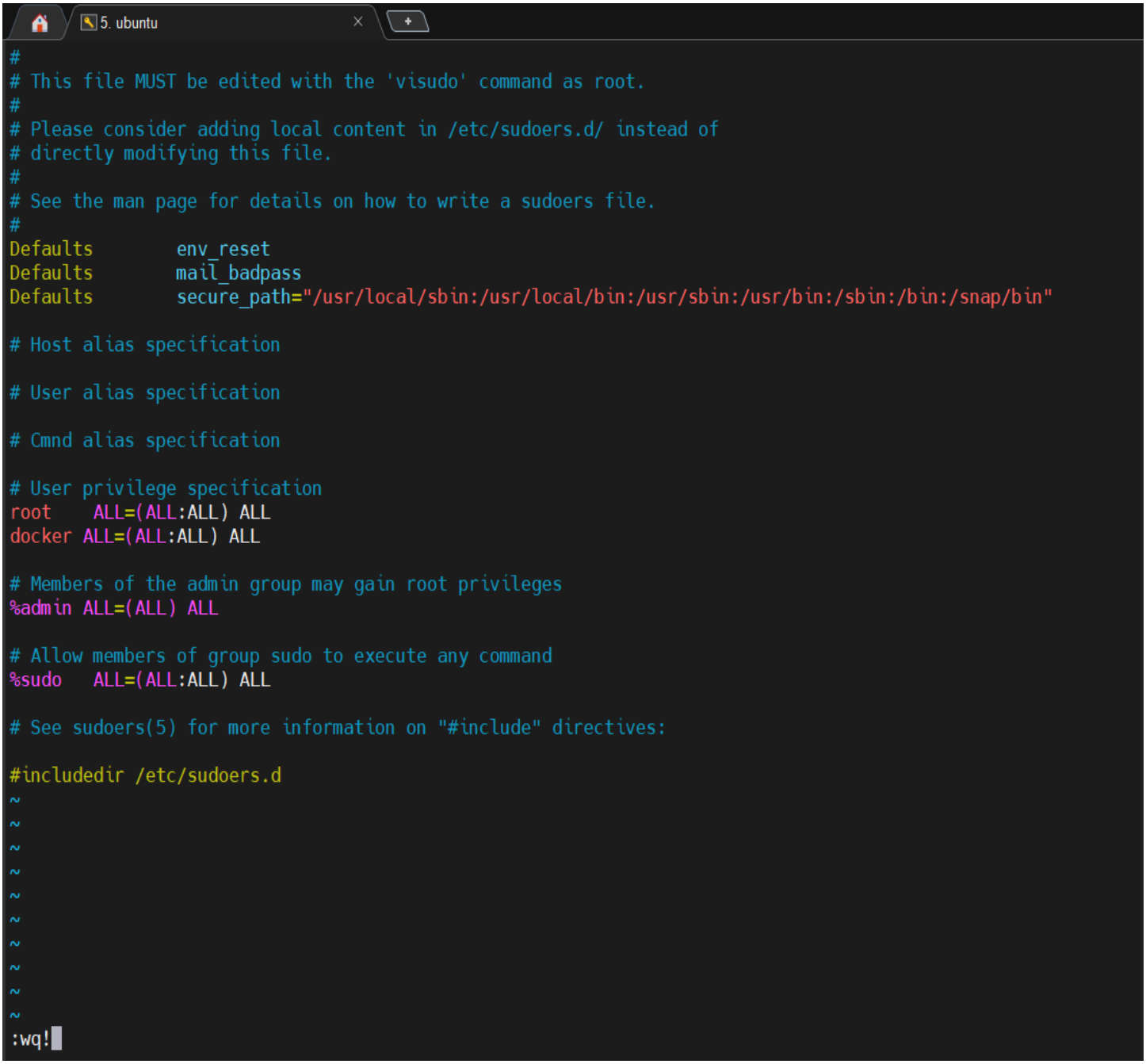
4) Add user to root (Command: # sudo adduser username):

```
root@ip-172-31-28-184:~# sudo adduser docker
Adding user `docker' ...
Adding new group `docker' (1001) ...
Adding new user `docker' (1001) with group `docker' ...
Creating home directory `/home/docker' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for docker
Enter the new value, or press ENTER for the default
        Full Name []: Priyanka
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] Y
root@ip-172-31-28-184:~#
```

