

DEVOPS TASK- 2

Step 1: Installation of Docker:

Install the docker.io

CODE :

```
sudo apt install docker.io Docker --version
```

```
sudo systemctl start docker
```

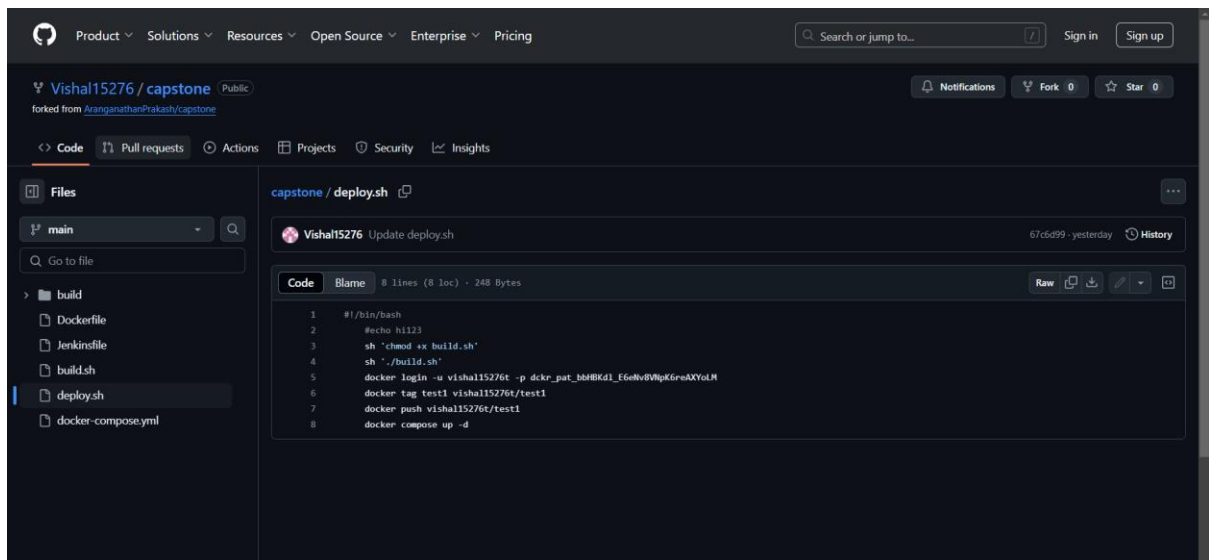
```
sudo systemctl enable docker
```

```
sudo systemctl status docker
```

```
root@LAPTOP-6V70H2B0:~# apt install docker.io
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
docker.io is already the newest version (26.1.3-0ubuntu1-24.04.1).
The following packages were automatically installed and are no longer required:
  libdrm-intel1 libpciaccess0 libsensors-config libsensors5
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 9 not upgraded.
root@LAPTOP-6V70H2B0:~# docker --version
Docker version 26.1.3, build 26.1.3-0ubuntu1-24.04.1
root@LAPTOP-6V70H2B0:~# sudo systemctl start docker
root@LAPTOP-6V70H2B0:~# sudo systemctl enable docker
root@LAPTOP-6V70H2B0:~# sudo systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/usr/lib/systemd/system/docker.service; enabled; preset: enabled)
   Active: active (running) since Thu 2025-03-20 06:44:32 UTC; 1h 32min ago
     TriggeredBy: ● docker.socket
       Docs: https://docs.docker.com
      Main PID: 9561 (dockerd)
        Tasks: 30
       Memory: 62.0M (-)
      CGROUP: /system.slice/docker.service
               └─ 9561 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock
                  └─ 10253 /usr/bin/docker-proxy -proto tcp -host-ip 0.0.0.0 -host-port 70 -container-ip 172.17.0.2 -con
                     └─ 10261 /usr/bin/docker-proxy -proto tcp -host-ip :: -host-port 70 -container-ip 172.17.0.2 -con
Mar 20 06:44:32 LAPTOP-6V70H2B0 dockerd[9561]: time="2025-03-20T06:44:32.185097971Z" level=warning msg="WARNIN
Mar 20 06:44:32 LAPTOP-6V70H2B0 dockerd[9561]: time="2025-03-20T06:44:32.185409232Z" level=warning msg="WARNIN
Mar 20 06:44:32 LAPTOP-6V70H2B0 dockerd[9561]: time="2025-03-20T06:44:32.185440810Z" level=warning msg="WARNIN
Mar 20 06:44:32 LAPTOP-6V70H2B0 dockerd[9561]: time="2025-03-20T06:44:32.185455418Z" level=warning msg="WARNIN
Mar 20 06:44:32 LAPTOP-6V70H2B0 dockerd[9561]: time="2025-03-20T06:44:32.185498240Z" level=info msg="Docker d
Mar 20 06:44:32 LAPTOP-6V70H2B0 dockerd[9561]: time="2025-03-20T06:44:32.185845402Z" level=info msg="Daemon h
Mar 20 06:44:32 LAPTOP-6V70H2B0 dockerd[9561]: time="2025-03-20T06:44:32.379205869Z" level=info msg="API list
Mar 20 06:44:32 LAPTOP-6V70H2B0 systemd[1]: Started docker.service - Docker Application Container Engine.
Mar 20 06:45:16 LAPTOP-6V70H2B0 dockerd[9561]: time="2025-03-20T06:45:16.405075970Z" level=info msg="Layer sh
Mar 20 06:45:16 LAPTOP-6V70H2B0 dockerd[9561]: time="2025-03-20T06:45:16.550116575Z" level=info msg="Layer sh
lines 1-23/23 (END) ... skipping...
● docker.service - Docker Application Container Engine
   Loaded: loaded (/usr/lib/systemd/system/docker.service; enabled; preset: enabled)
```

