Docker Task

TASKS

Work Flow:

- Create an EC2 instance with the help of AWS Management Console with linux OS of required configuration.
- Now, Connect an EC2 instance with an help of Windows Terminal or Gitbash or Vbox.
- To connect an EC2 instance the command is:
  - ssh -i “**key_file**” ec2-user@“**Public_IP_address**”

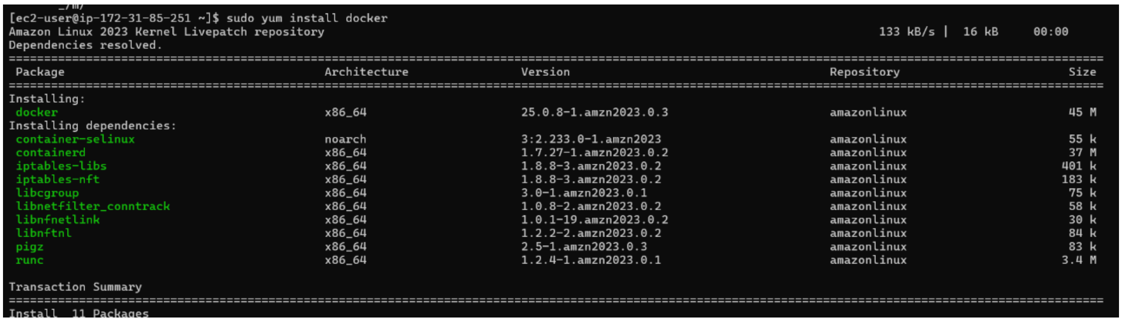
Key_file --- Key file of the instance with the extension .pem

Public_IP_address --- Public IP address of the instance.

1. Install docker on EC2 and explore the docker commands (docker images, containers, volumes, network)

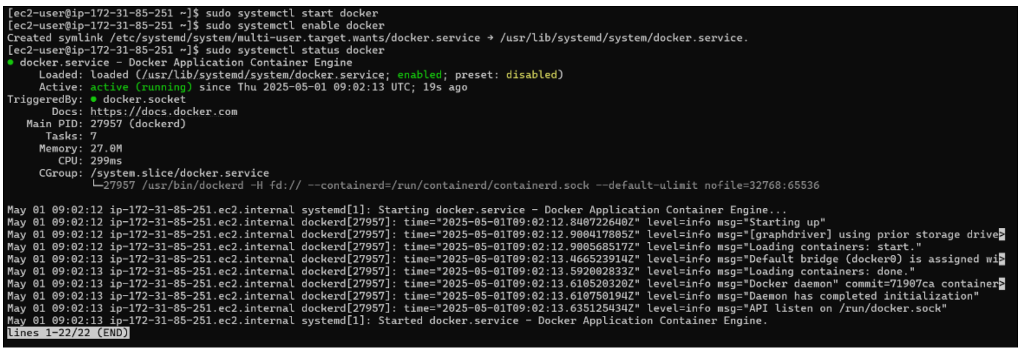
Step 1: Install an Docker in an EC2 instance

✓ To install an docker in linux machine, the command is:
- **sudo yum install docker**



✓ To start and enable an docker service, The command is:
- **sudo systemctl start docker**
- **sudo systemctl enable docker**

✓ To check the status of the docker service, The command is:
- **sudo systemctl status docker**



✓ To add an ec2-user to docker group, the command is:
- **sudo usermod -aG docker ec2-user**

✓ To check an version of the docker and to verify an installation, the command is:

- **docker --version**

```
[ec2-user@ip-172-31-85-251 ~]$ docker --version
Docker version 25.0.8, build 0bab007
```

Docker Image:

✓ The command to downloads the latest nginx image from Docker Hub, The command is:

- **docker pull nginx**

```
[ec2-user@ip-172-31-85-251 ~]$ sudo docker pull nginx
Using default tag: latest
latest: Pulling from library/nginx
254e724d7786: Pull complete
913115292750: Pull complete
3e544d53ce49: Pull complete
4f21ed9ac0c0: Pull complete
d38f2ef2d6f2: Pull complete
40a6e9f4e456: Pull complete
d3dc5ec71e9d: Pull complete
Digest: sha256:c15da6c91de8d2f436196f3a768483ad32c258ed4e1beb3d367a27ed67253e66
Status: Downloaded newer image for nginx:latest
docker.io/library/nginx:latest
[ec2-user@ip-172-31-85-251 ~]$
```

✓ If you give an command without mentioning an tag(version) it will download an latest tag(version) of the service from the docker hub.

