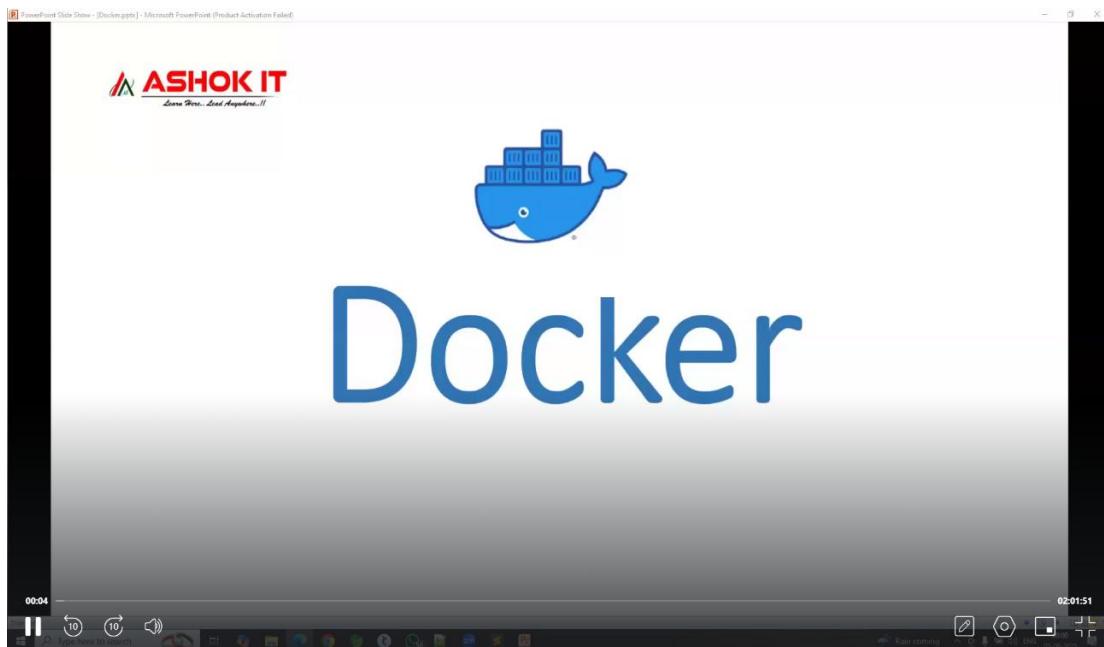


# Docker

## What is Docker ?



- Docker is a free & open source software
- Docker is used for containerization
  - Container = Execute application as a package (code + required s/w)
- With the help of docker, we can run our application in any machine very easily.
- Docker will take care of dependencies installation required for app execution.
- We can make our application portable using Docker.

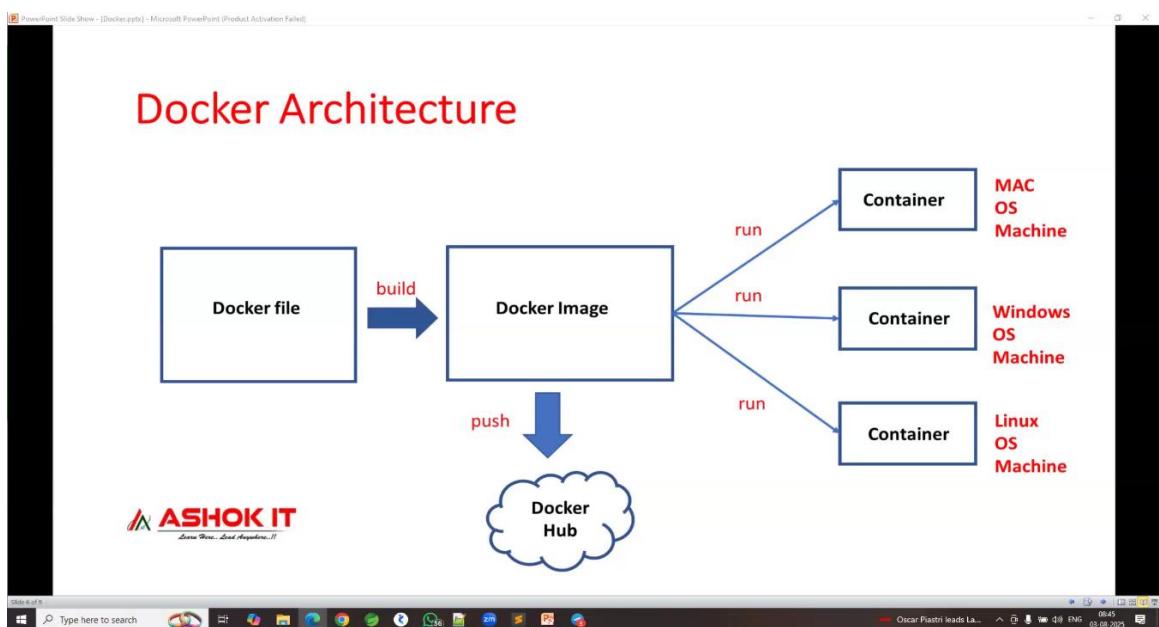
# Docker Architecture

1) Dockerfile

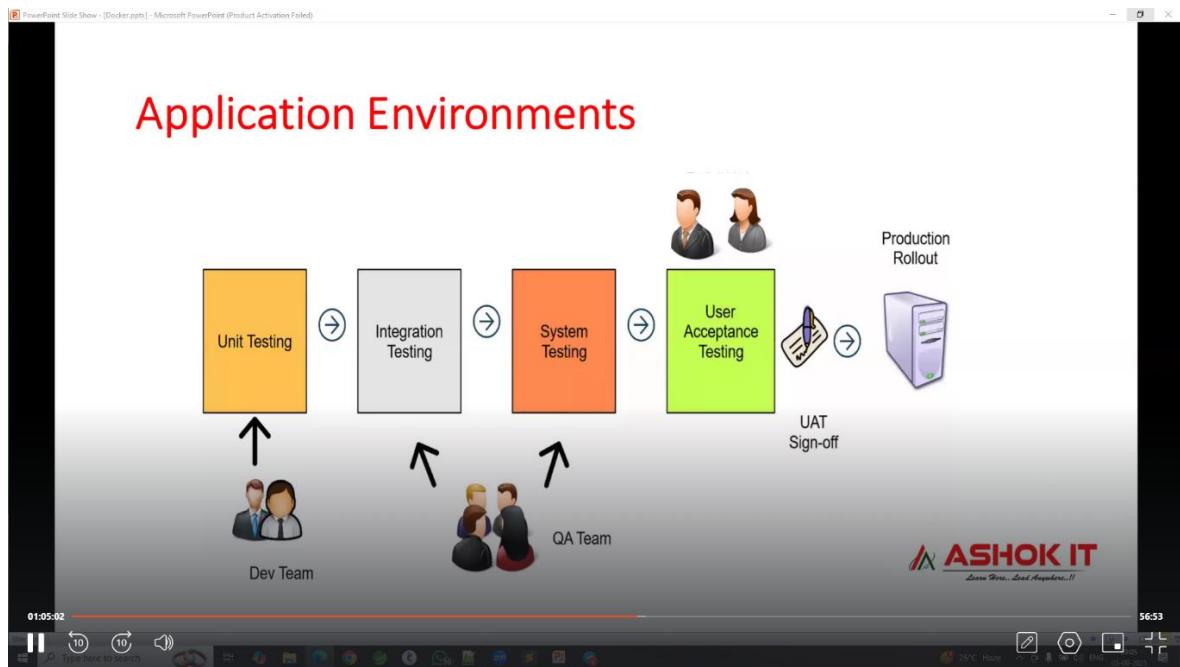
2) Docker Image

3) Docker Registry

4) Docker Container



- **Docker file** is used to specify where is our app-code and what dependencies (softwares) are required for our application execution.
- **Docker Image** is a package which contains (app\_code + dependencies)
- Note: **Dockerfile** is required to build docker image.
- **Docker Registry** is used to store Docker Images.
- Note: When we run **docker image** then Docker container will be created. Docker container is a linux virtual machine.
- **Docker Container** is used to run our application.



## Setup Linux VM (Amazon Linux / Ubuntu)

1. Login into AWS Cloud account
2. Create Linux VM and connect to it using MobaXterm

### Install Docker In Amazon Linux VM

```
sudo yum update -y
sudo yum install docker -y
sudo service docker start
sudo usermod -aG docker ec2-user
exit
```

### Install Docker In Ubuntu VM

```
sudo apt update
curl -fsSL get.docker.com | /bin/bash
sudo usermod -aG docker ubuntu
exit
```

### Verify docker installation

```
docker -v
```

# EC2 - Docker-VM created

The screenshot shows the AWS EC2 Instances page. On the left, a navigation sidebar lists categories like Dashboard, EC2 Global View, Events, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images (selected), AMIs, and AMI Catalog. The main content area displays a table titled 'Instances (1/1)'. The table has columns for Name, Instance ID, Instance state, Instance type, Status check, Alarm status, and Availability Zone. One row is visible for an instance named 'Docker-VM' with the ID 'i-066f92dce4e25c6cf', which is 'Running' and has an 't2.micro' instance type. Below this, a detailed view for the instance 'i-066f92dce4e25c6cf (Docker-VM)' is shown with tabs for Details, Status and alarms, Monitoring, Security, Networking, Storage, and Tags. The 'Details' tab is selected, showing sections for Instance summary, Public IPv4 address (13.232.5.213), Private IPv4 addresses (172.31.33.189), and Public DNS (ec2-13-232-5-213.ap-south-1.compute.amazonaws.com).

# Install Docker in Amazon linux VM

```
ec2-user@ip-172-31-33-189:~ [ec2-user@ip-172-31-33-189 ~]$ sudo yum update -y  
sudo yum install docker -y  
sudo service docker start  
sudo usermod -aG docker ec2-user  
exit
```

## After Docker Installation:

## Docker Commands

- **docker images** : To display docker images available in our system.
- **docker pull** : To download docker image from docker hub
  - \$ docker pull <image-name/image-id>
- **docker run** : To create docker container      based on docker image
  - \$ docker run <image-name/image-id>
- **docker ps** : to display running docker containers
- **docker ps -a** : To display running + stopped docker containers.
- **docker ps -q** : To display running containers ids.
- **docker ps -a -f status=exited -q** : To display stopped containers ids.
- **docker stop** : To stop running container
  - \$ docker stop <container-id>
- **docker rm** : To delete stopped container
  - \$ docker rm <container-id>
- **docker start** : To start the containers which is in stopped state.
  - \$ docker start <container-id>
- **docker rmi** : To delete docker image
  - \$ docker rmi <image-name/image-id>
- **docker logs** : To display container logs
  - \$ docker logs <container-id>
- **docker system prune** : delete stopped containers + un-used images
  - \$ docker system prune -a

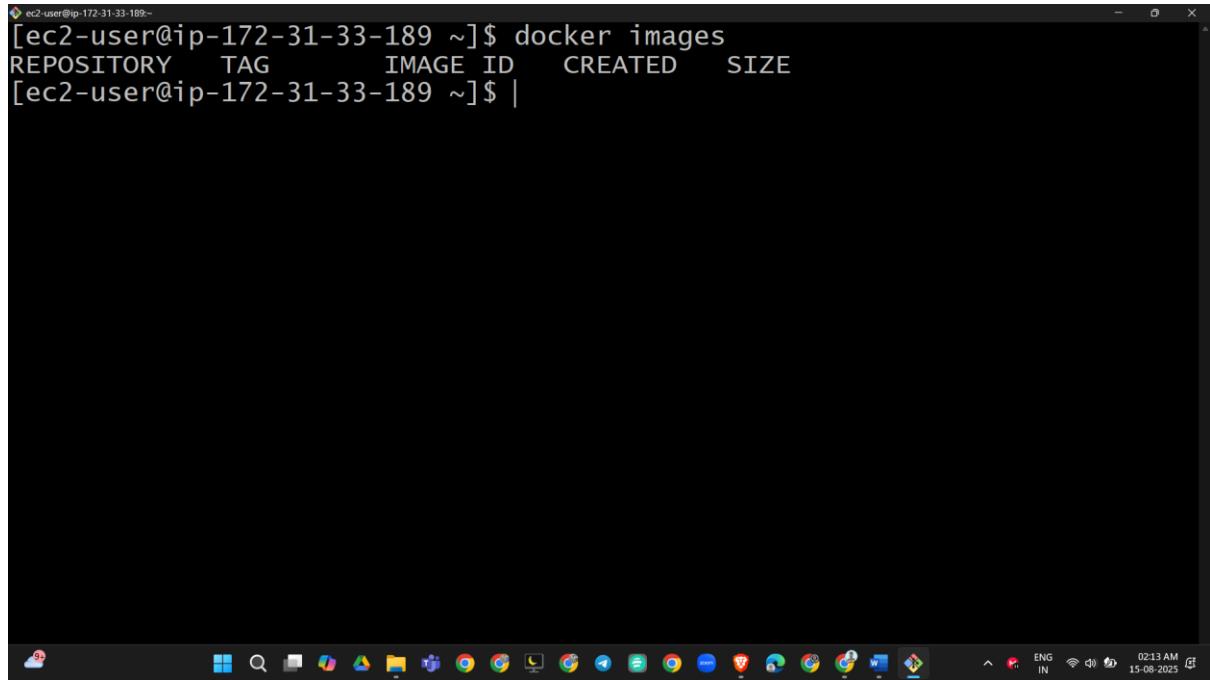
# Dockerhub

The screenshot shows the Docker Hub homepage. On the left, there's a sidebar for the user 'himanshur3888' with options like Repositories, Collaborations, Settings, Default privacy, Notifications, Billing, Usage, Pulls, and Storage. The main area features a large blue banner with the text 'Welcome to Docker' and 'Download the desktop application'. It includes a 'Download for Windows' button and notes that it's also available for Mac and Linux. Below the banner are three cards: 'Create a Repository' (Push container images to a repository on Docker Hub), 'Docker Hub Basics' (Watch the guide on how to create and push your first image into a Docker Hub repository), and 'Language-Specific Guides' (Learn how to containerize language-specific applications using Docker). At the bottom, there are sections for 'Spotlight', 'AI MEETS COMPOSE', 'CLOUD DEVELOPMENT', and 'SOFTWARE SUPPLY CHAIN'.

