DEVOPS TRAINING

DAY – 2

Step 1: Install Docker & Docker Compose

Step 1.1 Install Docker

Run the following commands to install Docker:

sudo apt update

sudo apt install -y docker.io

Enable and start Docker:

sudo systemctl enable docker

sudo systemctl start docker

Step 1.2 Install Docker Compose

Download and install Docker Compose:

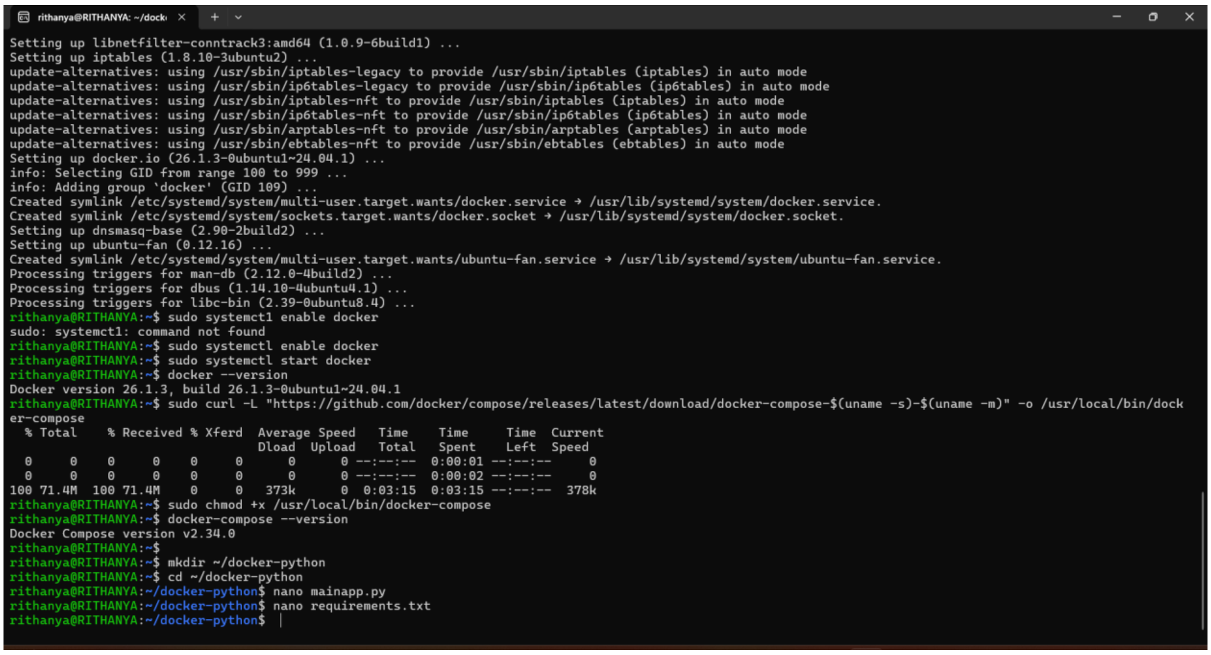
sudo curl -L

"https://github.com/docker/compose/releases/latest/download/docker-compose-

$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose

Give execution permission:

sudo chmod +x /usr/local/bin/docker-compose



```
Setting up libnetfilter-conntrack3:amd64 (1.0.9-6build1) ...
Setting up iptables (1.8.10-3ubuntu2) ...
update-alternatives: using /usr/sbin/iptables-legacy to provide /usr/sbin/iptables (iptables) in auto mode
update-alternatives: using /usr/sbin/ip6tables-legacy to provide /usr/sbin/ip6tables (ip6tables) in auto mode
update-alternatives: using /usr/sbin/iptables-nft to provide /usr/sbin/iptables (iptables) in auto mode
update-alternatives: using /usr/sbin/ip6tables-nft to provide /usr/sbin/ip6tables (ip6tables) in auto mode
update-alternatives: using /usr/sbin/arptables-nft to provide /usr/sbin/arptables (arptables) in auto mode
update-alternatives: using /usr/sbin/ebtables-nft to provide /usr/sbin/ebtables (ebtables) in auto mode
Setting up docker.io (26.1.3-0ubuntu1~24.04.1) ...
info: Selecting GID from range 100 to 999 ...
info: Adding group `docker' (GID 109) ...
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /usr/lib/systemd/system/docker.service.
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket → /usr/lib/systemd/system/docker.socket.
Setting up dnsmasq-base (2.90-2build2) ...
Setting up ubuntu-fan (0.12.16) ...
Created symlink /etc/systemd/system/multi-user.target.wants/ubuntu-fan.service → /usr/lib/systemd/system/ubuntu-fan.service.
Processing triggers for man-db (2.12.0-4build2) ...
Processing triggers for dbus (1.14.10-4ubuntu4.1) ...
Processing triggers for libc-bin (2.39-0ubuntu8.4) ...
rithanya@RITHANYA:~$ sudo systemct1 enable docker
sudo: systemct1: command not found
rithanya@RITHANYA:~$ sudo systemctl enable docker
rithanya@RITHANYA:~$ sudo systemctl start docker
rithanya@RITHANYA:~$ docker --version
Docker version 26.1.3, build 26.1.3-0ubuntu1~24.04.1
rithanya@RITHANYA:~$ sudo curl -L "https://github.com/docker/compose/releases/latest/download/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/dock
er-compose
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
  0     0    0     0    0     0      0      0 --:--:--  0:00:01 --:--:--     0
  0     0    0     0    0     0      0      0 --:--:--  0:00:02 --:--:--     0
100 71.4M  100 71.4M    0     0   373k      0  0:03:15  0:03:15 --:--:--  378k
rithanya@RITHANYA:~$ sudo chmod +x /usr/local/bin/docker-compose
rithanya@RITHANYA:~$ docker-compose --version
Docker Compose version v2.34.0
rithanya@RITHANYA:~$
rithanya@RITHANYA:~$ mkdir ~/docker-python
rithanya@RITHANYA:~$ cd ~/docker-python
rithanya@RITHANYA:~/docker-python$ nano mainapp.py
rithanya@RITHANYA:~/docker-python$ nano requirements.txt
rithanya@RITHANYA:~/docker-python$
```

Step 2: create the python application file

2.1 Create a Project Directory

mkdir ~/docker-python-app

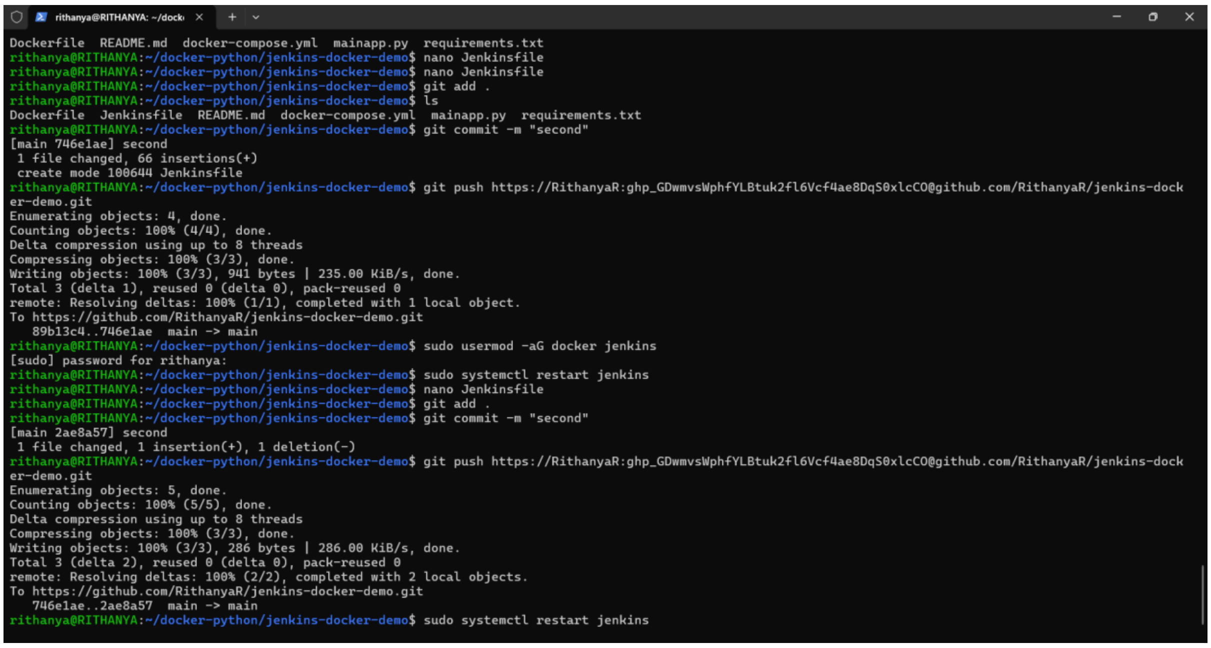
cd ~/docker-python-app

2.2 Create the Python Application File

Step 3: Create a Requirements File

Step 4: Create a Dockerfile

Step 5: Create a Docker Compose File



```
Dockerfile  README.md  docker-compose.yml  mainapp.py  requirements.txt
rithanya@RITHANYA:~/docker-python/jenkins-docker-demo$ nano Jenkinsfile
rithanya@RITHANYA:~/docker-python/jenkins-docker-demo$ nano Jenkinsfile
rithanya@RITHANYA:~/docker-python/jenkins-docker-demo$ git add .
rithanya@RITHANYA:~/docker-python/jenkins-docker-demo$ ls
Dockerfile  Jenkinsfile  README.md  docker-compose.yml  mainapp.py  requirements.txt
rithanya@RITHANYA:~/docker-python/jenkins-docker-demo$ git commit -m "second"
[main 746e1ae] second
 1 file changed, 66 insertions(+)
 create mode 100644 Jenkinsfile
rithanya@RITHANYA:~/docker-python/jenkins-docker-demo$ git push https://RithanyaR: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@github.com/RithanyaR/jenkins-dock
er-demo.git
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 941 bytes | 235.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/RithanyaR/jenkins-docker-demo.git
   89b13c4..746e1ae  main -> main
rithanya@RITHANYA:~/docker-python/jenkins-docker-demo$ sudo usermod -aG docker jenkins
[sudo] password for rithanya:
rithanya@RITHANYA:~/docker-python/jenkins-docker-demo$ sudo systemctl restart jenkins
rithanya@RITHANYA:~/docker-python/jenkins-docker-demo$ nano Jenkinsfile
rithanya@RITHANYA:~/docker-python/jenkins-docker-demo$ git add .
rithanya@RITHANYA:~/docker-python/jenkins-docker-demo$ git commit -m "second"
[main 2ae8a57] second
 1 file changed, 1 insertion(+), 1 deletion(-)
rithanya@RITHANYA:~/docker-python/jenkins-docker-demo$ git push https://RithanyaR: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@github.com/RithanyaR/jenkins-dock
er-demo.git
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 286 bytes | 286.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/RithanyaR/jenkins-docker-demo.git
   746e1ae..2ae8a57  main -> main
rithanya@RITHANYA:~/docker-python/jenkins-docker-demo$ sudo systemctl restart jenkins
```

Step 6: Build and Run the Docker Container

6.1 Build the Image

6.2 Start the Container

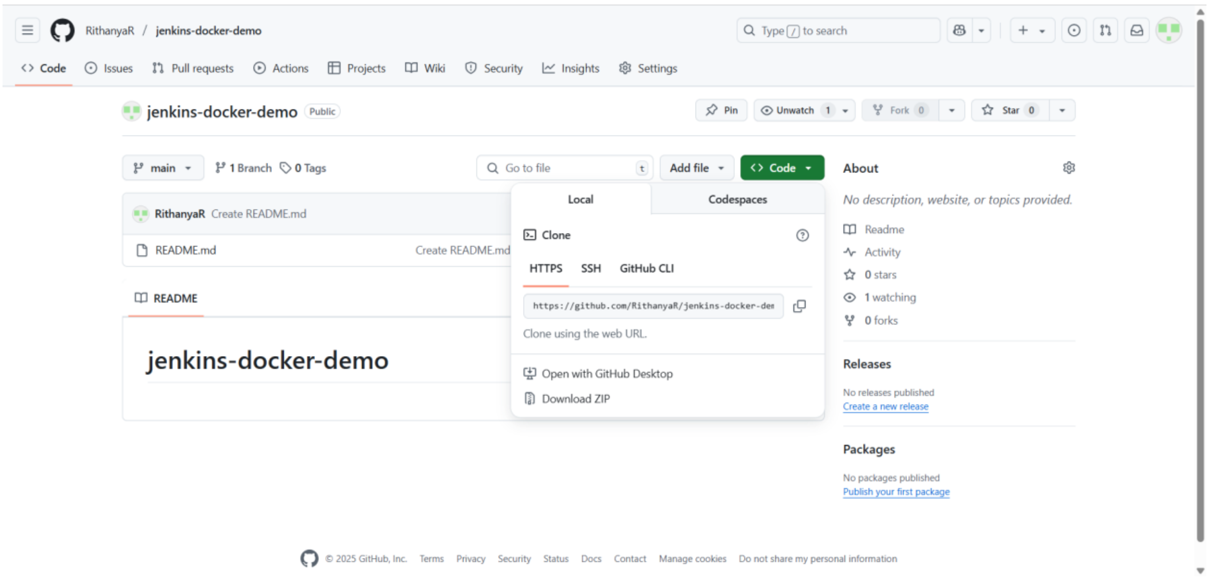
Step 7: Test the Application

Access the application in your web browser:

http://localhost:5000

Step 8:

Create a repository



```
https://github.com/RithanyaR/jenkins-docker-dem
```

Clone using the web URL.

No description, website, or topics provided.

No releases published

No packages published

Generate the token

1. Generate **Personal access tokens(PAT) (classic)** for git access-

Url: https://github.com/settings/tokens

select repo, admin:repo_hook and workflow

2. Upload JenkinsFile to github repo

git fetch

git pull

git status

git add –all

git commit -m “message”

git push

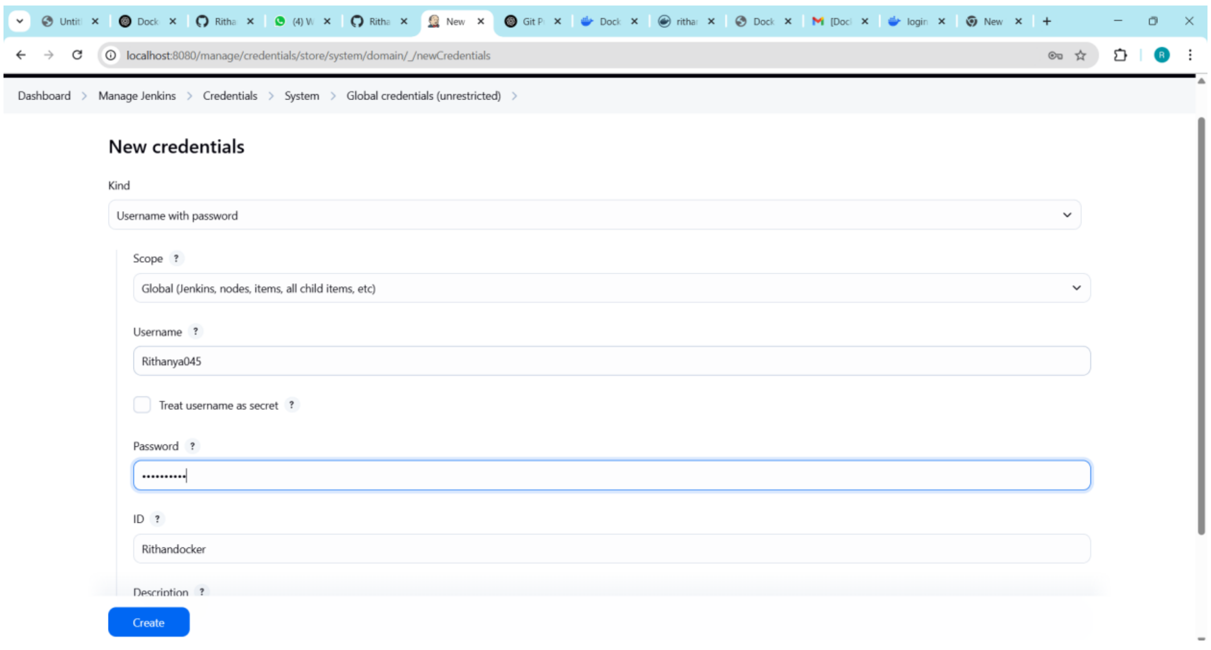
Error on git push? Try to execute below command

