

## DEVOPS TASK- 2

## 1) Installation of Docker:

**CODE :**

```
sudo apt install docker.io
```

## Docker –version

```
sudo systemctl start docker
```

```
sudo systemctl enable docker
```

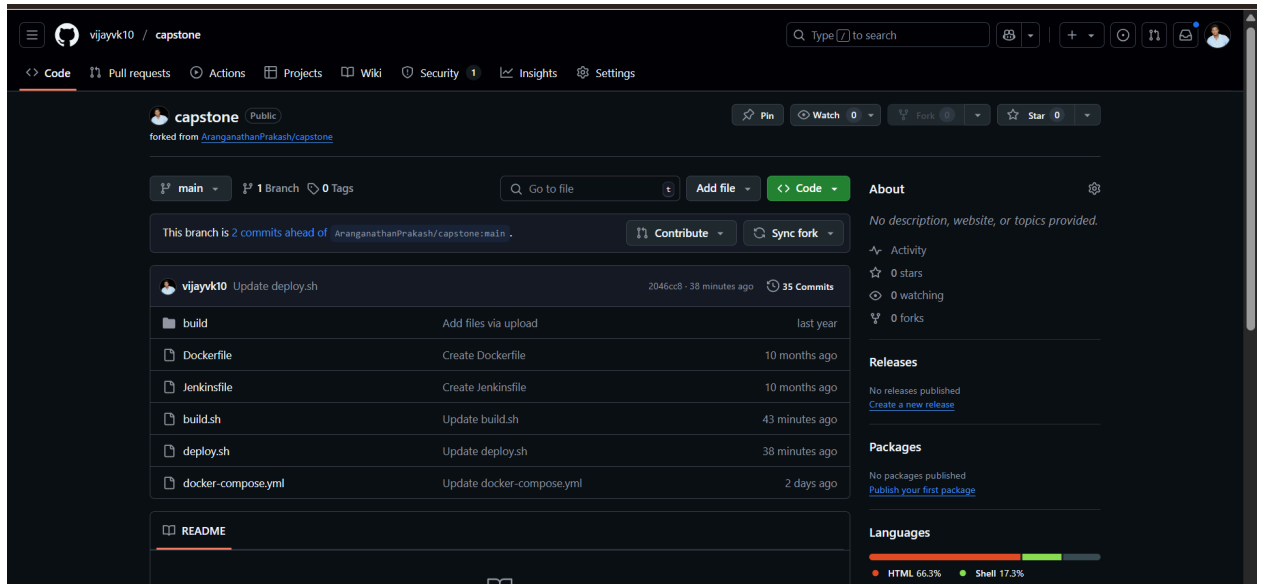
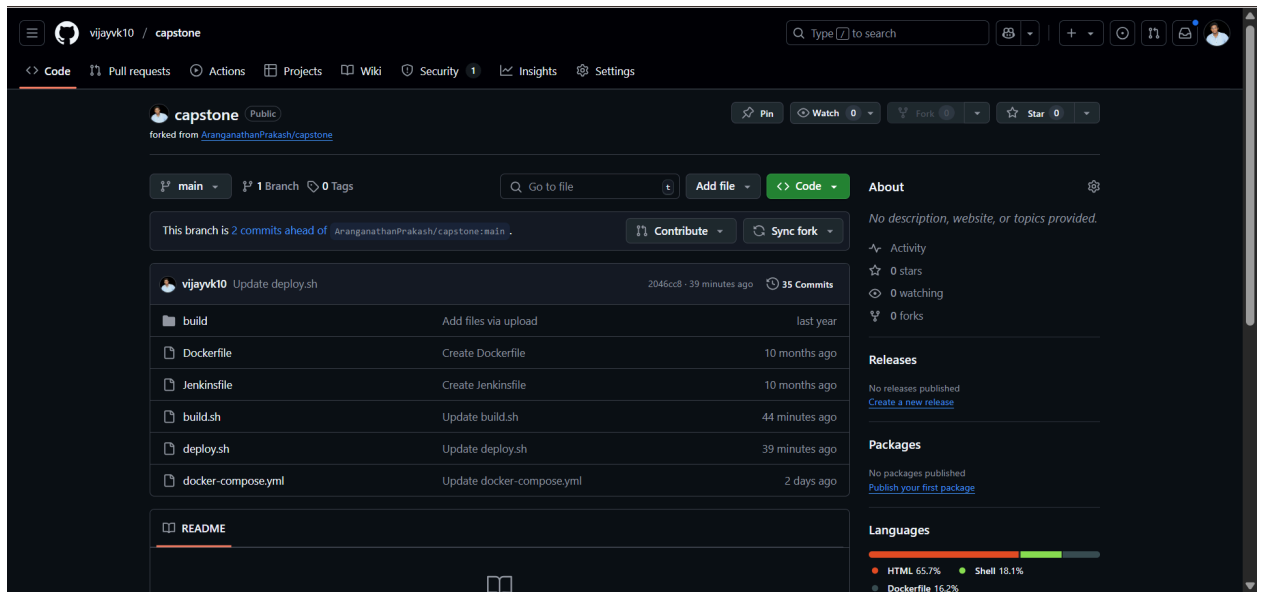
```
sudo systemctl status docker
```