5) Open root permissions /etc/sudoers in vi editor (Command: # vi /etc/sudoers)

```
root@ip-172-31-28-184:~# vi /etc/sudoers
```

6) Assign permission to user created in step-4 (Type: username ALL=(ALL:ALL) ALL) and save them:



7) Check if the permissions are assigned or not (Command: # cat /etc/sudoers):

```
root@ip-172-31-28-184:~# cat /etc/sudoers
#
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults        env_reset
Defaults        mail_badpass
Defaults        secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin"

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification
root    ALL=(ALL:ALL) ALL
docker ALL=(ALL:ALL) ALL

# Members of the admin group may gain root privileges
%admin ALL=(ALL) ALL

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "#include" directives:

#includedir /etc/sudoers.d
root@ip-172-31-28-184:~#
```

8) Move out of root to docker (Command: # su docker):

```
root@ip-172-31-28-184:~# su docker
docker@ip-172-31-28-184:/root$
```

9) Install Docker using the repository:
   ➔ Update the apt package index (Command: $ sudo apt-get update):

```
docker@ip-172-31-28-184:/root$ sudo apt-get update
[sudo] password for docker:
Hit:1 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal InRelease
Get:2 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Get:3 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-backports InRelease [108 kB]
Get:4 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal/universe amd64 Packages [8628 kB]
Get:5 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Get:6 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal/universe Translation-en [5124 kB]
Get:7 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal/universe amd64 c-n-f Metadata [265 kB]
Get:8 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal/multiverse amd64 Packages [144 kB]
Get:9 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal/multiverse Translation-en [104 kB]
Get:10 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal/multiverse amd64 c-n-f Metadata [9136 B]
Get:11 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-updates/main amd64 Packages [2192 kB]
Get:12 http://security.ubuntu.com/ubuntu focal-security/main amd64 Packages [1818 kB]
Get:13 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-updates/main Translation-en [385 kB]
Get:14 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-updates/main amd64 c-n-f Metadata [16.0 kB]
Get:15 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-updates/restricted amd64 Packages [1372 kB]
Get:16 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-updates/restricted Translation-en [194 kB]
Get:17 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-updates/restricted amd64 c-n-f Metadata [600 B]
Get:18 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-updates/universe amd64 Packages [971 kB]
Get:19 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-updates/universe Translation-en [221 kB]
Get:20 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-updates/universe amd64 c-n-f Metadata [21.8 kB]
Get:21 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-updates/multiverse amd64 Packages [29.9 kB]
Get:22 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-updates/multiverse Translation-en [7940 B]
Get:23 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-updates/multiverse amd64 c-n-f Metadata [664 B]
Get:24 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-backports/main amd64 Packages [69.5 kB]
Get:25 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-backports/main Translation-en [16.3 kB]
Get:26 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-backports/main amd64 c-n-f Metadata [1420 B]
Get:27 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-backports/restricted amd64 c-n-f Metadata [116 B]
Get:28 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-backports/universe amd64 Packages [27.1 kB]
Get:29 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-backports/universe Translation-en [16.0 kB]
Get:30 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-backports/universe amd64 c-n-f Metadata [860 B]
Get:31 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-backports/multiverse amd64 c-n-f Metadata [116 B]
Get:32 http://security.ubuntu.com/ubuntu focal-security/main Translation-en [300 kB]
Get:33 http://security.ubuntu.com/ubuntu focal-security/main amd64 c-n-f Metadata [11.2 kB]
Get:34 http://security.ubuntu.com/ubuntu focal-security/restricted amd64 Packages [1282 kB]
Get:35 http://security.ubuntu.com/ubuntu focal-security/restricted Translation-en [182 kB]
Get:36 http://security.ubuntu.com/ubuntu focal-security/restricted amd64 c-n-f Metadata [596 B]
Get:37 http://security.ubuntu.com/ubuntu focal-security/universe amd64 Packages [742 kB]
Get:38 http://security.ubuntu.com/ubuntu focal-security/universe Translation-en [137 kB]
Get:39 http://security.ubuntu.com/ubuntu focal-security/universe amd64 c-n-f Metadata [15.3 kB]
Get:40 http://security.ubuntu.com/ubuntu focal-security/multiverse amd64 Packages [22.2 kB]
```

```
Get:40 http://security.ubuntu.com/ubuntu focal-security/multiverse amd64 Packages [22.2 kB]
Get:41 http://security.ubuntu.com/ubuntu focal-security/multiverse Translation-en [5376 B]
Get:42 http://security.ubuntu.com/ubuntu focal-security/multiverse amd64 c-n-f Metadata [508 B]
Fetched 24.7 MB in 4s (6152 kB/s)
Reading package lists... Done
docker@ip-172-31-28-184:/root$
```

