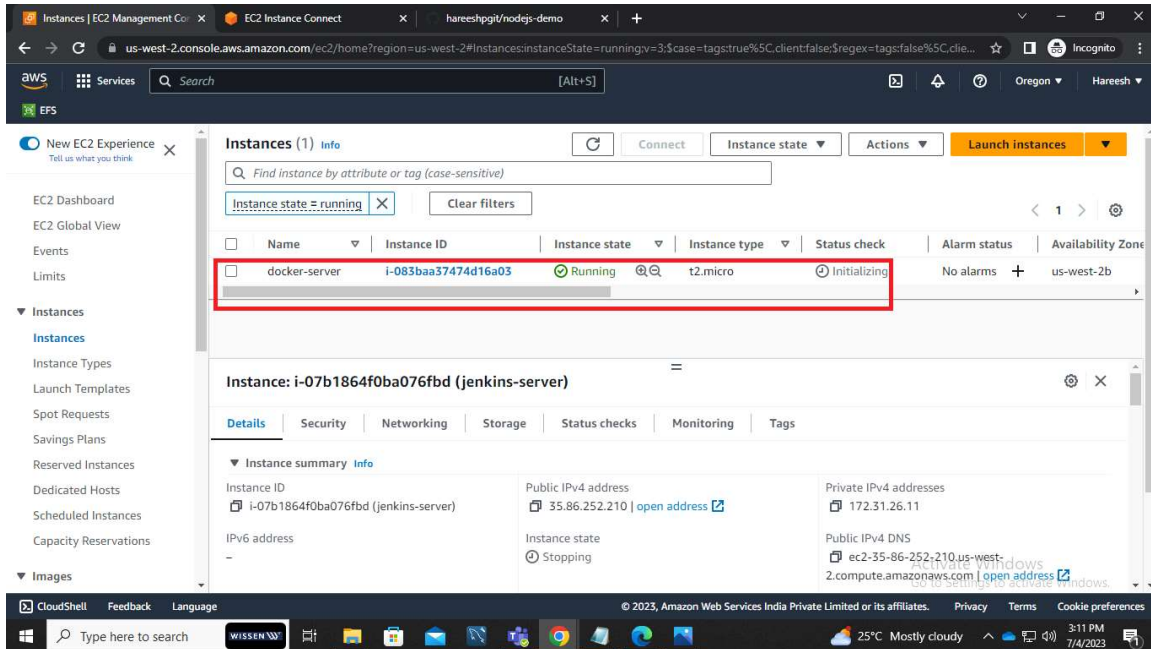


Creation of docker image using dockerfile and creation of docker image using jenkins pipeline:

### Task-1 -- create docker image using Dockerfile

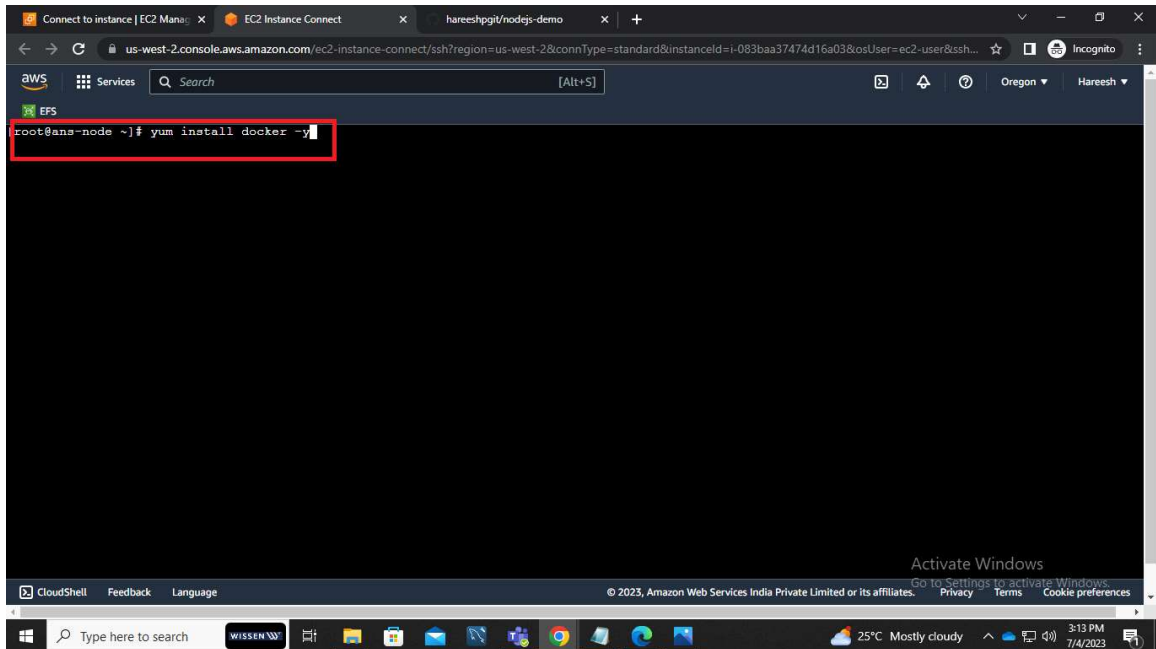
- Login to aws management console and launch the Docker-server



- Navigate to docker-server terminal and install the Docker

**CMD** --- sudo su -

--- yum install docker -y



- check the status of docker service using below commands
- After that start the docker service and enable the service as well

**CMD** -- systemctl status docker

-- systemctl start docker

-- systemctl enable docker

- create a docker file using required instructions to creation of image
- create a index.html file and write some content
- Here we taken nginx server inside dockerfile and copied the index.html content into nginx server