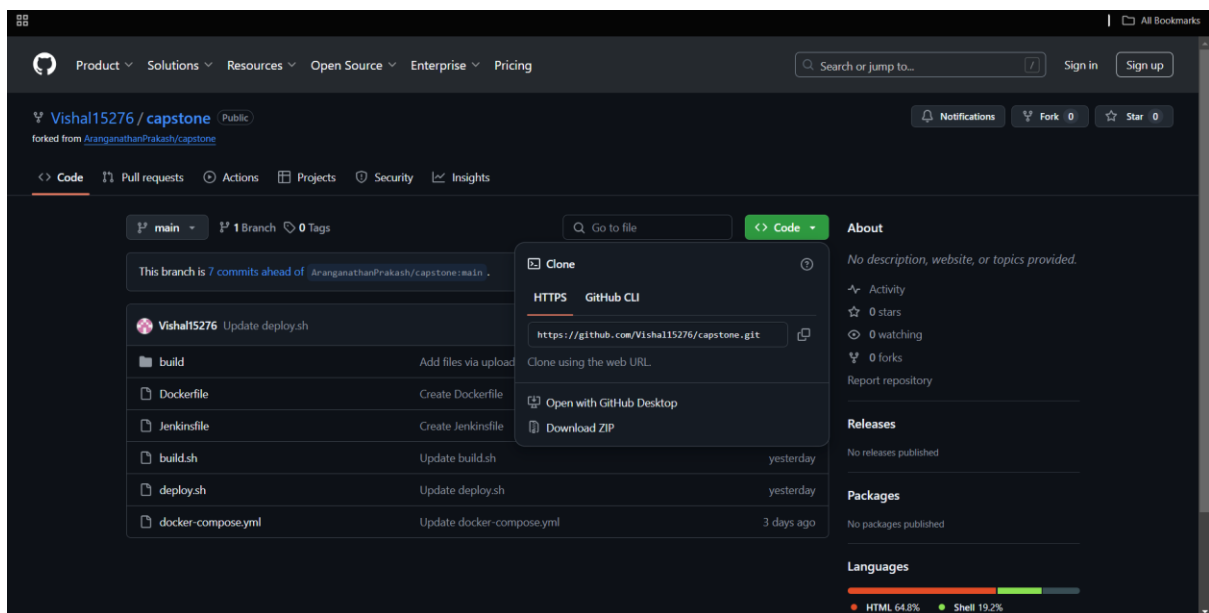
Step 2:Fork

Fork a copy of a GitHub repo which contains the necessary files which will result in the clone of that repo in our own repository . Then change the token and repo name of the docker Hub in the deploy.sh file which is in our repository.



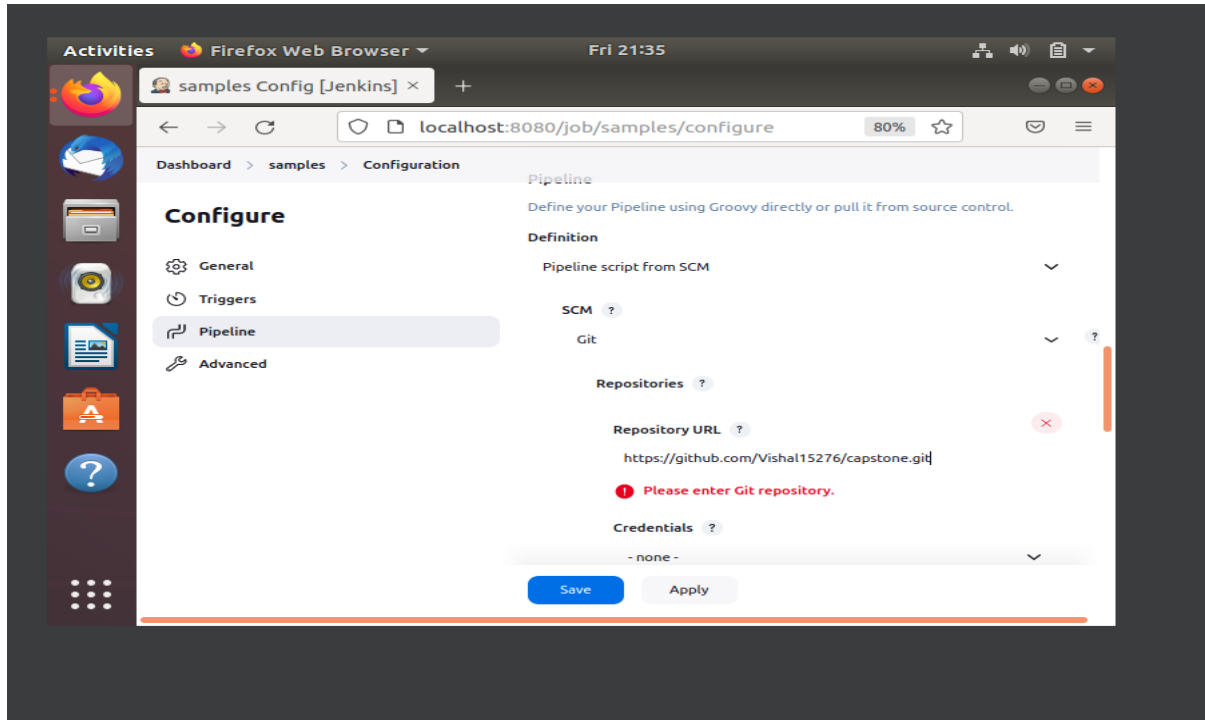
Step 3:GitHub Link

Then copy the GitHub link of the repository and go to Jenkins.