➔ Install packages to allow apt to use a repository over https (Command: $ sudo apt-get install \
ca-certificates \
curl \
gnupg \
lsb-release)

```
docker@ip-172-31-28-184:/root$ sudo apt-get install \
> ca-certificates \
> curl \
> gnupg \
> lsb-release
Reading package lists ... Done
Building dependency tree
Reading state information ... Done
lsb-release is already the newest version (11.1.0ubuntu2).
lsb-release set to manually installed.
ca-certificates is already the newest version (20211016~20.04.1).
ca-certificates set to manually installed.
curl is already the newest version (7.68.0-1ubuntu2.13).
curl set to manually installed.
gnupg is already the newest version (2.2.19-3ubuntu2.2).
gnupg set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 48 not upgraded.
docker@ip-172-31-28-184:/root$
```

➔ Add docker's official GPG key (Command: $ sudo mkdir -p /etc/apt/keyrings
$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/ docker.gpg)

```
docker@ip-172-31-28-184:/root$ sudo mkdir -p /etc/apt/keyrings
docker@ip-172-31-28-184:/root$
```

```
docker@ip-172-31-28-184:/root$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
```

➔ Setup a repository (Command: $ echo \
"deb [arch=$(dpkg –print-architecture) signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/ubuntu \
$(lsb_realease -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null

```
docker@ip-172-31-28-184:/root$ echo \
> "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/ubuntu \
> $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
docker@ip-172-31-28-184:/root$
```

10) Install Docker Engine:

➔ Update the apt package index (Command: $ sudo apt-get update):

```
docker@ip-172-31-28-184:/root$ sudo apt-get update
Hit:1 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal InRelease
Hit:2 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-updates InRelease
Hit:3 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-backports InRelease
Get:4 https://download.docker.com/linux/ubuntu focal InRelease [57.7 kB]
Get:5 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Get:6 https://download.docker.com/linux/ubuntu focal/stable amd64 Packages [20.8 kB]
Fetched 192 kB in 1s (354 kB/s)
Reading package lists... Done
docker@ip-172-31-28-184:/root$
```