✓ To list the images which is locally available in your machine , the command is:

- **docker image ls**

```
[ec2-user@ip-172-31-85-251 ~]$ docker image ls
REPOSITORY   TAG       IMAGE ID       CREATED       SIZE
nginx        latest    a830707172e8   2 weeks ago   192MB
```

✓ To copy and creating an new image from an existing image, The command is:

- **docker tag nginx:latest webserver:v1**

```
[ec2-user@ip-172-31-85-251 ~]$ docker tag nginx:latest webserver:v1
[ec2-user@ip-172-31-85-251 ~]$ docker image ls
REPOSITORY   TAG       IMAGE ID       CREATED       SIZE
nginx        latest    a830707172e8   2 weeks ago   192MB
webserver    v1        a830707172e8   2 weeks ago   192MB
```

✓ To remove or delete an particular image locally, The command is:

- **docker image rm nginx:latest**

```
[ec2-user@ip-172-31-85-251 ~]$ docker image rm webserver:v1
Untagged: webserver:v1
[ec2-user@ip-172-31-85-251 ~]$ docker image ls
REPOSITORY   TAG       IMAGE ID       CREATED       SIZE
nginx        latest    a830707172e8   2 weeks ago   192MB
```

Docker Container:

- ✓ To create and run an container from image, The command is:
  - **docker run --name webserver httpd**

```
[ec2-user@ip-172-31-85-251 ~]$ docker run --name webserver httpd
AH00558: httpd: Could not reliably determine the server's fully qualified domain name, using 172.17.0.2. Set the 'ServerName' directive globally to suppress
 this message
AH00558: httpd: Could not reliably determine the server's fully qualified domain name, using 172.17.0.2. Set the 'ServerName' directive globally to suppress
 this message
[Sat May 03 16:52:08.233368 2025] [mpm_event:notice] [pid 1:tid 1] AH00489: Apache/2.4.63 (Unix) configured -- resuming normal operations
[Sat May 03 16:52:08.233889 2025] [core:notice] [pid 1:tid 1] AH00094: Command line: 'httpd -D FOREGROUND'
```

- ✓ To list the container which are running, The command is:
  - **docker container ls**

```
[ec2-user@ip-172-31-85-251 ~]$ docker container ls
CONTAINER ID   IMAGE     COMMAND              CREATED          STATUS          PORTS                                   NAMES
62391d60518a   httpd     "httpd-foreground"   22 minutes ago   Up 22 minutes   0.0.0.0:80->80/tcp, :::80->80/tcp   webserver3
8b57d9edf3ba   httpd     "httpd-foreground"   25 minutes ago   Up 25 minutes   80/tcp                              musing_swartz
80a1478d033b   httpd     "httpd-foreground"   30 minutes ago   Up 11 seconds   80/tcp                              webserver
```

- ✓ To list the all container no matter about the status(running /stopped), The command is:
  - **docker container ls -a**

```
[ec2-user@ip-172-31-85-251 ~]$ docker container ls -a
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS                      PORTS     NAMES
065de5f0bbc4   nginx     "/docker-entrypoint.…"   29 seconds ago   Exited (0) 13 seconds ago             webserver1
80a1478d033b   httpd     "httpd-foreground"       4 minutes ago    Exited (0) 3 minutes ago              webserver
a9bbed43831d   httpd     "httpd-foreground"       13 minutes ago   Exited (0) 13 minutes ago             clever_hugle
```