Dockerfile--- inside docker file content

FROM NGINX

COPY index.html /usr/share/nginx/html

- Build the image from dockerfile using below command

**CMD** -- docker build -t docker-image .

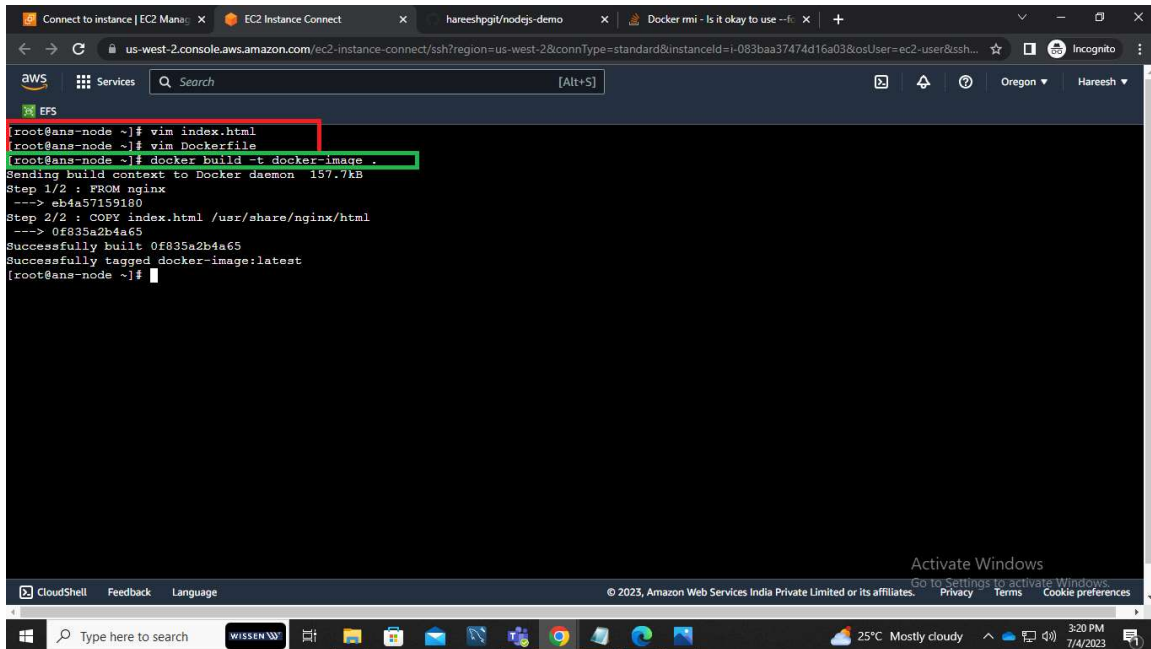
-- Here docker command start with docker

--build means building the image from dockerfile reference

-- -t means terminal

-- docker image means from these name image will be create after build happens--here we can give any meaning full name to docker image

-- . dot represents from current directory



The screenshot shows a terminal window within the AWS CloudShell interface. The terminal prompt is `root@ana-node ~`. The user has executed the following commands and received the following output:

```
root@ana-node ~]# vim index.html
root@ana-node ~]# vim Dockerfile
root@ana-node ~]# docker build -t docker-image .
Sending build context to Docker daemon 157.7kB
Step 1/2 : FROM nginx
--> eb4a57159180
Step 2/2 : COPY index.html /usr/share/nginx/html
--> 0f835a2b4a65
Successfully built 0f835a2b4a65
Successfully tagged docker-image:latest
root@ana-node ~]#
```

- Earlier we don't have any images in our Docker server; we want to see how many images are present for that we use the below command

**CMD--** `docker images`

- How many containers are present inside the Docker server; for that we use the below command

**CMD--** `docker ps -a`

- How many running containers are available inside the Docker server; we use the below command

**CMD--** `docker ps`

- After building the Dockerfile, we will get images; for that we use the `docker image` command

## Task-2 -- Run the container from Docker image

- To run the container from a Docker image, we use the below command

**CMD--** `docker run -d --name docker-container -p 80:80 docker-image`

--docker run means create image

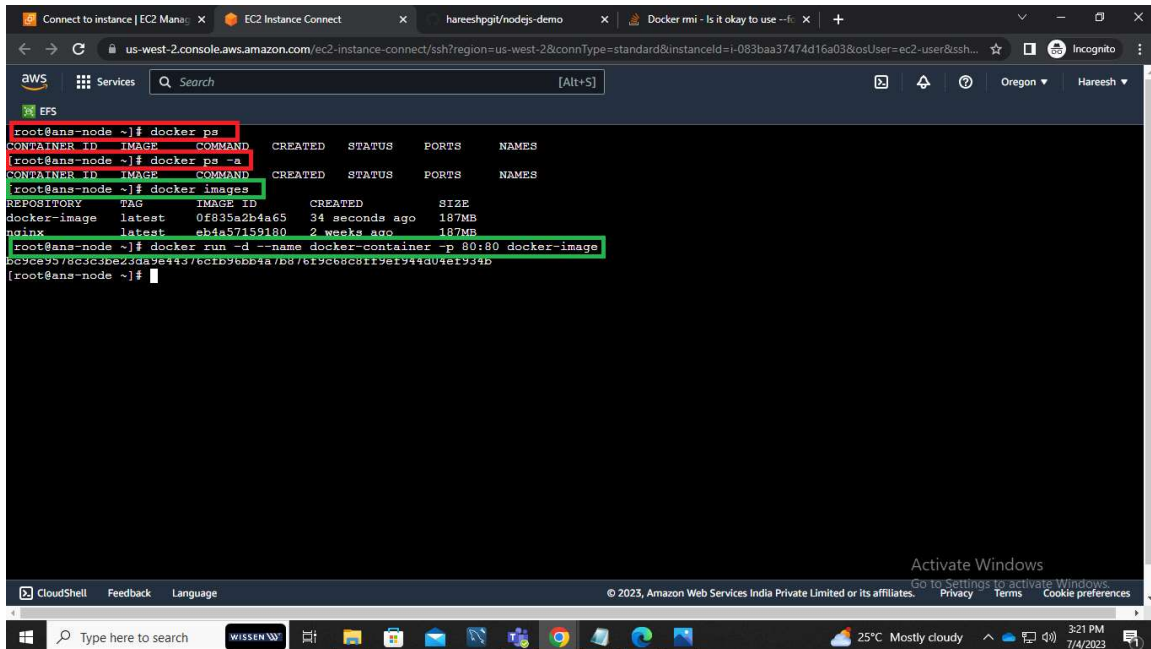
-- -d means detach mode

-- docker-container means container name while running the image; we can give a container name; otherwise, it will take random funny names.

-- -p means port -- here we can add the required port number to access the application inside the

running container

-- docker-image means from this image we create a container



The screenshot shows a terminal window with the following commands and output:

```
root@sns-node ~]# docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
root@sns-node ~]# docker ps -a
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
root@sns-node ~]# docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
docker-image   latest    0f835a2b4a65   34 seconds ago 187MB
nginx         latest    eb457159180    2 weeks ago   187MB
root@sns-node ~]# docker run -d --name docker-container -p 80:80 docker-image
dc9ce9576c3c3be23da9e44376c1b96b4a7b57619c68c61f961944d04e1934b
[root@sns-node ~]#
```

- Check the containers available in docker server
- For enter into inside running container we use below command

**CMD--** docker exec -it docker-container /bin/bash

-- docker exec is a command that allows the execution of any given command within a Docker container

-- -it --- -i means interactive and -t means terminal

-- /bin is a directory

-- /bash means shell

- After run the above command we enter into inside running container
- Here we can see list of files and directories
- we can create files and run applications

The screenshot shows a terminal window within the AWS CloudShell interface. The terminal output is as follows:

```
root@sns-node ~# docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                    NAMES
bc9ce9578c3c   docker-image   "/docker-entrypoint..." 14 seconds ago Up 13 seconds  0.0.0.0:80->80/tcp, :::80->80/tcp  docker-container

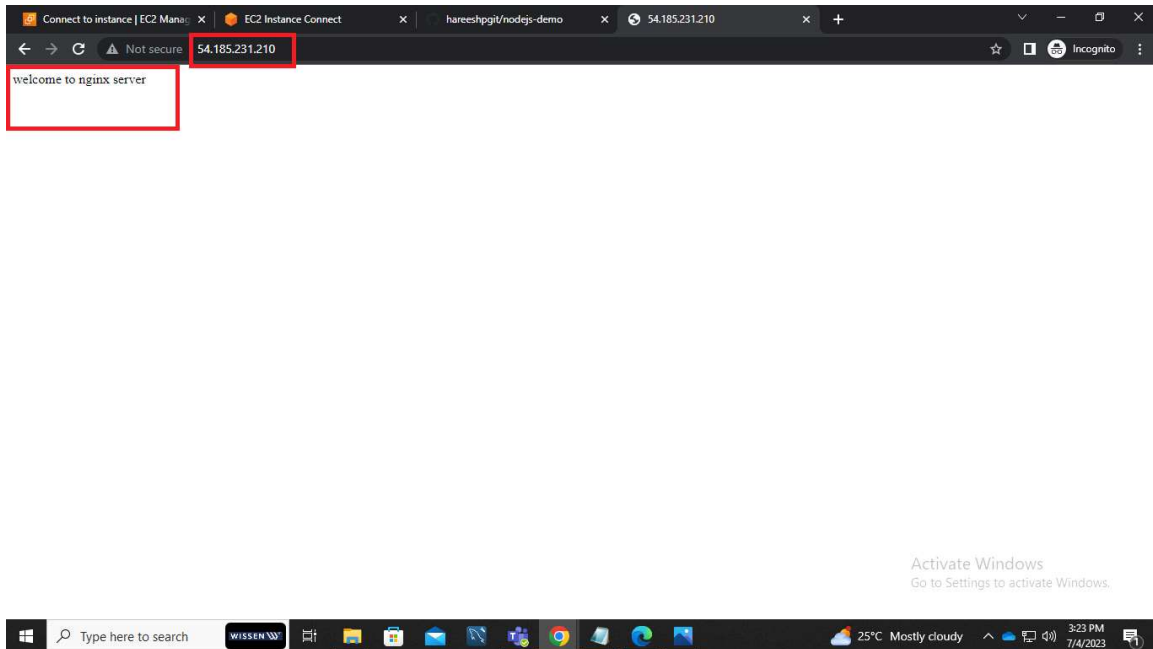
root@sns-node ~# docker exec -it docker-container /bin/bash
root@bc9ce9578c3c:/# whoami
root
root@bc9ce9578c3c:/# ls
bin      dev      docker-entrypoint.sh  home  lib32  libx32  mnt  proc  run  srv  tmp  var
boot    docker-entrypoint.d  etc      lib   lib64  media   opt  root  sbin sys  usr
```

The screenshot shows the same terminal window with the following additional output:

```
root@bc9ce9578c3c:/# touch file1
root@bc9ce9578c3c:/# ls
bin      dev      docker-entrypoint.sh  file1  lib   lib64  media   opt  root  sbin  sys  usr
boot    docker-entrypoint.d  etc      home   lib32 libx32  mnt     proc  run  srv   tmp  var
```

- Take the public ip of docker-server and paste in the browser
- our nginx server is running and what ever content we kept inside the index.html file

it was displaying

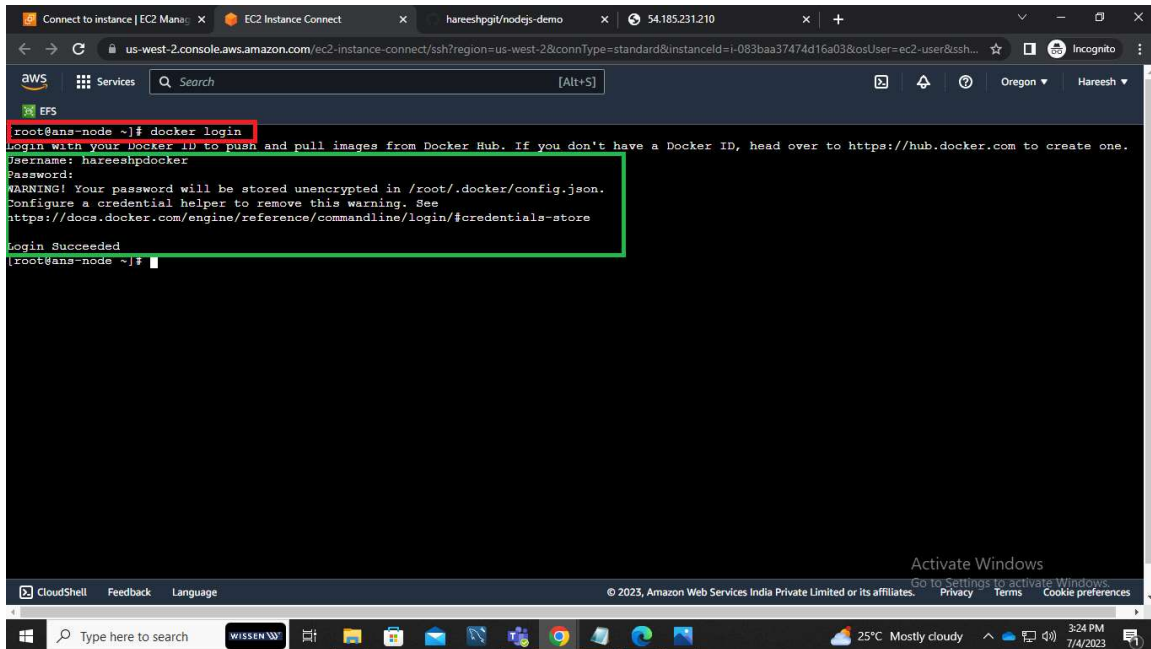


### Task -3 -- push docker image to dockerhub

- For push the docker image to docker hub we need to login the docker hub
- For that we use below command

**CMD--** docker login

- docker login command is use to login into dockerhub
- it will ask the user name and password of dockerhub
- When we enter credentials only we can push the images



```
root@ans-node ~]# 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: hareeshpdocker
Password:
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

login Succeeded
[root@ans-node ~]#
```

- Before push the image check the images available in our local server
- Here we need to tag the image then after we need to push the image
- For that we use commands

**CMD--** `docker tag docker-image:latest hareeshpdocker/nginx-server-repo:newtag`

`-- docker push hareeshpdocker/nginx-server-repo:newtag`

- `docker push` is push the image to dockerhub

`-- hareeshpdocker` is our docker server user name

`-- nginx-server-repo` is our dockerhub repository we need to create that repository

`-- newtag` is a tag to that image



```
Connect to instance | EC2 Manag... x EC2 Instance Connect x hareeshpgit/nodejs-demo x 54.185.231.210 x +
us-west-2.console.aws.amazon.com/ec2-instance-connect/ssh?region=us-west-2&connType=standard&instanceId=i-083baa3747d16a03&osUser=ec2-user&ssh...
AWS Services Search [Alt+S] Oregon Hareesh
EFS
Username: hareeshpdocker
Password:
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store
Login Succeeded
root@ans-node ~# docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
docker-image latest 0f835a2b4a65 5 minutes ago 187MB
nginx latest eb4a57159180 2 weeks ago 187MB
root@ans-node ~# docker tag docker-image:latest hareeshpdocker/nginx-server-repo:newtag
root@ans-node ~# docker push hareeshpdocker/nginx-server-repo:newtag
The push refers to repository [docker.io/hareeshpdocker/nginx-server-repo]
a55312067dd9: Pushed
9e9e226c58e7: Mounted from hareeshpdocker/docker-nginx
12a568acc014: Mounted from hareeshpdocker/docker-nginx
77b7099e19d2: Mounted from hareeshpdocker/docker-nginx
bf8b62fb2f13: Mounted from hareeshpdocker/docker-nginx
4ca29ffc4a01: Mounted from hareeshpdocker/docker-nginx
a83110139647: Mounted from hareeshpdocker/docker-nginx
ac4d164fef90: Mounted from hareeshpdocker/docker-nginx
newtag: digest: sha256:3079d30bfb106f71f8ae19a04fad26c80437af3659e1d0b75bd342de9a2d2d58 size: 1985
[root@ans-node ~]#
```

- login to docker hub and verify the images

Adrenalin MAX x Email - Hareesh Pathi x Instances | EC2 Manag... x nodejs-demo/enkins... x Using Docker with Pip... x hareeshpdocker/nginx... x +

hub.docker.com/repository/docker/hareeshpdocker/nginx-server-repo/general

DockerCon 2023: Our annual developer event is back – online & in person. [Learn more.](#)

Search Docker Hub Explore Repositories Organizations Help Upgrade hareeshpdocker

hareeshpdocker Repositories nginx-server-repo General Using 0 of 1 private repositories. [Get more](#)

General Tags Builds Collaborators Webhooks Settings

**hareeshpdocker / nginx-server-repo**

Description

This repository does not have a description

Last pushed: a few seconds ago

Docker commands

To push a new tag to this repository,

```
docker push hareeshpdocker/nginx-server-repo:tagname
```