➔ Install Docker Engine, container and Docker Compose (Command: $ sudo apt-get install docker-ce docker-ce-cli containerd.io docker-compose-plugin):

```
docker@ip-172-31-28-184:/root$ sudo apt-get install docker-ce docker-ce-cli containerd.io docker-compose-plugin
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  docker-ce-rootless-extras docker-scan-plugin pigz slirp4netns
Suggested packages:
  aufs-tools cgroupfs-mount | cgroup-lite
The following NEW packages will be installed:
  containerd.io docker-ce docker-ce-cli docker-ce-rootless-extras docker-compose-plugin docker-scan-plugin pigz slirp4netns
0 upgraded, 8 newly installed, 0 to remove and 48 not upgraded.
Need to get 111 MB of archives.
After this operation, 428 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal/universe amd64 pigz amd64 2.4-1 [57.4 kB]
Get:2 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal/universe amd64 slirp4netns amd64 0.4.3-1 [74.3 kB]
Get:3 https://download.docker.com/linux/ubuntu focal/stable amd64 containerd.io amd64 1.6.9-1 [27.7 MB]
Get:4 https://download.docker.com/linux/ubuntu focal/stable amd64 docker-ce-cli amd64 5:20.10.21~3-0~ubuntu-focal [41.5 MB]
Get:5 https://download.docker.com/linux/ubuntu focal/stable amd64 docker-ce amd64 5:20.10.21~3-0~ubuntu-focal [20.5 MB]
Get:6 https://download.docker.com/linux/ubuntu focal/stable amd64 docker-ce-rootless-extras amd64 5:20.10.21~3-0~ubuntu-focal [8394 kB]
Get:7 https://download.docker.com/linux/ubuntu focal/stable amd64 docker-compose-plugin amd64 2.12.2~ubuntu-focal [9567 kB]
Get:8 https://download.docker.com/linux/ubuntu focal/stable amd64 docker-scan-plugin amd64 0.21.0~ubuntu-focal [3622 kB]
Fetched 111 MB in 4s (30.2 MB/s)
Selecting previously unselected package pigz.
(Reading database ... 61718 files and directories currently installed.)
Preparing to unpack .../0-pigz_2.4-1_amd64.deb ...
Unpacking pigz (2.4-1) ...
Selecting previously unselected package containerd.io.
Preparing to unpack .../1-containerd.io_1.6.9-1_amd64.deb ...
Unpacking containerd.io (1.6.9-1) ...
Selecting previously unselected package docker-ce-cli.
Preparing to unpack .../2-docker-ce-cli_5%3a20.10.21~3-0~ubuntu-focal_amd64.deb ...
Unpacking docker-ce-cli (5:20.10.21~3-0~ubuntu-focal) ...
Selecting previously unselected package docker-ce.
Preparing to unpack .../3-docker-ce_5%3a20.10.21~3-0~ubuntu-focal_amd64.deb ...
Unpacking docker-ce (5:20.10.21~3-0~ubuntu-focal) ...
Selecting previously unselected package docker-ce-rootless-extras.
Preparing to unpack .../4-docker-ce-rootless-extras_5%3a20.10.21~3-0~ubuntu-focal_amd64.deb ...
Unpacking docker-ce-rootless-extras (5:20.10.21~3-0~ubuntu-focal) ...
Selecting previously unselected package docker-compose-plugin.
Preparing to unpack .../5-docker-compose-plugin_2.12.2~ubuntu-focal_amd64.deb ...
```

```
Preparing to unpack .../6-docker-scan-plugin_0.21.0~ubuntu-focal_amd64.deb ...
Unpacking docker-scan-plugin (0.21.0~ubuntu-focal) ...
Selecting previously unselected package slirp4netns.
Preparing to unpack .../7-slirp4netns_0.4.3-1_amd64.deb ...
Unpacking slirp4netns (0.4.3-1) ...
Setting up slirp4netns (0.4.3-1) ...
Setting up docker-scan-plugin (0.21.0~ubuntu-focal) ...
Setting up containerd.io (1.6.9-1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/containerd.service → /lib/systemd/system/containerd.service.
Setting up docker-compose-plugin (2.12.2~ubuntu-focal) ...
Setting up docker-ce-cli (5:20.10.21~3-0~ubuntu-focal) ...
Setting up pigz (2.4-1) ...
Setting up docker-ce-rootless-extras (5:20.10.21~3-0~ubuntu-focal) ...
Setting up docker-ce (5:20.10.21~3-0~ubuntu-focal) ...
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /lib/systemd/system/docker.service.
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket → /lib/systemd/system/docker.socket.
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for systemd (245.4-4ubuntu3.17) ...
docker@ip-172-31-28-184:/root$
```

RUNNING BASIC COMMANDS IN DOCKER

1) Check Docker version (Command: $ docker –version)

```
docker@ip-172-31-28-184:/root$ docker --version
Docker version 20.10.21, build baeda1f
docker@ip-172-31-28-184:/root$
```

2) Check Docker version and other information like about engine etc. (Command: $ docker version):

```
docker@ip-172-31-28-184:/root$ docker version
Client: Docker Engine - Community
 Version:           20.10.21
 API version:       1.41
 Go version:        go1.18.7
 Git commit:        baeda1f
 Built:             Tue Oct 25 18:02:21 2022
 OS/Arch:           linux/amd64
 Context:           default
 Experimental:      true

Server: Docker Engine - Community
 Engine:
  Version:          20.10.21
  API version:      1.41 (minimum version 1.12)
  Go version:       go1.18.7
  Git commit:       3056208
  Built:            Tue Oct 25 18:00:04 2022
  OS/Arch:          linux/amd64
  Experimental:     false
 containerd:
  Version:          1.6.9
  GitCommit:        1c90a442489720eec95342e1789ee8a5e1b9536f
 runc:
  Version:          1.1.4
  GitCommit:        v1.1.4-0-g5fd4c4d
 docker-init:
  Version:          0.19.0
  GitCommit:        de40ad0
docker@ip-172-31-28-184:/root$
```