- ✓ To run an container in an background, The command is:
  - **docker run -d httpd**

```
[ec2-user@ip-172-31-85-251 ~]$ docker run -d httpd
8b57d9edf3bad3c96ee0f086c97165b342a8475938f4c9f12aeab9ed7ede97ae
```

- ✓ To run a container with port mapping, The command is:
  - **docker run --name webserver3 -p 80:80 -d httpd**

```
[ec2-user@ip-172-31-85-251 ~]$ docker run --name webserver3 -p 80:80 -d httpd
62391d60518a3383c780d82a58359e1594c53c777638643bb023e57a369dc8a2
[ec2-user@ip-172-31-85-251 ~]$ docker container ls -a
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS                      PORTS                                   NAMES
62391d60518a   httpd     "httpd-foreground"       21 seconds ago   Up 20 seconds               0.0.0.0:80->80/tcp, :::80->80/tcp   webserver3
8b57d9edf3ba   httpd     "httpd-foreground"       3 minutes ago    Up 3 minutes                80/tcp                              musing_swartz
065de5f0bbc4   nginx     "/docker-entrypoint.…"   4 minutes ago    Exited (0) 4 minutes ago                                        webserver1
80a1478d033b   httpd     "httpd-foreground"       8 minutes ago    Exited (0) 8 minutes ago                                        webserver
a9bbed43831d   httpd     "httpd-foreground"       17 minutes ago   Exited (0) 17 minutes ago                                       clever_hugle
```

- ✓ To open an shell / login inside an running container, The command is:
  - **docker exec -it webserver bash**

```
[ec2-user@ip-172-31-85-251 ~]$ docker exec -it webserver bash
root@80a1478d033b:/usr/local/apache2#
```

- ✓ To start the container locally, The command is:
  - **docker container start webserver**

```
[ec2-user@ip-172-31-85-251 ~]$ docker container start webserver
webserver
[ec2-user@ip-172-31-85-251 ~]$ docker container ls
CONTAINER ID   IMAGE     COMMAND              CREATED          STATUS          PORTS                                   NAMES
62391d60518a   httpd     "httpd-foreground"   22 minutes ago   Up 22 minutes   0.0.0.0:80->80/tcp, :::80->80/tcp   webserver3
8b57d9edf3ba   httpd     "httpd-foreground"   25 minutes ago   Up 25 minutes   80/tcp                              musing_swartz
80a1478d033b   httpd     "httpd-foreground"   30 minutes ago   Up 11 seconds   80/tcp                              webserver
```

✓ To stop the running container locally, The command is:

- **docker container stop webserver**

```
[ec2-user@ip-172-31-85-251 ~]$ docker container stop webserver
webserver
[ec2-user@ip-172-31-85-251 ~]$ docker container ls
CONTAINER ID   IMAGE     COMMAND              CREATED          STATUS          PORTS                                   NAMES
62391d60518a   httpd     "httpd-foreground"   23 minutes ago   Up 23 minutes   0.0.0.0:80->80/tcp, :::80->80/tcp   webserver3
8b57d9edf3ba   httpd     "httpd-foreground"   26 minutes ago   Up 26 minutes   80/tcp                              musing_swartz
```

✓ To fetch and follow the logs of a container., The command is:

- **docker logs -f webserver**

```
[ec2-user@ip-172-31-85-251 ~]$ docker logs -f webserver
AH00558: httpd: Could not reliably determine the server's fully qualified domain name, using 172.17.0.2. Set the 'ServerName' directive globally to suppress
 this message
AH00558: httpd: Could not reliably determine the server's fully qualified domain name, using 172.17.0.2. Set the 'ServerName' directive globally to suppress
 this message
[Sat May 03 16:52:08.233368 2025] [mpm_event:notice] [pid 1:tid 1] AH00489: Apache/2.4.63 (Unix) configured -- resuming normal operations
[Sat May 03 16:52:08.233889 2025] [core:notice] [pid 1:tid 1] AH00094: Command line: 'httpd -D FOREGROUND'
[Sat May 03 16:52:50.432815 2025] [mpm_event:notice] [pid 1:tid 1] AH00491: caught SIGTERM, shutting down
AH00558: httpd: Could not reliably determine the server's fully qualified domain name, using 172.17.0.4. Set the 'ServerName' directive globally to suppress
 this message
AH00558: httpd: Could not reliably determine the server's fully qualified domain name, using 172.17.0.4. Set the 'ServerName' directive globally to suppress
 this message
[Sat May 03 17:22:35.481937 2025] [mpm_event:notice] [pid 1:tid 1] AH00489: Apache/2.4.63 (Unix) configured -- resuming normal operations
[Sat May 03 17:22:35.482329 2025] [core:notice] [pid 1:tid 1] AH00094: Command line: 'httpd -D FOREGROUND'
[Sat May 03 17:23:56.084640 2025] [mpm_event:notice] [pid 1:tid 1] AH00492: caught SIGWINCH, shutting down gracefully
```

✓ To inspect a running container, The command is:

- **docker inspect webserver3**

```
[ec2-user@ip-172-31-85-251 ~]$ docker inspect webserver3
[
    {
        "Id": "62391d60518a3383c780d82a58359e1594c53c777638643bb023e57a369dc8a2",
        "Created": "2025-05-03T17:00:45.558701063Z",
        "Path": "httpd-foreground",
        "Args": [],
        "State": {
            "Status": "running",
            "Running": true,
            "Paused": false,
            "Restarting": false,
            "OOMKilled": false,
            "Dead": false,
            "Pid": 136856,
            "ExitCode": 0,
            "Error": "",
            "StartedAt": "2025-05-03T17:00:45.952505411Z",
            "FinishedAt": "0001-01-01T00:00:00Z"
        },
        "Image": "sha256:0208f149a449a6a055f08add91cbbc004e486f78bf02a76554ff3cd235ced6fe",
        "ResolvConfPath": "/var/lib/docker/containers/62391d60518a3383c780d82a58359e1594c53c777638643bb023e57a369dc8a2/resolv.conf",
        "HostnamePath": "/var/lib/docker/containers/62391d60518a3383c780d82a58359e1594c53c777638643bb023e57a369dc8a2/hostname",
        "HostsPath": "/var/lib/docker/containers/62391d60518a3383c780d82a58359e1594c53c777638643bb023e57a369dc8a2/hosts",
        "LogPath": "/var/lib/docker/containers/62391d60518a3383c780d82a58359e1594c53c777638643bb023e57a369dc8a2/62391d60518a3383c780d82a58359e1594c53c777638
643bb023e57a369dc8a2-json.log",
        "Name": "/webserver3",
        "RestartCount": 0,
        "Driver": "overlay2",
        "Platform": "linux",
        "MountLabel": "",
        "ProcessLabel": "",
        "AppArmorProfile": "",
        "ExecIDs": null,
        "HostConfig": {
            "Binds": null,
            "ContainerIDFile": "",
            "LogConfig": {
                "Type": "json-file",
                "Config": {}
            },
```

✓ To view resource usage stats,The command is:

- **docker container stats**

```
CONTAINER ID   NAME           CPU %     MEM USAGE / LIMIT     MEM %     NET I/O       BLOCK I/O    PIDS
62391d60518a   webserver3     0.00%     6.07MiB / 949.4MiB    0.64%     1.08kB / 0B   0B / 4.1kB   82
8b57d9edf3ba   musing_swartz  0.01%     6.102MiB / 949.4MiB   0.64%     1.3kB / 0B    0B / 4.1kB   82
```

✓ To remove an container locally, The command is:

- **docker container rm web**

```
[ec2-user@ip-172-31-85-251 ~]$ docker container ls -a
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS                      PORTS     NAMES
62391d60518a   httpd     "httpd-foreground"       32 minutes ago   Exited (0) 7 seconds ago              webserver3
8b57d9edf3ba   httpd     "httpd-foreground"       36 minutes ago   Up 36 minutes               80/tcp    musing_swartz
065de5f0bbc4   nginx     "/docker-entrypoint.…"   37 minutes ago   Exited (0) 37 minutes ago             webserver1
80a1478d033b   httpd     "httpd-foreground"       41 minutes ago   Exited (0) 9 minutes ago              webserver
a9bbed43831d   httpd     "httpd-foreground"       50 minutes ago   Exited (0) 50 minutes ago             clever_hugle
[ec2-user@ip-172-31-85-251 ~]$ docker container rm webserver3
webserver3
[ec2-user@ip-172-31-85-251 ~]$ docker container ls -a
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS                      PORTS     NAMES
8b57d9edf3ba   httpd     "httpd-foreground"       36 minutes ago   Up 36 minutes               80/tcp    musing_swartz
065de5f0bbc4   nginx     "/docker-entrypoint.…"   37 minutes ago   Exited (0) 37 minutes ago             webserver1
80a1478d033b   httpd     "httpd-foreground"       41 minutes ago   Exited (0) 9 minutes ago              webserver
a9bbed43831d   httpd     "httpd-foreground"       50 minutes ago   Exited (0) 50 minutes ago             clever_hugle
[ec2-user@ip-172-31-85-251 ~]$
```

Docker Volumes

- ✓ To create an new docker volume, The command is:
  - **docker volume create myvol**

```
[ec2-user@ip-172-31-85-251 ~]$ docker volume create myvol
myvol
```

- ✓ To list all the docker volume, The command is:
  - **docker volume ls**

```
[ec2-user@ip-172-31-85-251 ~]$ docker volume create myvol
myvol
[ec2-user@ip-172-31-85-251 ~]$ docker volume ls
DRIVER    VOLUME NAME
local     myvol
```

- ✓ To remove / delete an volume, The command is:
  - **docker volume rm myvol**

```
[ec2-user@ip-172-31-85-251 ~]$ docker volume rm myvol
myvol
[ec2-user@ip-172-31-85-251 ~]$ docker volume ls
DRIVER    VOLUME NAME
```

- ✓ To display the information about the volume, The command is:
  - **docker volume inspect myvol**

```
[ec2-user@ip-172-31-85-251 ~]$ docker volume inspect myvol
[
    {
        "CreatedAt": "2025-05-03T17:35:56Z",
        "Driver": "local",
        "Labels": null,
        "Mountpoint": "/var/lib/docker/volumes/myvol/_data",
        "Name": "myvol",
        "Options": null,
        "Scope": "local"
    }
]
```

- ✓ To run a volume with mapping, The command is:
  - **docker run -v /opt/backup_dir:/var/lib/httpd httpd**

```
[ec2-user@ip-172-31-85-251 ~]$ docker run -v /opt/backup_dir:/var/lib/httpd httpd
AH00558: httpd: Could not reliably determine the server's fully qualified domain name, using 172.17.0.3. Set the 'ServerName' directive globally to suppress
 this message
AH00558: httpd: Could not reliably determine the server's fully qualified domain name, using 172.17.0.3. Set the 'ServerName' directive globally to suppress
 this message
[Sat May 03 17:39:32.100250 2025] [mpm_event:notice] [pid 1:tid 1] AH00489: Apache/2.4.63 (Unix) configured -- resuming normal operations
[Sat May 03 17:39:32.100551 2025] [core:notice] [pid 1:tid 1] AH00094: Command line: 'httpd -D FOREGROUND'
```

Docker Network

- ✓ To create an new docker network, The command is:
  - **docker network create mynetwork**

```
[ec2-user@ip-172-31-85-251 ~]$ docker network create mynetwork
ad04ea710ee1d64bdec83b7a9ed6c355b6aaf3f4e639dfb82490f09eba3707e3
```

- ✓ To connect an container to an network, The command is:
  - **docker network connect mynetwork <cont_id>**