Public View

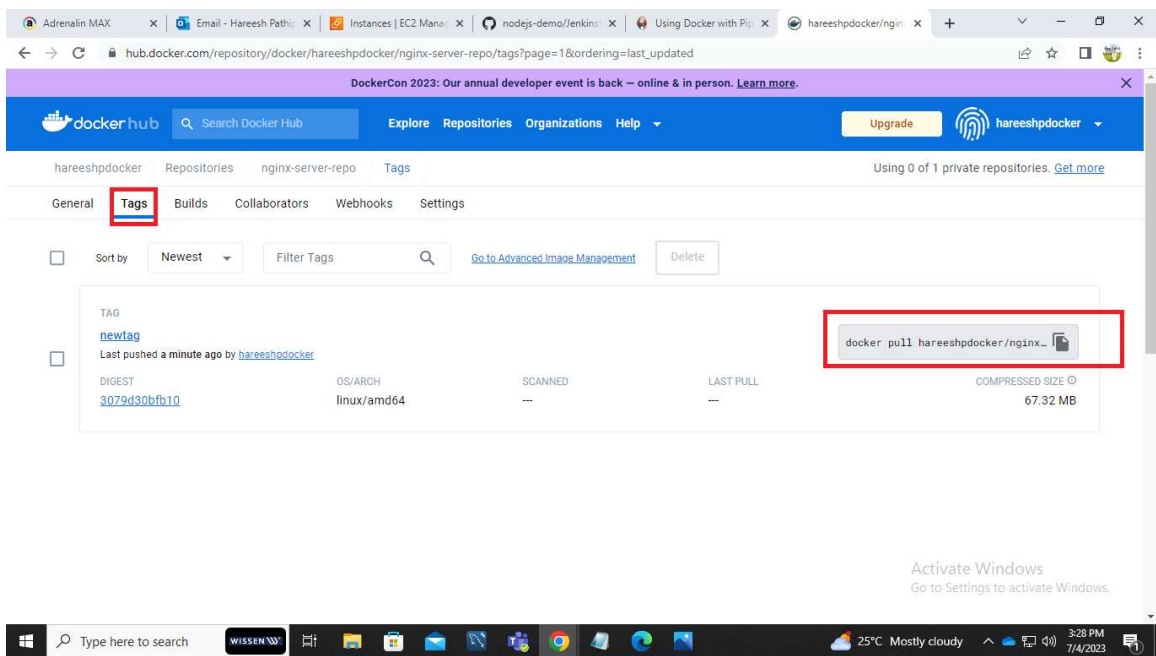
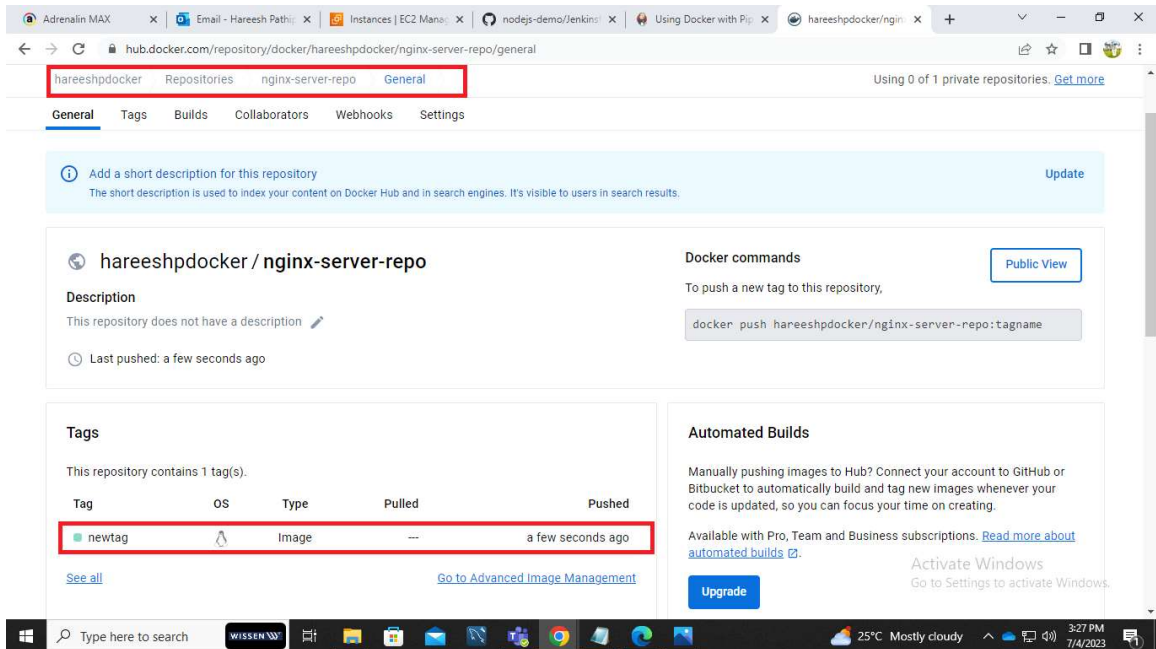
Tags

This repository contains 1 tag(s).

Tag	OS	Type	Pulled	Pushed
newtag				

Automated Builds

Manually pushing images to Hub? Connect your account to GitHub or Bitbucket to automatically build and tag new images whenever your code is updated, so you can focus your time on creating.



#### Task-4 --pull image from docker hub

- pull the image from docker hub to local server we use below command

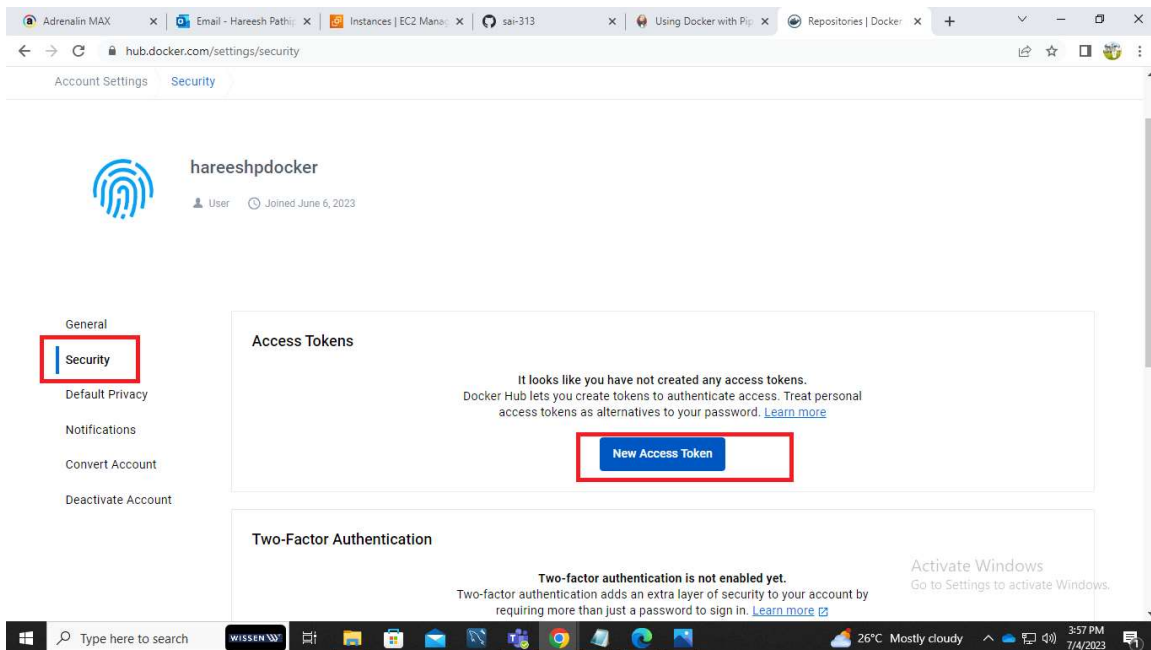
**CMD--** docker pull hareeshpdocker/nginx-server-repo:newtag

```
root@ans-node ~]# docker pull hareeshpdocker/nginx-server-repo:newtag
newtag: Pulling from hareeshpdocker/nginx-server-repo
Digest: sha256:3079d30bfb106f71f8ae19a04fad26c80437af3659e1d0b75bd342de9a2d2d58
Status: Image is up to date for hareeshpdocker/nginx-server-repo:newtag
docker.io/hareeshpdocker/nginx-server-repo:newtag
[root@ans-node ~]# docker images
```

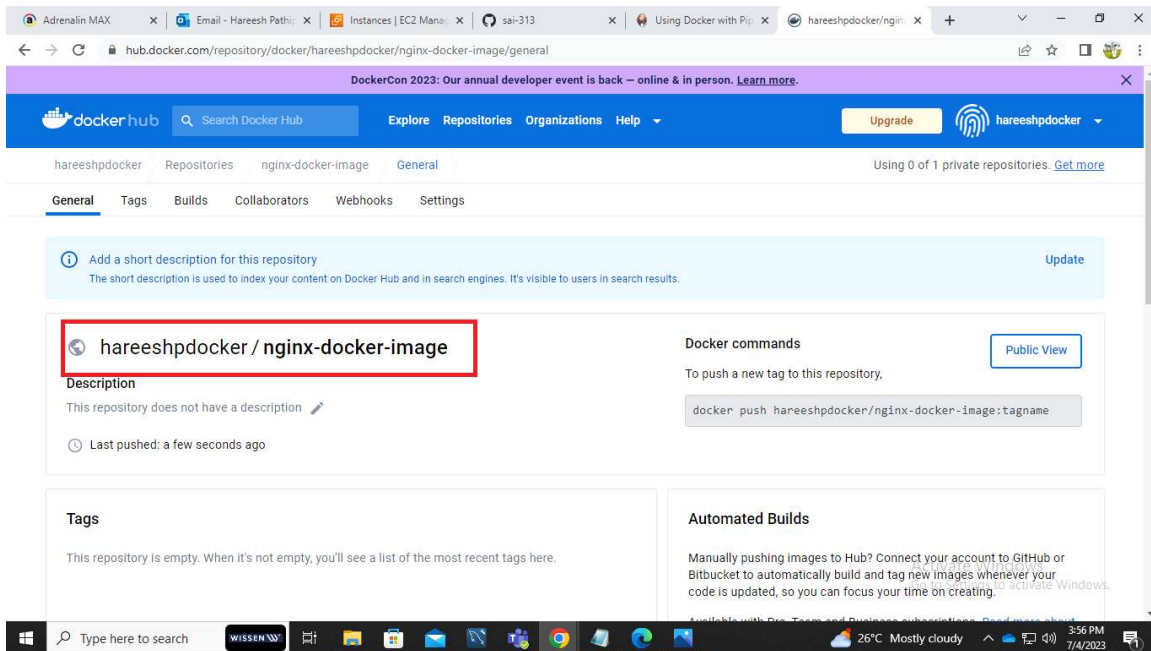
REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
hareeshpdocker/nginx-server-repo	newtag	0f835a2b4a65	8 minutes ago	187MB
docker-image	latest	0f835a2b4a65	8 minutes ago	187MB
nginx	latest	eb4a57159180	2 weeks ago	187MB

## Task-5-- docker image creation using jenkins pipeline

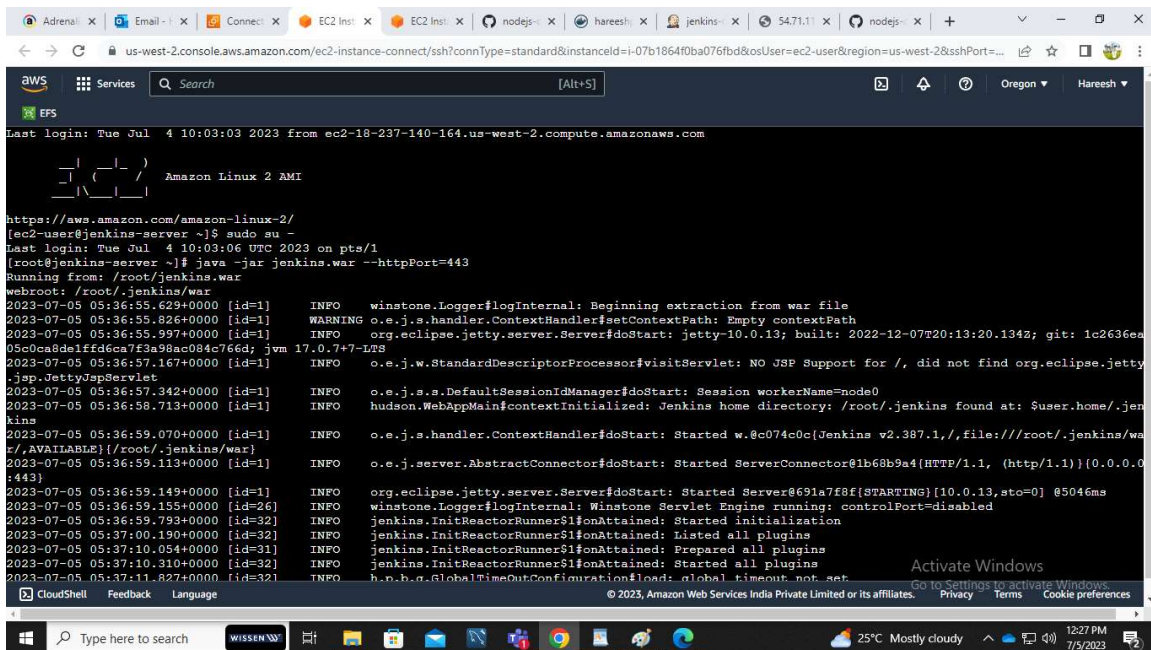
- In this task we need to add docker hub credentials inside jenkins server
- For that we need to create a token inside docker hub
- create a new token and copy that token in to credentials of jenkins



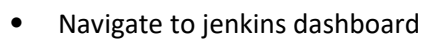
- Before creating the docker image we need to create a repository in docker hub
- During push the image it will added into created repository in side docker hub



