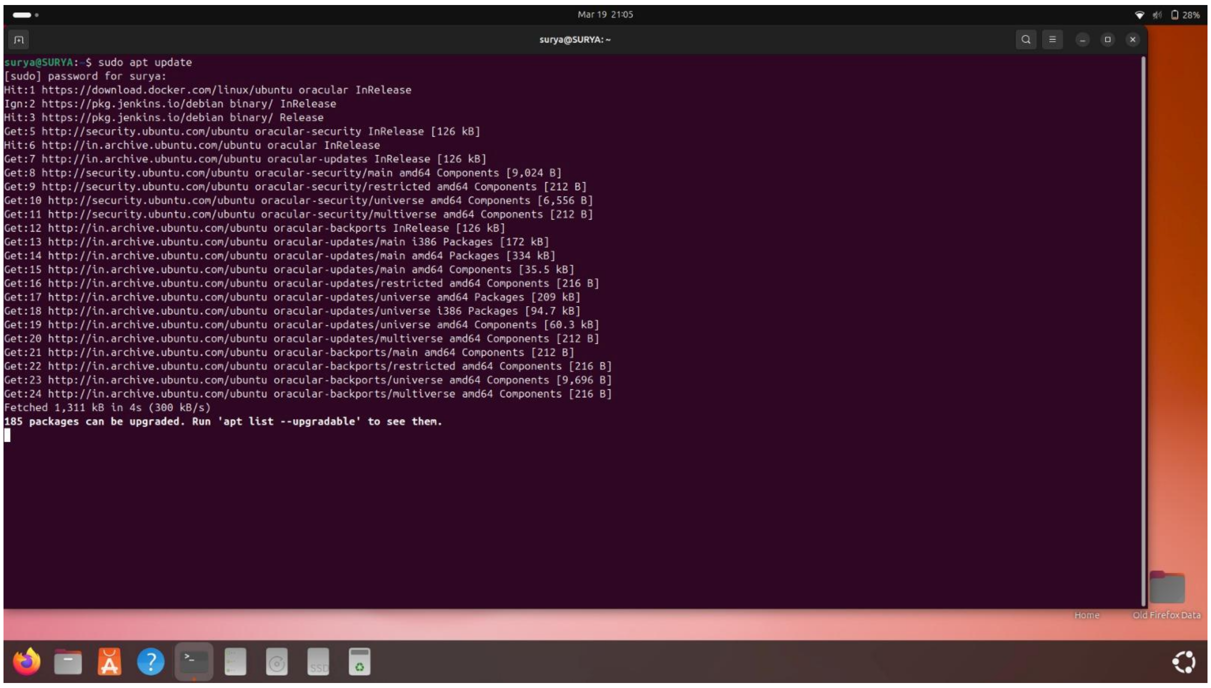
DAY-2

DEVOPS

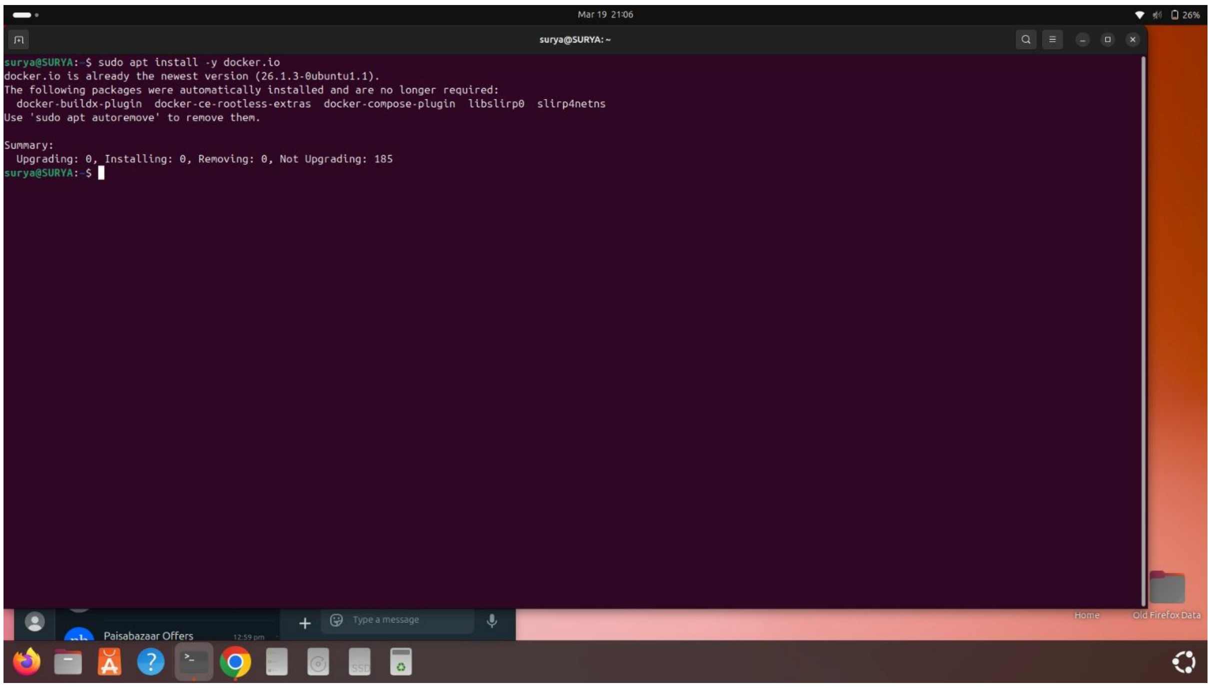
STEP -1 :INSTALL DOCKER

1) sudo apt update



```
surya@SURYA:~$ sudo apt update
[sudo] password for surya:
Hit:1 https://download.docker.com/linux/ubuntu oracular InRelease
Ign:2 https://pkg.jenkins.io/debian binary/ InRelease
Hit:3 https://pkg.jenkins.io/debian binary/ Release
Get:5 http://security.ubuntu.com/ubuntu oracular-security InRelease [126 kB]
Hit:6 http://in.archive.ubuntu.com/ubuntu oracular InRelease
Get:7 http://in.archive.ubuntu.com/ubuntu oracular-updates InRelease [126 kB]
Get:8 http://security.ubuntu.com/ubuntu oracular-security/main amd64 Components [9,024 B]
Get:9 http://security.ubuntu.com/ubuntu oracular-security/restricted amd64 Components [212 B]
Get:10 http://security.ubuntu.com/ubuntu oracular-security/universe amd64 Components [6,556 B]
Get:11 http://security.ubuntu.com/ubuntu oracular-security/multiverse amd64 Components [212 B]
Get:12 http://in.archive.ubuntu.com/ubuntu oracular-backports InRelease [126 kB]
Get:13 http://in.archive.ubuntu.com/ubuntu oracular-updates/main i386 Packages [172 kB]
Get:14 http://in.archive.ubuntu.com/ubuntu oracular-updates/main amd64 Packages [334 kB]
Get:15 http://in.archive.ubuntu.com/ubuntu oracular-updates/main amd64 Components [35.5 kB]
Get:16 http://in.archive.ubuntu.com/ubuntu oracular-updates/restricted amd64 Components [216 B]
