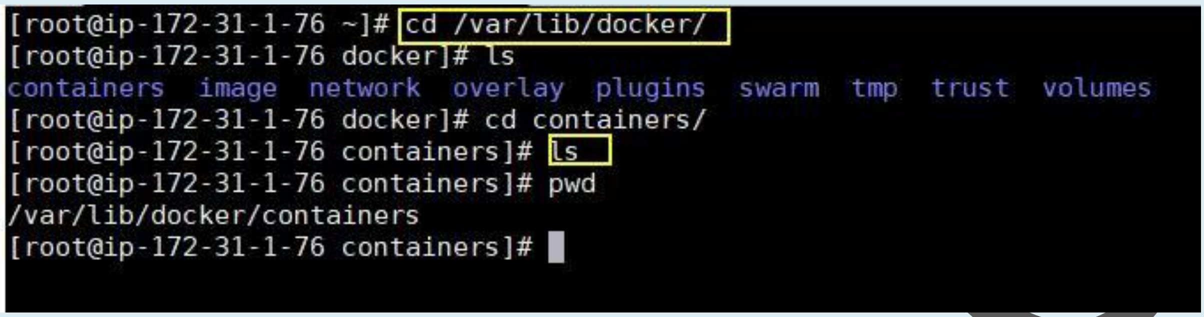
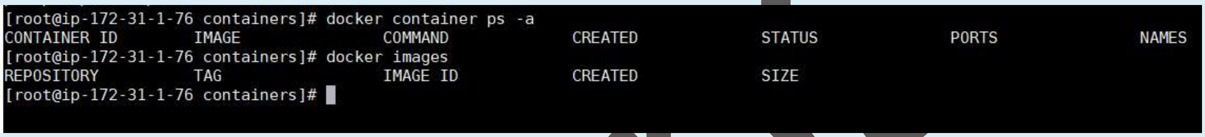
Step 1 : Observe the directory Before creation of container

```
ls -l /var/lib/docker/containers/
```



```
[root@ip-172-31-1-76 ~]# cd /var/lib/docker/
[root@ip-172-31-1-76 docker]# ls
containers  image  network  overlay  plugins  swarm  tmp  trust  volumes
[root@ip-172-31-1-76 docker]# cd containers/
[root@ip-172-31-1-76 containers]# ls
[root@ip-172-31-1-76 containers]# pwd
/var/lib/docker/containers
[root@ip-172-31-1-76 containers]#
```

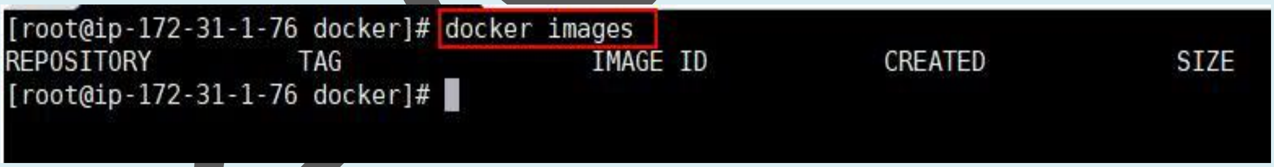
In below diagram we can see no docker images or container is present



```
[root@ip-172-31-1-76 containers]# docker container ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS               NAMES
[root@ip-172-31-1-76 containers]# docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
[root@ip-172-31-1-76 containers]#
```

Step 1 : Listing the Docker images

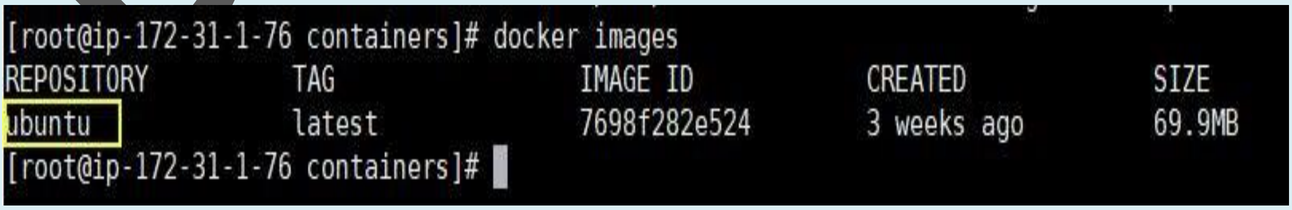
```
docker images
```



```
[root@ip-172-31-1-76 docker]# docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
[root@ip-172-31-1-76 docker]#
```