Cmd : [Docker -v](#)

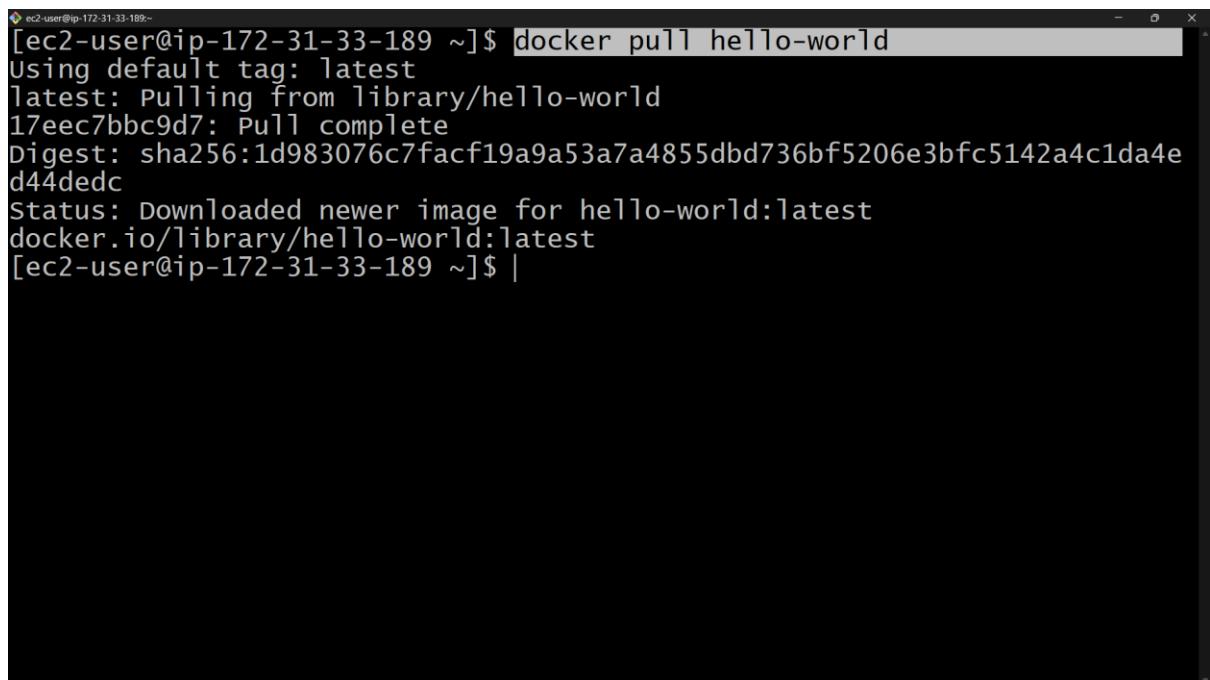
```
[ec2-user@ip-172-31-33-189 ~]$ docker -v
Docker version 25.0.8, build 0bab007
[ec2-user@ip-172-31-33-189 ~]$
```

## Cmd : Docker images



```
[ec2-user@ip-172-31-33-189 ~]$ docker images
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
[ec2-user@ip-172-31-33-189 ~]$ |
```

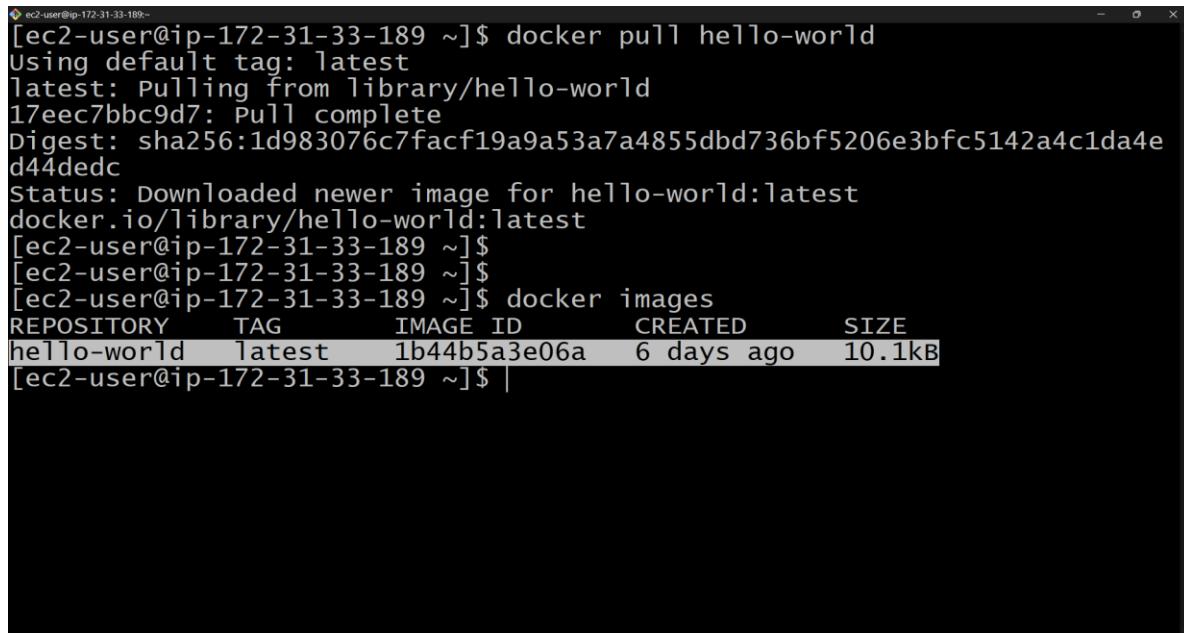
## Cmd : Docker pull hello-world



```
[ec2-user@ip-172-31-33-189 ~]$ docker pull hello-world
Using default tag: latest
latest: Pulling from library/hello-world
17eec7bbc9d7: Pull complete
Digest: sha256:1d983076c7facf19a9a53a7a4855dbd736bf5206e3bfc5142a4c1da4e
d44dedc
Status: Downloaded newer image for hello-world:latest
docker.io/library/hello-world:latest
[ec2-user@ip-172-31-33-189 ~]$ |
```

## Verification after docker pull

### Cmd: Docker images

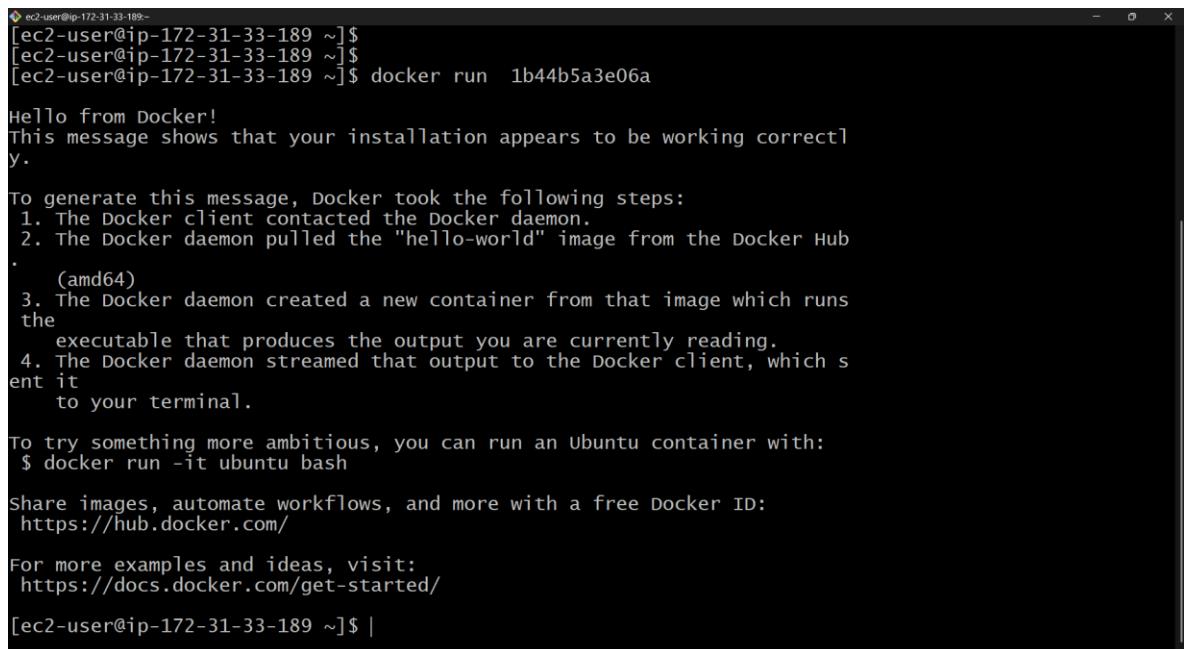


```
[ec2-user@ip-172-31-33-189 ~]$ docker pull hello-world
Using default tag: latest
latest: Pulling from library/hello-world
17eec7bbc9d7: Pull complete
Digest: sha256:1d983076c7facf19a9a53a7a4855dbd736bf5206e3bfc5142a4c1da4e
d44dedc
Status: Downloaded newer image for hello-world:latest
docker.io/library/hello-world:latest
[ec2-user@ip-172-31-33-189 ~]$
[ec2-user@ip-172-31-33-189 ~]$
[ec2-user@ip-172-31-33-189 ~]$ docker images
REPOSITORY      TAG          IMAGE ID      CREATED       SIZE
hello-world    latest        1b44b5a3e06a   6 days ago   10.1kB
[ec2-user@ip-172-31-33-189 ~]$ |
```

## To Run Docker image

### Cmd: docker run (Image id)

#### Docker run 1b44b5a3e06a



```
[ec2-user@ip-172-31-33-189 ~]$
[ec2-user@ip-172-31-33-189 ~]$
[ec2-user@ip-172-31-33-189 ~]$ docker run 1b44b5a3e06a
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
    .
 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 s
ent 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/
[ec2-user@ip-172-31-33-189 ~]$ |
```

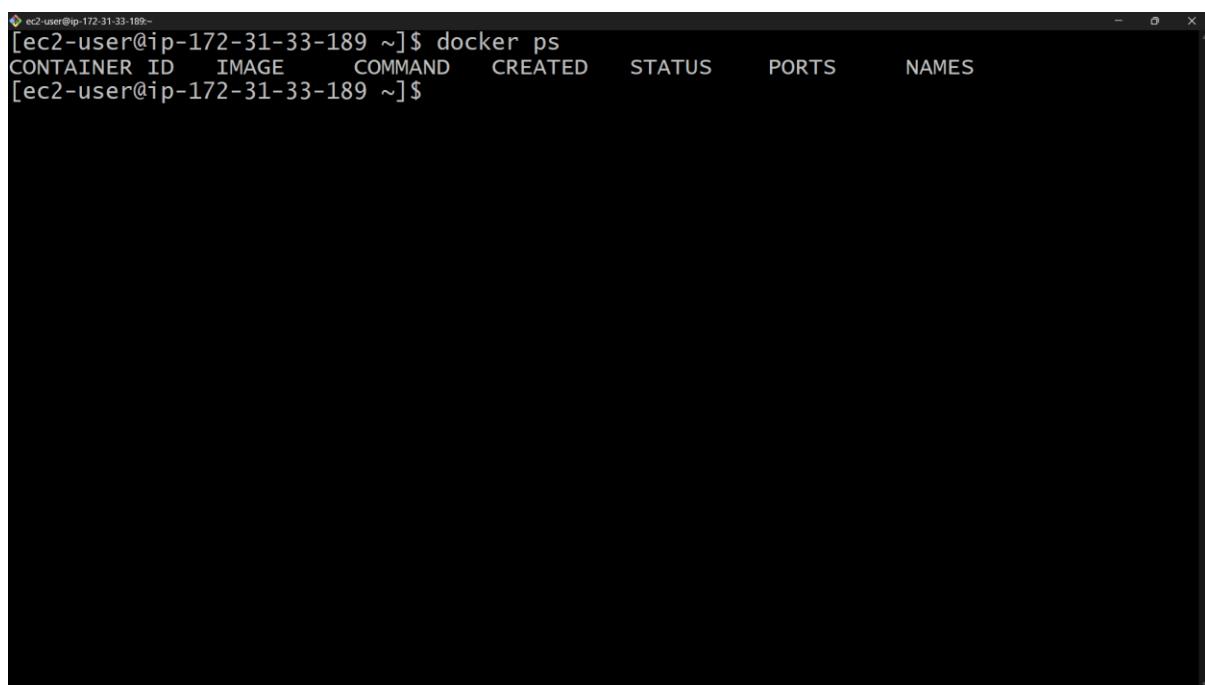
**To generate this message, Docker took the following steps:**

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.

**Cmd : docker ps**

**Shows running container**

Currently container is not running

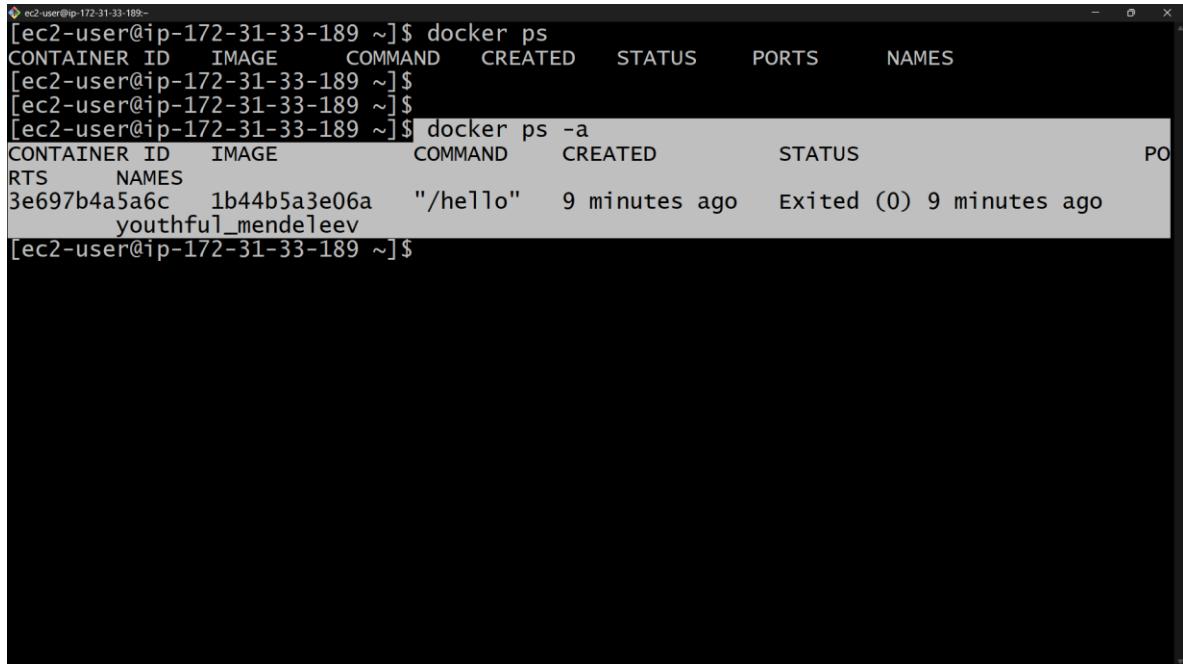


A screenshot of a terminal window titled 'Terminal' with a dark background. The window shows the command [ec2-user@ip-172-31-33-189 ~]\$ docker ps being run. The output is as follows:

```
[ec2-user@ip-172-31-33-189 ~]$ docker ps
CONTAINER ID        IMAGE               COMMAND       CREATED          STATUS          PORTS      NAMES
[ec2-user@ip-172-31-33-189 ~]$
```

**Cmd : docker ps -a**

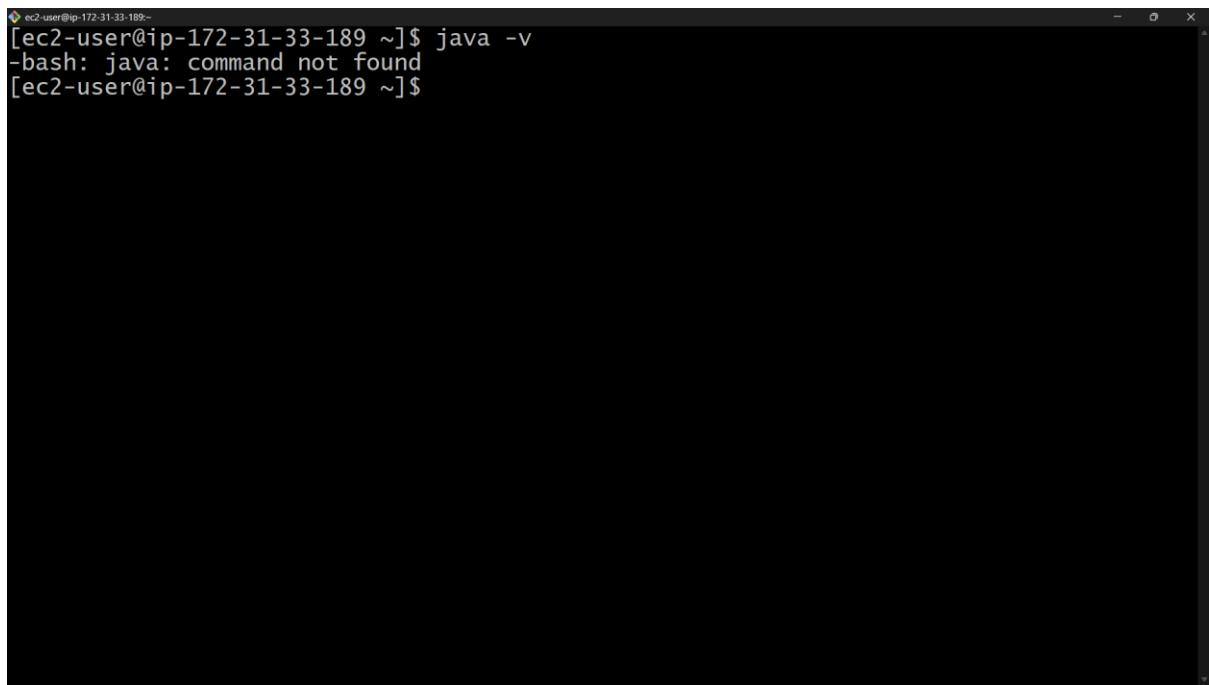
**Shows running and stopped container both .**



```
[ec2-user@ip-172-31-33-189 ~]$ docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
[ec2-user@ip-172-31-33-189 ~]$
[ec2-user@ip-172-31-33-189 ~]$
[ec2-user@ip-172-31-33-189 ~]$ docker ps -a
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
[ec2-user@ip-172-31-33-189 ~]$
```

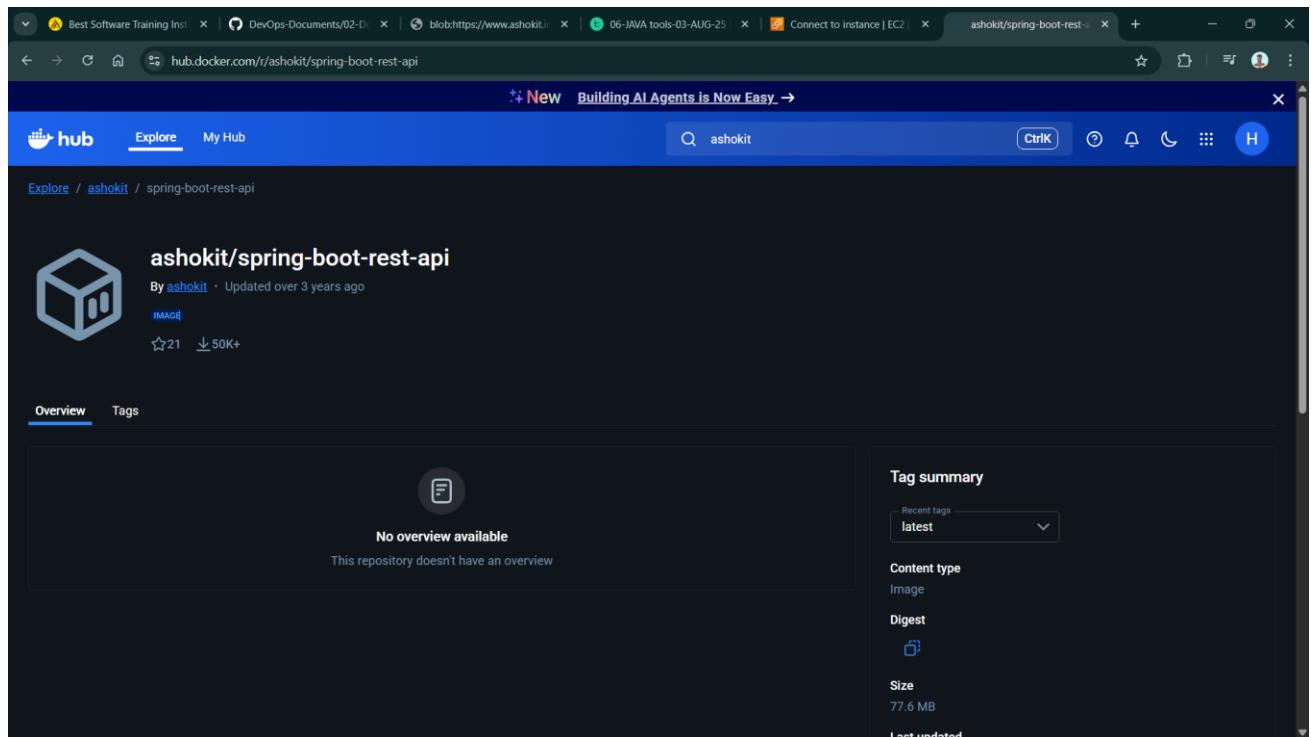
**Java not installed in docker**

**Cmd : java -v**



```
[ec2-user@ip-172-31-33-189 ~]$ java -v
-bash: java: command not found
[ec2-user@ip-172-31-33-189 ~]$
```

## Docker hub – Ashok it SpringBoot docker image

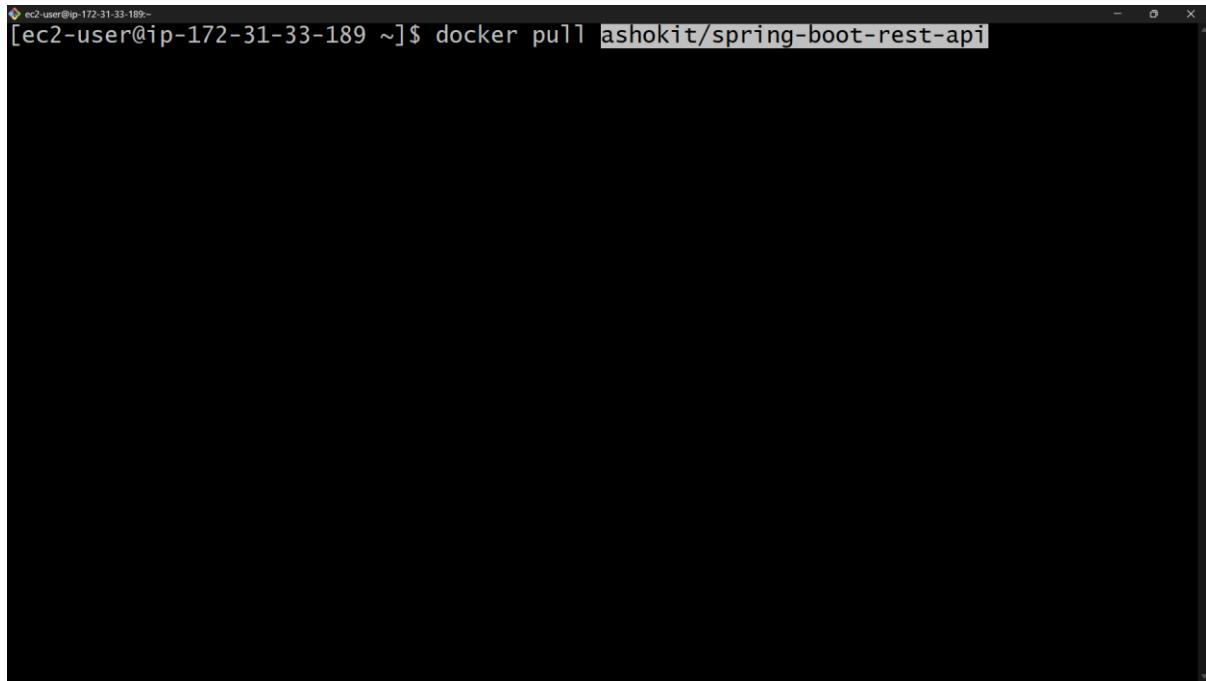


## Checked docker images

### Cmd: Docker images

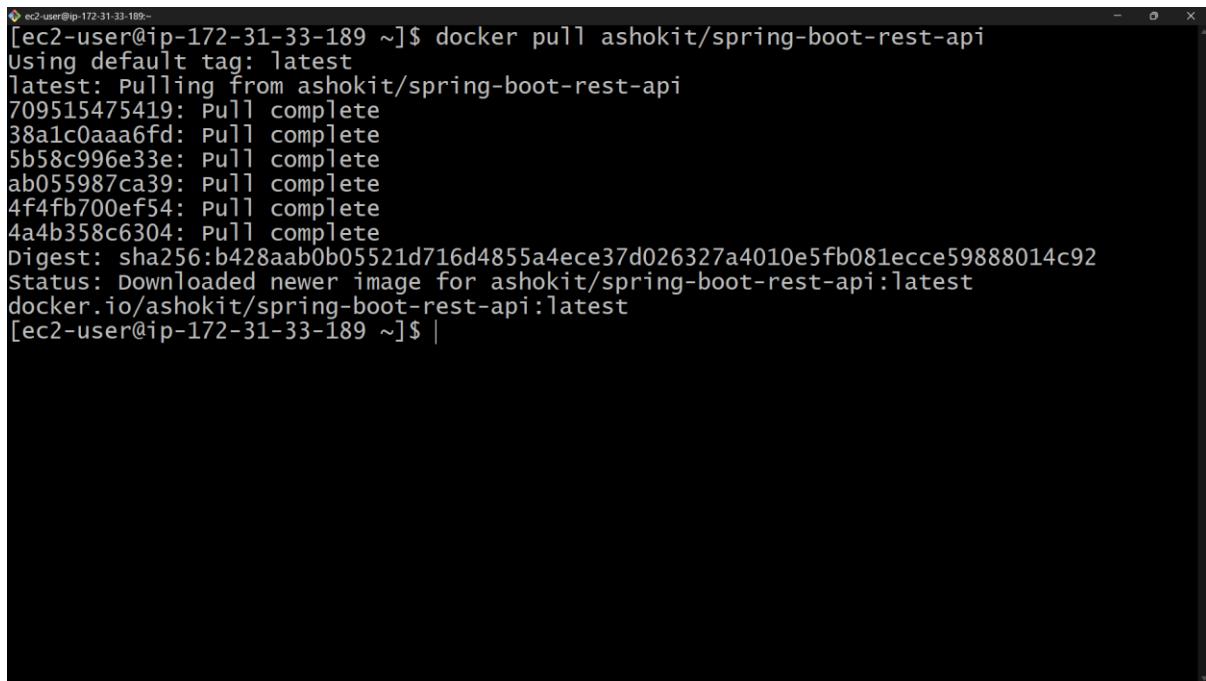
```
ec2-user@ip-172-31-33-189:~]$ java -v
-bash: java: command not found
[ec2-user@ip-172-31-33-189:~]$ docker images
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
hello-world    latest    1b44b5a3e06a   6 days ago   10.1kB
[ec2-user@ip-172-31-33-189:~]$
```