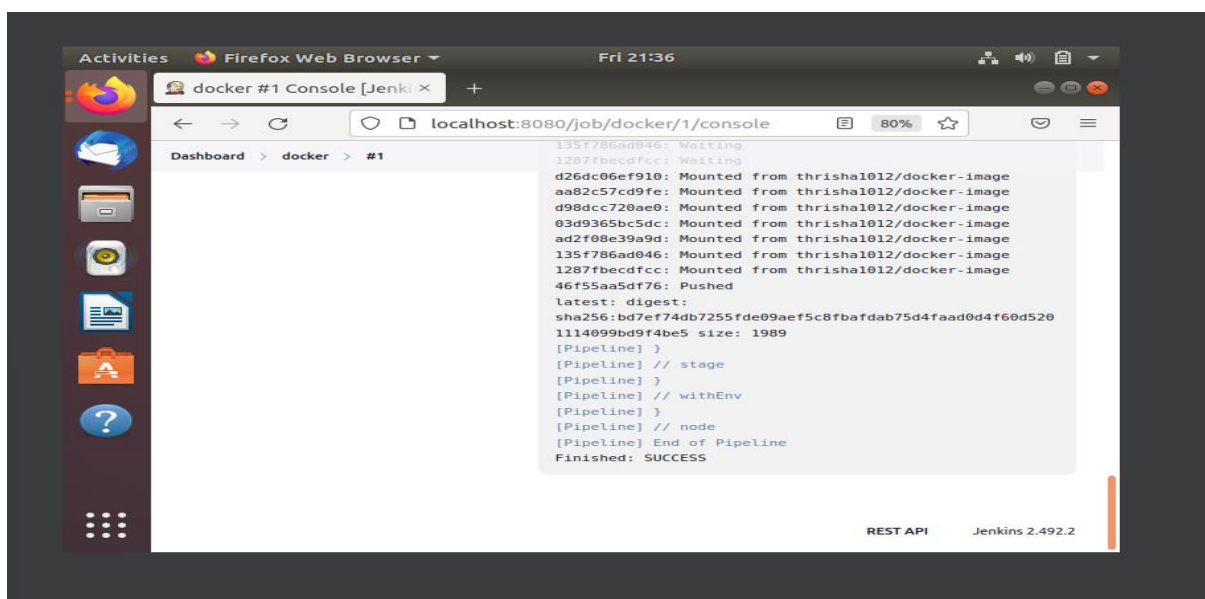
Step 4: New Item

In Jenkins, create a new item (Job) with a type pipeline and add the copied GitHub url to it with the correct branch and Jenkinsfile.



Step 5: Build And Image Creation

After Creating the job, build it and it will give the console output and the docker image will be created.



Step 6: Build Image with Port number

Now Built this docker image in the terminal with desired port number to it.

CODE:

docker images

docker build -itd -p 70:80 test1

```

Password:
Error saving credentials: error storing credentials - err: exit status 1, out: 'error storing credentials - err: exit status 1, out: 'exit status 1: gpg: suganth0
gpg: [stdin]: encryption failed: No public key
Password encryption aborted.`
suganth@suganth-debian: $ cd ~/docker/config.json
suganth@suganth-debian: $ docker login -u suganth07

Info - A Personal Access Token (PAT) can be used instead.
To create a PAT, visit https://app.docker.com/settings

Password:

WARNING! Your credentials are stored unencrypted in '/home/suganth/.docker/config.json'.
Configure a credential helper to remove this warning. See
https://docs.docker.com/go/credential-store/

Login Succeeded
suganth@suganth-debian: $ docker images
REPOSITORY          TAG         IMAGE ID      CREATED       SIZE
suganth07/devops     latest      d96625e7ec0b  2 hours ago   195MB
test1                latest      d96625e7ec0b  2 hours ago   195MB
hello-world          latest      74c54e27dc4   8 weeks ago   10.1kB
suganth@suganth-debian: $ docker run -itd -p 70:80 test1
09183a957145c5486005700a48291d3153d2148d50390f0fbc9a31afcf27eb68
suganth@suganth-debian: $ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                               NAMES
09183a957145   test1     "/docker-entrypoint. ..." About a minute ago Up About a minute   0.0.0.0:70->80/tcp, [::]:70->80/tcp   flamboyant_chatterjee
suganth@suganth-debian: $ color a
bash: color: command not found
suganth@suganth-debian: $ history
1008 git add
1009 git commit -m "vercel hosting updated"
```

Step 7: Output

Go to the Browser and search for localhost:<PORT_NUMBER> and the respective application will be hosted.