3) Get proper information about the Docker Client (Command: $ docker info):

```
docker@ip-172-31-28-184:/root$ docker info
Client:
 Context:    default
 Debug Mode: false
 Plugins:
  app: Docker App (Docker Inc., v0.9.1-beta3)
  buildx: Docker Buildx (Docker Inc., v0.9.1-docker)
  compose: Docker Compose (Docker Inc., v2.12.2)
  scan: Docker Scan (Docker Inc., v0.21.0)

Server:
 Containers: 0
  Running: 0
  Paused: 0
  Stopped: 0
 Images: 0
 Server Version: 20.10.21
 Storage Driver: overlay2
  Backing Filesystem: extfs
  Supports d_type: true
  Native Overlay Diff: true
  userxattr: false
 Logging Driver: json-file
 Cgroup Driver: cgroupfs
 Cgroup Version: 1
 Plugins:
  Volume: local
  Network: bridge host ipvlan macvlan null overlay
  Log: awslogs fluentd gcplogs gelf journald json-file local logentries splunk syslog
 Swarm: inactive
 Runtimes: runc io.containerd.runc.v2 io.containerd.runtime.v1.linux
 Default Runtime: runc
 Init Binary: docker-init
 containerd version: 1c90a442489720eec95342e1789ee8a5e1b9536f
 runc version: v1.1.4-0-g5fd4c4d
 init version: de40ad0
 Security Options:
  apparmor
  seccomp
   Profile: default
 Kernel Version: 5.15.0-1019-aws
 Operating System: Ubuntu 20.04.5 LTS
 OSType: linux
 Architecture: x86_64
 CPUs: 2
 Total Memory: 3.829GiB
 Name: ip-172-31-28-184
 ID: 644P:6JGA:DBTU:ACEV:NXLI:DXL2:5XT4:2C3U:FHUR:UX6I:PCLA:JQK2
 Docker Root Dir: /var/lib/docker
 Debug Mode: false
 Registry: https://index.docker.io/v1/
 Labels:
 Experimental: false
 Insecure Registries:
  127.0.0.0/8
 Live Restore Enabled: false

docker@ip-172-31-28-184:/root$
```

4) Get help in command line (Command: $ docker help):

```
docker@ip-172-31-28-184:/root$ docker help

Usage:  docker [OPTIONS] COMMAND

A self-sufficient runtime for containers

Options:
      --config string      Location of client config files (default "/home/docker/.docker")
  -c, --context string     Name of the context to use to connect to the daemon (overrides DOCKER_HOST env var and default context set with "docker
                           context use")
  -D, --debug              Enable debug mode
  -H, --host list          Daemon socket(s) to connect to
  -l, --log-level string   Set the logging level ("debug"|"info"|"warn"|"error"|"fatal") (default "info")
      --tls                Use TLS; implied by --tlsverify
      --tlscacert string   Trust certs signed only by this CA (default "/home/docker/.docker/ca.pem")
      --tlscert string     Path to TLS certificate file (default "/home/docker/.docker/cert.pem")
      --tlskey string      Path to TLS key file (default "/home/docker/.docker/key.pem")
      --tlsverify          Use TLS and verify the remote
  -v, --version            Print version information and quit

Management Commands:
  app*        Docker App (Docker Inc., v0.9.1-beta3)
  builder     Manage builds
  buildx*     Docker Buildx (Docker Inc., v0.9.1-docker)
  compose*    Docker Compose (Docker Inc., v2.12.2)
  config      Manage Docker configs
  container   Manage containers
  context     Manage contexts
  image       Manage images
  manifest    Manage Docker image manifests and manifest lists
  network     Manage networks
  node        Manage Swarm nodes
  plugin      Manage plugins
  scan*       Docker Scan (Docker Inc., v0.21.0)
  secret      Manage Docker secrets
  service     Manage services
  stack       Manage Docker stacks
  swarm       Manage Swarm
  system      Manage Docker
  trust       Manage trust on Docker images
  volume      Manage volumes
```

```
Commands:
  attach      Attach local standard input, output, and error streams to a running container
  build       Build an image from a Dockerfile
  commit      Create a new image from a container's changes
  cp          Copy files/folders between a container and the local filesystem
  create      Create a new container
  diff        Inspect changes to files or directories on a container's filesystem
  events      Get real time events from the server
  exec        Run a command in a running container
  export      Export a container's filesystem as a tar archive
  history     Show the history of an image
  images      List images
  import      Import the contents from a tarball to create a filesystem image
  info        Display system-wide information
  inspect     Return low-level information on Docker objects
  kill        Kill one or more running containers
  load        Load an image from a tar archive or STDIN
  login       Log in to a Docker registry
  logout      Log out from a Docker registry
  logs        Fetch the logs of a container
  pause       Pause all processes within one or more containers
  port        List port mappings or a specific mapping for the container
  ps          List containers
  pull        Pull an image or a repository from a registry
  push        Push an image or a repository to a registry
  rename      Rename a container
  restart     Restart one or more containers
  rm          Remove one or more containers
  rmi         Remove one or more images
  run         Run a command in a new container
  save        Save one or more images to a tar archive (streamed to STDOUT by default)
  search      Search the Docker Hub for images
  start       Start one or more stopped containers
  stats       Display a live stream of container(s) resource usage statistics
  stop        Stop one or more running containers
  tag         Create a tag TARGET_IMAGE that refers to SOURCE_IMAGE
  top         Display the running processes of a container
  unpause     Unpause all processes within one or more containers
  update      Update configuration of one or more containers
```

```
  version     Show the Docker version information
  wait        Block until one or more containers stop, then print their exit codes

Run 'docker COMMAND --help' for more information on a command.

To get more help with docker, check out our guides at https://docs.docker.com/go/guides/
docker@ip-172-31-28-184:/root$
```

5) List the running containers (Command: $ docker ps):

```
docker@ip-172-31-28-184:/root$ docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
docker@ip-172-31-28-184:/root$
```

6) List running and exited containers (Command: $ docker ps -a):

```
docker@ip-172-31-28-184:/root$ docker ps -a
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
docker@ip-172-31-28-184:/root$
```

7) Login to docker hub (Command: $ docker login):

```
docker@ip-172-31-28-184:/root$ 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: priyankaceq
Password:
WARNING! Your password will be stored unencrypted in /home/docker/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
docker@ip-172-31-28-184:/root$
```

8) Add image to our machine (Command: $ docker run imagename):

```
docker@ip-172-31-28-184:/root$ docker run mysql
Unable to find image 'mysql:latest' locally
latest: Pulling from library/mysql
50cbc88660a5: Pull complete
92ca853f7184: Pull complete
9a2047696230: Pull complete
fe3fea56f9fb: Pull complete
b058249d3104: Pull complete
9d5014a20163: Pull complete
906aa7388ee2: Pull complete
86b5e2150967: Pull complete
7c6b15dcdf4e: Pull complete
21de4337b977: Pull complete
35dab154f2ae: Pull complete
Digest: sha256:06314a7a220f6043436cfd72fd9c7f174fd58ef69fe4b788625fa53be4ab66aa
Status: Downloaded newer image for mysql:latest
2022-10-26 05:49:35+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 8.0.31-1.el8 started.
2022-10-26 05:49:35+00:00 [Note] [Entrypoint]: Switching to dedicated user 'mysql'
2022-10-26 05:49:35+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 8.0.31-1.el8 started.
2022-10-26 05:49:35+00:00 [ERROR] [Entrypoint]: Database is uninitialized and password option is not specified
    You need to specify one of the following as an environment variable:
    - MYSQL_ROOT_PASSWORD
    - MYSQL_ALLOW_EMPTY_PASSWORD
    - MYSQL_RANDOM_ROOT_PASSWORD
docker@ip-172-31-28-184:/root$
```

9) Get images in our machine (Command: $ docker images):

```
docker@ip-172-31-28-184:/root$ docker images
REPOSITORY   TAG       IMAGE ID       CREATED       SIZE
mysql        latest    8fad08b3c84b   4 days ago    535MB
docker@ip-172-31-28-184:/root$
```

10) Forcefully remove the image from the machine (Command: $ docker rmi -f imagename):

```
docker@ip-172-31-28-184:/root$ docker rmi -f mysql
Untagged: mysql:latest
Untagged: mysql@sha256:06314a7a220f6043436cfd72fd9c7f174fd58ef69fe4b788625fa53be4ab66aa
Deleted: sha256:8fad08b3c84be3e9164f86153224ab616bf71ee2c79677154c2e5cd3179cccfe
docker@ip-172-31-28-184:/root$
```

```
docker@ip-172-31-28-184:/root$ docker images
REPOSITORY   TAG       IMAGE ID   CREATED   SIZE
docker@ip-172-31-28-184:/root$
```

11) Logout of docker hub (Command: $ docker logout):

```
docker@ip-172-31-28-184:/root$ docker logout
Removing login credentials for https://index.docker.io/v1/
docker@ip-172-31-28-184:/root$
```