## Cmd :[docker pull ashokit/spring-boot-rest-api](#)



```
[ec2-user@ip-172-31-33-189 ~]$ docker pull ashokit/spring-boot-rest-api
```

## After cmd: [docker run ashokit/spring-boot-rest-api](#)



```
[ec2-user@ip-172-31-33-189 ~]$ docker pull ashokit/spring-boot-rest-api
Using default tag: latest
latest: Pulling from ashokit/spring-boot-rest-api
709515475419: Pull complete
38a1c0aaa6fd: Pull complete
5b58c996e33e: Pull complete
ab055987ca39: Pull complete
4f4fb700ef54: Pull complete
4a4b358c6304: Pull complete
Digest: sha256:b428aab0b05521d716d4855a4ece37d026327a4010e5fb081ecce59888014c92
Status: Downloaded newer image for ashokit/spring-boot-rest-api:latest
docker.io/ashokit/spring-boot-rest-api:latest
[ec2-user@ip-172-31-33-189 ~]$ |
```

# Verify Docker images

## Cmd: Docker images

```
[ec2-user@ip-172-31-33-189 ~]$ docker pull ashokit/spring-boot-rest-api
Using default tag: latest
latest: Pulling from ashokit/spring-boot-rest-api
709515475419: Pull complete
38a1c0aaa6fd: Pull complete
5b58c996e33e: Pull complete
ab055987ca39: Pull complete
4f4fb700ef54: Pull complete
4a4b358c6304: Pull complete
Digest: sha256:b428aab0b05521d716d4855a4ece37d026327a4010e5fb081ecce59888014c92
Status: Downloaded newer image for ashokit/spring-boot-rest-api:latest
docker.io/ashokit/spring-boot-rest-api:latest
[ec2-user@ip-172-31-33-189 ~]$
[ec2-user@ip-172-31-33-189 ~]$
[ec2-user@ip-172-31-33-189 ~]$ docker images
REPOSITORY          TAG      IMAGE ID      CREATED       SIZE
hello-world         latest   1b44b5a3e06a  6 days ago   10.1kB
ashokit/spring-boot-rest-api  latest   fcb4841c55d1  3 years ago  178MB
[ec2-user@ip-172-31-33-189 ~]$ |
```

# Now run Docker image

**Cmd : docker run ashokit/spring-boot-rest-api**

## **APPLICATION RUNNING:-**

**Note : without installing java**

- Running Real-world applications using docker images

public docker image name (java springboot app) : **ashokit/spring-boot-rest-api**

```
docker pull ashokit/spring-boot-rest-api
```

```
docker run ashokit/spring-boot-rest-api
```

## Docker system prune -a :- delete stopped containers + unused images

```
[ec2-user@ip-172-31-33-189 ~]$ docker system prune -a
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] y
Deleted Containers:
accaca2c4df7f21322b9f7506e0025c0e3567731f2c448b2fdfccc7b49af77e7
3e697b4a5a6c2c0dc2523658880828fb772d1bec570e69450c3f00c579be22ce

Deleted Images:
untagged: ashokit/spring-boot-rest-api:latest
untagged: ashokit/spring-boot-rest-api@sha256:b428aab0b05521d716d4855a4ece37d026
327a4010e5fb081ecc59888014c92
deleted: sha256:fcb4841c55d10837c8b7d22f96ad09fefeb80fba286a7c2487ccc20a705d3f62
deleted: sha256:64433a79a768d40420f88677fc449adf3deb08f43cc73b94ff23cf69be1c54c
deleted: sha256:54bafde522f5ad1b7fbde52f742ba41d67ffa1bd362d36d6195ef652422d7950
deleted: sha256:91ec3f9643e1434e81656ba36202f1a3b7f5f62baef3ac2e657883eae89b11796
deleted: sha256:be3703612910bd0c49e7d69c76b580ca4cdbf15e2e30b384a2cb5b0eaeac2144
deleted: sha256:de2a0834244ea0d56bb79f8a978c917b6674c585df7b74e99e9ff4581912b36d
deleted: sha256:9f8566ee5135862dd980160c27bd7721448a6f7f385bb81f7f001f1b78a5fbf
untagged: hello-world:latest
untagged: hello-world@sha256:1d983076c7facf19a9a53a7a4855dbd736bf5206e3bfc5142a4
c1da4ed44dedc
deleted: sha256:1b44b5a3e06a9aae883e7bf25e45c100be0bb81a0e01b32de604f3ac44711634
deleted: sha256:53d204b3dc5ddbc129df4ce71996b8168711e211274c785de5e0d4eb68ec3851

Total reclaimed space: 178.2MB
[ec2-user@ip-172-31-33-189 ~]$
```

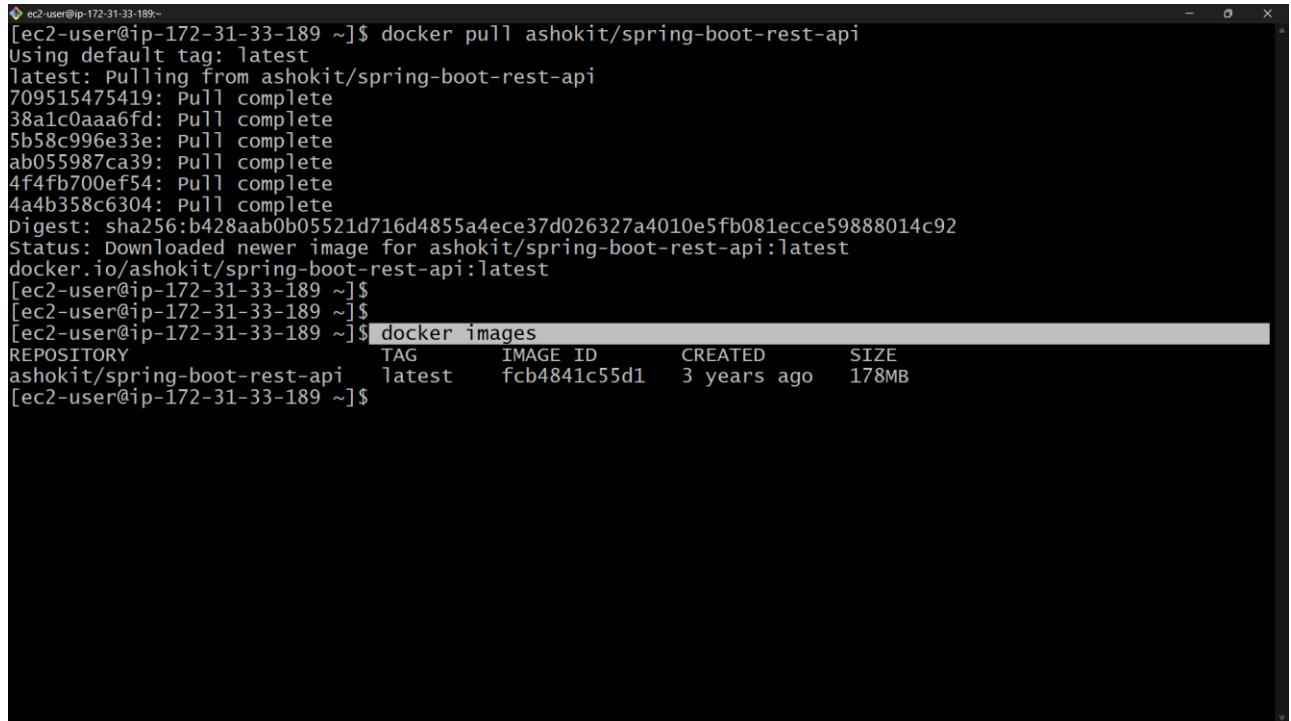
After prune command

Run : Docker pull [ashokit/spring-boot-rest-api](#)

```
[ec2-user@ip-172-31-33-189 ~]$ docker pull ashokit/spring-boot-rest-api
Using default tag: latest
latest: Pulling from ashokit/spring-boot-rest-api
709515475419: Pull complete
38a1c0aaa6fd: Pull complete
5b58c996e33e: Pull complete
ab055987ca39: Pull complete
4f4fb700ef54: Pull complete
4a4b358c6304: Pull complete
Digest: sha256:b428aab0b05521d716d4855a4ece37d026327a4010e5fb081ecc59888014c92
Status: Downloaded newer image for ashokit/spring-boot-rest-api:latest
docker.io/ashokit/spring-boot-rest-api:latest
[ec2-user@ip-172-31-33-189 ~]$ 
[ec2-user@ip-172-31-33-189 ~]$ 
[ec2-user@ip-172-31-33-189 ~]$ docker images
REPOSITORY          TAG      IMAGE ID      CREATED       SIZE
ashokit/spring-boot-rest-api    latest   fcb4841c55d1   3 years ago   178MB
[ec2-user@ip-172-31-33-189 ~]$ |
```

## Check docker images

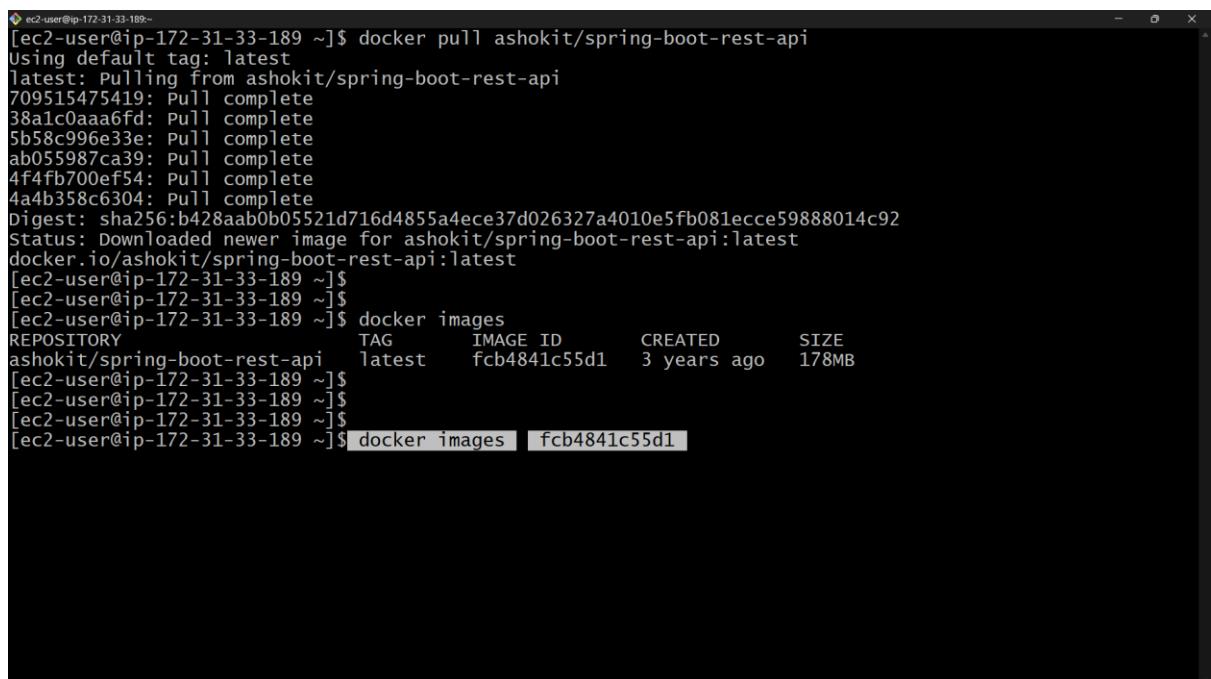
Run : [docker images](#)



```
[ec2-user@ip-172-31-33-189 ~]$ docker pull ashokit/spring-boot-rest-api
Using default tag: latest
latest: Pulling from ashokit/spring-boot-rest-api
709515475419: Pull complete
38a1c0aaa6fd: Pull complete
5b58c996e33e: Pull complete
ab055987ca39: Pull complete
4f4fb700ef54: Pull complete
4a4b358c6304: Pull complete
Digest: sha256:b428aab0b05521d716d4855a4ece37d026327a4010e5fb081ecce59888014c92
Status: Downloaded newer image for ashokit/spring-boot-rest-api:latest
docker.io/ashokit/spring-boot-rest-api:latest
[ec2-user@ip-172-31-33-189 ~]$
[ec2-user@ip-172-31-33-189 ~]$
[ec2-user@ip-172-31-33-189 ~]$ docker images
REPOSITORY          TAG      IMAGE ID      CREATED       SIZE
ashokit/spring-boot-rest-api    latest    fcb4841c55d1   3 years ago   178MB
[ec2-user@ip-172-31-33-189 ~]$
```

Note: We can not run our application outside of the machine

Cmd : [docker images fcb4841c55d1](#)



```
[ec2-user@ip-172-31-33-189 ~]$ docker pull ashokit/spring-boot-rest-api
Using default tag: latest
latest: Pulling from ashokit/spring-boot-rest-api
709515475419: Pull complete
38a1c0aaa6fd: Pull complete
5b58c996e33e: Pull complete
ab055987ca39: Pull complete
4f4fb700ef54: Pull complete
4a4b358c6304: Pull complete
Digest: sha256:b428aab0b05521d716d4855a4ece37d026327a4010e5fb081ecce59888014c92
Status: Downloaded newer image for ashokit/spring-boot-rest-api:latest
docker.io/ashokit/spring-boot-rest-api:latest
[ec2-user@ip-172-31-33-189 ~]$
[ec2-user@ip-172-31-33-189 ~]$
[ec2-user@ip-172-31-33-189 ~]$ docker images
REPOSITORY          TAG      IMAGE ID      CREATED       SIZE
ashokit/spring-boot-rest-api    latest    fcb4841c55d1   3 years ago   178MB
[ec2-user@ip-172-31-33-189 ~]$
[ec2-user@ip-172-31-33-189 ~]$
[ec2-user@ip-172-31-33-189 ~]$ docker images | fcb4841c55d1
[ec2-user@ip-172-31-33-189 ~]$
```

## No relation between container and local host Ec-2

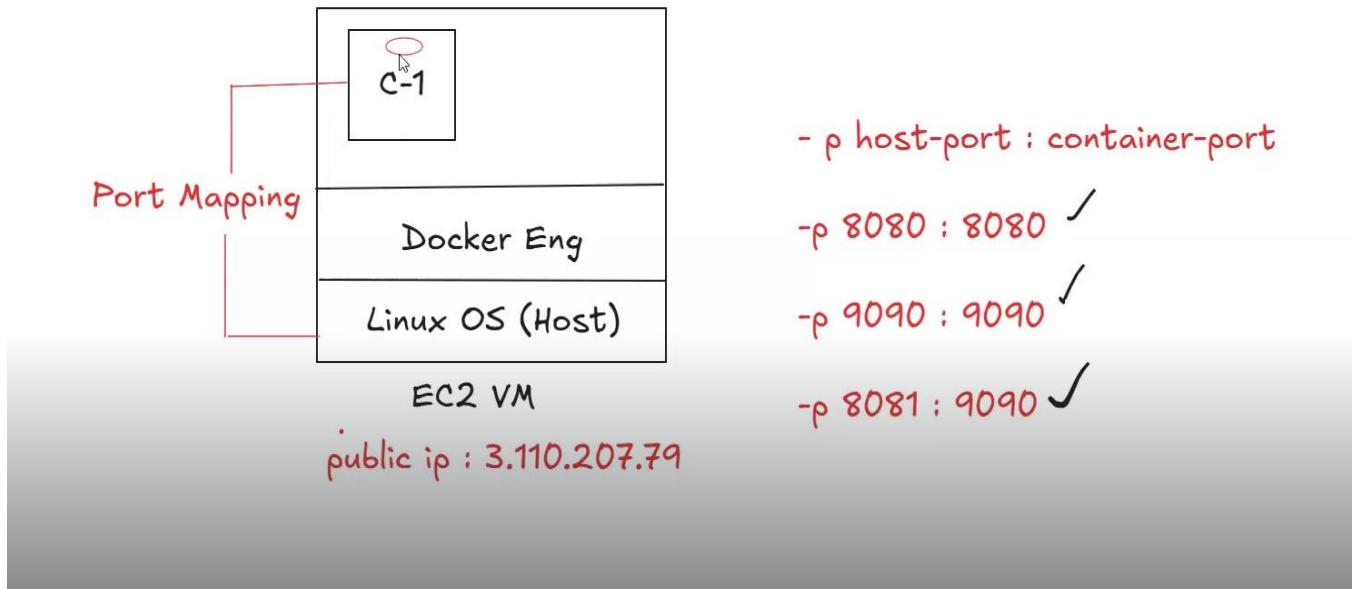
```
[ec2-user@ip-172-31-33-189 ~]$ docker run fcb4841c55d1

          _.-_,-_,-_,-_,-_,-_,-_,-_,-_,-_,-_
         (( ))(( ))(( ))(( ))(( ))(( ))
        / \ \ / \ \ / \ \ / \ \ / \ \ / \ \ / \ \ / \ \ / \ \
       |   | |   | |   | |   | |   | |   | |   | |
      ==|==|==|==|==|==|==|==|==|==|==|==|==|==|
      :: Spring Boot ::      (v2.3.8.RELEASE)

2025-08-15 05:01:01.439  INFO 1 --- [           main] in.ashokit.SbRestAppApplication      : Starting
SbRestAppApplication v0.0.1-SNAPSHOT on 354164a6d37d with PID 1 (/usr/app/spring-boot-docker-app.jar sta
rted by root in /usr/app)
2025-08-15 05:01:01.457  INFO 1 --- [           main] in.ashokit.SbRestAppApplication      : No activ
e profile set, falling back to default profiles: default
2025-08-15 05:01:04.119  INFO 1 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat i
nitialized with port(s): 9090 (http)
2025-08-15 05:01:04.154  INFO 1 --- [           main] o.apache.catalina.core.StandardService   : Starting
service [Tomcat]
2025-08-15 05:01:04.155  INFO 1 --- [           main] org.apache.catalina.core.StandardEngine  : Starting
Servlet engine: [Apache Tomcat/9.0.41]
2025-08-15 05:01:04.335  INFO 1 --- [           main] o.a.c.c.C.[Tomcat].[localhost].[]        : Initiali
zing Spring embedded WebApplicationContext
2025-08-15 05:01:04.336  INFO 1 --- [           main] w.s.c.ServletWebServerApplicationContext : Root Web
ApplicationContext: initialization completed in 2725 ms
2025-08-15 05:01:05.088  INFO 1 --- [           main] in.ashokit.rest.WelcomeRestController    : ***** We
lcomeRestController::Constructor *****
2025-08-15 05:01:05.490  INFO 1 --- [           main] o.s.s.concurrent.ThreadPoolTaskExecutor  : Initiali
zing ExecutorService 'applicationTaskExecutor'
2025-08-15 05:01:05.995  INFO 1 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat s
tarted on port(s): 9090 (http) with context path ''
2025-08-15 05:01:06.010  INFO 1 --- [           main] in.ashokit.SbRestAppApplication      : Started
SbRestAppApplication in 5.625 seconds (JVM running for 6.632)
```

# What is Port Mapping ?

- Note: By default, services running inside Docker containers are isolated and not accessible from outside.
    - Docker port mapping is the process of linking container port to host machine port.
    - It is used to allow external access to applications running inside the container.
  - Syntax : `docker run -p <host_port>:<container_port> image_name`
  - Note: host port and container port no need to be same.



Ex : Ex: docker run -p host-port:container-port <image-name>

```
$ docker run -p 9090:9090 ashokit/spring-boot-rest-api
```

```
$ docker run -p 9091:9090 ashokit/spring-boot-rest-api
```

```
$ docker run -p 9092:9090 ashokit/spring-boot-rest-api
```

**Note: Host port number we need to enable in ec2-vm security group inbound rules to allow the traffic.**

**Note: To access application running in the container we will use below URL**

## Docker run with port mapping

Cmd: Docker run -p 9090:9090 fcb4841c55d1

```
ec2-user@ip-172-31-33-189:~$ docker run -p 9090:9090 fcb4841c55d1
:: Spring Boot ::      (v2.3.8.RELEASE)

2025-08-15 05:05:14.144  INFO 1 --- [           main] in.ashokit.SbRestAppApplication      : Starting SbRestAppApplication on 6526c9acc8db with PID 1 (/usr/app/spring-boot-docker-app.jar started by root in /usr/app)
2025-08-15 05:05:14.150  INFO 1 --- [           main] in.ashokit.SbRestAppApplication      : No active profile set, falling back to default profiles: default
2025-08-15 05:05:16.659  INFO 1 --- [       th port(s): 9090 (http)]
2025-08-15 05:05:16.683  INFO 1 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 9090 (http)
2025-08-15 05:05:16.695  INFO 1 --- [        ne: [Apache Tomcat/9.0.41]]
2025-08-15 05:05:16.853  INFO 1 --- [mbbedded webApplicationContext
2025-08-15 05:05:16.862  INFO 1 --- [ntext: initialization completed in 2577 ms
2025-08-15 05:05:17.530  INFO 1 --- [main] in.ashokit.rest.WelcomeRestController : ***** WelcomeRestController
2025-08-15 05:05:17.888  INFO 1 --- [main] o.s.s.concurrent.ThreadPoolTaskExecutor : Initializing ExecutorService 'applicationTaskExecutor'
2025-08-15 05:05:18.342  INFO 1 --- [t(s): 9090 (http) with context path
2025-08-15 05:05:18.361  INFO 1 --- [           main] in.ashokit.SbRestAppApplication      : Started SbRestAppApplication in 5.248 seconds (JVM running for 6.233)
```

Now change port in EC-2 VM in security bounding

The screenshot shows the AWS CloudWatch Instances console. On the left, there's a navigation sidebar with options like EC2, Instances, Images, Elastic Block Store, and Network & Security. The main area displays the 'Instances (1/1)' section. A table lists one instance: 'Docker-VM' (Instance ID: i-066f92dce4e25c6cf, State: Running, Type: t2.micro). Below this, the details for the 'Docker-VM' instance are shown. Under the 'Security' tab, it lists the IAM Role (empty), Owner ID (846759222787), and Security groups (sg-02548d5e1b346c4ff (launch-wizard-10)). The 'Launch time' is Fri Aug 15 2025 10:09:40 GMT+0530 (India Standard Time). At the bottom, there are links for CloudShell and Feedback.

Screenshot of the AWS EC2 Security Groups console showing the details of a security group named "sg-02548d5e1b346c4ff - launch-wizard-10".

**Details:**

Security group name launch-wizard-10	Security group ID sg-02548d5e1b346c4ff	Description launch-wizard-10 created 2025-07-20T12:53:57.556Z	VPC ID vpc-08b19d1fc556e4a2b
Owner 846759222787	Inbound rules count 2 Permission entries	Outbound rules count 1 Permission entry	

**Inbound rules (2):**

Name	Security group rule ID	IP version	Type	Protocol	Port range
-	sgr-03567e1f096ff6503	IPv4	SSH	TCP	22
-	sgr-03ca1b31a154f4d46	IPv4	HTTP	TCP	80

Screenshot of the AWS EC2 ModifyInboundSecurityGroupRules console showing the edit inbound rules page for the same security group.

**Edit inbound rules**

Inbound rules control the incoming traffic that's allowed to reach the instance.

**Inbound rules**

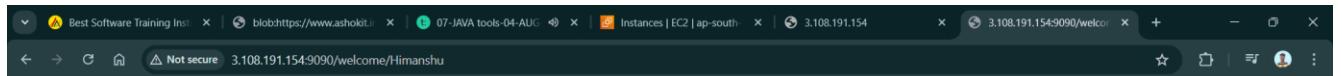
Security group rule ID	Type	Protocol	Port range	Source	Description - optional
sgr-03567e1f096ff6503	SSH	TCP	22	Custom	0.0.0.0/0
sgr-03ca1b31a154f4d46	Custom TCP	TCP	9090	Custom	0.0.0.0/0

**Warning:** Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

**Buttons:** Cancel, Preview changes, Save rules.

**SB App URL : <http://host-public-ip:host-port/welcome/{name}>**

**url: <http://3.108.191.154:9090/welcome/Himanshu>**



---

Himanshu, Welcome to Ashok IT..!!

**Docker container stopped when we use terminal**

**Cmd: Crtl + c**

We have to use container in **detached mode**

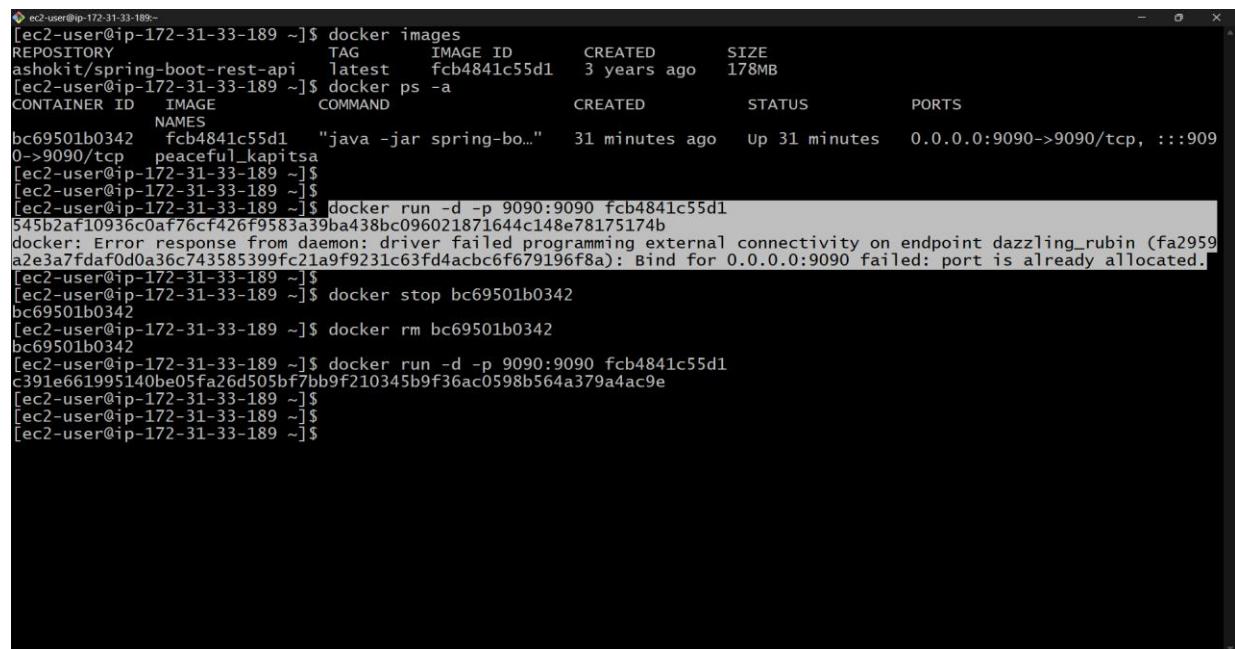
## What is detached mode ?

- Detached mode in Docker means running a container in the background rather than attaching to its input/output in the terminal.
- When you run a container in detached mode, Docker starts it and then immediately gives you back control of the terminal. The container continues running in the background.

**\$ docker run -d -p host-port:container-port <image-name>**

**Cmd: docker run -d -p 9090:9090 fcb4841c55d1**

## ERROR IN EC-2



```
[ec2-user@ip-172-31-33-189 ~]$ docker images
REPOSITORY          TAG      IMAGE ID      CREATED        SIZE
ashokit/spring-boot-rest-api   latest   fcb4841c55d1    3 years ago   178MB
[ec2-user@ip-172-31-33-189 ~]$ docker ps -a
CONTAINER ID        IMAGE               COMMAND                  CREATED        STATUS        PORTS
 NAMES
bc69501b0342      fcb4841c55d1    "java -jar spring-bo..."   31 minutes ago   Up 31 minutes   0.0.0.0:9090->9090/tcp, :::9090
0->9090/tcp        peaceful_kapitsa
[ec2-user@ip-172-31-33-189 ~]$
[ec2-user@ip-172-31-33-189 ~]$
[ec2-user@ip-172-31-33-189 ~]$ docker run -d -p 9090:9090 fcb4841c55d1
545b2af10936c0af76cf426f9583a39ba438bc096021871644c148e78175174b
docker: Error response from daemon: driver failed programming external connectivity on endpoint dazzling_rubin (fa2959a2e3a7fdaf0da0a36c743585399fc21a9f9231c63fd4acbc6f679196f8a): Bind for 0.0.0.0:9090 failed: port is already allocated.
[ec2-user@ip-172-31-33-189 ~]$
[ec2-user@ip-172-31-33-189 ~]$ docker stop bc69501b0342
bc69501b0342
[ec2-user@ip-172-31-33-189 ~]$ docker rm bc69501b0342
bc69501b0342
[ec2-user@ip-172-31-33-189 ~]$ docker run -d -p 9090:9090 fcb4841c55d1
c391e661995140be05fa26d505bf7bb9f210345b9f36ac0598b564a379a4ac9e
[ec2-user@ip-172-31-33-189 ~]$
[ec2-user@ip-172-31-33-189 ~]$
[ec2-user@ip-172-31-33-189 ~]$
```

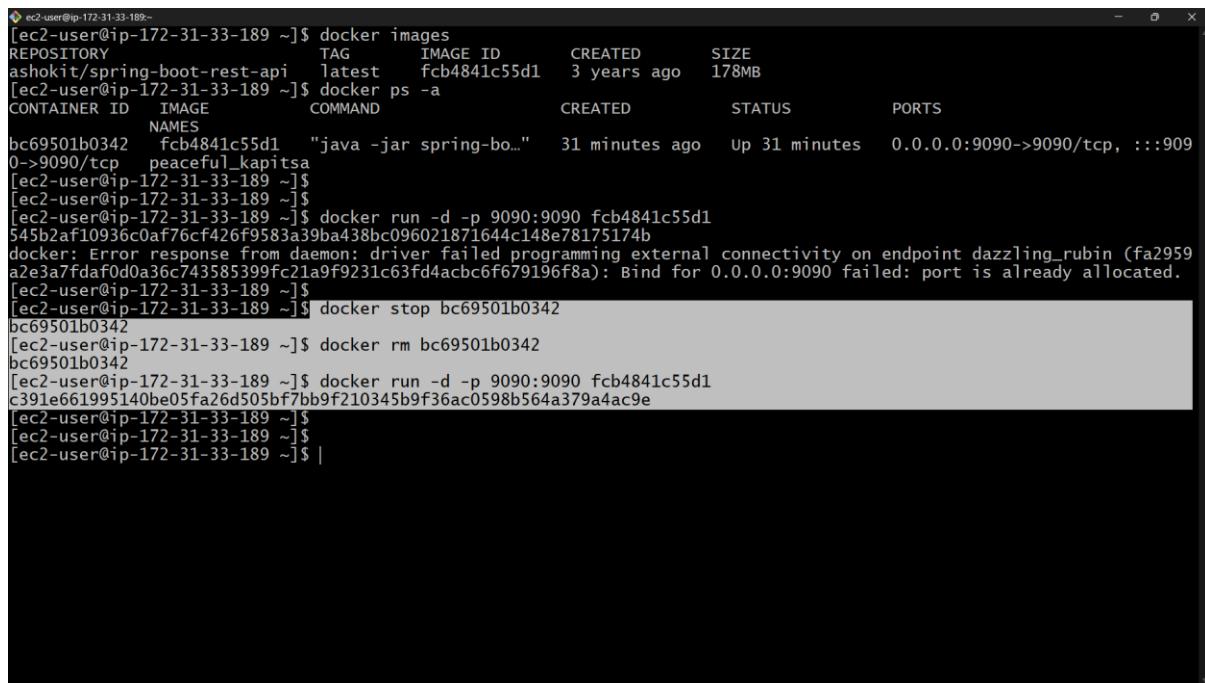
## Step to solve the Error :-

CMD:-

**docker stop bc69501b0342**

**docker rm bc69501b0342**

**docker run -d -p 9090:9090 fcb4841c55d1**



```
[ec2-user@ip-172-31-33-189 ~]$ docker images
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
ashokit/spring-boot-rest-api  latest   fcb4841c55d1  3 years ago  178MB
[ec2-user@ip-172-31-33-189 ~]$ docker ps -a
CONTAINER ID   IMAGE      COMMAND      CREATED      STATUS      PORTS
 NAMES
bc69501b0342   fcb4841c55d1  "java -jar spring-bo..."  31 minutes ago  Up 31 minutes  0.0.0.0:9090->9090/tcp, :::9090
0->9090/tcp    peaceful_kapitsa
[ec2-user@ip-172-31-33-189 ~]$
[ec2-user@ip-172-31-33-189 ~]$
[ec2-user@ip-172-31-33-189 ~]$ docker run -d -p 9090:9090 fcb4841c55d1
545b2af10936c0af76cf426f9583a39ba438bc096021871644c148e78175174b
docker: Error response from daemon: driver failed programming external connectivity on endpoint dazzling_rubin (fa2959a2e3a7fdaf0da36c743585399fc21a9f90231c63fd4acbc6f679196f8a): Bind for 0.0.0.0:9090 failed: port is already allocated.
[ec2-user@ip-172-31-33-189 ~]$
[ec2-user@ip-172-31-33-189 ~]$ docker stop bc69501b0342
bc69501b0342
[ec2-user@ip-172-31-33-189 ~]$ docker rm bc69501b0342
bc69501b0342
[ec2-user@ip-172-31-33-189 ~]$ docker run -d -p 9090:9090 fcb4841c55d1
c391e661995140be05fa26d505bf7bb9f210345b9f36ac0598b564a379a4ac9e
[ec2-user@ip-172-31-33-189 ~]$
[ec2-user@ip-172-31-33-189 ~]$
[ec2-user@ip-172-31-33-189 ~]$ |
```

**Random name they are using for container**

```
[ec2-user@ip-172-31-33-189 ~]
545b2af10936c0af76cf426f9583a39ba438bc096021871644c148e78175174b
docker: Error response from daemon: driver failed programming external connectivity on endpoint dazzling_rubin (fa2959
a2e3a7fdf00da36c743585399fc21a9f9231c63fd4acbc6f679196f8a): Bind for 0.0.0.0:9090 failed: port is already allocated.
[ec2-user@ip-172-31-33-189 ~]$
[ec2-user@ip-172-31-33-189 ~]$ docker stop bc69501b0342
[ec2-user@ip-172-31-33-189 ~]$ docker rm bc69501b0342
[ec2-user@ip-172-31-33-189 ~]$ docker run -d -p 9090:9090 fcbb4841c55d1
c391e661995140be05fa26d505bf7bb9f210345b9f36ac0598b564a379a4ac9e
[ec2-user@ip-172-31-33-189 ~]$
[ec2-user@ip-172-31-33-189 ~]$
[ec2-user@ip-172-31-33-189 ~]$ Read from remote host ec2-3-108-191-154.ap-south-1.compute.amazonaws.com: Connection re
set by peer
Connection to ec2-3-108-191-154.ap-south-1.compute.amazonaws.com closed.
client_loop: send disconnect: Connection reset by peer

himanshu12@himanshu MINGW64 /d/OneDrive/Desktop/pem
$ ssh -i "HR.pem" ec2-user@ec2-3-108-191-154.ap-south-1.compute.amazonaws.com
,
#_
~~ \#### Amazon Linux 2023
~~ \#####
~~ \#####
~~ \#/
~~ V-, '-->
~~
~~ .-
~~ /'-
~~ /'-
~~ /m'
Last login: Fri Aug 15 05:29:36 2025 from 152.58.32.57
[ec2-user@ip-172-31-33-189 ~]$ docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS
NAMES
c391e6619951 fcbb4841c55d1 "java -jar spring-bo..." 6 minutes ago Up 6 minutes 0.0.0.0:9090->9090/tcp, :::9090->9090/tcp
[ec2-user@ip-172-31-33-189 ~]$
```

## Docker ps contain container id

```
[ec2-user@ip-172-31-33-189 ~]$ docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED             STATUS              PORTS
 NAMES
c391e6619951        fcb4841c55d1   "java -jar spring-bo..."   8 minutes ago     Up 8 minutes   0.0.0.0:9090->9090/tcp, :::9090-
>9090/tcp          blissful_williams

[ec2-user@ip-172-31-33-189 ~]$
```

Check docker log using container id :-

Cmd: `docker logs c391e6619951`

```
ec2-user@ip-172-31-33-189:~$ c391e6619951  fcb4841c55d1  "java -jar spring-bo..."  10 minutes ago   Up 10 minutes  0.0.0.0:9090->9090/tcp, :::9090
0->9090/tcp  blissful_williams
[ec2-user@ip-172-31-33-189 ~]$ [ec2-user@ip-172-31-33-189 ~]$ docker logs c391e6619951

=====
:: Spring Boot ::      (v2.3.8.RELEASE)

2025-08-15 05:55:29.516  INFO 1 --- [           main] in.ashokit.SbRestAppApplication      : Starting SbRestAppApp
lication v0.0.1-SNAPSHOT on c391e6619951 with PID 1 (/usr/app/spring-boot-docker-app.jar started by root in /usr/app)
2025-08-15 05:55:29.527  INFO 1 --- [           main] in.ashokit.SbRestAppApplication      : No active profile set
, falling back to default profiles: default
2025-08-15 05:55:32.209  INFO 1 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized wi
th port(s): 9090 (http)
2025-08-15 05:55:32.240  INFO 1 --- [           main] o.apache.catalina.core.StandardService : Starting service [Tom
cat]
2025-08-15 05:55:32.240  INFO 1 --- [           main] org.apache.catalina.core.StandardEngine  : Starting Servlet engi
ne: [Apache Tomcat/9.0.41]
2025-08-15 05:55:32.403  INFO 1 --- [           main] o.a.c.c.C.[Tomcat].[localhost].[]        : Initializing Spring e
mbbed webApplicationContext
2025-08-15 05:55:32.415  INFO 1 --- [           main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationCo
ntext: initialization completed in 2742 ms
2025-08-15 05:55:33.108  INFO 1 --- [           main] in.ashokit.rest.WelcomeRestController    : ***** WelcomeRestCont
roller::Constructor *****
2025-08-15 05:55:33.461  INFO 1 --- [           main] o.s.s.concurrent.ThreadPoolTaskExecutor : Initializing Executor
Service 'applicationTaskExecutor'
2025-08-15 05:55:33.914  INFO 1 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on por
t(s): 9090 (http) with context path
2025-08-15 05:55:33.934  INFO 1 --- [           main] in.ashokit.SbRestAppApplication      : Started SbRestAppAppl
ication in 5.48 seconds (JVM running for 6.484)
[ec2-user@ip-172-31-33-189 ~]$ |
```