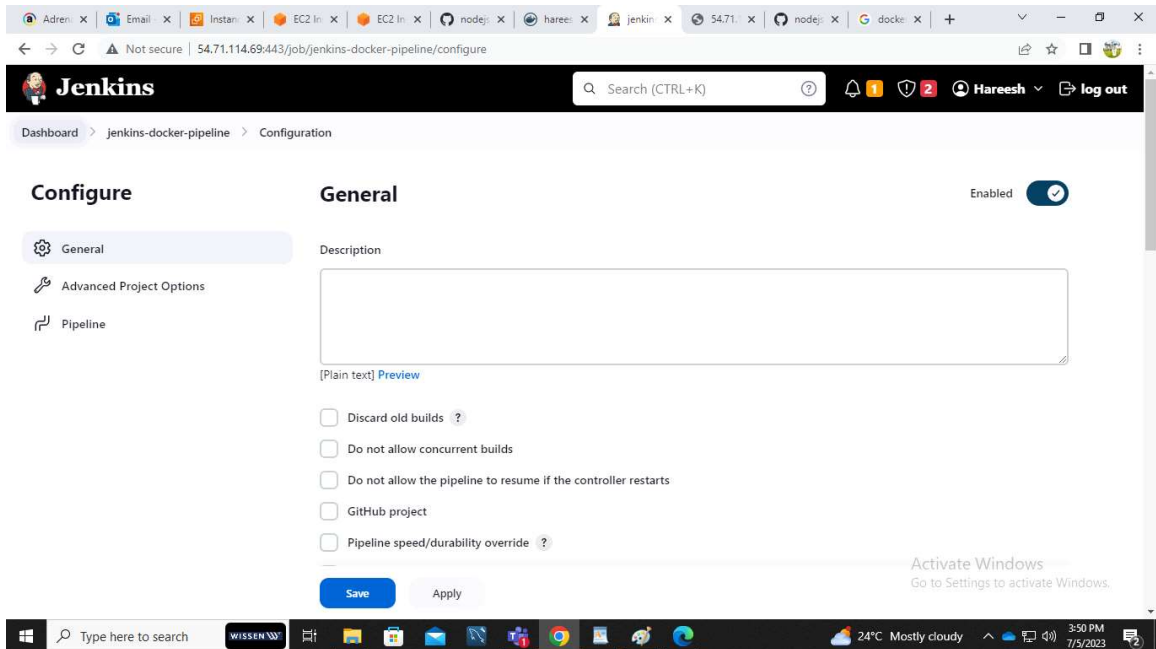
- create a jenkins-server and run jenkins-server



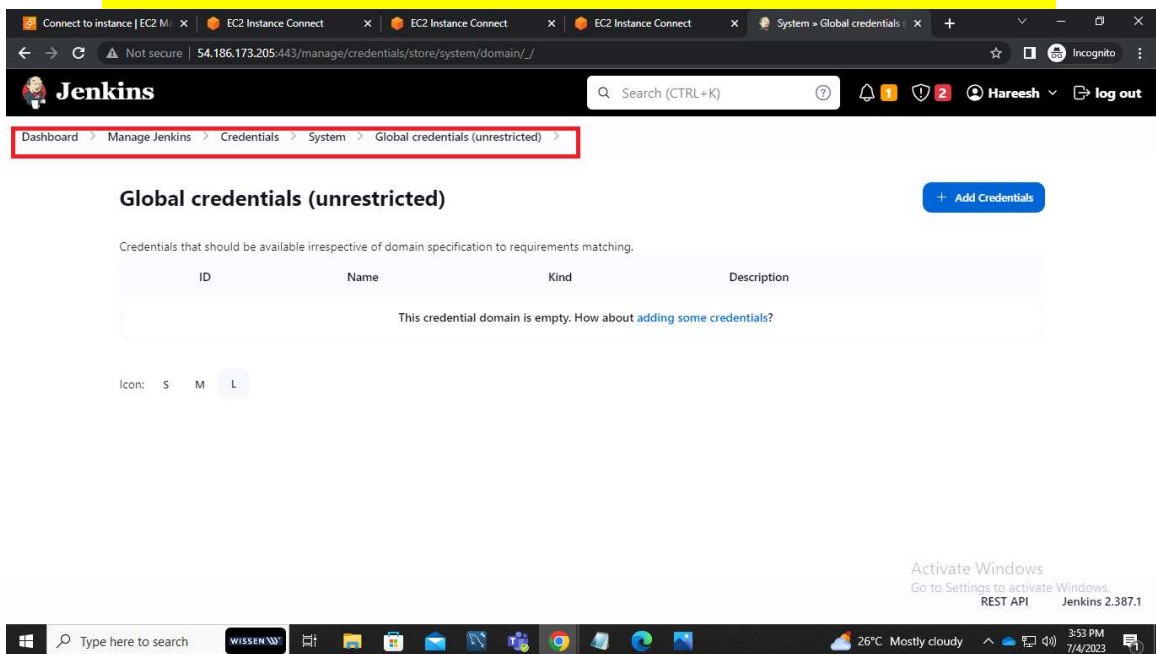
- ```
-- docker ps -a
```







- Before creation of job we need to set the credentials for that follow the below options
- Select -Dashboard > Manage jenkins > Credentials > system > Global credentials



- Here we add the credentials of docker server
- Here username section we can add docker hub user name

- password section we can add token of docker hub

Dashboard > Manage Jenkins > Credentials > System > Global credentials (unrestricted)

### New credentials

Kind: Username with password

Scope: Global (Jenkins, nodes, items, all child items, etc)

Username: hareeshpdocker

☐ Treat username as secret

Password: [masked]

ID: [empty]

**Create**

- Finally we get credential we use this one in our jenkins file to creation of docker images and login purpose