Get:17 http://in.archive.ubuntu.com/ubuntu oracular-updates/universe amd64 Packages [209 kB]
Get:18 http://in.archive.ubuntu.com/ubuntu oracular-updates/universe i386 Packages [94.7 kB]
Get:19 http://in.archive.ubuntu.com/ubuntu oracular-updates/universe amd64 Components [60.3 kB]
Get:20 http://in.archive.ubuntu.com/ubuntu oracular-updates/multiverse amd64 Components [212 B]
Get:21 http://in.archive.ubuntu.com/ubuntu oracular-backports/main amd64 Components [212 B]
Get:22 http://in.archive.ubuntu.com/ubuntu oracular-backports/restricted amd64 Components [216 B]
Get:23 http://in.archive.ubuntu.com/ubuntu oracular-backports/universe amd64 Components [9,696 B]
Get:24 http://in.archive.ubuntu.com/ubuntu oracular-backports/multiverse amd64 Components [216 B]
Fetched 1,311 kB in 4s (300 kB/s)
185 packages can be upgraded. Run 'apt list --upgradable' to see them.
```

2) sudo apt install -y docker.io



```
surya@SURYA:~$ sudo apt install -y docker.io
docker.io is already the newest version (26.1.3-0ubuntu1.1).
The following packages were automatically installed and are no longer required:
  docker-buildx-plugin  docker-ce-rootless-extras  docker-compose-plugin  libslirp0  slirp4netns
Use 'sudo apt autoremove' to remove them.

Summary:
  Upgrading: 0, Installing: 0, Removing: 0, Not Upgrading: 185
surya@SURYA:~$
```

3)

STEP 2: ENABLE AND DISABLE

1) sudo systemctl enable docker

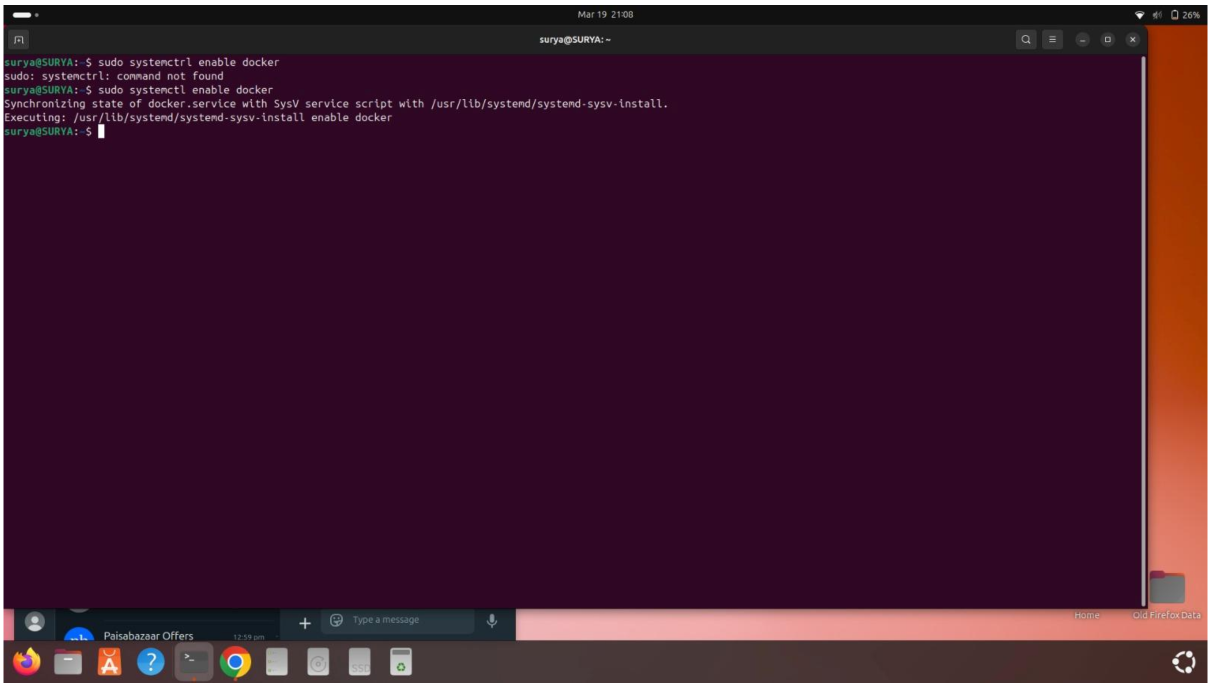
2)sudo systemctl start docker

STEP 3:VERIFY THE INSTALLATION:

docker –version

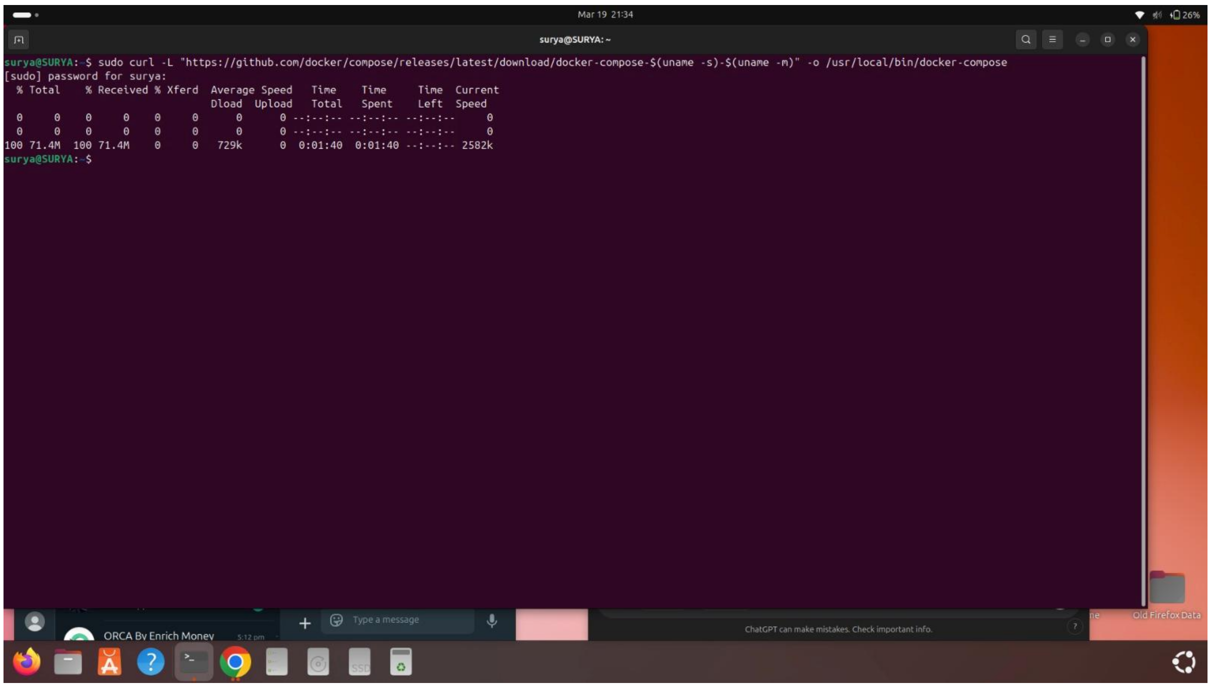
STEP 4:INSTALL DOCKER COMPOSE

sudo curl -L "https://github.com/docker/compose/releases/latest/download/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose



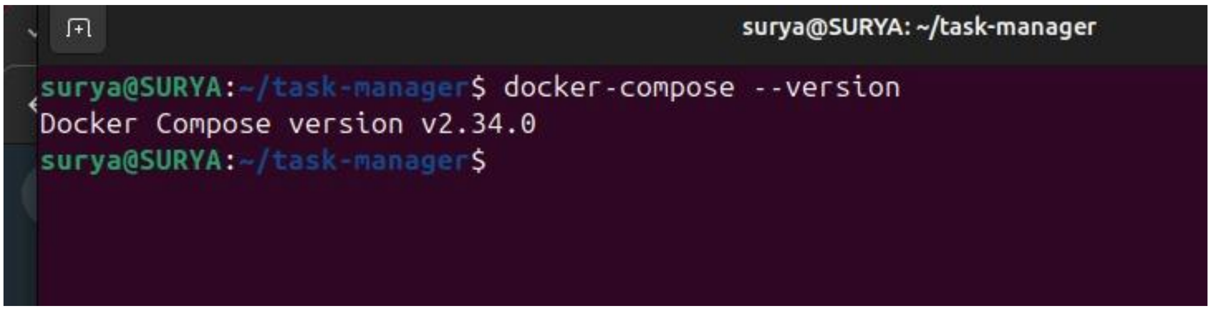
```
surya@SURYA:~$ sudo systemctrl enable docker
sudo: systemctrl: command not found
surya@SURYA:~$ sudo systemctl enable docker
Synchronizing state of docker.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable docker
surya@SURYA:~$ 
```

Give execution permission:



```
surya@SURYA:~$ sudo curl -L "https://github.com/docker/compose/releases/latest/download/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
[sudo] password for surya:
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0
100 71.4M  100 71.4M    0     0   729k      0  0:01:40  0:01:40 --:--:-- 2582k
surya@SURYA:~$
```

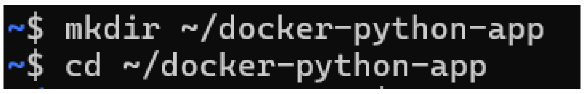
VERIFY INSTALLATION



```
surya@SURYA:~/task-manager$ docker-compose --version
Docker Compose version v2.34.0
surya@SURYA:~/task-manager$
```

CREATE AN “HELLO WOLRD: APPLICATION

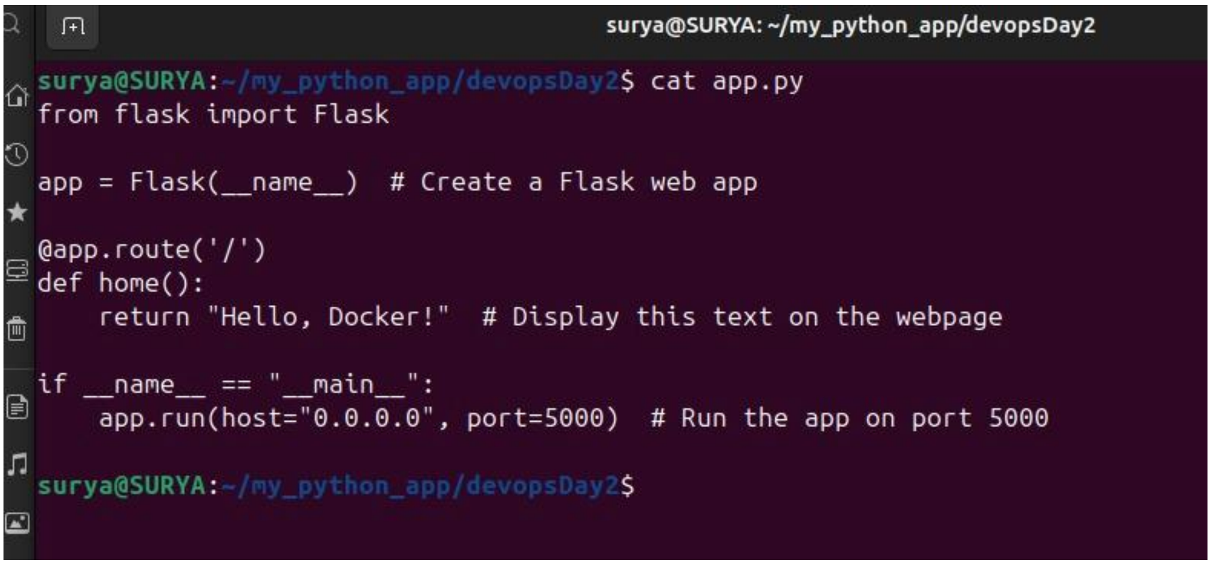
Create a project directory



```
~$ mkdir ~/docker-python-app
~$ cd ~/docker-python-app
```

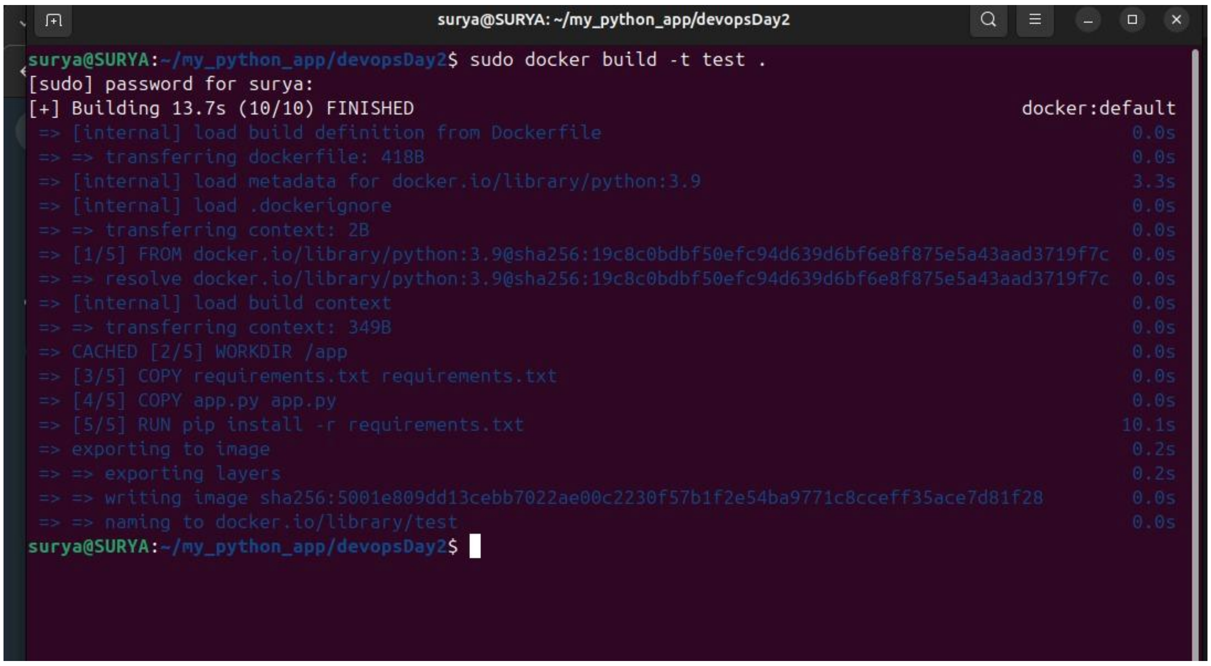
Create the python Application File

Create a file

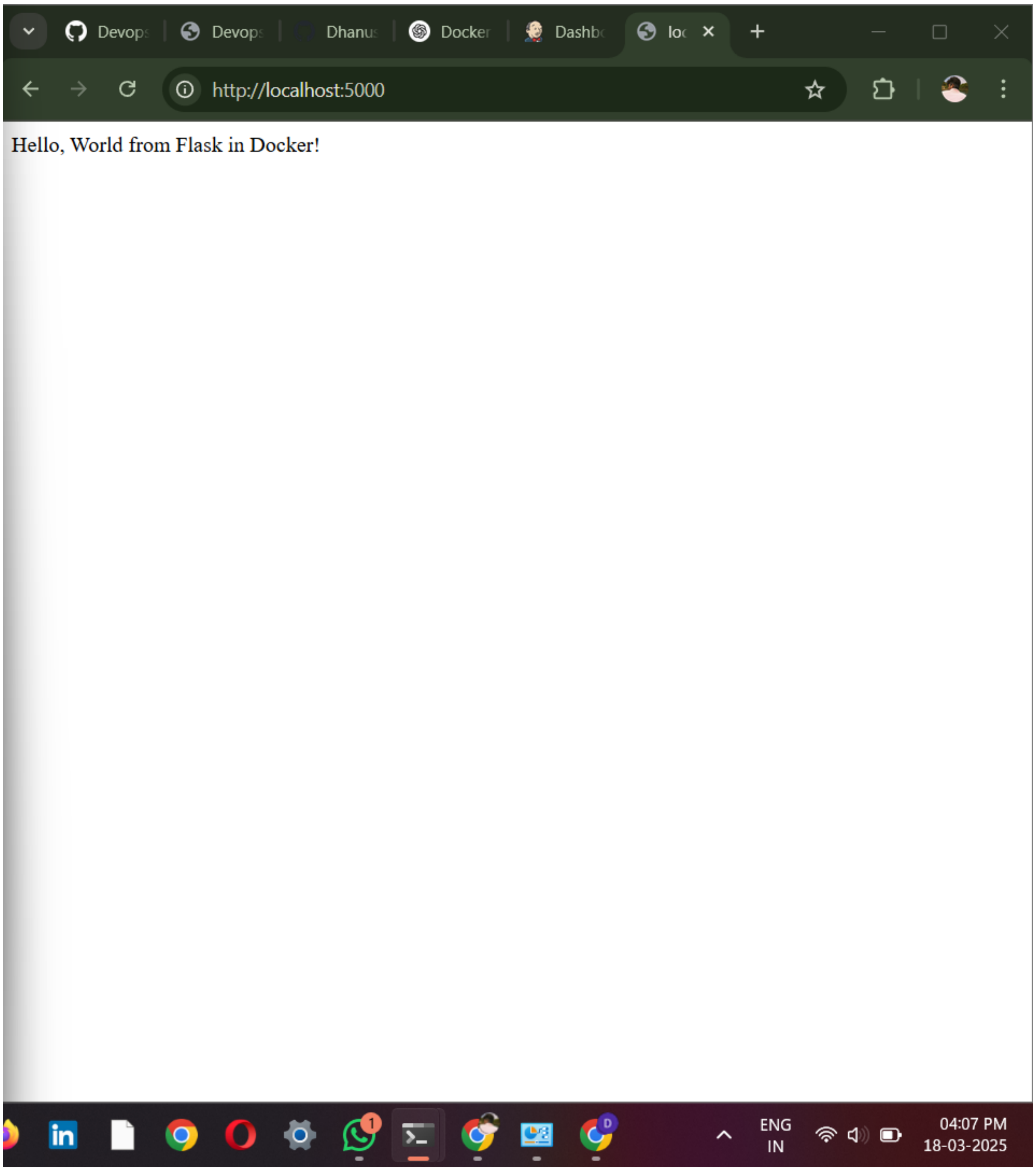

surya@SURYA: ~/my_python_app/devopsDay2

```
surya@SURYA:~/my_python_app/devopsDay2$ cat app.py
from flask import Flask

app = Flask(__name__)  # Create a Flask web app

@app.route('/')
def home():
    return "Hello, Docker!"  # Display this text on the webpage

if __name__ == "__main__":
    app.run(host="0.0.0.0", port=5000)  # Run the app on port 5000

surya@SURYA:~/my_python_app/devopsDay2$
```