**public docker image name (python app) : ashokit/python-flask-app**

`$ docker run -d -p 5000:5000 ashokit/python-flask-app`

**Note: Host port number we need to enable in ec2-vm security group inbound rules to allow the traffic.**

@@@ Python App URL : <http://host-public-ip:host-port/>



## Running python application as docker container

```
ec2-user@ip-172-31-33-189~$ Status: Downloaded newer image for ashokit/python-flask-app:latest
docker.io/ashokit/python-flask-app:latest
[ec2-user@ip-172-31-33-189 ~]$ docker images
REPOSITORY          TAG      IMAGE ID      CREATED     SIZE
ashokit/python-flask-app    latest   82781d5652bc  2 years ago  913MB
ashokit/spring-boot-rest-api latest   fcb4841c55d1  3 years ago  178MB
[ec2-user@ip-172-31-33-189 ~]$ docker run -d -p 5000:5000 82781d5652bc
160dc77139327b820f66ead019677a059249af02e3026a8944b0bb0ff2cb95c0
[ec2-user@ip-172-31-33-189 ~]$ |
```

## Python -v

```
ec2-user@ip-172-31-33-189~$ [ec2-user@ip-172-31-33-189 ~]$ docker images
REPOSITORY          TAG      IMAGE ID      CREATED     SIZE
ashokit/python-flask-app    latest   82781d5652bc  2 years ago  913MB
ashokit/spring-boot-rest-api latest   fcb4841c55d1  3 years ago  178MB
[ec2-user@ip-172-31-33-189 ~]$ docker run -d -p 5000:5000 82781d5652bc
160dc77139327b820f66ead019677a059249af02e3026a8944b0bb0ff2cb95c0
[ec2-user@ip-172-31-33-189 ~]$ python -v
[ec2-user@ip-172-31-33-189 ~]$ bash: python: command not found
[ec2-user@ip-172-31-33-189 ~]$ |
```

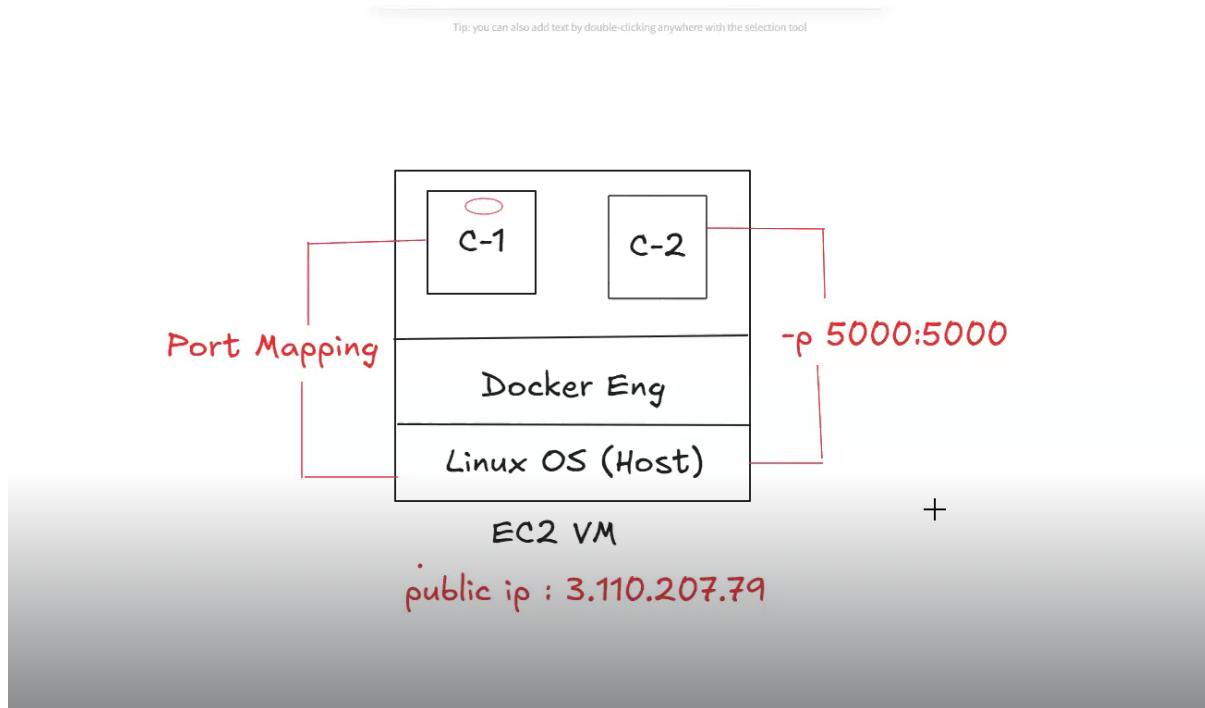
## Cmd: docker ps

## To see container in running mode

## Docker logs (container id)

```
[ec2-user@ip-172-31-33-189 ~]$ [ec2-user@ip-172-31-33-189 ~]$ [ec2-user@ip-172-31-33-189 ~]$ [ec2-user@ip-172-31-33-189 ~]$ [ec2-user@ip-172-31-33-189 ~]$ python -v -bash: python: command not found [ec2-user@ip-172-31-33-189 ~]$ docker ps CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES 160dc7713932 82781d5652bc "python app.py" About a minute ago Up About a minute 0.0.0.0:5000->5000/tcp , ::5000->5000/tcp great_rhodes c391e6619951 fcb4841c55d1 "java -jar spring-bo..." 29 minutes ago Up 29 minutes 0.0.0.0:9090->9090/tcp , ::9090->9090/tcp blissful_williams [ec2-user@ip-172-31-33-189 ~]$ docker logs 160dc7713932 * Serving Flask app 'app' (lazy loading) * Environment: production WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead. * Debug mode: on * Running on all addresses. WARNING: This is a development server. Do not use it in a production deployment. * Running on http://172.17.0.3:5000/ (Press CTRL+C to quit) * Restarting with stat * Debugger is active! * Debugger PIN: 119-742-448 [ec2-user@ip-172-31-33-189 ~]$ |
```

# Port Mapping



## Edit inbound rules:

aws Search [Alt+S] Account ID: 8467-5922-2787 himanshu raghore

EC2 > Security Groups > sg-02548d5e1b346c4ff - launch-wizard-10 > Edit inbound rules

**Edit inbound rules** info

Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules <small>info</small>	Type <small>Info</small>	Protocol <small>Info</small>	Port range <small>Info</small>	Source <small>Info</small>	Description - optional <small>Info</small>
sgr-03567e1f096ff6503	SSH	TCP	22	Custom	<input type="text"/> 0.0.0.0/0 <span>Delete</span>
sgr-03ca1b31a154f4d46	Custom TCP	TCP	9090	Custom	<input type="text"/> 0.0.0.0/0 <span>Delete</span>
-	Custom TCP	TCP	5000	Anywhere	<input type="text"/> 0.0.0.0/0 <span>Delete</span>

[Add rule](#)

Cancel [Preview changes](#) Save rules

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

## Public id : 5000

The screenshot shows the AWS EC2 Instances page. On the left, there's a navigation sidebar with 'EC2' selected. Under 'Instances', it lists 'Instances', 'Instance Types', 'Launch Templates', 'Spot Requests', 'Savings Plans', 'Reserved Instances', 'Dedicated Hosts', and 'Capacity Reservations'. Under 'Images', it lists 'AMIs' and 'AMI Catalog'. Under 'Elastic Block Store', it lists 'Volumes' and 'Snapshots'. Under 'Network & Security', it lists 'CloudShell' and 'Feedback'. The main content area shows a table titled 'Instances (1/1)'. It has columns for 'Name' (with a dropdown menu), 'Instance ID', 'Instance state', 'Instance type', 'Status check', 'Alarm status', and 'Availability Zone'. A single row is selected for 'Docker-VM' with Instance ID i-066f92dce4e25c6cf, showing 'Running' in all columns except 'Status check' which says '2/2 checks passed'. At the bottom of the table, it says 'Last updated less than a minute ago'. There are buttons for 'Connect', 'Actions', and 'Launch instances'. Below the table, there's a detailed view for 'i-066f92dce4e25c6cf (Docker-VM)'. It shows 'instance summary' with fields like 'Instance ID' (i-066f92dce4e25c6cf), 'IPv6 address' (empty), 'Hostname type' (IP name: ip-172-31-33-189.ap-south-1.compute.internal), and 'Instance type' (t2.micro). It also shows 'Public IPv4 address' (3.108.191.154), 'Private IP DNS name (IPv4 only)' (ip-172-31-33-189.ap-south-1.compute.internal), 'Public DNS' (ec2-3-108-191-154.ap-south-1.compute.amazonaws.com), and 'Private IPv4 addresses' (172.31.33.189). The bottom right of the page includes a footer with '© 2025, Amazon Web Services, Inc. or its affiliates.', 'Privacy', 'Terms', and 'Cookie preferences'.

url - <http://3.108.191.154:5000/>

The screenshot shows a web browser window with the URL '3.108.191.154:5000' in the address bar. The page content is 'Welcome to Python Flask Application - Ashok IT'.

Welcome to Python Flask Application - Ashok IT

## Same image with different port we can run container

```
ec2-user@ip-172-31-33-189:~$ docker logs 08ef99a6765d
2025-08-15 06:46:21.600 INFO 1 --- [           main] in.ashokit.SbRestAppApplication      : Starting SbRestAppApplication v0.0.1-SNAPSHOT on 08ef99a6765d
with PID 1 (/usr/app/spring-boot-docker-app.jar started by root in /usr/app)
2025-08-15 06:46:21.610 INFO 1 --- [           main] in.ashokit.SbRestAppApplication      : No active profile set, falling back to default profiles: defa
ult
2025-08-15 06:46:24.425 INFO 1 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 9090 (http)
2025-08-15 06:46:24.448 INFO 1 --- [           main] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2025-08-15 06:46:24.455 INFO 1 --- [           main] org.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/9.0.41]
2025-08-15 06:46:24.637 INFO 1 --- [           main] o.a.c.c.[Tomcat].[localhost].[] : Initializing Spring embedded WebApplicationContext
2025-08-15 06:46:24.638 INFO 1 --- [           main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 2873
ms
2025-08-15 06:46:25.377 INFO 1 --- [           main] in.ashokit.rest.WelcomeRestController : ***** WelcomeRestController::Constructor *****
2025-08-15 06:46:25.745 INFO 1 --- [           main] o.s.s.concurrent.ThreadPoolTaskExecutor : Initializing ExecutorService 'applicationTaskExecutor'
2025-08-15 06:46:26.207 INFO 1 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 9090 (http) with context path ''
2025-08-15 06:46:26.240 INFO 1 --- [           main] in.ashokit.SbRestAppApplication      : Started SbRestAppApplication in 5.744 seconds (JVM running fo
r 6.779)

[ec2-user@ip-172-31-33-189 ~]$ docker images
REPOSITORY          TAG      IMAGE ID      CREATED        SIZE
ashokit/python-flask-app   latest   82781d5652bc  2 years ago   913MB
ashokit/spring-boot-rest-api   latest   fcb4841c55d1  3 years ago   178MB
[ec2-user@ip-172-31-33-189 ~]$ docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED          STATUS            PORTS
08ef99a6765d        ashokit/spring-boot-rest-api   "java -jar spring-bo..."   About a minute ago   Exited (130) 17 seconds ago
25b145c50a15        ashokit/spring-boot-rest-api   "java -jar spring-bo..."   2 minutes ago    Exited (130) About a minute ago
160dc7713932        82781d5652bc     "python app.py"       24 minutes ago   Up 24 minutes    0.0.0.0:5000->5000/tcp, :::5000->5000/tcp      great_rhodes
c391e6619951        fcb4841c55d1     "java -jar spring-bo..."   52 minutes ago   Up 52 minutes    0.0.0.0:9090->9090/tcp, :::9090->9090/tcp      blissful_williams
[ec2-user@ip-172-31-33-189 ~]$ docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED          STATUS            PORTS
08ef99a6765d        ashokit/spring-boot-rest-api   "java -jar spring-bo..."   About a minute ago   Exited (130) 17 seconds ago
25b145c50a15        ashokit/spring-boot-rest-api   "java -jar spring-bo..."   2 minutes ago    Exited (130) About a minute ago
160dc7713932        82781d5652bc     "python app.py"       24 minutes ago   Up 24 minutes    0.0.0.0:5000->5000/tcp, :::5000-
>5000/tcp      great_rhodes
c391e6619951        fcb4841c55d1     "java -jar spring-bo..."   52 minutes ago   Up 52 minutes    0.0.0.0:9090->9090/tcp, :::9090-
>9090/tcp      blissful_williams
545b2af10936        fcb4841c55d1     "java -jar spring-bo..."   54 minutes ago   Created
[ec2-user@ip-172-31-33-189 ~]$ |
```