Step 1.A: Pulling docker images from dockerhub

```
docker pull ubuntu
```



```
[root@ip-172-31-1-76 containers]# docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
ubuntu              latest              7698f282e524        3 weeks ago         69.9MB
[root@ip-172-31-1-76 containers]#
```

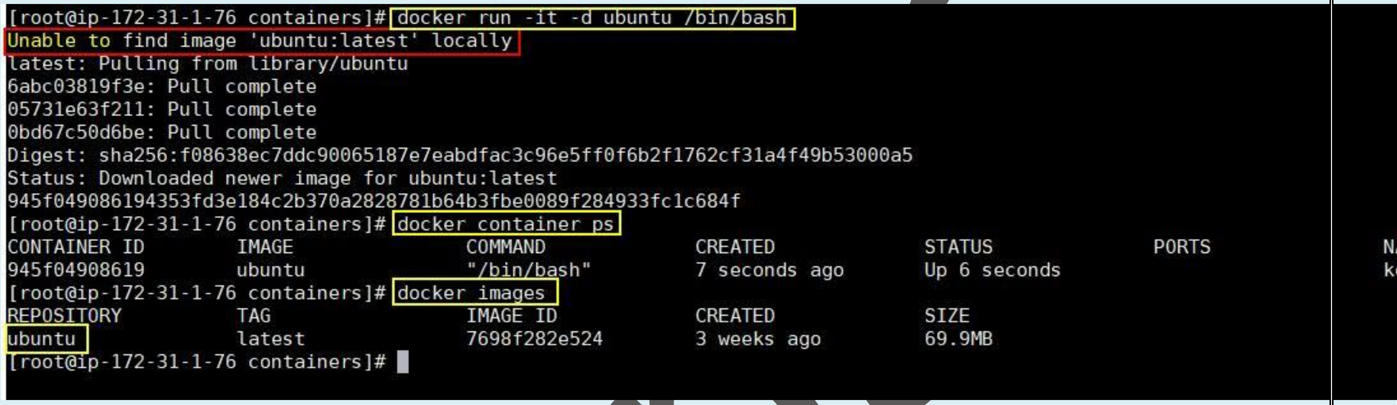
Step 2: Deploy a ubuntu server container

```
docker run -it -d ubuntu /bin/bash
```