To Run an Docker

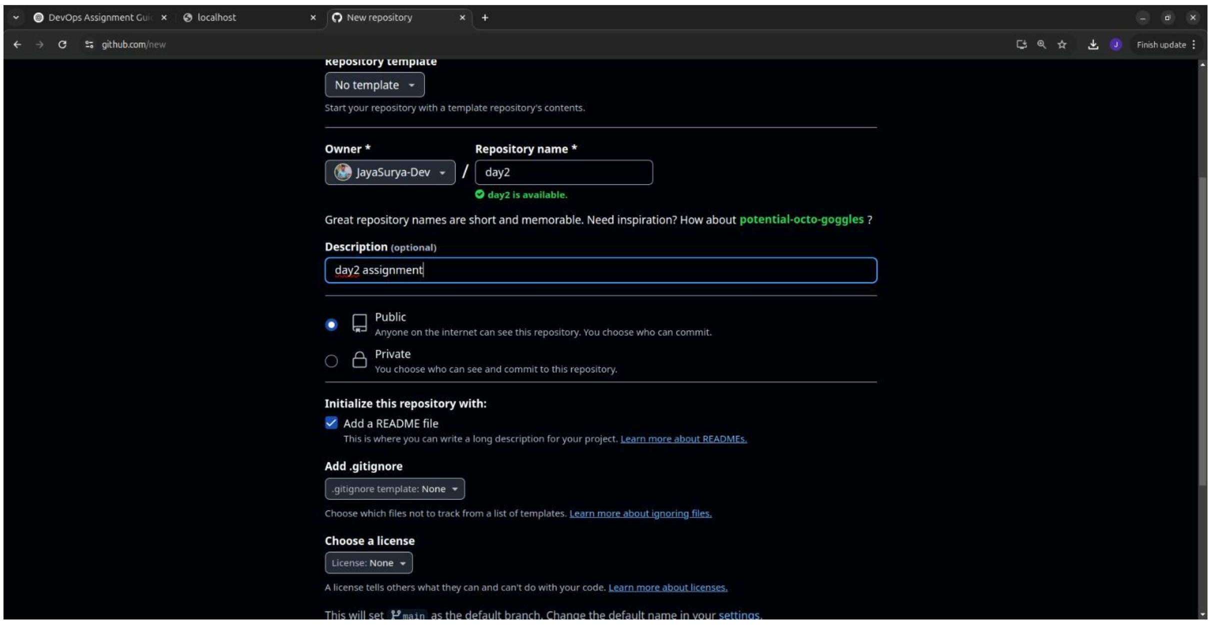

surya@SURYA: ~/my_python_app/devopsDay2

```
surya@SURYA:~/my_python_app/devopsDay2$ sudo docker build -t test .
[sudo] password for surya:
[+] Building 13.7s (10/10) FINISHED                                                              docker:default
 => [internal] load build definition from Dockerfile                                                       0.0s
 => => transferring dockerfile: 418B                                                                       0.0s
 => [internal] load metadata for docker.io/library/python:3.9                                              3.3s
 => [internal] load .dockerignore                                                                          0.0s
 => => transferring context: 2B                                                                            0.0s
 => [1/5] FROM docker.io/library/python:3.9@sha256:19c8c0bdbf50efc94d639d6bf6e8f875e5a43aad3719f7c   0.0s
 => => resolve docker.io/library/python:3.9@sha256:19c8c0bdbf50efc94d639d6bf6e8f875e5a43aad3719f7c   0.0s
 => [internal] load build context                                                                          0.0s
 => => transferring context: 349B                                                                          0.0s
 => CACHED [2/5] WORKDIR /app                                                                              0.0s
 => [3/5] COPY requirements.txt requirements.txt                                                           0.0s
 => [4/5] COPY app.py app.py                                                                               0.0s
 => [5/5] RUN pip install -r requirements.txt                                                             10.1s
 => exporting to image                                                                                     0.2s
 => => exporting layers                                                                                    0.2s
 => => writing image sha256:5001e809dd13cebb7022ae00c2230f57b1f2e54ba9771c8cceff35ace7d81f28              0.0s
 => => naming to docker.io/library/test                                                                    0.0s
surya@SURYA:~/my_python_app/devopsDay2$
```



```
Hello, World from Flask in Docker!
```

Devops Jenkins-Docker



Repository template

No template

Start your repository with a template repository's contents.

Owner * JayaSurya-Dev / Repository name * day2

day2 is available.

Great repository names are short and memorable. Need inspiration? How about potential-octo-goggles ?

Description (optional)

day2 assignment

[✓ Public] Anyone on the internet can see this repository. You choose who can commit.

Private You choose who can see and commit to this repository.

Initialize this repository with:

[✓] Add a README file

This is where you can write a long description for your project. Learn more about READMEs.

Add .gitignore

.gitignore template: None

Choose which files not to track from a list of templates. Learn more about ignoring files.

Choose a license

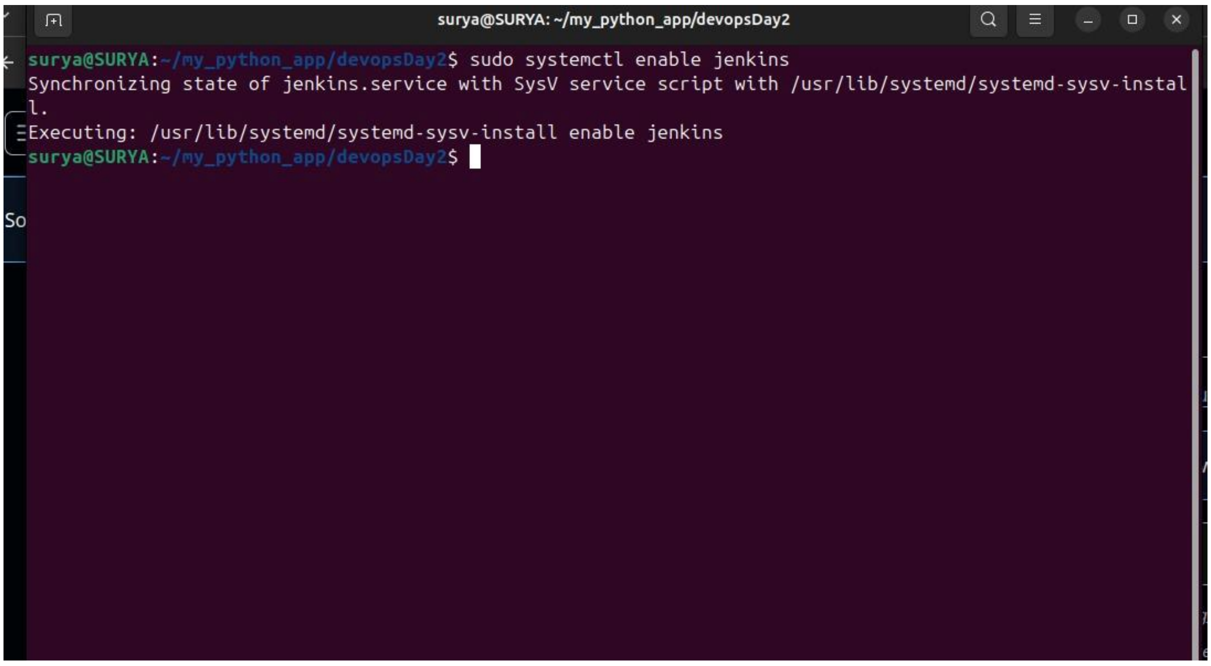
License: None

A license tells others what they can and can't do with your code. Learn more about licenses.

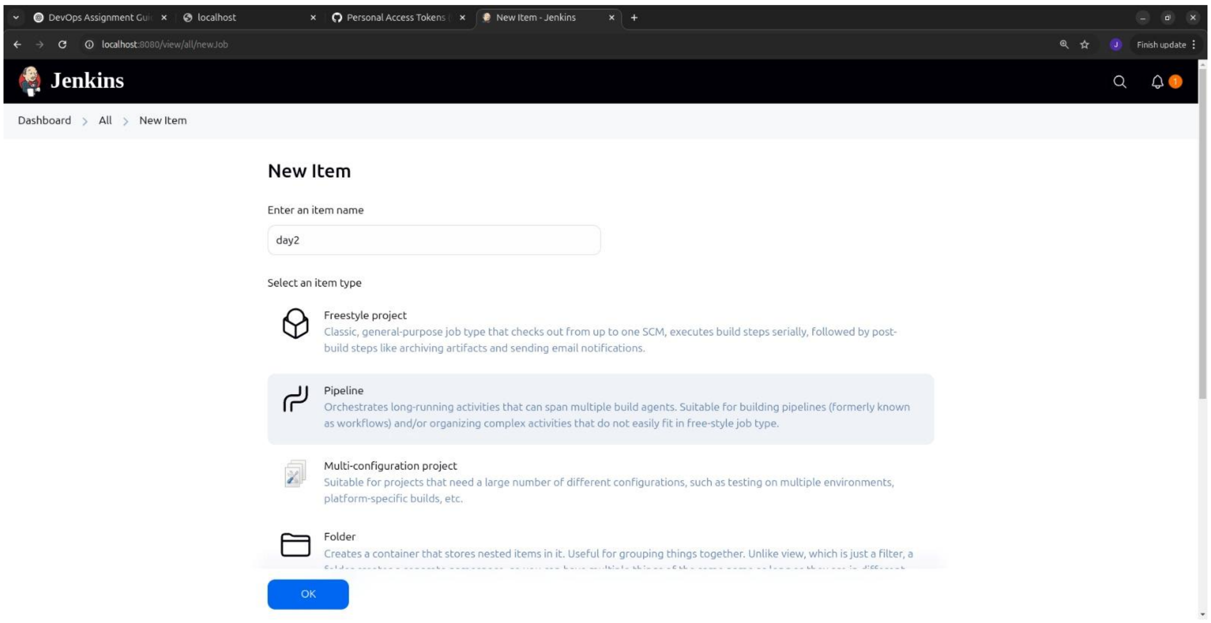
This will set main as the default branch. Change the default name in your settings.

Personal Token : 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

Start Jenkins



```
surya@SURYA:~/my_python_app/devopsDay2$ sudo systemctl enable jenkins
Synchronizing state of jenkins.service with SysV service script with /usr/lib/systemd/systemd-sysv-instal
l.
Executing: /usr/lib/systemd/systemd-sysv-install enable jenkins
surya@SURYA:~/my_python_app/devopsDay2$
```


localhost:8080/view/all/newJob

Dashboard > All > New Item

New Item

Enter an item name

day2

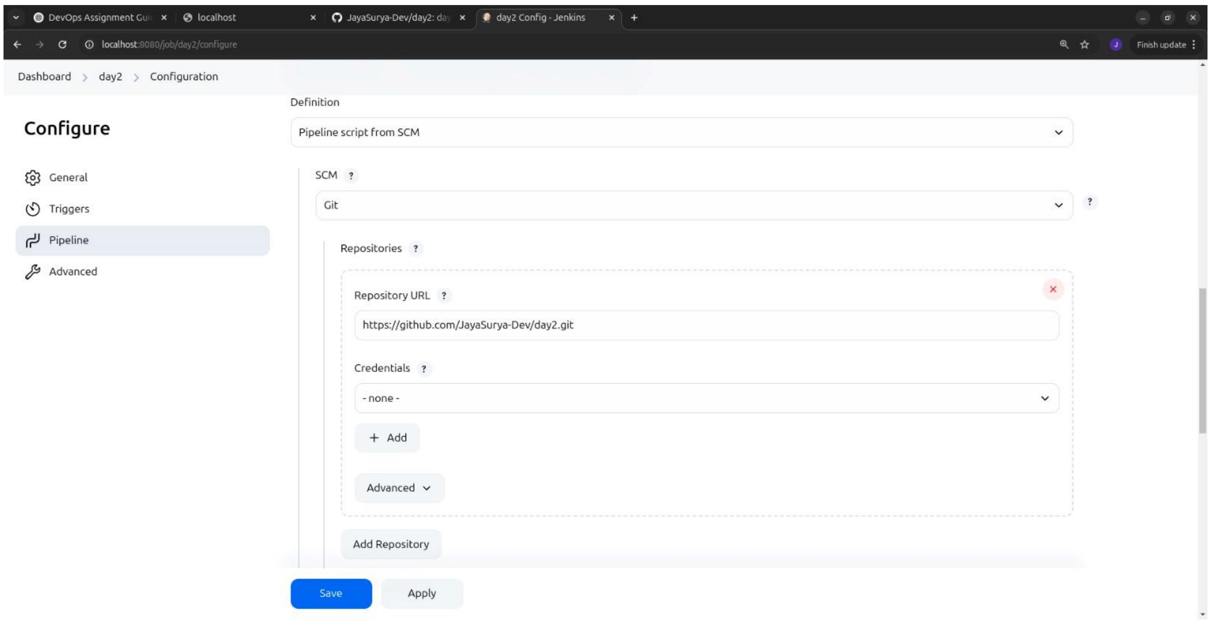
Select an item type

Freestyle project
Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.

Pipeline
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.

Multi-configuration project
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.