## Cms:-Docker ps -q :- running docker images

```
ec2-user@ip-172-31-33-189:~$ docker ps -q
c391e6619951      fcb4841c55d1      "java -jar spring-bo..."   52 minutes ago   Up 52 minutes    0.0.0.0:9090->9090/tcp, :::9090-
>9090/tcp      blissful_williams
545b2af10936      fcb4841c55d1      "java -jar spring-bo..."   54 minutes ago   Created
[ec2-user@ip-172-31-33-189 ~]$ Read from remote host ec2-3-108-191-154.ap-south-1.compute.amazonaws.com: Connection reset by peer
Connection to ec2-3-108-191-154.ap-south-1.compute.amazonaws.com closed.
client_loop: send disconnect: Connection reset by peer

himanshu12@himanshu MINGW64 /d/OneDrive/Desktop/pem
$ 

himanshu12@himanshu MINGW64 /d/OneDrive/Desktop/pem
$ ssh -i "HR.pem" ec2-user@ec2-3-108-191-154.ap-south-1.compute.amazonaws.com
, _#_
~- \###_ Amazon Linux 2023
~- \###_
~- \##_
~- \|_,--> https://aws.amazon.com/linux/amazon-linux-2023
~~_/
~~_/_/ 
Last login: Fri Aug 15 06:44:41 2025 from 152.58.32.136
[ec2-user@ip-172-31-33-189 ~]$ 
[ec2-user@ip-172-31-33-189 ~]$ docker ps -q
160dc7713932
c391e6619951
[ec2-user@ip-172-31-33-189 ~]$ 
[ec2-user@ip-172-31-33-189 ~]$ 
[ec2-user@ip-172-31-33-189 ~]$ docker stop $(docker ps -q)
160dc7713932
c391e6619951
[ec2-user@ip-172-31-33-189 ~]$ |
```

Cmd : docker stop \$(docker ps -q)

## To stop all running container

# Docker ps

## Docker ps -a

**Cmd:** docker ps -a -f status=exited -q

# Stopped state container

## Delete container

```
Docker rm $(docker ps -a -f status=exited -q)
```

```
ec2-user@ip-172-31-33-189:~$ client_loop: send disconnect: Connection reset by peer
himanshu12@himanshu MINGW64 /d/onedrive/Desktop/pem
$ ssh -i "HR.pem" ec2-user@ec2-3-108-191-154.ap-south-1.compute.amazonaws.com
,           #
~~\_ #####_      Amazon Linux 2023
~~ \#####\
~~ \###|
~~   \|#/| https://aws.amazon.com/linux/amazon-linux-2023
~~   V~,-->
~~   ~~~
~~   ~~-.
~~   _/`_/
~~   _/m/
Last login: Fri Aug 15 07:15:08 2025 from 152.58.32.252
[ec2-user@ip-172-31-33-189 ~]$ docker rm $(docker ps -a -f status=exited -q)
08ef99a6765d
25b145c50a15
160dc7713932
c391e6619951
[ec2-user@ip-172-31-33-189 ~]$ docker ps
CONTAINER ID   IMAGE      COMMAND     CREATED      STATUS      PORTS      NAMES
[ec2-user@ip-172-31-33-189 ~]$ docker ps -a
CONTAINER ID   IMAGE      COMMAND     CREATED      STATUS      PORTS      NAMES
545b2af10936   fcb4841c55d1   "java -jar spring-bo..."   2 hours ago   Created
[ec2-user@ip-172-31-33-189 ~]$ |
```

# Docker images

```
c391e6619951
[ec2-user@ip-172-31-33-189 ~]$ Read from remote host ec2-3-108-191-154.compute.amazonaws.com: Connection to ec2-3-108-191-154.compute.amazonaws.com closed.
client_loop: send disconnect: Connection reset by peer
himanshu12@himanshu MINGW64 /d/OneDrive/Desktop/pem
$ ssh -i "HR.pem" ec2-user@ec2-3-108-191-154.compute.amazonaws.com
'          #_
~\_\_ #####_      Amazon Linux 2023
~~ \_\_\#\#\#\|_
~~ \#\#\#|_
~~ \#/ ,--> https://aws.amazon.com/linux/amazon-linux-2023
~~ V~ ,-->
~~ \_\_\_/
~~ \_\_\_/
~~ \_\_\_/
Last login: Fri Aug 15 07:15:08 2025 from 152.58.32.252
[ec2-user@ip-172-31-33-189 ~]$ docker rm $(docker ps -a -f status=exited -q)
08ef99a6765d
25b145c50a15
160dc7713932
c391e6619951
[ec2-user@ip-172-31-33-189 ~]$ docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
[ec2-user@ip-172-31-33-189 ~]$ docker ps -a
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
545b2af10936 fcb4841c55d1 "java -jar spring-bo..." 2 hours ago Created
[ec2-user@ip-172-31-33-189 ~]$ docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
ashokit/python-flask-app latest 82781d5652bc 2 years ago 913MB
ashokit/spring-boot-rest-api latest fcb4841c55d1 3 years ago 178MB
[ec2-user@ip-172-31-33-189 ~]$ 
[ec2-user@ip-172-31-33-189 ~]$
```

## Delete docker images

### Docker rmi (imageid)

```
/m'
Last login: Fri Aug 15 07:15:08 2025 from 152.58.32.252
[ec2-user@ip-172-31-33-189 ~]$ docker rm $(docker ps -a -f status=exited -q)
08ef99a6765d
25b145c50a15
160dc7713932
c391e6619951
[ec2-user@ip-172-31-33-189 ~]$ docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
[ec2-user@ip-172-31-33-189 ~]$ docker ps -a
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
545b2af10936 fcb4841c55d1 "java -jar spring-bo..." 2 hours ago Created
[ec2-user@ip-172-31-33-189 ~]$ docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
ashokit/python-flask-app latest 82781d5652bc 2 years ago 913MB
ashokit/spring-boot-rest-api latest fcb4841c55d1 3 years ago 178MB
[ec2-user@ip-172-31-33-189 ~]$ 
[ec2-user@ip-172-31-33-189 ~]$ docker rmi 82781d5652bc
Untagged: ashokit/python-flask-app:latest
Untagged: ashokit/python-flask-app@sha256:d455d0eb17a6723c540bda9ad01eee109cc9ccc2dd3db25082cf4741171e8dee
Deleted: sha256:82781d5652bcc595e7b3d624cf70cab40597d2fd992e01e46b09661c6db0d1
Deleted: sha256:ab05876319db52fb4b6742fd24d0bfef0ce5ab3e0774bb8b757d6224b6fd360b
Deleted: sha256:75744206862ad073451baeee3a306608e83332afeb065336b8b8868d1cb264a78f
Deleted: sha256:a455b5e6736316466237846f4c95e7bcc54ef1cb54c5c6f784f5485fb55cf4a
Deleted: sha256:378ed93bb92e9b5885debcf4b524f39ab0c470ff6eac35a01dab7a958a64552d
Deleted: sha256:a17a7c140e63f7aa416d2c8866dfa1d9d755fb1b41fdf79ea21d1591c461b3f
Deleted: sha256:007f5d362fd1bb395c9229c89593a3727fe3e600f508057d79b27c69aa8c492
Deleted: sha256:29a8de6a9ba3858e830b20b810f426bef144944a5c998ce4c53762ab9165e2d6
Deleted: sha256:cbece712ed17923285239f9d9c052894ae065b7413d68a0290e2c8eecc98f4a
Deleted: sha256:aa56d037ee5925ebf11127c3e1f617874c4ce8bae6b6af7d132b7f7a4a606e6f
Deleted: sha256:97e5f44efb543d466c5847602654a8cb22c9466b61d04988d47ec44b197ea874
Deleted: sha256:11936051f93baf5a4fb090a8fa0999309b8173556f7826598e235e8a82127bce
[ec2-user@ip-172-31-33-189 ~]$
```

# Delete docker images at a time

## Docker system prune -a

```
[ec2-user@ip-172-31-33-189~]$ Deleted: sha256:aa56d037ee5925ebf11127c3e1f617874c4ce8bae6b6af7d132b7f7a4a606e6f
Deleted: sha256:97e5f44efb543d466c5847602654a8cb22c9466b61d04988d47ec44b197ea874
Deleted: sha256:11936051f93baf5a4fb090a8fa0999309b8173556f7826598e235e8a82127bce
[ec2-user@ip-172-31-33-189~]$ [ec2-user@ip-172-31-33-189~]$ [ec2-user@ip-172-31-33-189~]$ [ec2-user@ip-172-31-33-189~]$ [ec2-user@ip-172-31-33-189~]$ [ec2-user@ip-172-31-33-189~]$ [ec2-user@ip-172-31-33-189~]$ docker system prune -a
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] y
Deleted Containers:
545b2af10936c0af76cf426f9583a39ba438bc096021871644c148e78175174b

Deleted Images:
untagged: ashokit/spring-boot-rest-api:latest
untagged: ashokit/spring-boot-rest-api@sha256:b428aab0b05521d716d4855a4ece37d026327a4010e5fb081ecce59888014c92
deleted: sha256:fcb4841c55d10837c8b7d22f96ad09fefeb80fba286a7c2487cc20a705d3f62
deleted: sha256:64433a79a768d40420f88677fcfd449adf3deb08f43cc73b94ff23cf69be1c54c
deleted: sha256:54bafde522f5ad1b7fbde52f742ba41d67ffa1bd362d36d6195ef652422d7950
deleted: sha256:91ec3f9643e1434e81656ba36202f1a3b7f5f62baf3ac2e657883eae89b11796
deleted: sha256:be3703612910bd0c49e7d69c76b580ca4cdbf15e2e30b384a2cb5b0eaeac2144
deleted: sha256:de2a0834244ea0d56bb79f8a978c917b6674c585df7b74e99eff4581912b36d
deleted: sha256:9f8566ee5135862dd980160c27bd7721448a6f7f385bbb81f7f001f1b78a5fbf

Total reclaimed space: 178.2MB
[ec2-user@ip-172-31-33-189~]$
```

## Docker images

```
[ec2-user@ip-172-31-33-189~]$ [ec2-user@ip-172-31-33-189~]$ [ec2-user@ip-172-31-33-189~]$ [ec2-user@ip-172-31-33-189~]$ [ec2-user@ip-172-31-33-189~]$ docker system prune -a
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] y
Deleted Containers:
545b2af10936c0af76cf426f9583a39ba438bc096021871644c148e78175174b

Deleted Images:
untagged: ashokit/spring-boot-rest-api:latest
untagged: ashokit/spring-boot-rest-api@sha256:b428aab0b05521d716d4855a4ece37d026327a4010e5fb081ecce59888014c92
deleted: sha256:fcb4841c55d10837c8b7d22f96ad09fefeb80fba286a7c2487cc20a705d3f62
deleted: sha256:64433a79a768d40420f88677fcfd449adf3deb08f43cc73b94ff23cf69be1c54c
deleted: sha256:54bafde522f5ad1b7fbde52f742ba41d67ffa1bd362d36d6195ef652422d7950
deleted: sha256:91ec3f9643e1434e81656ba36202f1a3b7f5f62baf3ac2e657883eae89b11796
deleted: sha256:be3703612910bd0c49e7d69c76b580ca4cdbf15e2e30b384a2cb5b0eaeac2144
deleted: sha256:de2a0834244ea0d56bb79f8a978c917b6674c585df7b74e99eff4581912b36d
deleted: sha256:9f8566ee5135862dd980160c27bd7721448a6f7f385bbb81f7f001f1b78a5fbf

Total reclaimed space: 178.2MB
[ec2-user@ip-172-31-33-189~]$ [ec2-user@ip-172-31-33-189~]$ [ec2-user@ip-172-31-33-189~]$ docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
[ec2-user@ip-172-31-33-189~]$ docker ps -a
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
[ec2-user@ip-172-31-33-189~]$ |
```

## Docker ps -a

### Docker container

```
ec2-user@ip-172-31-33-189:~$ docker ps -a
- 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] y
Deleted Containers:
545b2af10936c0af76cf426f9583a39ba438bc096021871644c148e78175174b

Deleted Images:
untagged: ashokit/spring-boot-rest-api:latest
untagged: ashokit/spring-boot-rest-api@sha256:b428aab0b05521d716d4855a4ece37d026327a4010e5
deleted: sha256:fcb4841c55d10837c8b7d22f96ad09fefeb80fba286a7c2487ccc20a705d3f62
deleted: sha256:64433a79a768d40420f88677fc449adf3deb08f43cc73b94ff23cf69be1c54c
deleted: sha256:54bafde522f5ad1b7fbde52f742ba41d67ffa1bd362d36d6195ef652422d7950
deleted: sha256:91ec3f9643e1434e81656ba36202f1a3b7f5f62baf3ac2e657883eae89b11796
deleted: sha256:be3703612910bd0c49e7d69c76b580ca4cdbf15e2e30b384a2cb5b0eaeac2144
deleted: sha256:de2a0834244ea0d56bb79f8a978c917b6674c585df7b74e99e9ff4581912b36d
deleted: sha256:9f8566ee5135862dd980160c27bd7721448a6f7f385bbb81f7f001f1b78a5fbf

Total reclaimed space: 178.2MB
[ec2-user@ip-172-31-33-189 ~]$ 
[ec2-user@ip-172-31-33-189 ~]$ docker images
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
[ec2-user@ip-172-31-33-189 ~]$ docker ps -a
CONTAINER ID      IMAGE      COMMAND      CREATED      STATUS      PORTS      NAMES
[ec2-user@ip-172-31-33-189 ~]$
```

**Note:** we can run docker image directly without pulling