i → interactive

t → terminal or tty

d → detached mode or daemon



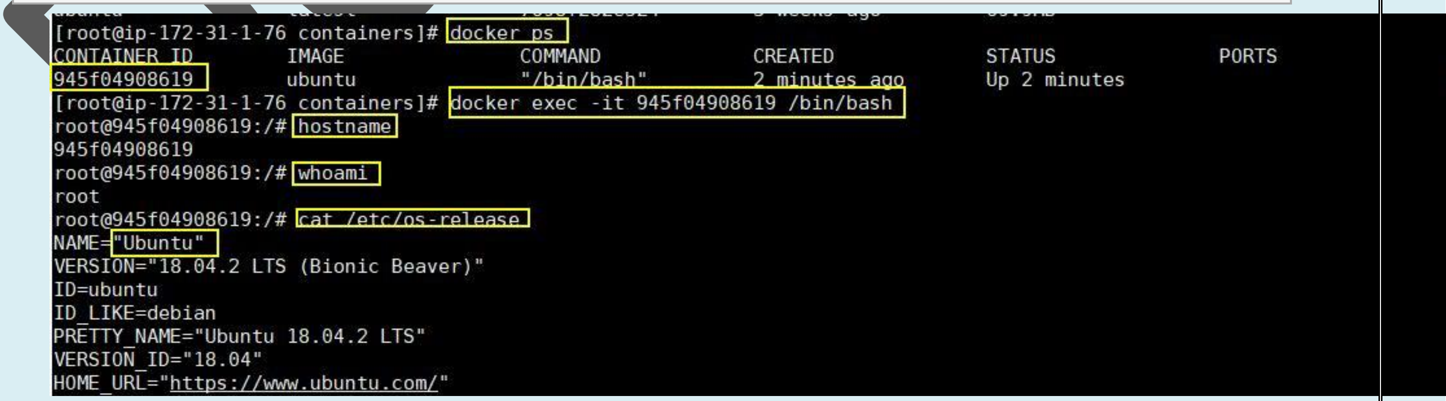
```
[root@ip-172-31-1-76 containers]# docker run -it -d ubuntu /bin/bash
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
6abc03819f3e: Pull complete
05731e63f211: Pull complete
0bd67c50d6be: Pull complete
Digest: sha256:f08638ec7ddc90065187e7eabdfac3c96e5ff0f6b2f1762cf31a4f49b53000a5
Status: Downloaded newer image for ubuntu:latest
945f049086194353fd3e184c2b370a2828781b64b3fbe0089f284933fc1c684f
[root@ip-172-31-1-76 containers]# docker container ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS               NA
945f04908619        ubuntu              "/bin/bash"         7 seconds ago       Up 6 seconds                            k
[root@ip-172-31-1-76 containers]# docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
ubuntu              latest              7698f282e524        3 weeks ago         69.9MB
[root@ip-172-31-1-76 containers]# █
```

Step 2.a: list the running container

```
docker ps
```

Step 2.b: move inside container

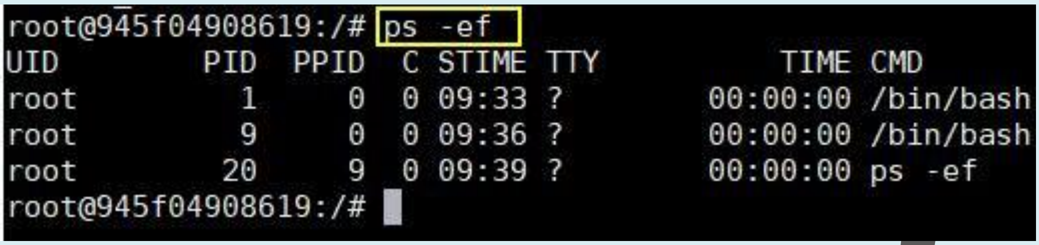
```
docker exec -it  <containerid> /bin/bash
```



```
[root@ip-172-31-1-76 containers]# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS
945f04908619        ubuntu              "/bin/bash"         2 minutes ago       Up 2 minutes
[root@ip-172-31-1-76 containers]# docker exec -it 945f04908619 /bin/bash
root@945f04908619:/# hostname
945f04908619
root@945f04908619:/# whoami
root
root@945f04908619:/# cat /etc/os-release
NAME="Ubuntu"
VERSION="18.04.2 LTS (Bionic Beaver)"
ID=ubuntu
ID_LIKE=debian
PRETTY_NAME="Ubuntu 18.04.2 LTS"
VERSION_ID="18.04"
HOME_URL="https://www.ubuntu.com/"
```

Step 2.C: list process running in container

```
root@04707a043ac9:/# ps -ef
```



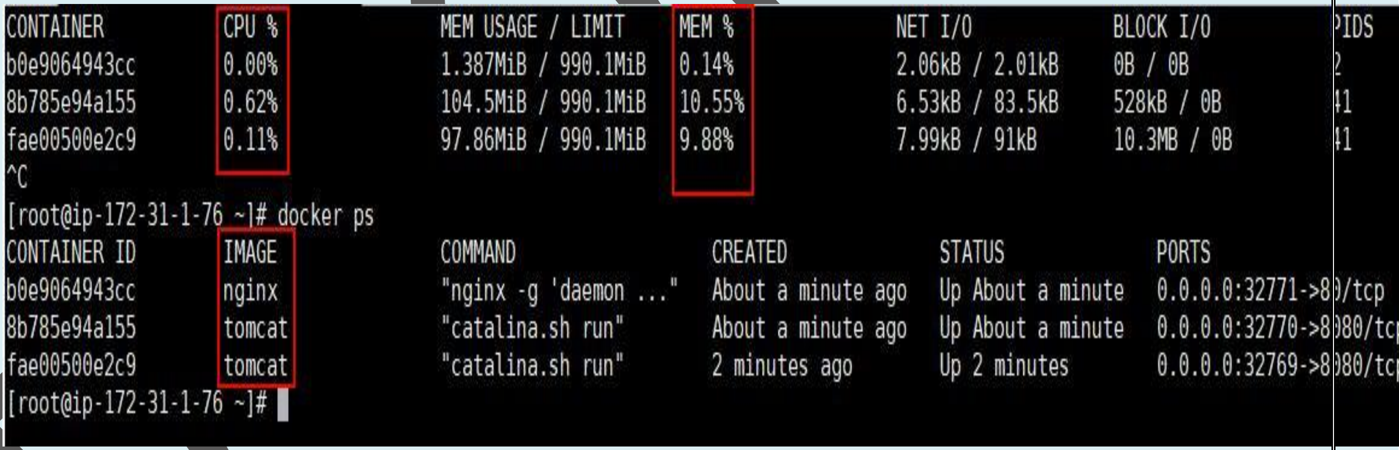
```
root@945f04908619:/# ps -ef
UID         PID  PPID  C STIME TTY          TIME CMD
root          1     0  0 09:33 ?        00:00:00 /bin/bash
root          9     0  0 09:36 ?        00:00:00 /bin/bash
root         20     9  0 09:39 ?        00:00:00 ps -ef
root@945f04908619:/#
```

Step 3 : exit from the container

```
root@04707a043ac9:/# exit
```

**** Note: - don't use exit command to move out of container as this will kill the container if no process runs, however make use of "ctrl+p+q" to come out of container**

Step 4: We can use the command "docker stats" to get the memory and cpu usage by container, here we are looking into resource usage by nginx and tomcat



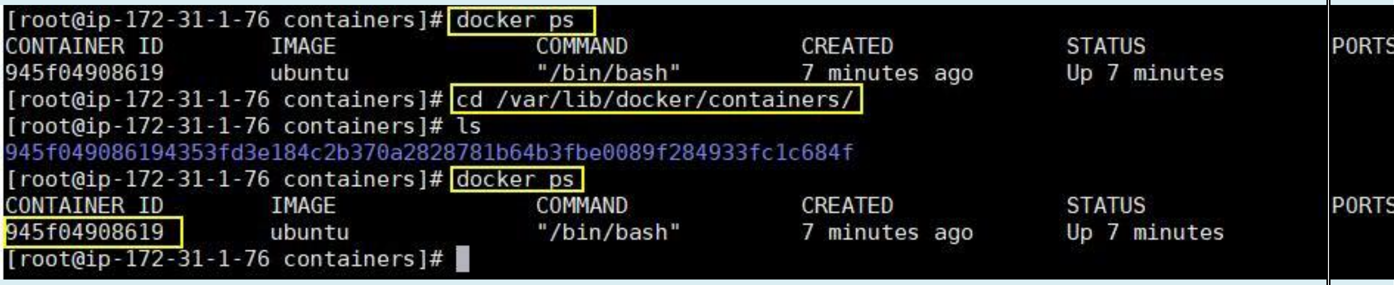
```
CONTAINER           CPU %               MEM USAGE / LIMIT     MEM %               NET I/O             BLOCK I/O           PIDS
b0e9064943cc        0.00%               1.387MiB / 990.1MiB   0.14%               2.06kB / 2.01kB     0B / 0B             2
8b785e94a155        0.62%               104.5MiB / 990.1MiB   10.55%              6.53kB / 83.5kB     528kB / 0B          41
fae00500e2c9        0.11%               97.86MiB / 990.1MiB   9.88%               7.99kB / 91kB       10.3MB / 0B         41
^C
[root@ip-172-31-1-76 ~]# docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED              STATUS              PORTS
b0e9064943cc        nginx               "nginx -g 'daemon ..."   About a minute ago   Up About a minute   0.0.0.0:32771->80/tcp
8b785e94a155        tomcat              "catalina.sh run"        About a minute ago   Up About a minute   0.0.0.0:32770->8080/tc
fae00500e2c9        tomcat              "catalina.sh run"        2 minutes ago        Up 2 minutes        0.0.0.0:32769->8080/tc
[root@ip-172-31-1-76 ~]#
```

Docker takes any random name from docker hub to run container, it doesn't mean the name given that user has created the image, and we can also give our name with below command: -

docker run -itd --name amit ubuntu

Step 4 : Observe the directory After creation of container on docker host login

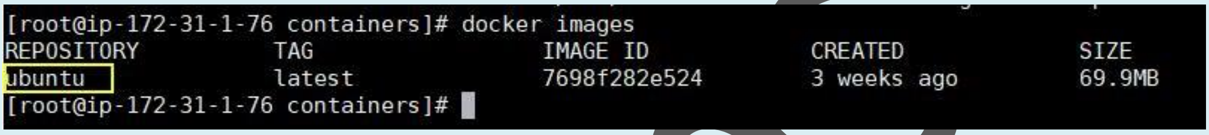
```
ls -l /var/lib/docker/containers/
```



```
[root@ip-172-31-1-76 containers]# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS
945f04908619        ubuntu              "/bin/bash"         7 minutes ago       Up 7 minutes
[root@ip-172-31-1-76 containers]# cd /var/lib/docker/containers/
[root@ip-172-31-1-76 containers]# ls
945f049086194353fd3e184c2b370a2828781b64b3fbe0089f284933fc1c684f
[root@ip-172-31-1-76 containers]# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS
945f04908619        ubuntu              "/bin/bash"         7 minutes ago       Up 7 minutes
[root@ip-172-31-1-76 containers]# 
```

Step 5: Listing the Docker images

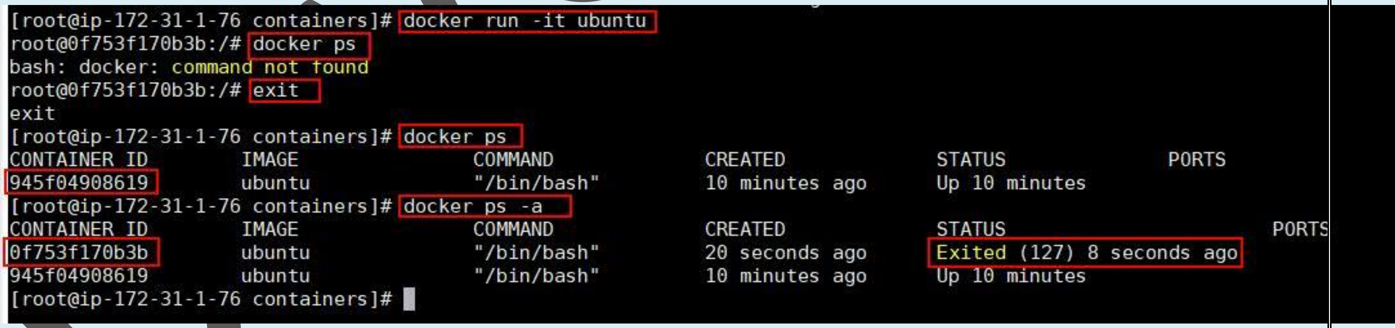
```
docker images
```



```
[root@ip-172-31-1-76 containers]# docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
ubuntu              latest              7698f282e524        3 weeks ago         69.9MB
[root@ip-172-31-1-76 containers]# 
```

Step 6: List the running containers

```
docker ps
```

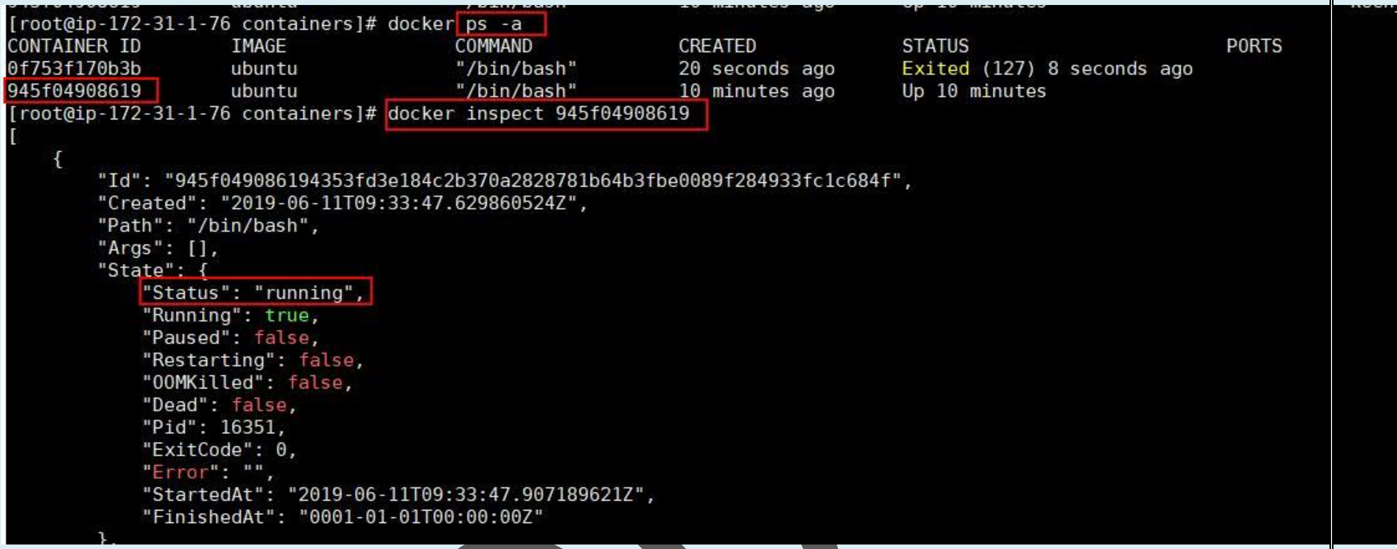


```
[root@ip-172-31-1-76 containers]# docker run -it ubuntu
root@0f753f170b3b:/# docker ps
bash: docker: command not found
root@0f753f170b3b:/# exit
exit
[root@ip-172-31-1-76 containers]# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS
945f04908619        ubuntu              "/bin/bash"         10 minutes ago      Up 10 minutes
[root@ip-172-31-1-76 containers]# docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS                          PORTS
0f753f170b3b        ubuntu              "/bin/bash"         20 seconds ago      Exited (127) 8 seconds ago
945f04908619        ubuntu              "/bin/bash"         10 minutes ago      Up 10 minutes
[root@ip-172-31-1-76 containers]# 
```

First we made a container without running it in detached mode so container created and immediately we came inside container, but when we moved out of container using command "exit", immediately the container got down and went in exited status, so if we are using docker ps , that died container we are not able to see however we can see total list of container using command "docker ps -a"

Step 7 : **Fetch info about container**

```
docker inspect 945f04908619
```



```
[root@ip-172-31-1-76 containers]# docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS                        PORTS
0f753f170b3b        ubuntu              "/bin/bash"         20 seconds ago      Exited (127) 8 seconds ago
945f04908619        ubuntu              "/bin/bash"         10 minutes ago      Up 10 minutes
[root@ip-172-31-1-76 containers]# docker inspect 945f04908619
[
    {
        "Id": "945f049086194353fd3e184c2b370a2828781b64b3fbe0089f284933fc1c684f",
        "Created": "2019-06-11T09:33:47.629860524Z",
        "Path": "/bin/bash",
        "Args": [],
        "State": {
            "Status": "running",
            "Running": true,
            "Paused": false,
            "Restarting": false,
            "OOMKilled": false,
            "Dead": false,
            "Pid": 16351,
            "ExitCode": 0,
            "Error": "",
            "StartedAt": "2019-06-11T09:33:47.907189621Z",
            "FinishedAt": "0001-01-01T00:00:00Z"
        },
```