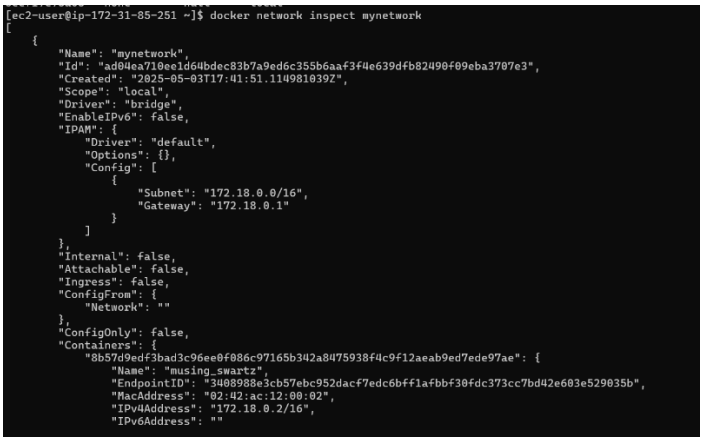
```
[ec2-user@ip-172-31-85-251 ~]$ docker network connect mynetwork 8b57d9edf3ba
[ec2-user@ip-172-31-85-251 ~]$ docker network ls
NETWORK ID     NAME        DRIVER    SCOPE
f18eaf4010e1   bridge      bridge    local
f85f195238b3   host        host      local
ad04ea710ee1   mynetwork   bridge    local
5ecf17c78a08   none        null      local
[ec2-user@ip-172-31-85-251 ~]$ docker network inspect mynetwork
[
    {
        "Name": "mynetwork",
        "Id": "ad04ea710ee1d64bdec83b7a9ed6c355b6aaf3f4e639dfb82490f09eba3707e3",
        "Created": "2025-05-03T17:41:51.114981039Z",
        "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": false,
        "Ingress": false,
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": {
            "8b57d9edf3bad3c96ee0f086c97165b342a8475938f4c9f12aeab9ed7ede97ae": {
                "Name": "musing_swartz",
                "EndpointID": "3408988e3cb57ebc952dacf7edc6bff1afbbf30fdc373cc7bd42e603e529035b",
                "MacAddress": "02:42:ac:12:00:02",
                "IPv4Address": "172.18.0.2/16",
                "IPv6Address": ""
```

- ✓ To disconnect a container from a network, The command is:
  - **docker network disconnect mynetwork <cont_id>**

```
[ec2-user@ip-172-31-85-251 ~]$ docker network disconnect mynetwork 8b57d9edf3ba
[ec2-user@ip-172-31-85-251 ~]$ docker network inspect mynetwork
[
    {
        "Name": "mynetwork",
        "Id": "ad04ea710ee1d64bdec83b7a9ed6c355b6aaf3f4e639dfb82490f09eba3707e3",
        "Created": "2025-05-03T17:41:51.114981039Z",
        "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": false,
        "Ingress": false,
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": {},
        "Options": {},
        "Labels": {}
    }
]
```

✓ To display information about a Docker network, The command is:

- **docker network inspect mynetwork**



✓ To list all the network, The command is:

- **docker network ls**

```
[ec2-user@ip-172-31-85-251 ~]$ docker network ls
NETWORK ID     NAME         DRIVER    SCOPE
f18eaf4010e1   bridge       bridge    local
f85f195238b3   host         host      local
ad04ea710ee1   mynetwork    bridge    local
5ecf17c78a08   none         null      local
```

✓ To remove one or more networks, The command is:

- **docker network rm mynetwork**

```
[ec2-user@ip-172-31-85-251 ~]$ docker network rm mynetwork
mynetwork
[ec2-user@ip-172-31-85-251 ~]$ docker network ls
NETWORK ID     NAME      DRIVER    SCOPE
f18eaf4010e1   bridge    bridge    local
f85f195238b3   host      host      local
5ecf17c78a08   none      null      local
```

********** **TASK COMPLETED** **********