**SCREENSHOT:**

```
root@APTF06-EVTM0801:~# apt install docker.io
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
docker.io is already the newest version (20.1.3~deb10~20.04.1).
The following packages were automatically installed and are no longer required:
  libklibc libklibc-bin libklibc-curl libklibc-dev libklibc-doc
Use 'dpkg --get-references' to remove them.
* upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
root@APTF06-EVTM0801:~# docker --version
Docker version 20.1.3, build 20.1.3~deb10~20.04.1
root@APTF06-EVTM0801:~# sudo systemctl start docker
root@APTF06-EVTM0801:~# sudo systemctl enable docker
root@APTF06-EVTM0801:~# 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 2020-01-28 06:08:32 UTC; 3h 34min ago
   TriggeredBy: ● docker.socket
   Docs: https://docs.docker.com
   Main PID: 9561 (dockerd)
   Tasks: 30
   Memory: 61.0M
   CGroup: /system.slice/docker.service
           └─ 9561 /usr/bin/dockerd --log-driver=json-file --config=/etc/docker/daemon.json
              18233 /usr/bin/docker-proxy -proto tcp --host-ip 0.0.0.0 --host-port 78 --container-ip 172.17.0.2
              18261 /usr/bin/docker-proxy -proto tcp --host-ip :: --host-port 78 --container-ip 172.17.0.2 --con
```

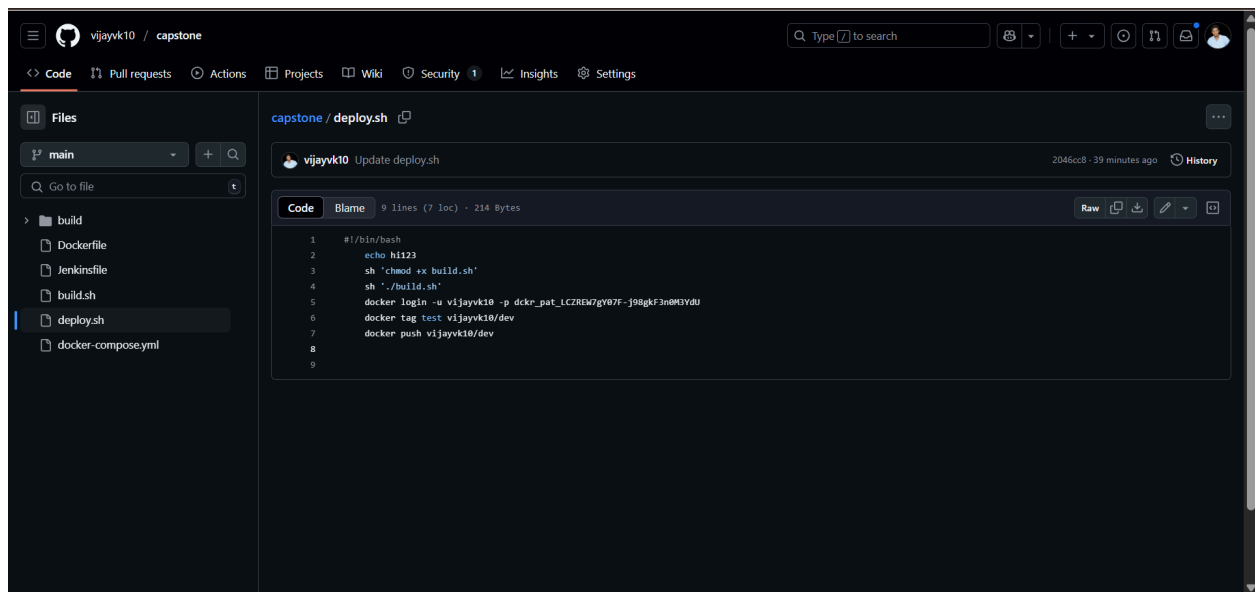
2) Fork a copy of a GitHub repo which contains the necessary files which will result in the clone of that repo in our own repository