Above info comes from

```
/var/lib/docker/containers/945f049086194353fd3e184c2b370a2828781b64b3fbe0089f284933fc1c684f/config.v2.json
```

Step 8: **Login to your container**

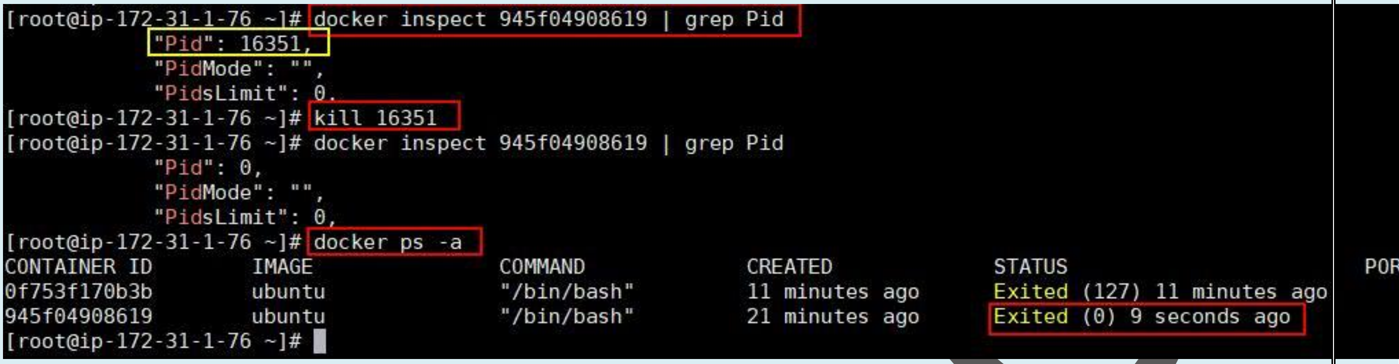
```
docker exec -i -t 945f04908619 /bin/bash
```

If you login successfully, you should be in your container bash login as shown below

```
root@ 945f04908619:/#
```

Another way to shutdown your container (login as root on your host)

```
docker inspect 945f04908619 | grep Pid
```

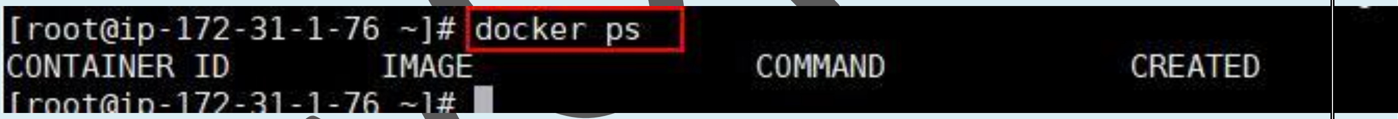


```
[root@ip-172-31-1-76 ~]# docker inspect 945f04908619 | grep Pid
            "Pid": 16351,
            "PidMode": "",
            "PidsLimit": 0,
[root@ip-172-31-1-76 ~]# kill 16351
[root@ip-172-31-1-76 ~]# docker inspect 945f04908619 | grep Pid
            "Pid": 0,
            "PidMode": "",
            "PidsLimit": 0,
[root@ip-172-31-1-76 ~]# docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS                       POR
0f753f170b3b        ubuntu              "/bin/bash"         11 minutes ago      Exited (127) 11 minutes ago
945f04908619        ubuntu              "/bin/bash"         21 minutes ago      Exited (0) 9 seconds ago
[root@ip-172-31-1-76 ~]#
```

```
kill 16351
```

You will notice container is stopped

```
docker ps
```



```
[root@ip-172-31-1-76 ~]# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED
[root@ip-172-31-1-76 ~]#
```