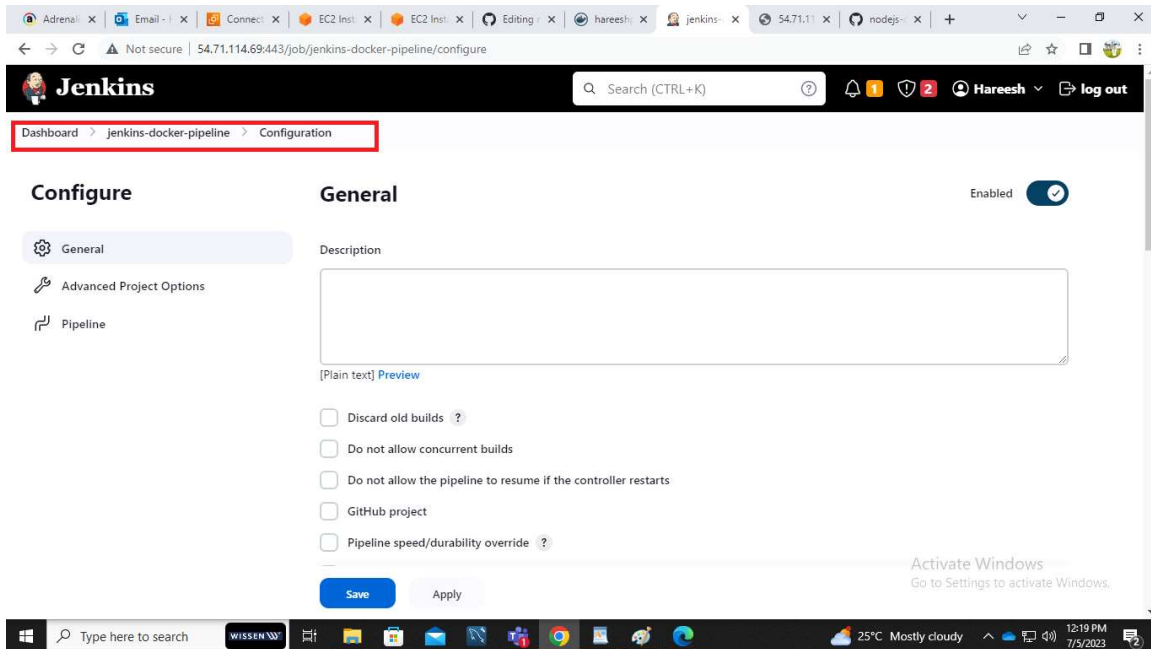
Dashboard > Manage Jenkins > Credentials > System > Global credentials (unrestricted) > hareeshpdocker/\*\*\*\*\* (docker-hub-hareeshpdocker)

### hareeshpdocker/\*\*\*\*\* (docker-hub-hareeshpdocker)

**Usage**

This credential has not been recorded as used anywhere.  
*Note: usage tracking requires the cooperation of plugins and consequently may not track every use.*

- Navigate to Dashboard and create job and configure that job



- Here we selected the pipeline job
- After we select the pipeline script section
- Here we can select pipeline script -- manually we can write script
- second one is directly we can add from SCM -- github
- Here we selected pipeline script
- In this script type is declarative pipeline
- In this script we added credentials of jenkins
- in the first stage we added scm - git hub url -- here we have docker file and jenkins file.  
inside docker file we write the instructions.

docker file--

FROM nginx

COPY index.html /usr/share/nginx/html

- second stage we need to build docker image
- third stage login to docker hub



- Adrenal X Email - X Connect X EC2 Inst X EC2 Inst X Editing X hareesh X jenkins X 54.71.114.69:443 X nodejs X +

Not secure | 54.71.114.69:443/job/jenkins-docker-pipeline/configure

Dashboard > jenkins-docker-pipeline > Configuration

## Configure

General

Advanced Project Options

Pipeline

Definition

Pipeline script

Pipeline script

Pipeline script from SCM

```
1 pipeline {
2   agent any
3   environment {
4     DOCKERHUB_CREDENTIALS = credentials('docker-hub-hareeshdocker')
5   }
6   stages {
7     stage('SCM Checkout') {
8       steps {
9         git 'https://github.com/hareeshpgit/nodejs-demo.git'
10      }
11    }
12
13    stage('Build docker image') {
14      steps {
15        sh 'docker build -t hareeshdocker/nginx-docker-image:$BUILD_NUMBER .'
16      }
17    }
18    stage('login to dockerhub') {
19      steps {
20        sh "echo $DOCKERHUB_CREDENTIALS_PSW | docker login -u $DOCKERHUB_CREDENTIALS_USR --password-stdin"
21      }
22    }
23    stage('push image') {
24      steps {
25        sh 'docker push hareeshdocker/nginx-docker-image:$BUILD_NUMBER'
26      }
27    }
28    stage('pull image') {
29      steps {
30        sh 'docker pull hareeshdocker/nginx-docker-image:$(cat /dev/urandom | tr -dc a-z0-9 | fold -w 32 | xargs)'
31      }
32    }
33  }
34 }
```

Activate Windows  
Go to Settings to activate Windows.

- Here we can see the output of our pipeline job and stages

Dashboard > jenkins-docker-pipeline >

Stage View

Build History

Filter builds...

#1 Jul 5, 2023, 6:22 AM

Atom feed for all Atom feed for failures

Permalinks

- Last build (#1), 36 sec ago
- Last stable build (#1), 36 sec ago
- Last successful build (#1), 36 sec ago
- Last completed build (#1), 36 sec ago

| SCM Checkout | Build docker image | login to dockerhub | push image | pull image | run image | Declarative: Post Actions |
|--------------|--------------------|--------------------|------------|------------|-----------|---------------------------|
| 1s           | 321ms              | 877ms              | 2s         | 1s         | 867ms     | 341ms                     |

- Take the public ip of docker server and paste in the browser see the output

hello team this docker image

- Login to docker hub verify the pushed image from pipeline.

Adrenaline x Email x Connect x EC2 Inst x EC2 Inst x nodejs x hareesh x jenkins x 54.71.11 x nodejs x

hub.docker.com/repository/docker/hareeshpdocker/nginx-docker-image/tags?page=1&ordering=last\_updated

DockerCon 2023: Our annual developer event is back – online & in person. [Learn more.](#)

dockerhub Search Docker Hub Explore Repositories Organizations Help Upgrade hareeshpdocker

hareeshpdocker Repositories nginx-docker-image Tags Using 0 of 1 private repositories. [Get more](#)

General **Tags** Builds Collaborators Webhooks Settings

☐ Sort by Newest Filter Tags [Go to Advanced Image Management](#) Delete

| TAG                          |                                                              |         |                |                 |                                                  |
|------------------------------|--------------------------------------------------------------|---------|----------------|-----------------|--------------------------------------------------|
| 1                            | Last pushed 29 minutes ago by <a href="#">hareeshpdocker</a> |         |                |                 | <code>docker pull hareeshpdocker/nginx...</code> |
| DIGEST                       | OS/ARCH                                                      | SCANNED | LAST PULL      | COMPRESSED SIZE |                                                  |
| <a href="#">4b66e6503e9e</a> | linux/amd64                                                  | ---     | 29 minutes ago | 67.32 MB        |                                                  |

Activate Windows  
Go to Settings to activate Windows.

Type here to search WISSEN 25°C Mostly cloudy 12:21 PM 7/5/2023