**SCREENSHOT :**



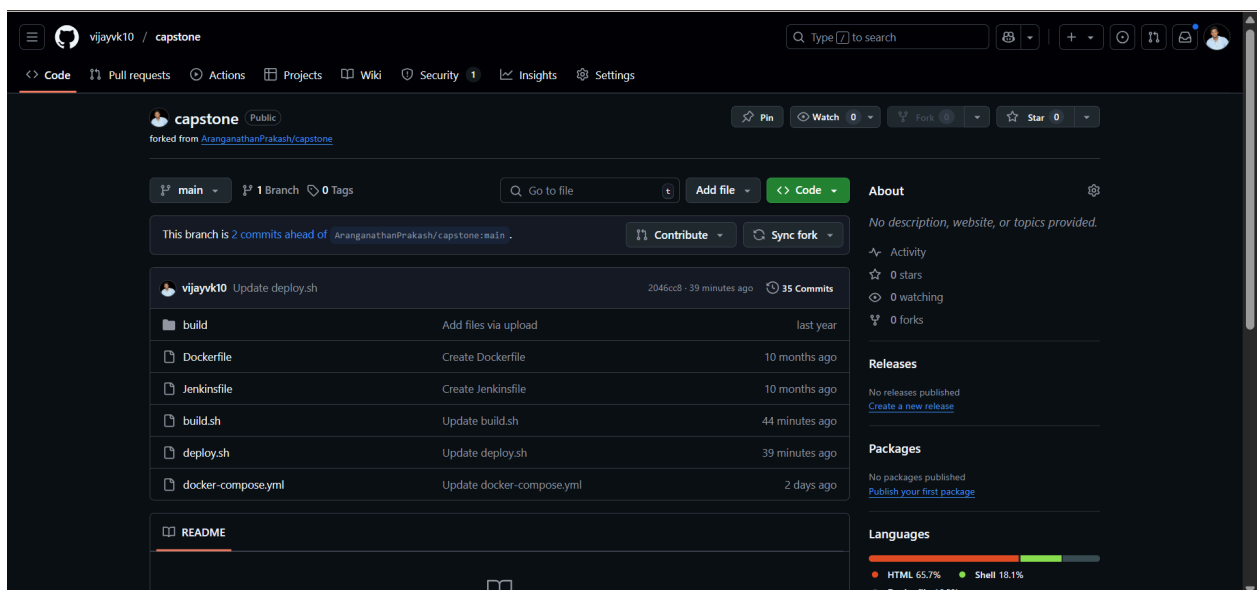
3) Then change the token and repo name of the docker Hub in the deploy.sh file which is in our repository.

**SCREENSHOT :**



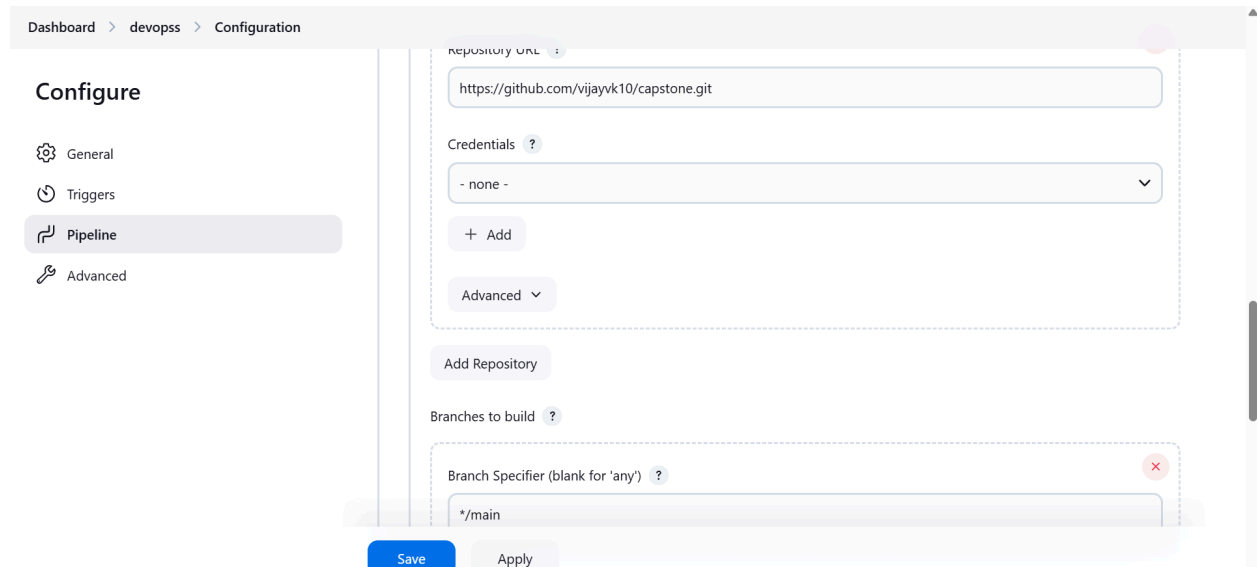
4) Then copy the GitHub link of the repository and go to