git push https://<username>:<PAT>@github.com/<user-name>/<repo-name>.git

Step 9:

For Pipeline Jobs (Jenkinsfile-Based):

1. Go to Jenkins → New Item → Select Pipeline.
2. Under Pipeline Definition, choose Pipeline script from SCM.
3. Select Git and enter the repository URL.
4. Click Add Credentials → Select your GitHub credentials.
5. Enter the Jenkinsfile path (e.g., Jenkinsfile).
6. Click Save and Build Now.



localhost:8080/manage/credentials/store/system/domain/_/newCredentials

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

New credentials

Kind: Username with password

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

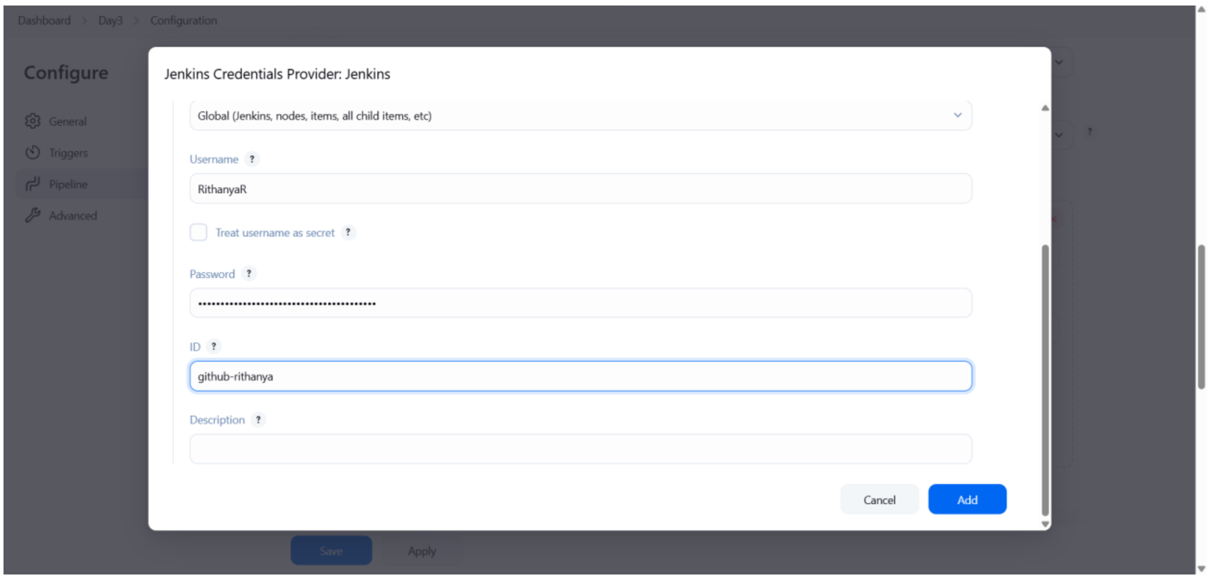
Username: Rithanya045

Treat username as secret

Password: ••••••••••

ID: Rithandocker

Description



Dashboard > Day3 > Configuration

Jenkins Credentials Provider: Jenkins

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

Username: RithanyaR

Treat username as secret

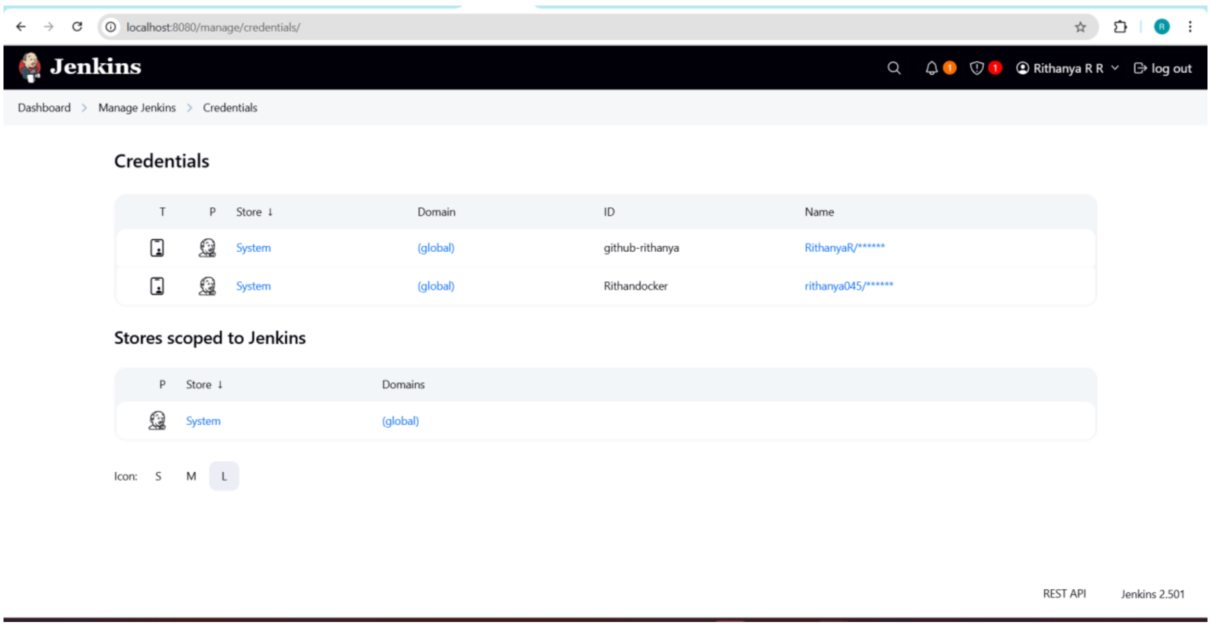
Password: ••

ID: github-rithanya

Description

1. Open Jenkins Credentials Management

1. Go to Jenkins Dashboard
2. Click Manage Jenkins → Manage Credentials
3. Choose (global) or a specific folder
4. Click Add Credentials



localhost:8080/manage/credentials/

Jenkins — Rithanya R R — log out

Dashboard > Manage Jenkins > Credentials

Credentials

T	P	Store ↓	Domain	ID	Name
		System	(global)	github-rithanya	RithanyaR/******
		System	(global)	Rithandocker	rithanya045/******

Stores scoped to Jenkins

P	Store ↓	Domains
	System	(global)

Icon: S M L

REST API Jenkins 2.501

☐ **Create the Required Files**

- Add a **Jenkinsfile** (to define the pipeline).
- Add a **Python script** (to be executed by Jenkins).

☐ **Initialize a Git Repository**

- Set up Git for version control in the directory.

☐ **Add the Files to Git**

- Stage the files to track changes.

☐ **Commit the Changes**

- Save the current state of the files with a commit message.

☐ **Connect to GitHub**

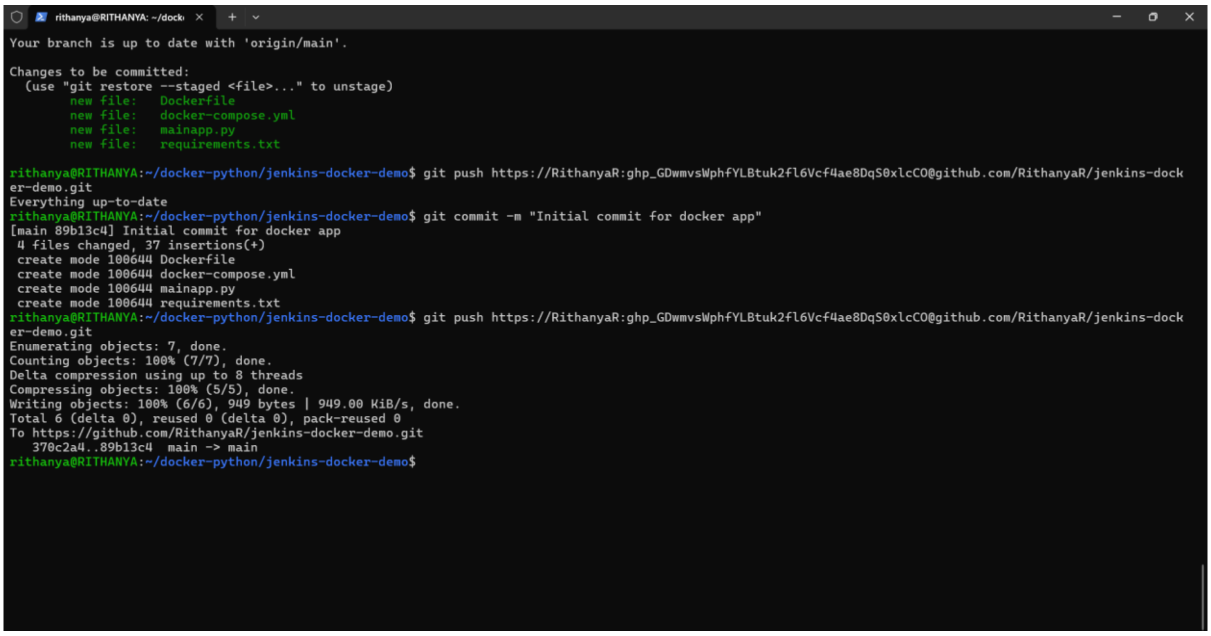
- Link the local repository to a GitHub repository.

☐ **Push the Files to GitHub**

- Upload the committed files to GitHub.

Verify on GitHub

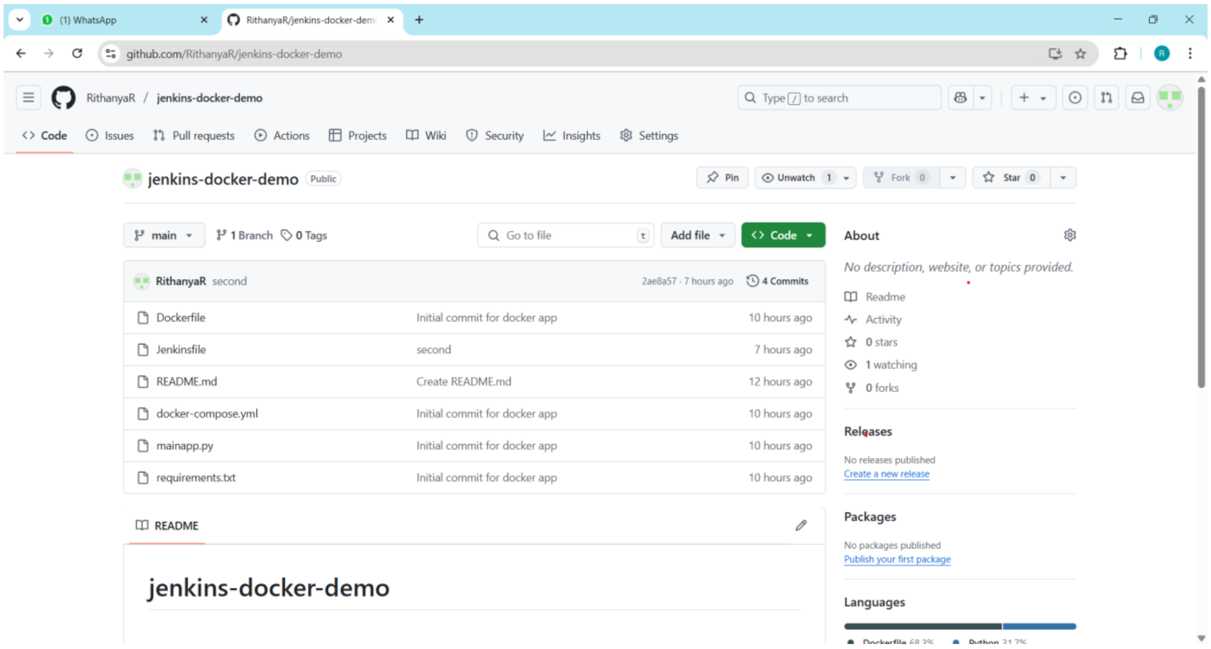
- Check the repository to confirm the files are uploaded.



```
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   Dockerfile
        new file:   docker-compose.yml
        new file:   mainapp.py
        new file:   requirements.txt

rithanya@RITHANYA:~/docker-python/jenkins-docker-demo$ git push https://RithanyaR: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@github.com/RithanyaR/jenkins-dock
er-demo.git
Everything up-to-date
rithanya@RITHANYA:~/docker-python/jenkins-docker-demo$ git commit -m "Initial commit for docker app"
[main 89b13c4] Initial commit for docker app
 4 files changed, 37 insertions(+)
 create mode 100644 Dockerfile
 create mode 100644 docker-compose.yml
 create mode 100644 mainapp.py
 create mode 100644 requirements.txt
rithanya@RITHANYA:~/docker-python/jenkins-docker-demo$ git push https://RithanyaR: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@github.com/RithanyaR/jenkins-dock
er-demo.git
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (6/6), 949 bytes | 949.00 KiB/s, done.
Total 6 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/RithanyaR/jenkins-docker-demo.git
   370c2a4..89b13c4  main -> main
rithanya@RITHANYA:~/docker-python/jenkins-docker-demo$
```



github.com/RithanyaR/jenkins-docker-demo

RithanyaR / jenkins-docker-demo

jenkins-docker-demo Public

main · 1 Branch · 0 Tags

RithanyaR second · 2ae8a57 · 7 hours ago · 4 Commits

Dockerfile	Initial commit for docker app	10 hours ago
Jenkinsfile	second	7 hours ago
README.md	Create README.md	12 hours ago
docker-compose.yml	Initial commit for docker app	10 hours ago
mainapp.py	Initial commit for docker app	10 hours ago
requirements.txt	Initial commit for docker app	10 hours ago

README

jenkins-docker-demo

About

No description, website, or topics provided.

Readme
Activity
0 stars
1 watching
0 forks

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

Languages

Dockerfile 68.3% · Python 31.7%

1. Open Jenkins Dashboard

- Log in to Jenkins.
- Navigate to the pipeline or job you created.

2. Trigger the Build

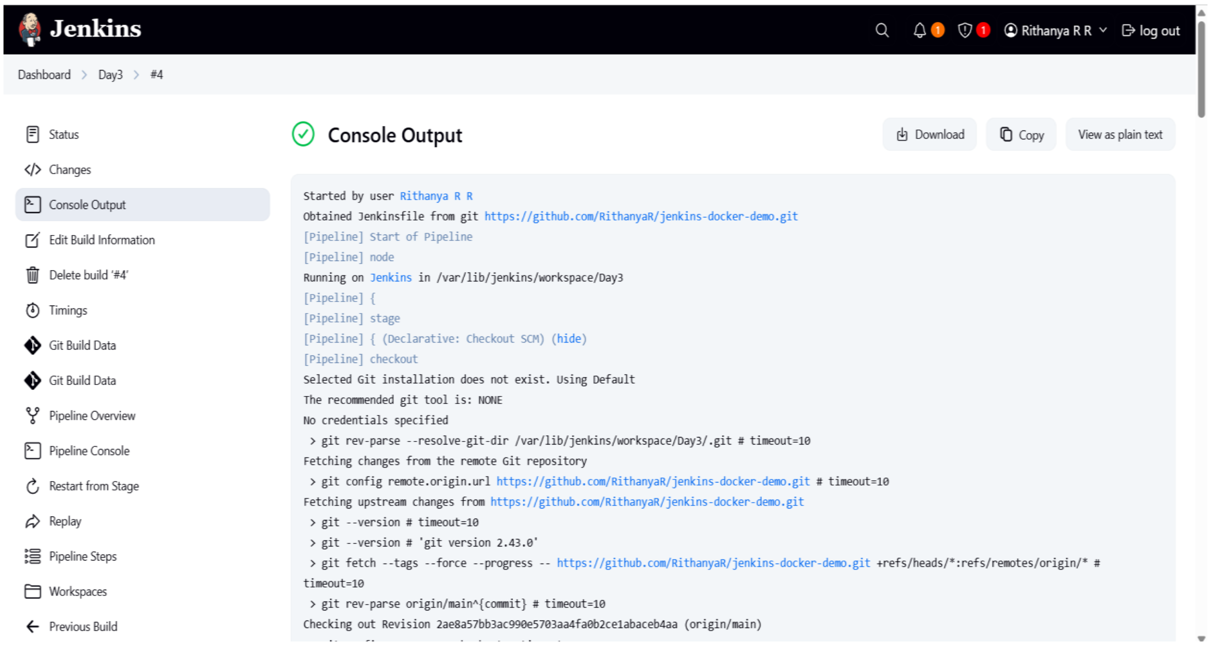
- Click on the **job name**.
- Click **Build Now** on the left panel.
- A new build (#1, #2, etc.) will appear in the **Build History** section.

3. View Console Output

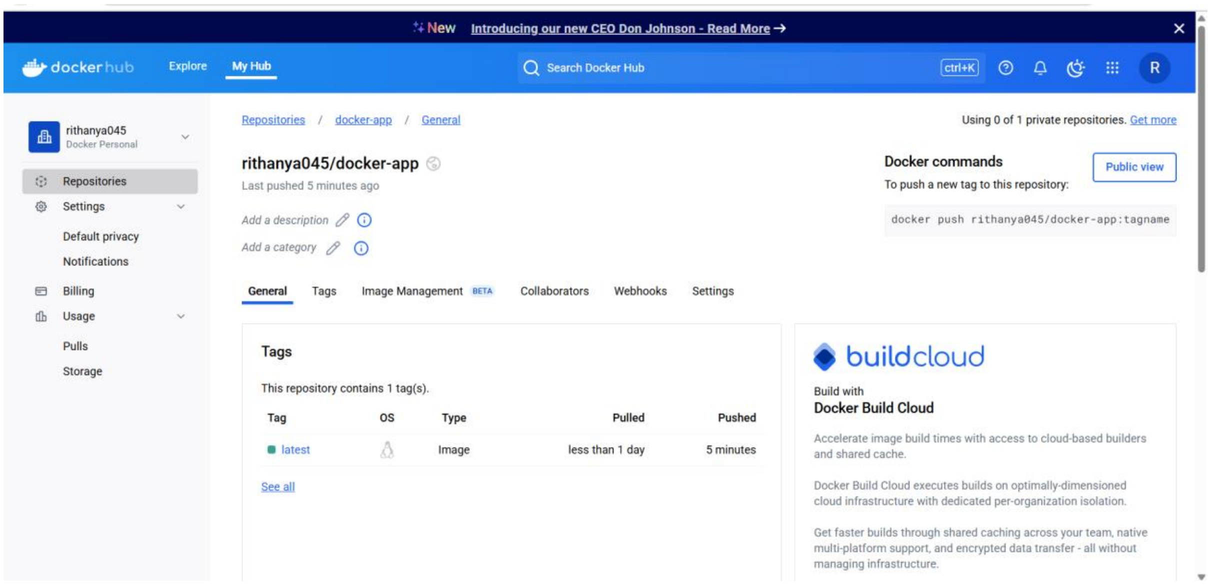
- Click on the latest build (#1, #2, etc.).
- Click **Console Output** in the left panel.

4. Check for Errors or Success

- If the build succeeds, you will see **"Finished: SUCCESS"** at the end.
- If it fails, check the logs for errors and troubleshoot.



```
Started by user Rithanya R R
Obtained Jenkinsfile from git https://github.com/RithanyaR/jenkins-docker-demo.git
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/lib/jenkins/workspace/Day3
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Declarative: Checkout SCM) (hide)
[Pipeline] checkout
Selected Git installation does not exist. Using Default
The recommended git tool is: NONE
No credentials specified
 > git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/Day3/.git # timeout=10
Fetching changes from the remote Git repository
 > git config remote.origin.url https://github.com/RithanyaR/jenkins-docker-demo.git # timeout=10
Fetching upstream changes from https://github.com/RithanyaR/jenkins-docker-demo.git
 > git --version # timeout=10
 > git --version # 'git version 2.43.0'
 > git fetch --tags --force --progress -- https://github.com/RithanyaR/jenkins-docker-demo.git +refs/heads/*:refs/remotes/origin/* #
timeout=10
 > git rev-parse origin/main^{commit} # timeout=10
Checking out Revision 2ae8a57bb3ac990e5703aa4fa0b2ce1abaceb4aa (origin/main)
```


Repositories / docker-app / General

Using 0 of 1 private repositories. Get more

rithanya045/docker-app

Last pushed 5 minutes ago

Docker commands

To push a new tag to this repository:

```
docker push rithanya045/docker-app:tagname
```

Tags

This repository contains 1 tag(s).

Tag	OS	Type	Pulled	Pushed
latest		Image	less than 1 day	5 minutes

See all

buildcloud

Build with Docker Build Cloud

Accelerate image build times with access to cloud-based builders and shared cache.

Docker Build Cloud executes builds on optimally-dimensioned cloud infrastructure with dedicated per-organization isolation.

Get faster builds through shared caching across your team, native multi-platform support, and encrypted data transfer - all without managing infrastructure.

Select the Repository

- Click on the repository name (**rithanya045/docker-app**) to open its details.

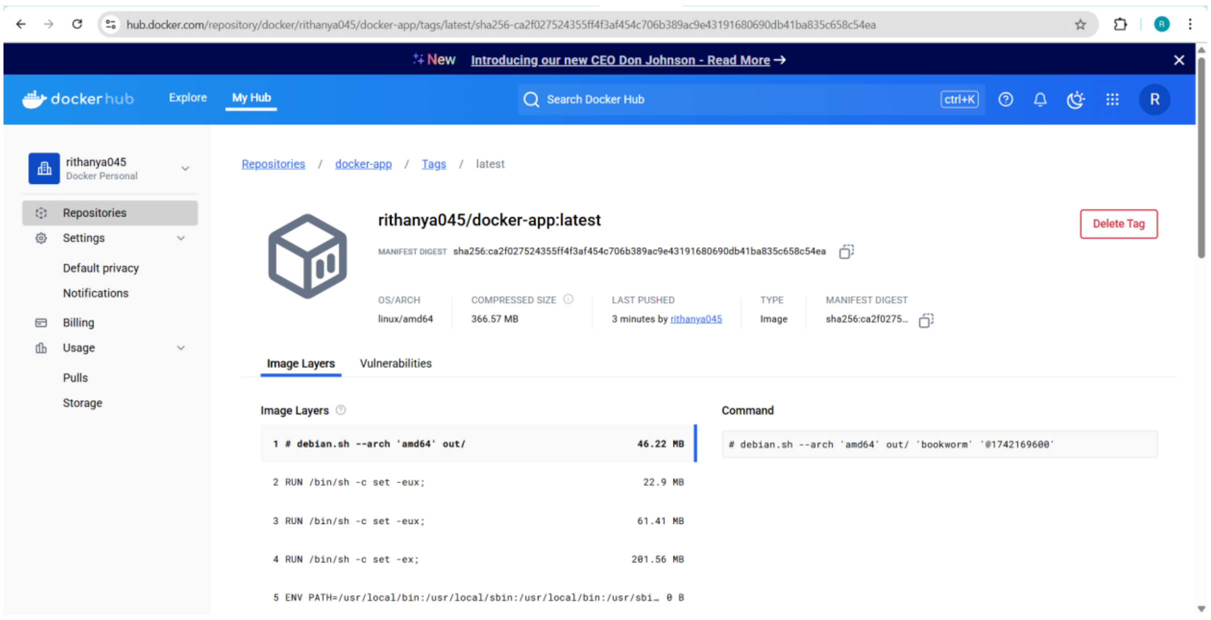
View the "latest" Tag

- Click on the **"Tags"** tab, then select the **"latest"** tag to view its details.

Last Pushed Time

Image Layers with commands used to build the image

Delete Tag button (if you want to remove the tag)


hub.docker.com/repository/docker/rithanya045/docker-app/tags/latest/sha256-ca2f027524355ff4f3af454c706b389ac9e43191680690db41ba835c658c54ea

Repositories / docker-app / Tags / latest

rithanya045/docker-app:latest

MANIFEST DIGEST sha256:ca2f027524355ff4f3af454c706b389ac9e43191680690db41ba835c658c54ea

OS/ARCH	COMPRESSED SIZE	LAST PUSHED	TYPE	MANIFEST DIGEST
linux/amd64	366.57 MB	3 minutes by rithanya045	Image	sha256:ca2f0275…

Image Layers

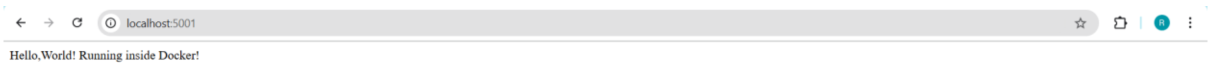
```
1 # debian.sh --arch 'amd64' out/                       46.22 MB
2 RUN /bin/sh -c set -eux;                                22.9 MB
3 RUN /bin/sh -c set -eux;                               61.41 MB
4 RUN /bin/sh -c set -ex;                               201.56 MB
5 ENV PATH=/usr/local/bin:/usr/local/sbin:/usr/local/bin:/usr/sbi… 0 B
```

Command

```
# debian.sh --arch 'amd64' out/ 'bookworm' '@1742169600'
```

Access the Application in a Web Browser

- Open a browser.
- Type **http://localhost:5000** in the address bar.
- Press **Enter**.



```
Hello,World! Running inside Docker!
```