Folder
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a


localhost:8080/job/day2/configure

Dashboard > day2 > Configuration

Configure

Definition: Pipeline script from SCM

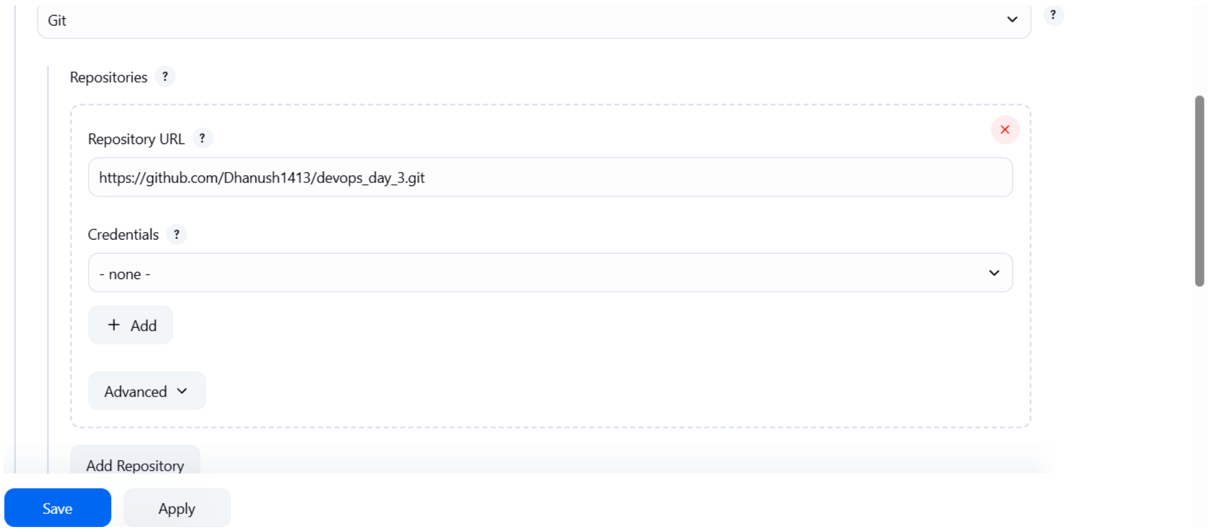
SCM: Git

Repositories

Repository URL: https://github.com/JayaSurya-Dev/day2.git

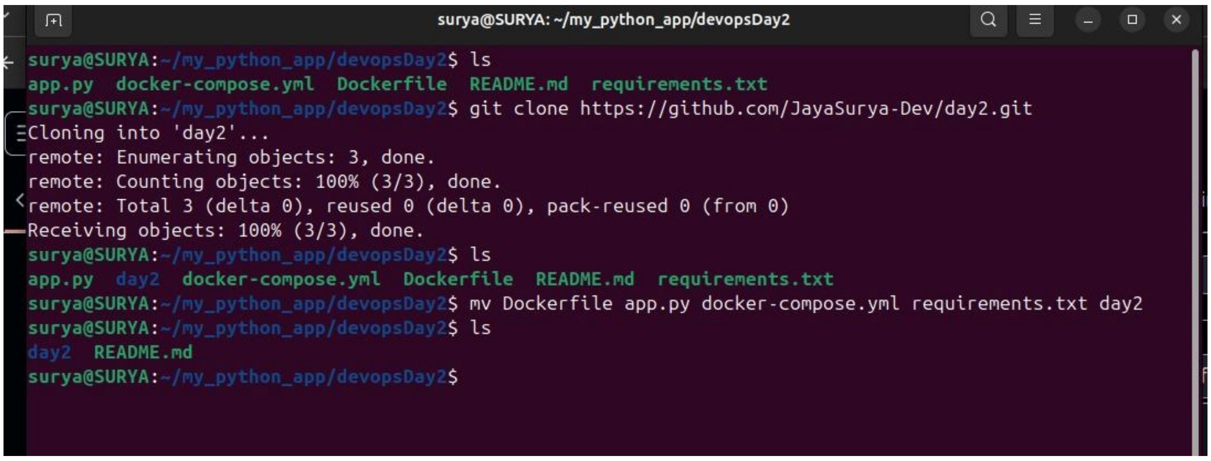
Credentials: - none -

Click on Add Credentials and Fill the details



In First Time, it Will have Password, in that we will give github token for it.

Clone the Git Repo in Terminal:

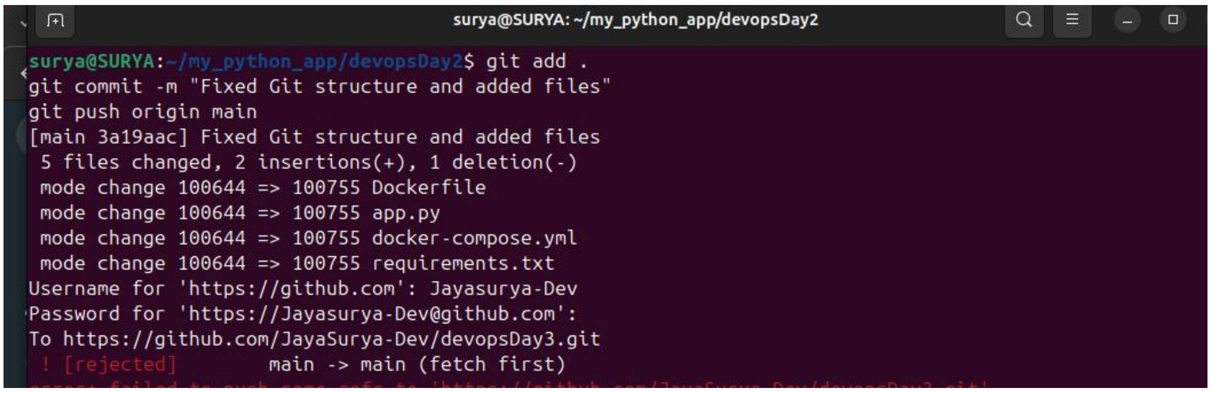


```
surya@SURYA:~/my_python_app/devopsDay2$ ls
app.py  docker-compose.yml  Dockerfile  README.md  requirements.txt
surya@SURYA:~/my_python_app/devopsDay2$ git clone https://github.com/JayaSurya-Dev/day2.git
Cloning into 'day2'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (3/3), done.
surya@SURYA:~/my_python_app/devopsDay2$ ls
app.py  day2  docker-compose.yml  Dockerfile  README.md  requirements.txt
surya@SURYA:~/my_python_app/devopsDay2$ mv Dockerfile app.py docker-compose.yml requirements.txt day2
surya@SURYA:~/my_python_app/devopsDay2$ ls
day2  README.md
surya@SURYA:~/my_python_app/devopsDay2$
```

And move all other file to github repo folder.

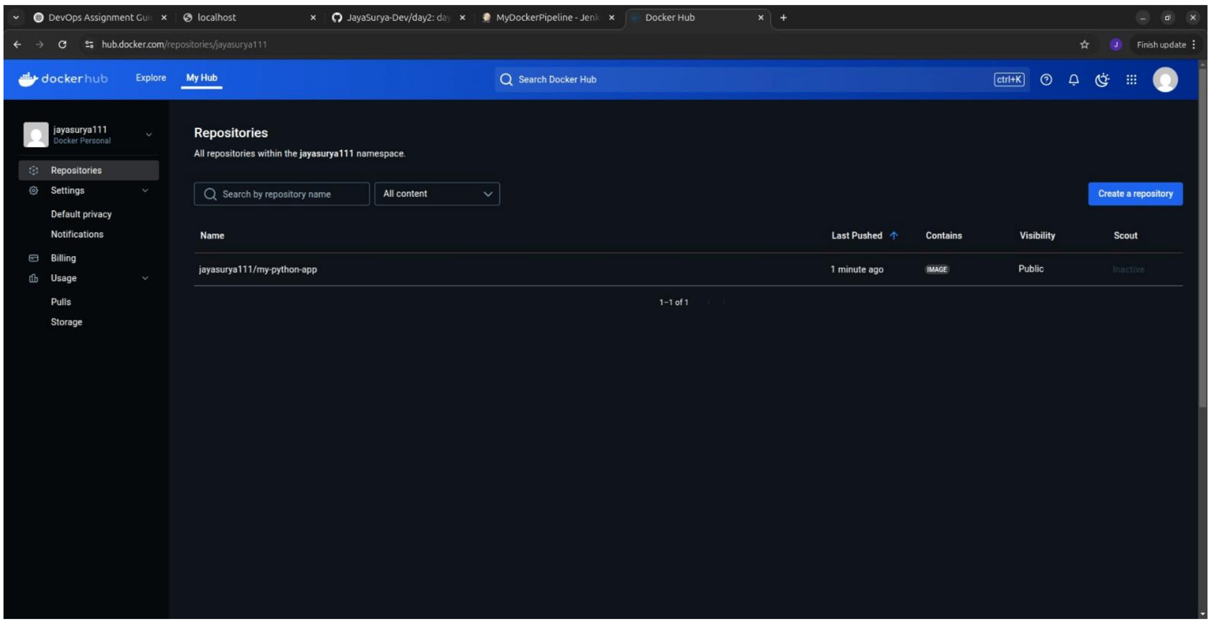
Git fetch – Remote repo Change and haven’t pulled in local.

Working to Push on GitHub Repo:



```
surya@SURYA:~/my_python_app/devopsDay2$ git add .
git commit -m "Fixed Git structure and added files"
git push origin main
[main 3a19aac] Fixed Git structure and added files
 5 files changed, 2 insertions(+), 1 deletion(-)
 mode change 100644 => 100755 Dockerfile
 mode change 100644 => 100755 app.py
 mode change 100644 => 100755 docker-compose.yml
 mode change 100644 => 100755 requirements.txt
Username for 'https://github.com': Jayasurya-Dev
Password for 'https://Jayasurya-Dev@github.com':
To https://github.com/JayaSurya-Dev/devopsDay3.git
 ! [rejected]        main -> main (fetch first)
```

Then create, Build now again and Click the repository in docker:



hub.docker.com/repositories/jayasurya111

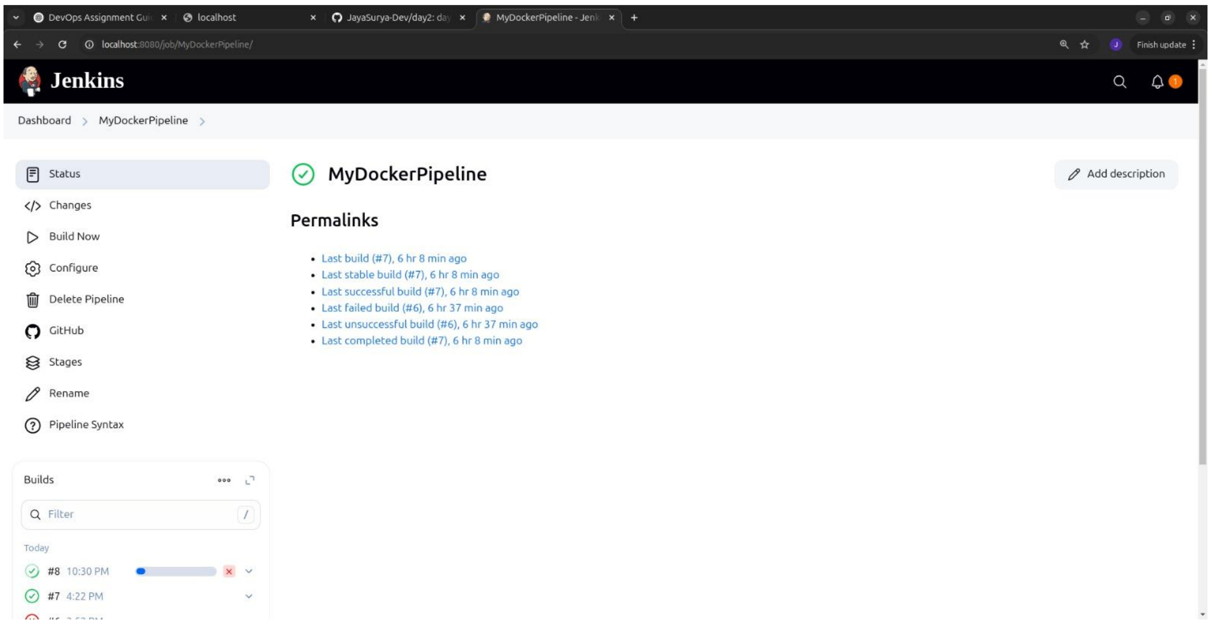
Repositories

All repositories within the **jayasurya111** namespace.

Name	Last Pushed	Contains	Visibility	Scout
jayasurya111/my-python-app	1 minute ago	IMAGE	Public	Inactive

1–1 of 1

Click the container that we create in Jenkins:



MyDockerPipeline

Permalinks

- Last build (#7), 6 hr 8 min ago
- Last stable build (#7), 6 hr 8 min ago
- Last successful build (#7), 6 hr 8 min ago
- Last failed build (#6), 6 hr 37 min ago
- Last unsuccessful build (#6), 6 hr 37 min ago
- Last completed build (#7), 6 hr 8 min ago

#8 10:30 PM
#7 4:22 PM