Jenkins. **SCREENSHOT:**



- 5) 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.

## SCREENSHOT:



The screenshot shows the Jenkins 'Configure' page for a new Pipeline job. The breadcrumb navigation at the top reads 'Dashboard > devopss > Configuration'. On the left, the 'Configure' section has four tabs: 'General', 'Triggers', 'Pipeline' (which is selected and highlighted), and 'Advanced'. The main configuration area contains the following fields:

- Repository URL:** A text input field containing 'https://github.com/vijayvk10/capstone.git'.
- Credentials:** A dropdown menu currently showing '- none -'. Below it are buttons for '+ Add' and 'Advanced'.
- Add Repository:** A button located below the Credentials section.
- Branches to build:** A section with a 'Branch Specifier (blank for 'any')' input field containing '\*/main'. A red 'x' icon is visible in the top right corner of this section's container.

At the bottom of the configuration area, there are two buttons: 'Save' (in blue) and 'Apply' (in light blue).

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

## SCREENSHOT:

**Jenkins**

Dashboard > devopss > #6

Console Output

Started by user [vijay](#)

Checking out git <https://github.com/vijayvk10/capstone.git> into /var/lib/jenkins/workspace/devopss@script/de4622d98d9f60d875cd9c554a978197b180734723b8a0381c2acd257fb87489 to read Jenkinsfile

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/devopss@script/de4622d98d9f60d875cd9c554a978197b180734723b8a0381c2acd257fb87489/.git # timeout=10

Fetching changes from the remote Git repository

> git config remote.origin.url <https://github.com/vijayvk10/capstone.git> # timeout=10

Fetching upstream changes from <https://github.com/vijayvk10/capstone.git>

> git --version # timeout=10

> git --version # 'git version 2.43.0'

> git fetch --tags --force --progress -- <https://github.com/vijayvk10/capstone.git> +refs/heads/\*:refs/remotes/origin/\* # timeout=10

7) Now Build this docker image in the terminal with desired port number to it. **CODE:**

docker images

docker build -itd -p 70:80 test1

**SCREENSHOT:**

```
vijay@LAPTOP-KFMKT43R: ~$ sudo docker images
[sudo] password for vijay:
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
test          latest   a353a37920c9   4 hours ago    192MB
nginx         latest   53a18edff809   6 weeks ago    192MB
vijayvk10/dev latest   53a18edff809   6 weeks ago    192MB
vijay@LAPTOP-KFMKT43R:~$ sudo usermod -ag docker jenkins
[sudo] password for vijay:

[1]+  Stopped                  sudo usermod -ag docker jenkins
vijay@LAPTOP-KFMKT43R:~$ sudo usermod -aG docker jenkins
[sudo] password for vijay:
vijay@LAPTOP-KFMKT43R:~$ sudo systemctl restart jenkins
vijay@LAPTOP-KFMKT43R:~$ sudo systemctl restart docker
vijay@LAPTOP-KFMKT43R:~$ docker images
permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Head "http://%2Fvar%2Frun%2Fdocker.sock/_ping": dial unix /var/run/docker.sock: connect: permission denied
vijay@LAPTOP-KFMKT43R:~$ sudo docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
test1         latest   875517f735c0   50 seconds ago 195MB
test          latest   a353a37920c9   5 hours ago    192MB
vijayvk10/dev latest   a353a37920c9   5 hours ago    192MB
nginx         latest   53a18edff809   6 weeks ago    192MB
vijayvk10/dev <none>  53a18edff809   6 weeks ago    192MB
vijay@LAPTOP-KFMKT43R:~$ sudo docker run -itd -p 70:80 test1
cabbda94e072b6e91ae2af07b718667f9fea8ef6ca25fb9f5e2760ddeb1b5eca
vijay@LAPTOP-KFMKT43R:~$
```

8) Go to the Browser and search for localhost:<PORT\_NUMBER> and the

respective application will be hosted.

## SCREENSHOT:

