Practical - 4

Collaborating and Cloning Using GitHub

Step 1. Clone a public repository:

git clone https://github.com/bhargavikeni12/Demo.git

cd Demo

```
HP@DESKTOP-PNISSJS MINGW64 ~
$ git clone https://github.com/bhargavikeni12/Demo.git
Cloning into 'Demo'...
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.

HP@DESKTOP-PNISSJS MINGW64 ~
$ cd Demo
```

Step 2: Create a New Branch

git checkout -b update-readme

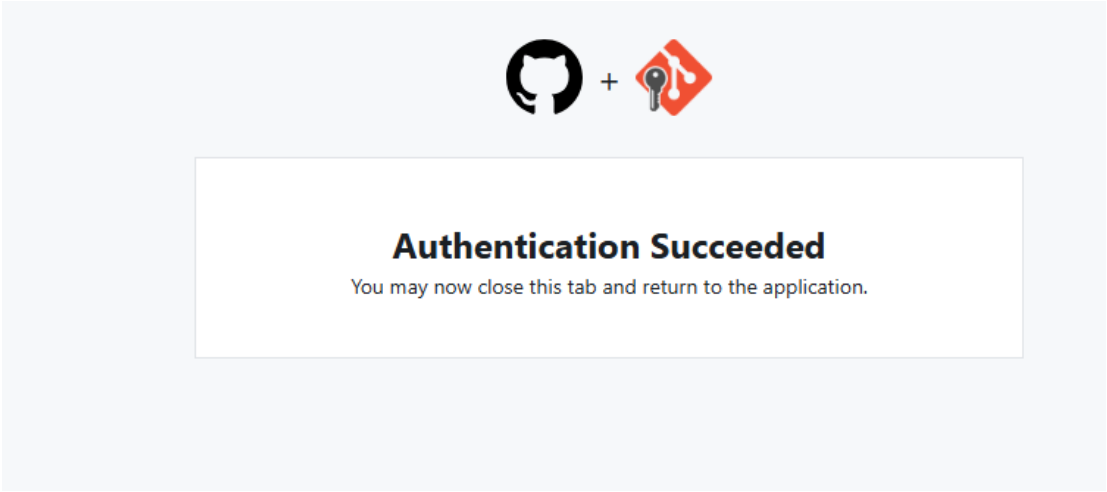
```
HP@DESKTOP-PNISSJS MINGW64 ~/Demo (main)
$ git checkout -b update-readme
Switched to a new branch 'update-readme'

HP@DESKTOP-PNISSJS MINGW64 ~/Demo (update-readme)
$ echo "Added a line for practical" >> README.md

HP@DESKTOP-PNISSJS MINGW64 ~/Demo (update-readme)
$ git add README.md
warning: in the working copy of 'README.md', LF will be replaced by CRLF the next time Git touches it
```

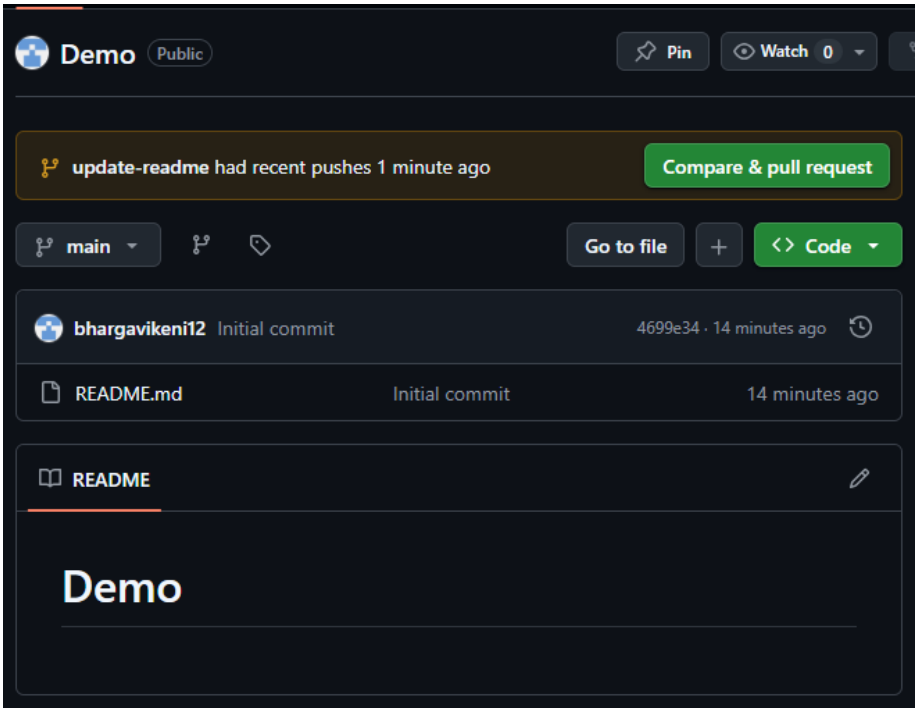
```
HP@DESKTOP-PNISSJS MINGW64 ~/Demo (update-readme)
$ git config --global user.name "Bhargavi Keni"

HP@DESKTOP-PNISSJS MINGW64 ~/Demo (update-readme)
$ git config --global user.email "bhargavikeni12@gmail.com"

HP@DESKTOP-PNISSJS MINGW64 ~/Demo (update-readme)
$ git commit -m "Updated README for practical"
[update-readme 4586019] Updated README for practical
 1 file changed, 1 insertion(+), 1 deletion(-)
```

Step 3: Push and Open pull request:

git push origin update-readme

```
HP@DESKTOP-PNISSJS MINGW64 ~/Demo (update-readme)
$ git push origin update-readme
info: please complete authentication in your browser...
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Writing objects: 100% (3/3), 299 bytes | 299.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'update-readme' on GitHub by visiting:
remote:      https://github.com/bhargavikeni12/Demo/pull/new/update-readme
remote:
To https://github.com/bhargavikeni12/Demo.git
 * [new branch]      update-readme -> update-readme
```





Name-Bhargavi-89

Practical No – 5

Using GitLab Web IDE

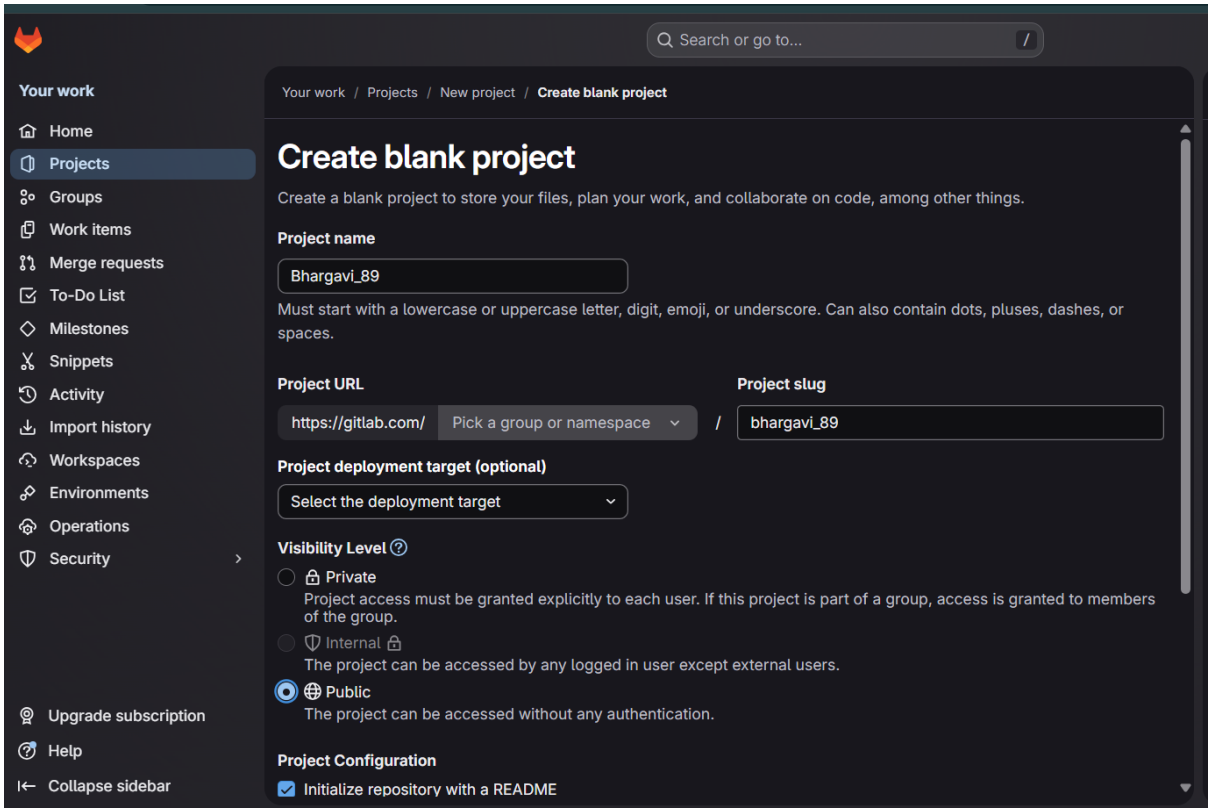
GitLab Web IDE

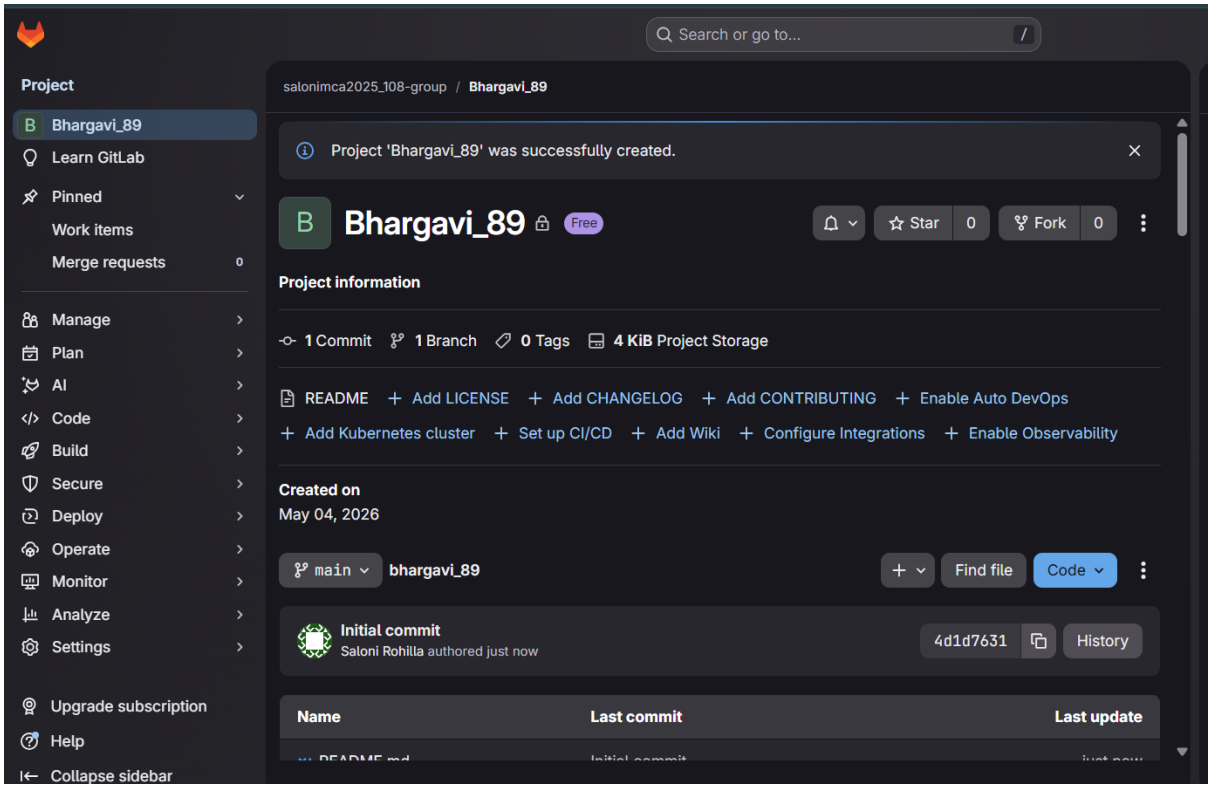
Steps:

1. Sign up at https://gitlab.com

2. Create a project.

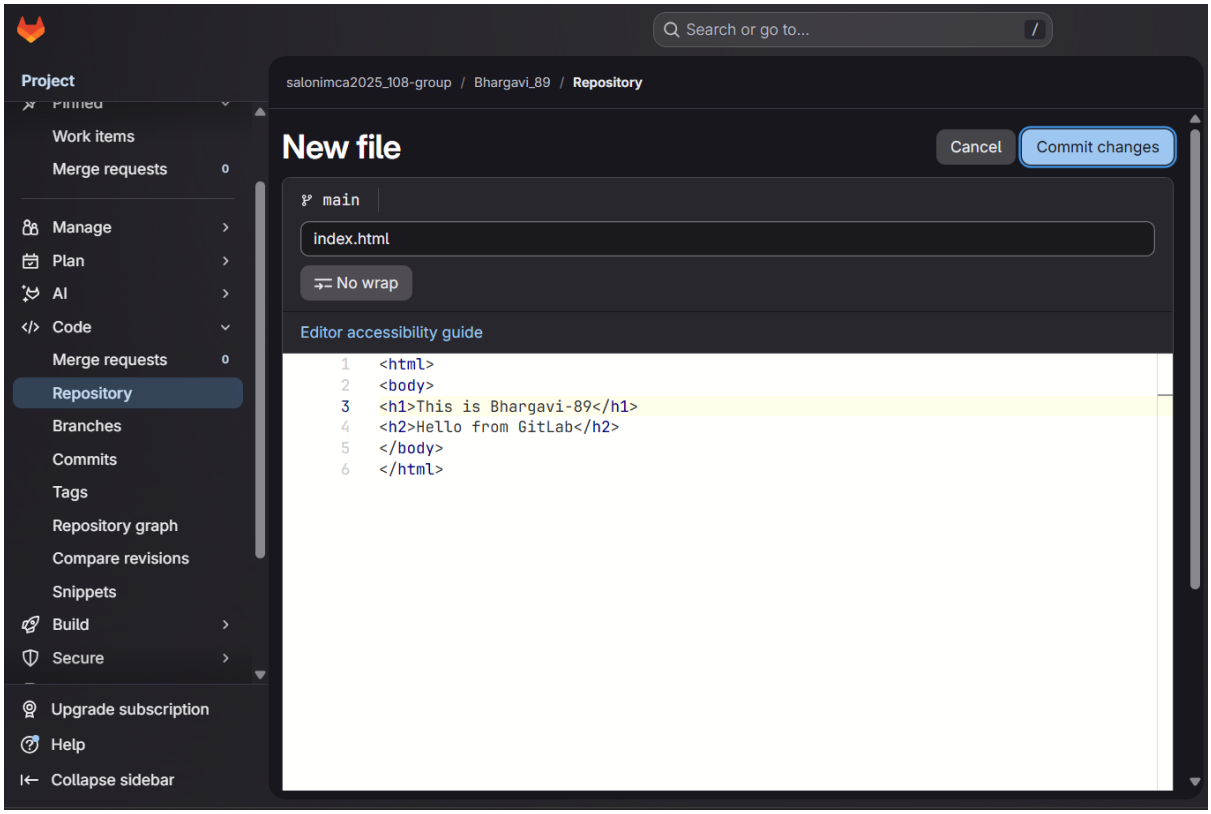
3. Click on Web IDE in your repository.



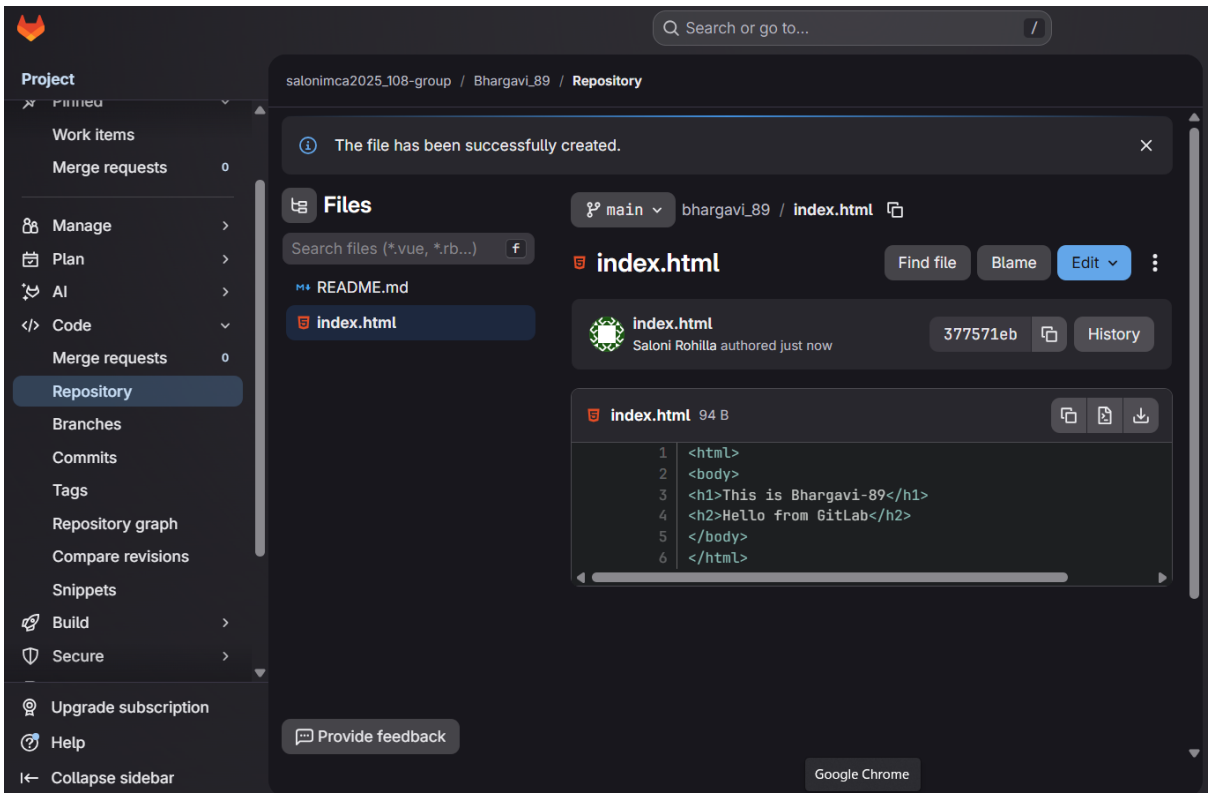


4. Create a file (index.html):

```
<html>
<body>
   <h1>Hello Prince</h1>
   <h2>Working on Gitlab</h2>
</body>
</html>
```



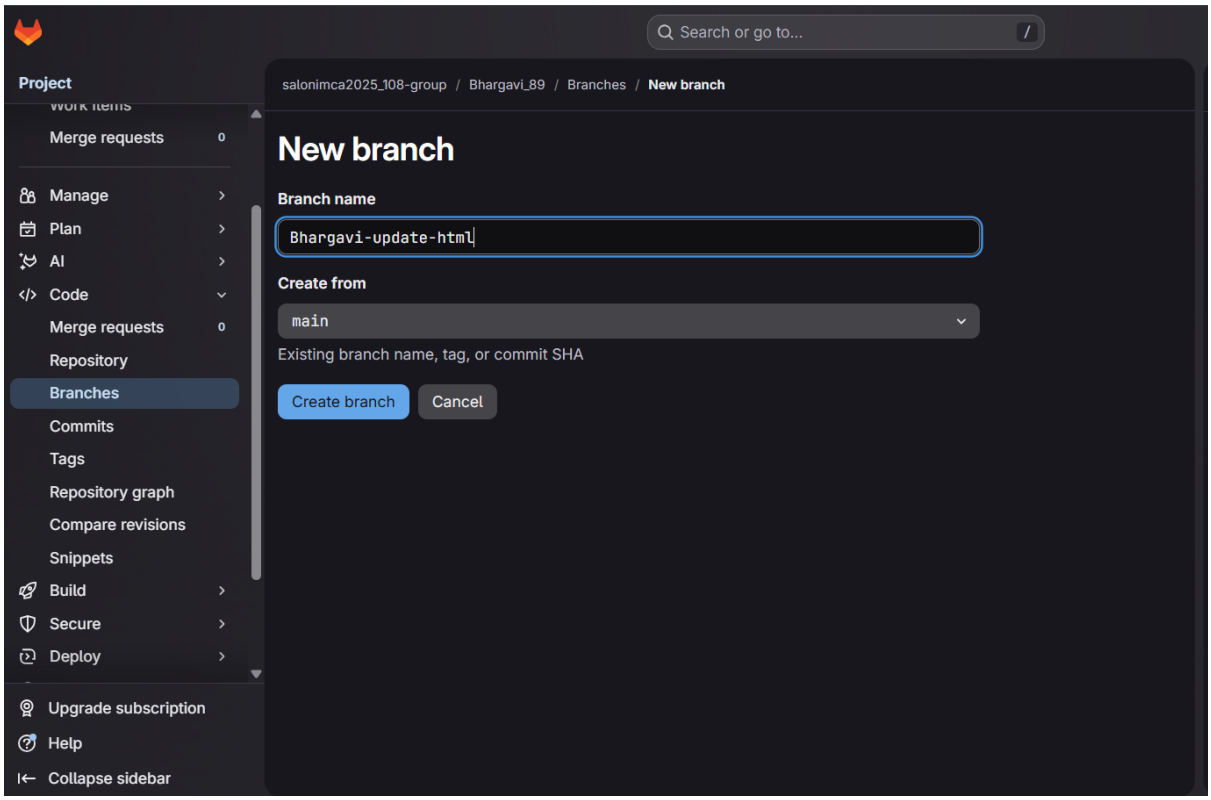
5. Click Commit and push changes

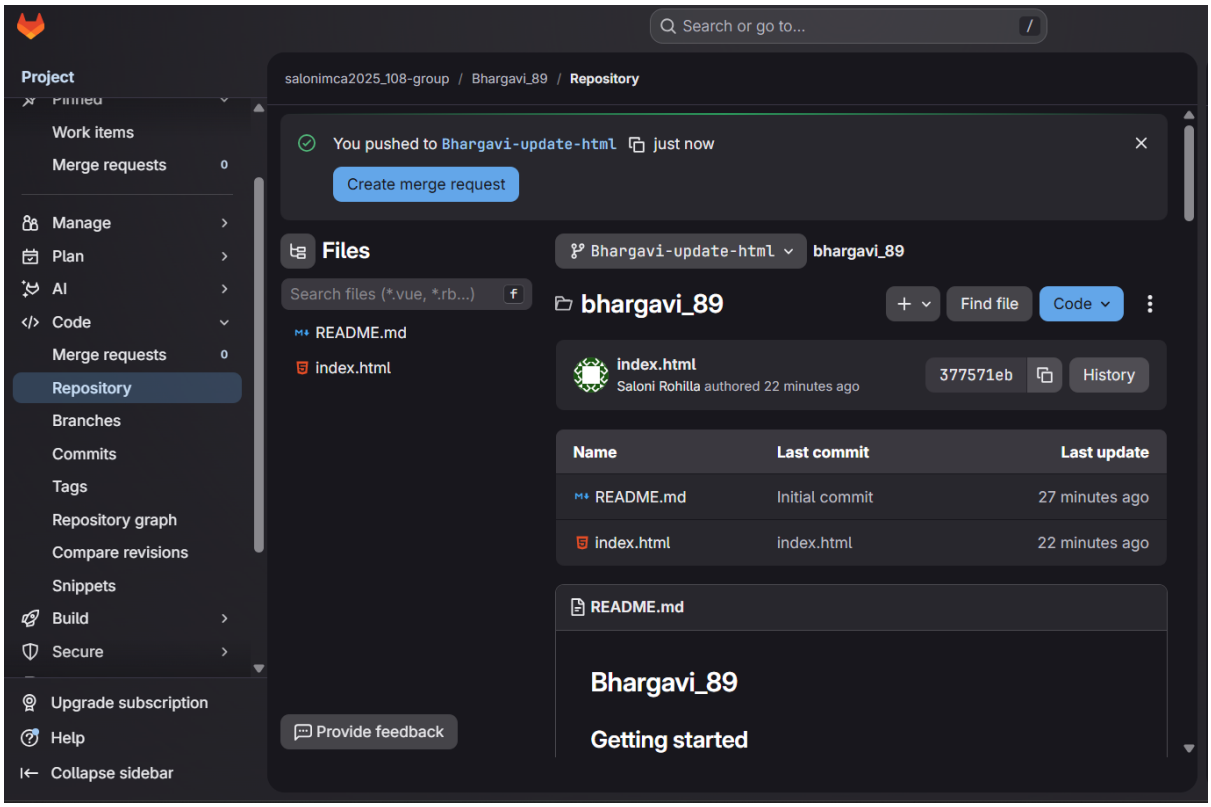


Practical – 6

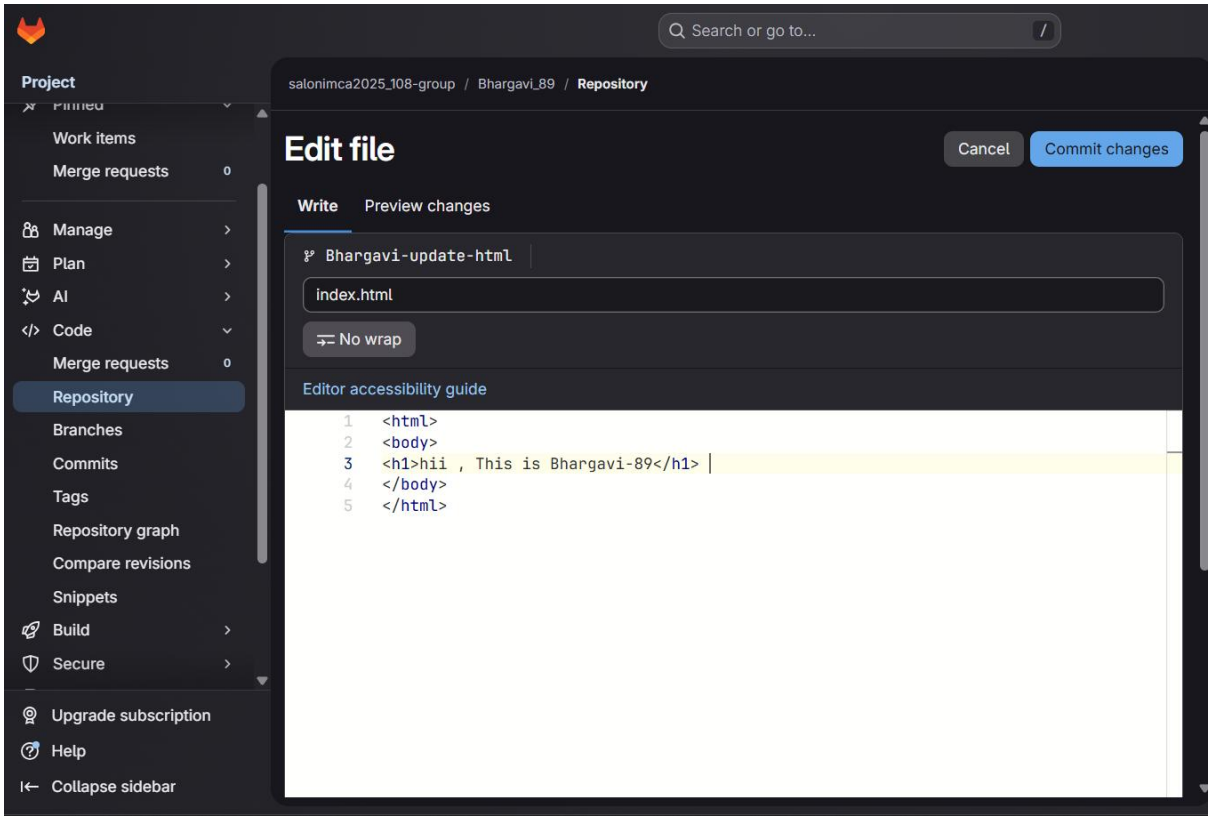
Performing Merge Requests Using GitLab

Step 1. Create a new branch in Web IDE

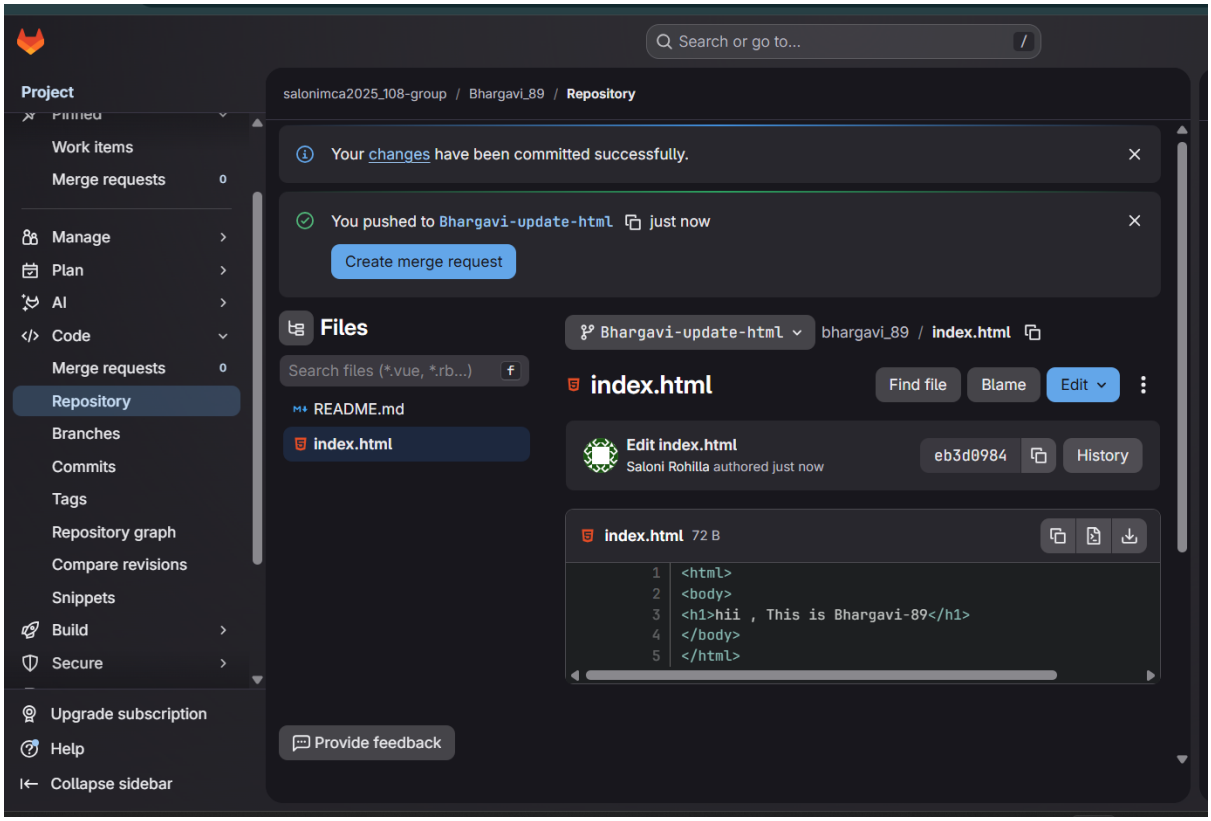


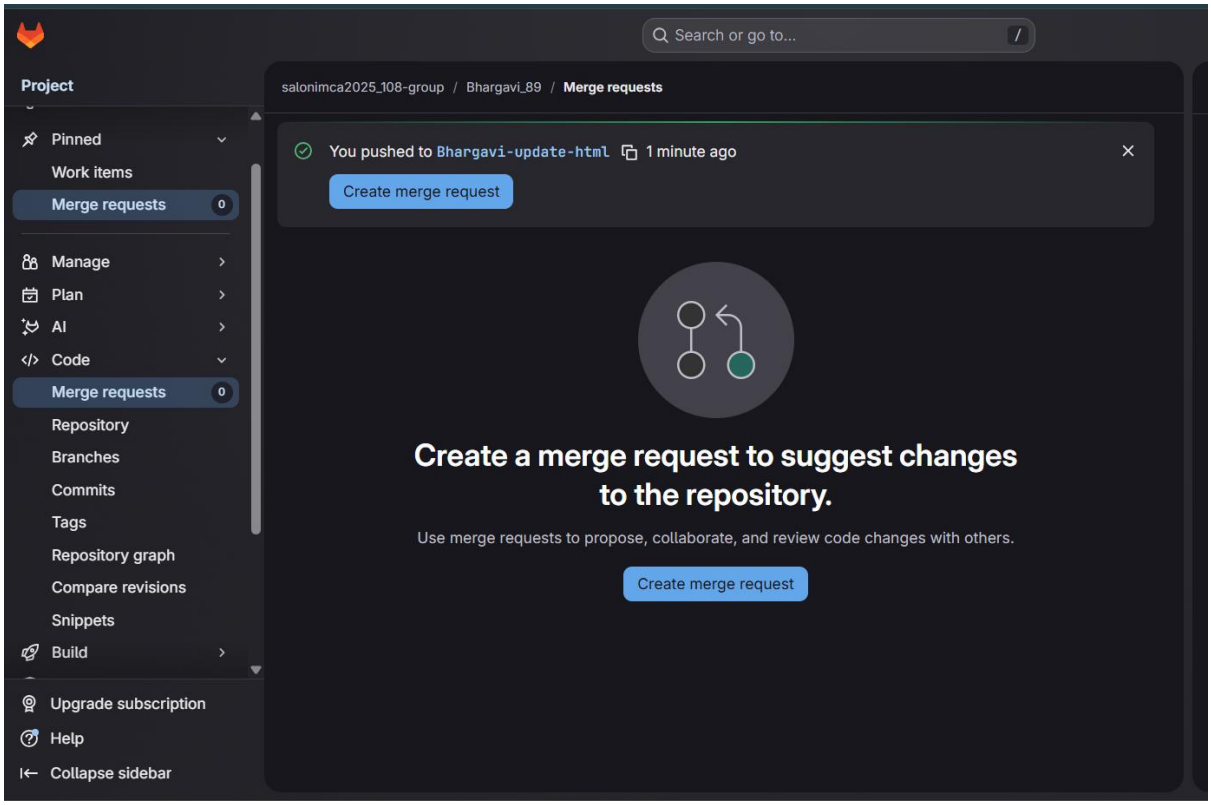


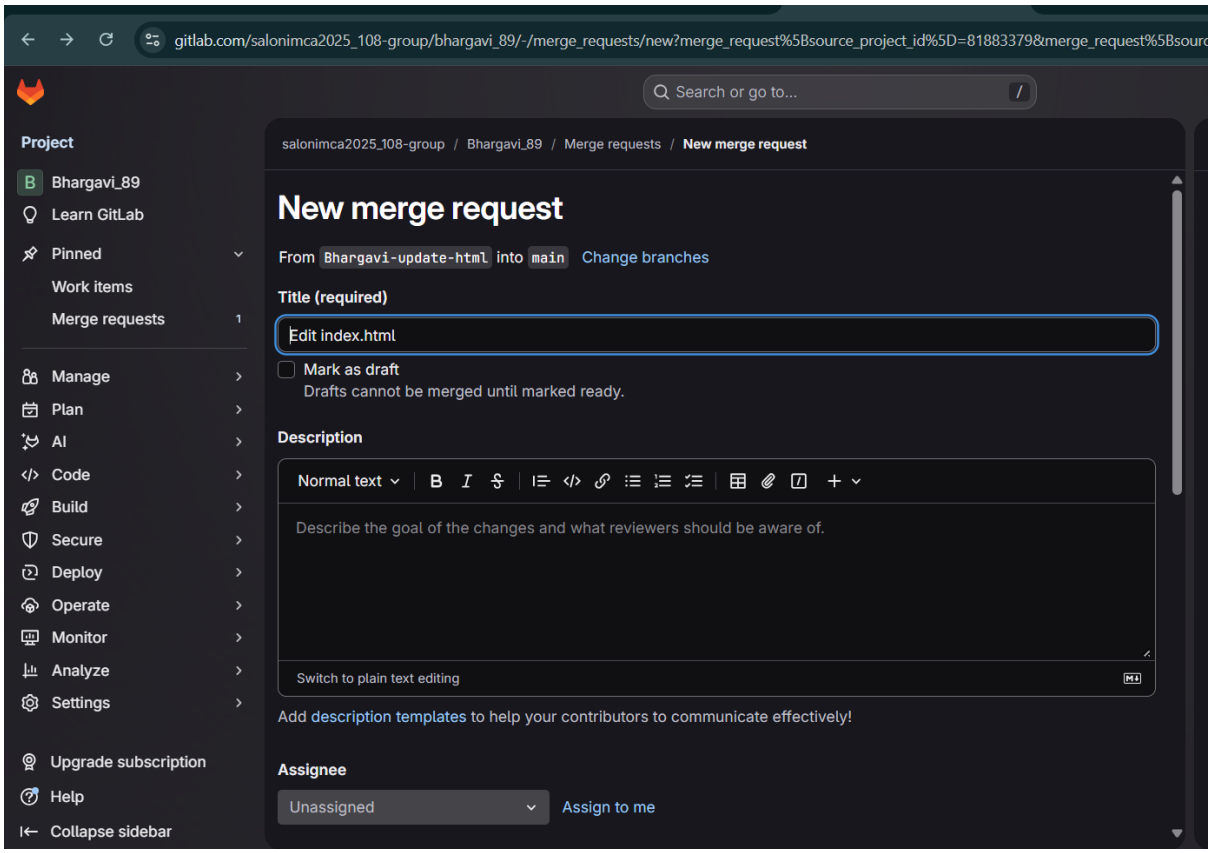
2. Add/edit a file and commit



3. Click on Merge Requests > New Merge Request.



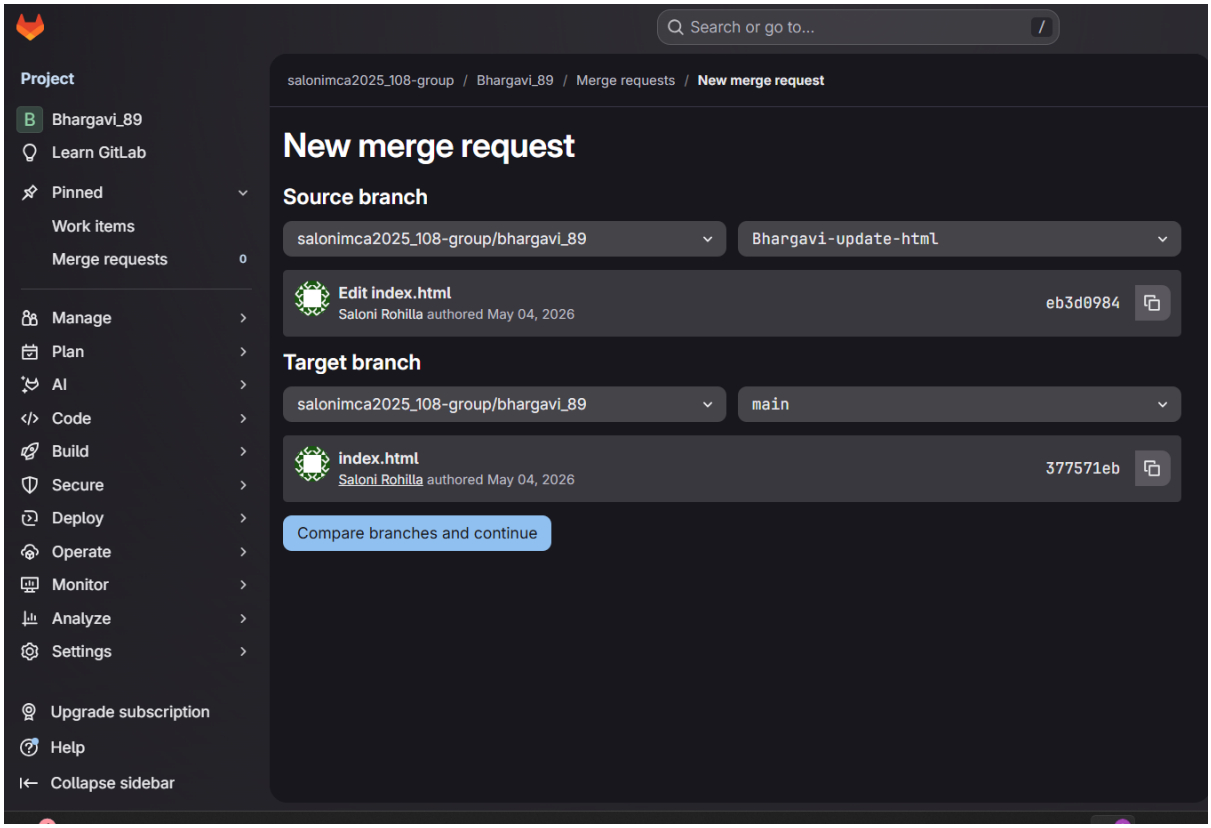


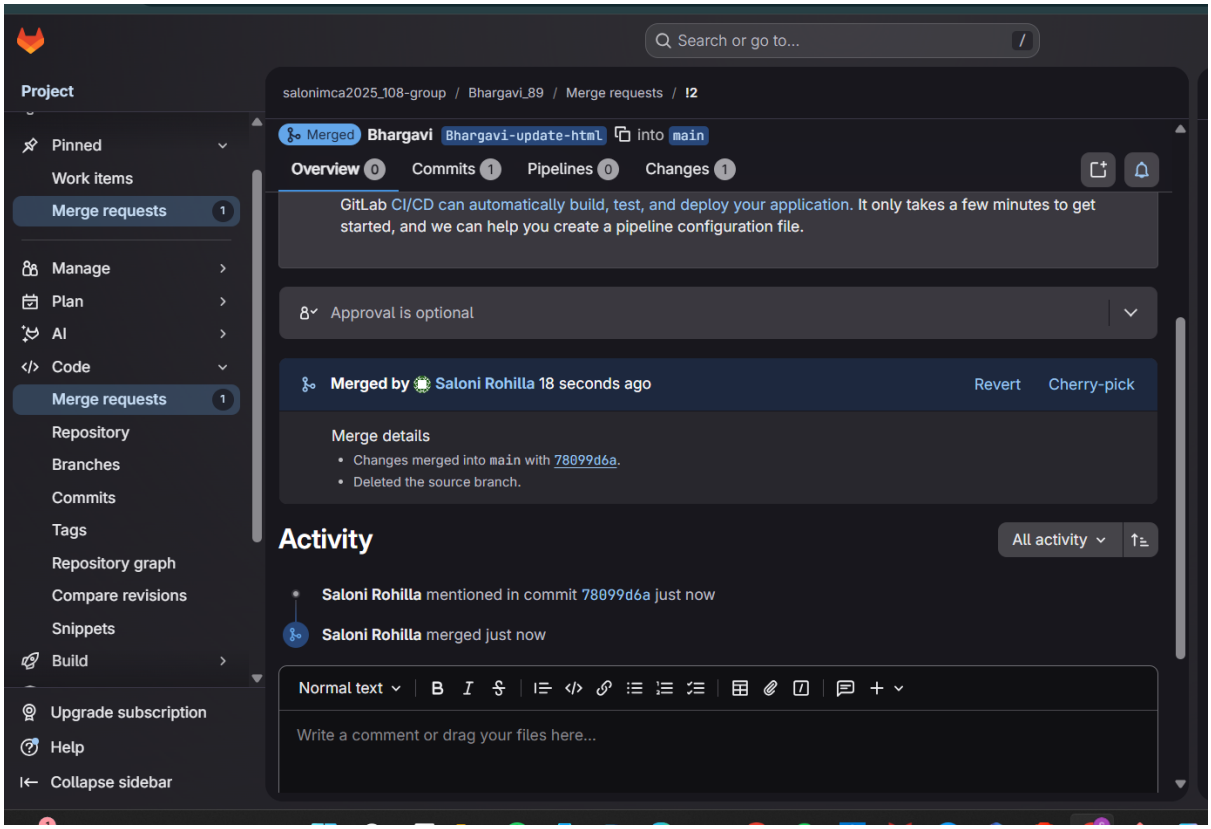


4. Select source and target branches

5. Submit and merge after review

Name-Bhargavi-89



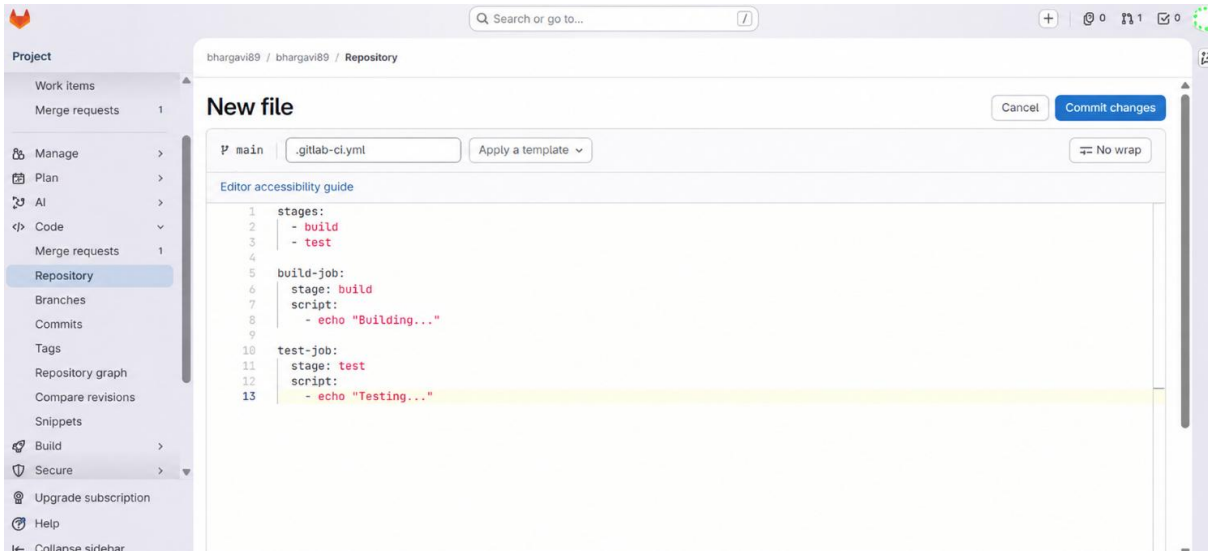


Practical No – 7

Workflow Management in GitLab

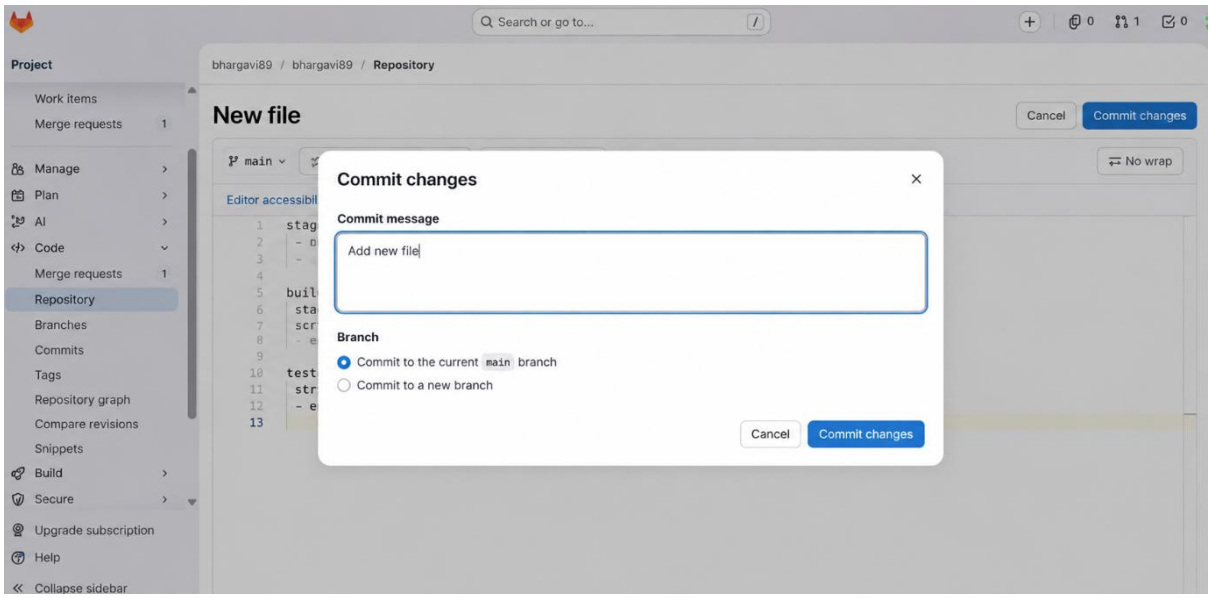
Steps1: In your repo, create .gitlab-ci.yml:

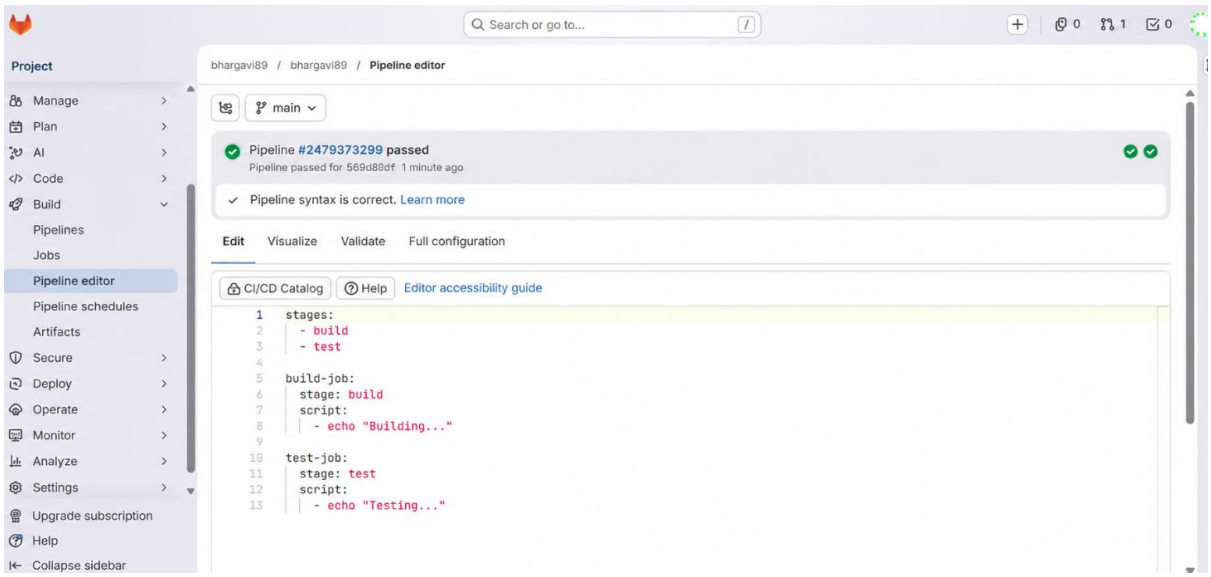
```
stages:
  - build
  - test
build-job:
  stage: build
  script:
    - echo "Building..."
test-job:
  stage: test
  script:
    - echo "Testing..."
```

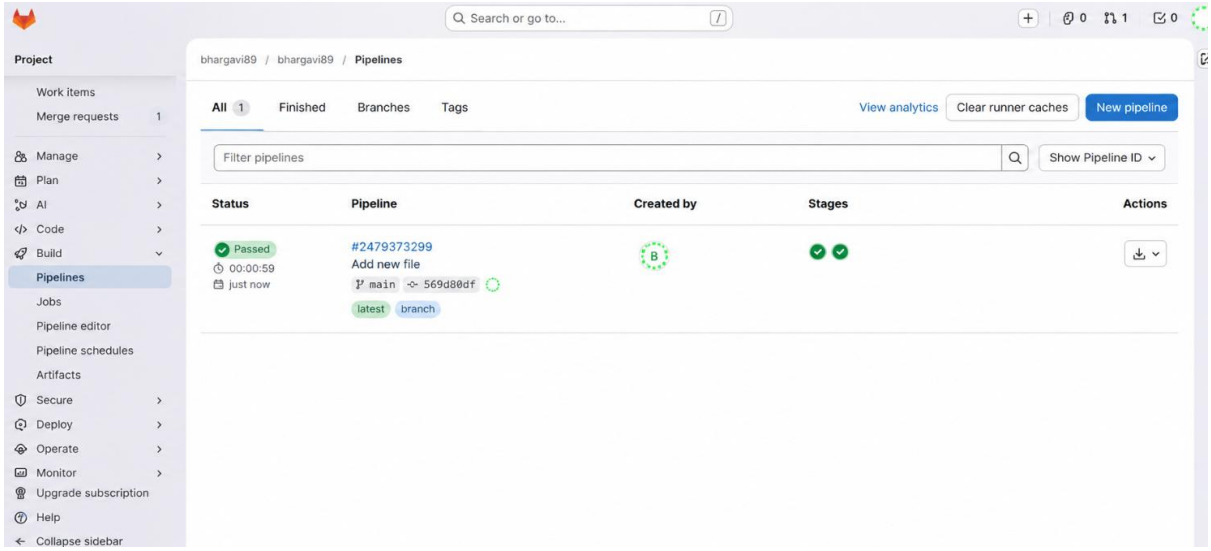


Name-Bhargavi-89

2. Commit and push.

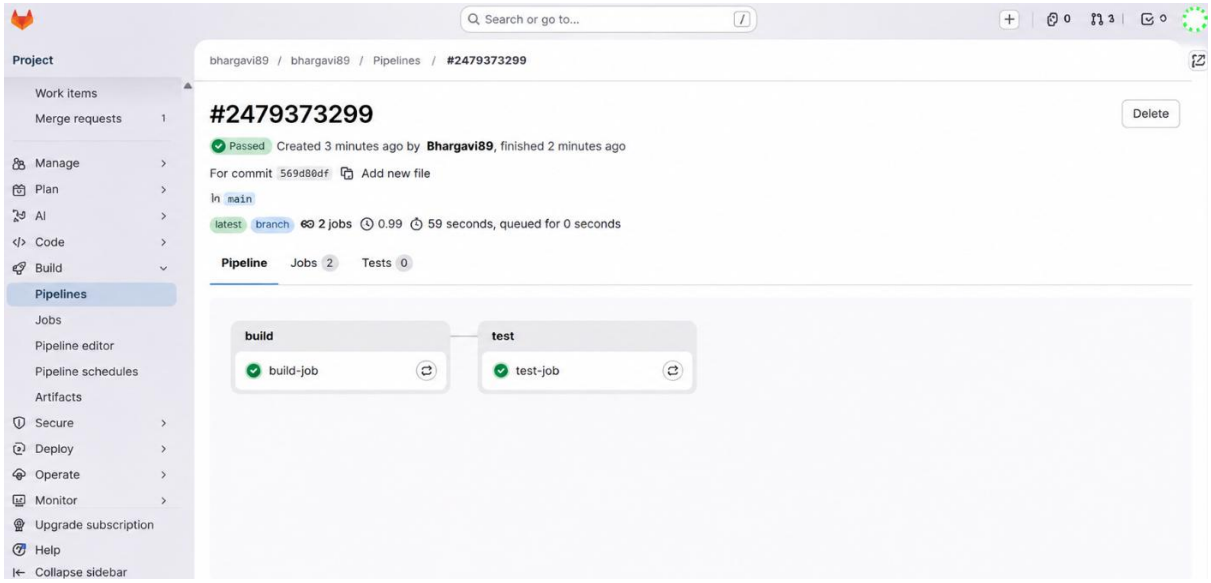






Name-Bhargavi-89

3. Go to CI/CD > Pipelines and view the build/test stages.



Name-Bhargavi-89

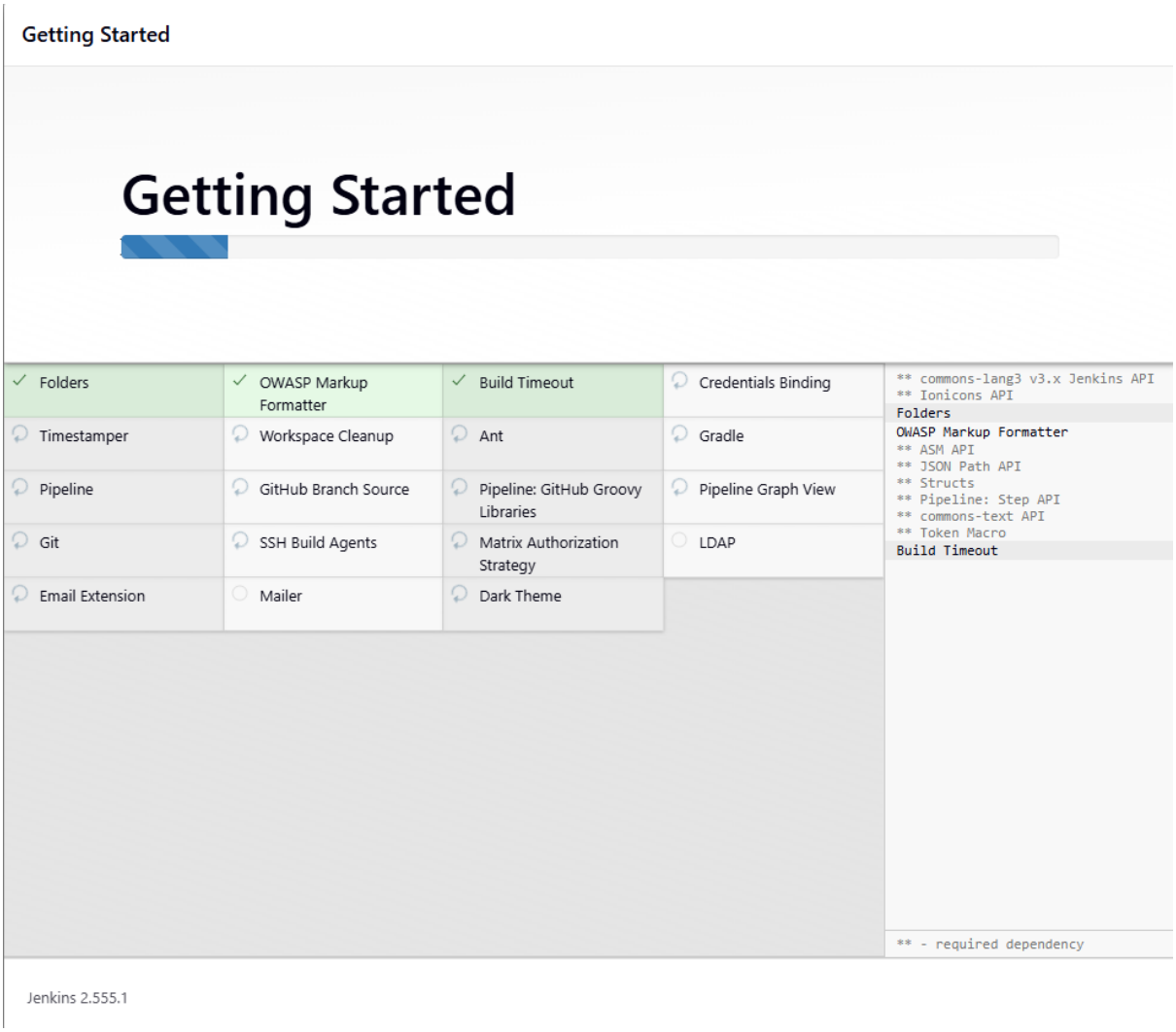
Practical No – 8

Continuous Integration and Development Using Jenkins

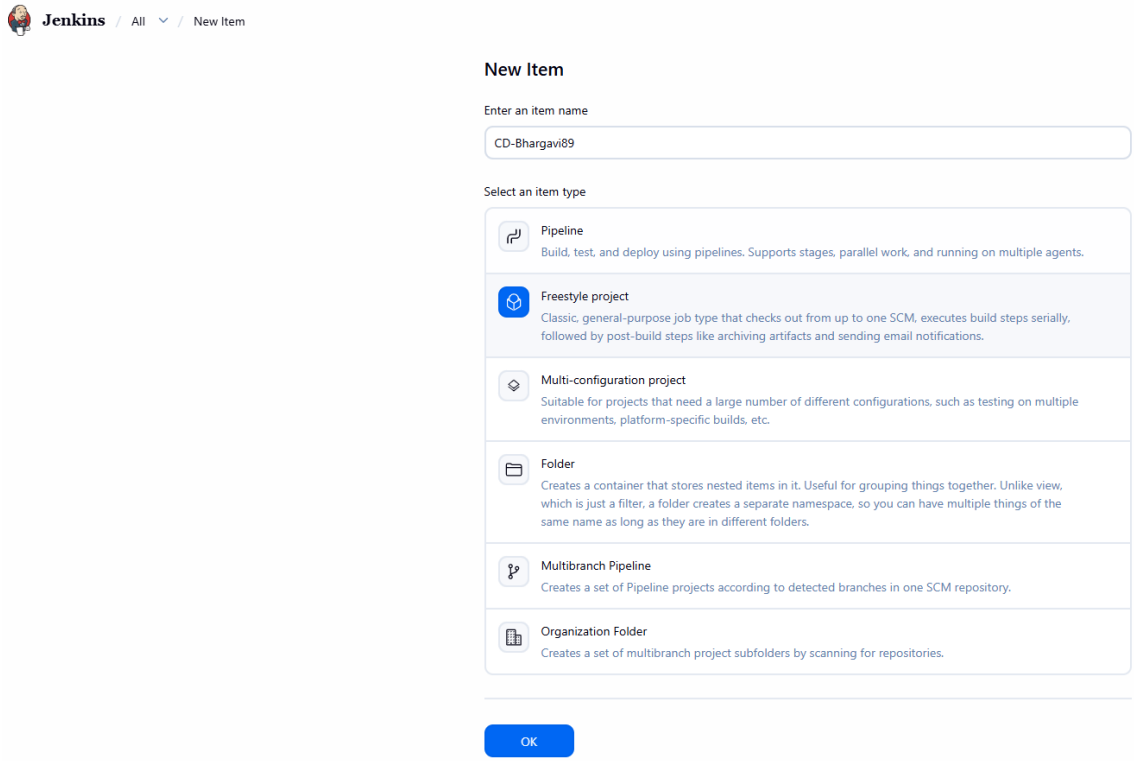
Steps:

1. Install Jenkins (visit https://www.jenkins.io)

2. Run Jenkins: http://localhost:8080



3. Create new Freestyle Project: CD-Bhargavi89



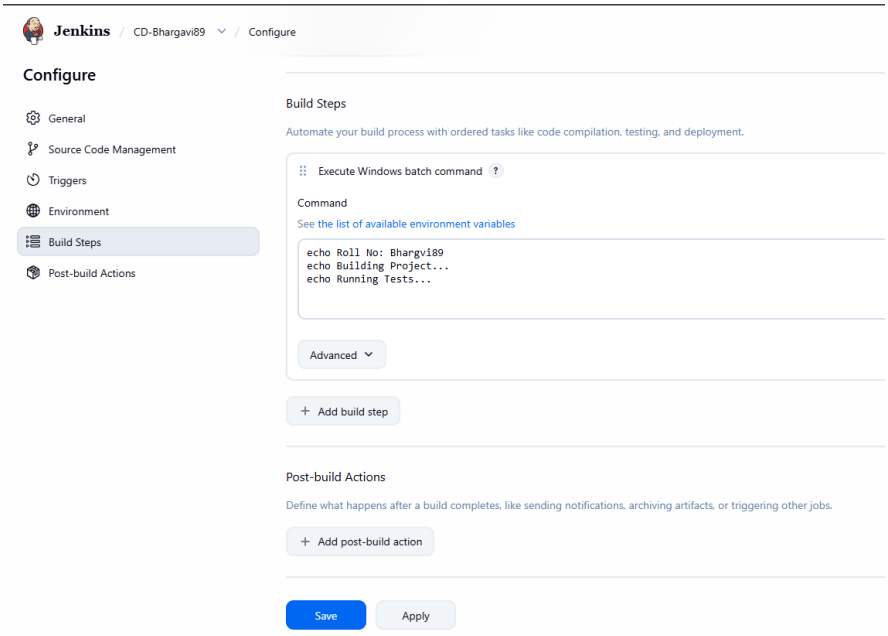
4. Under Source Code Management, choose Git and enter your repo URL.

5. Add Build Step > Execute Windows batch command:

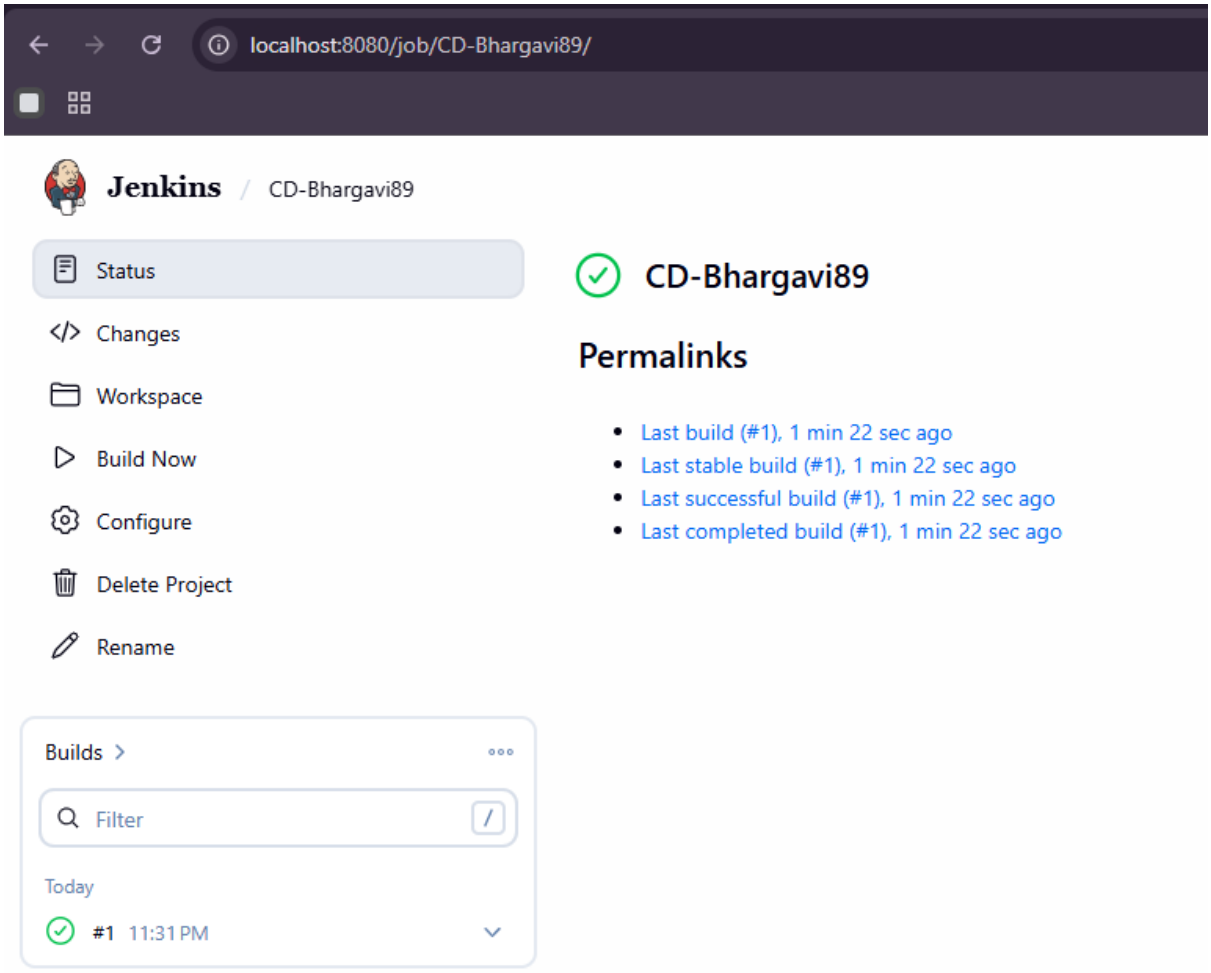
echo Roll No.: Bhargvi89

echo Building Project...

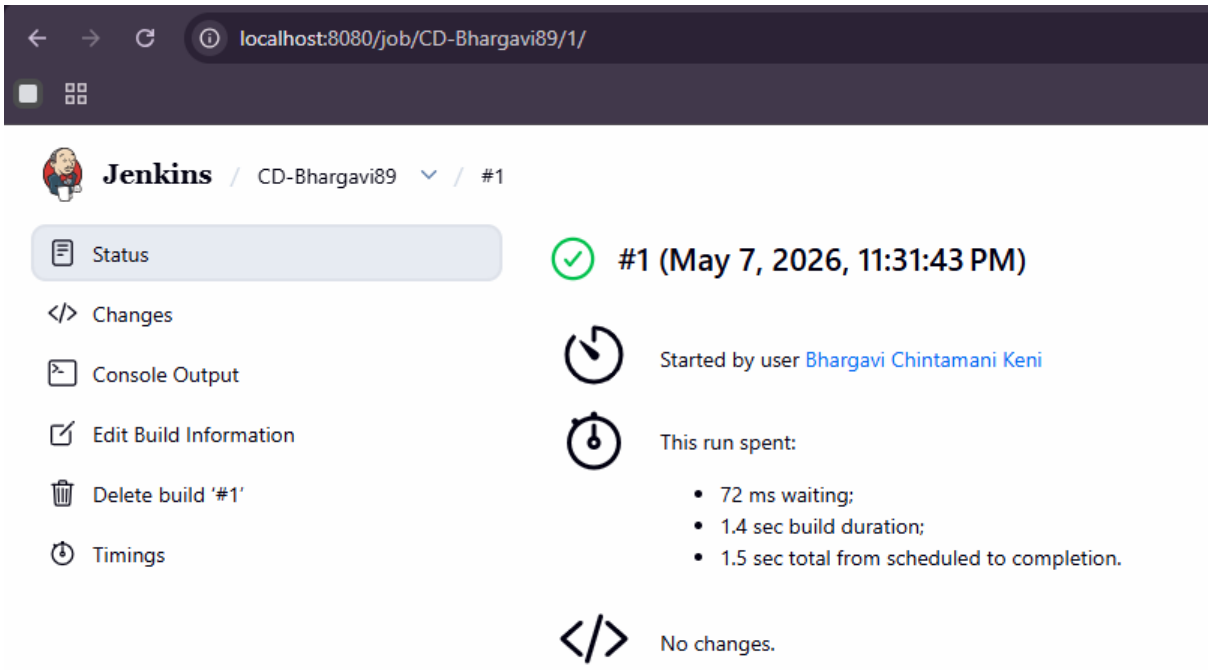
echo Run tests...

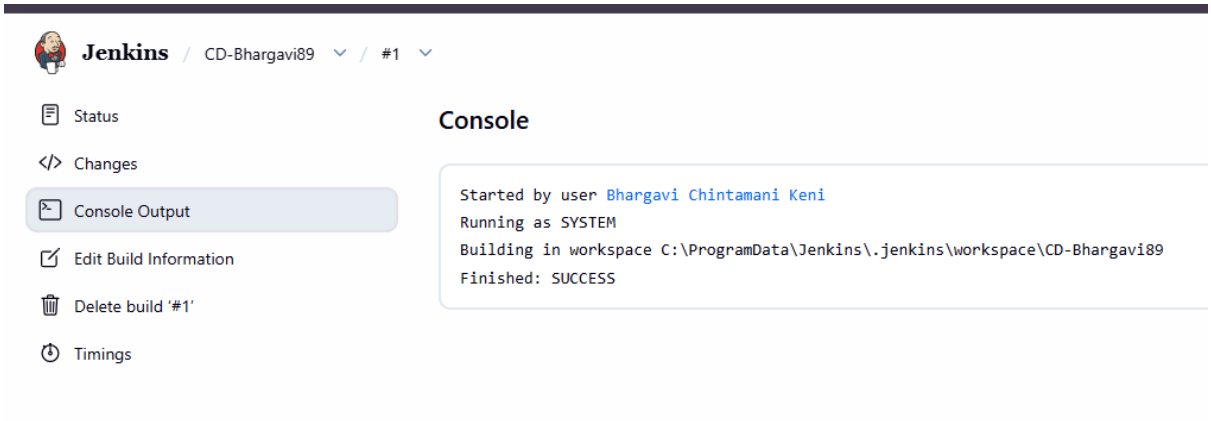


6. Save and click Build Now.



7. Check output in Console Output.





Practical 9

Explore Docker Commands for Content Management

1. Check Docker Version

docker version

2. Pull Nginx Image

docker pull nginx

```
C:\Users\HP>docker --version
Docker version 29.4.2, build 055a478

C:\Users\HP>docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
4f55086f7dd0: Pull complete
d5e71e642bf5: Download complete
Digest: sha256:f9078146db2e05e794366b1bfe584a14ea6317f44027d10ef7dad65279026885
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.
```

2. Pull Nginx Image

docker pull nginx

```
C:\Users\HP>docker pull nginx
Using default tag: latest
latest: Pulling from library/nginx
85c66128325a: Pull complete
ff048f1f2159: Pull complete
801a1ad15b4e: Pull complete
4677c2a9a3d4: Pull complete
677c63196868: Pull complete
ce776bbcda0d: Pull complete
3531af2bc2a9: Pull complete
54c38c75806e: Download complete
69989ccd189b: Download complete
Digest: sha256:6e23479198b998e5e25921dff8455837c7636a67111a04a635cf1bb363d199dc
Status: Downloaded newer image for nginx:latest
docker.io/library/nginx:latest

What's next:
    View a summary of image vulnerabilities and recommendations → docker scout quickview nginx
```

3. List Docker Images

docker images

4. Run Nginx Container

docker run -d -p 8081:80 --name mynginx nginx

```
C:\Users\HP>docker run -d -p 8081:80 --name mynginx nginx
13832fcb85425215306b6458d0d6604699054e11eaf6b51d12540ed0e6599efd
```

5. List Running Containers

docker ps

```
C:\Users\HP>docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED              STATUS              PORTS
                NAMES
13832fcb8542   nginx     "/docker-entrypoint.…"   About a minute ago   Up About a minute   0.0.0.0:8081->80/tcp, [::]:80
81->80/tcp   mynginx
```

6. Copy File from Host to Container

docker cp "C:\Users\HP\Documents\bhargavi89.txt"
mynginx:/usr/share/nginx/html/index.html

```
C:\Users\HP>docker cp "C:\Users\HP\Documents\bhargavi89.txt" mynginx:/usr/share/nginx/html/index.html
Successfully copied 139B (transferred 2.05kB) to mynginx:/usr/share/nginx/html/index.html
```

7. Copy File from Container to Host

docker cp mynginx:/usr/share/nginx/html/index.html
"C:\Users\HP\Documents\bhargavi89.txt"

```
C:\Users\HP>docker cp "C:\Users\HP\Documents\bhargavi89.txt" mynginx:/usr/share/nginx/html/index.html
Successfully copied 139B (transferred 2.05kB) to mynginx:/usr/share/nginx/html/index.html
```

8. Create Docker Volume

docker volume create mydata

```
C:\Users\HP>docker volume create mydata89
mydata89
```

9. List Docker Volumes

docker volume ls

```
C:\Users\HP>docker volume ls
DRIVER    VOLUME NAME
local     mydata89
```

Step 10: Remove Container

docker rm -f mynginx

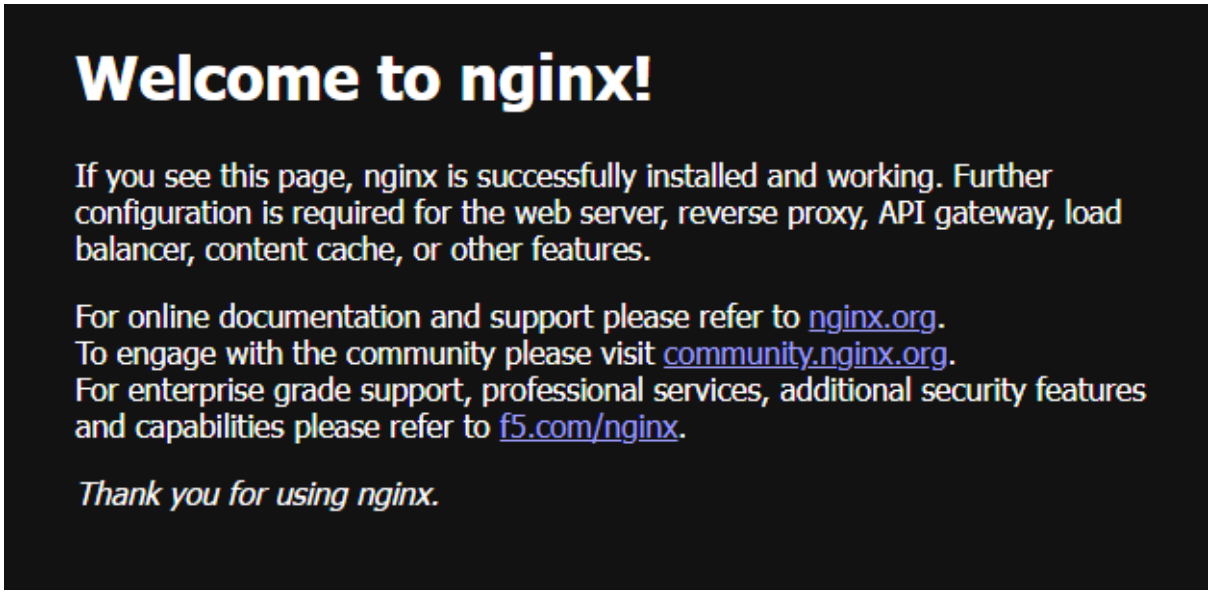
```
C:\Users\HP>docker rm -f mynginx
mynginx
```

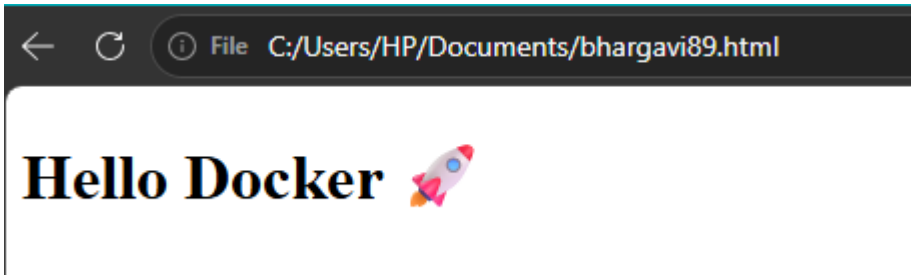
11. Remove Docker Image

docker rmi -f nginx

```
C:\Users\HP>docker rmi -f nginx
Untagged: nginx:latest
Deleted: sha256:6e23479198b998e5e25921dff8455837c7636a67111a04a635cf1bb363d199dc
```

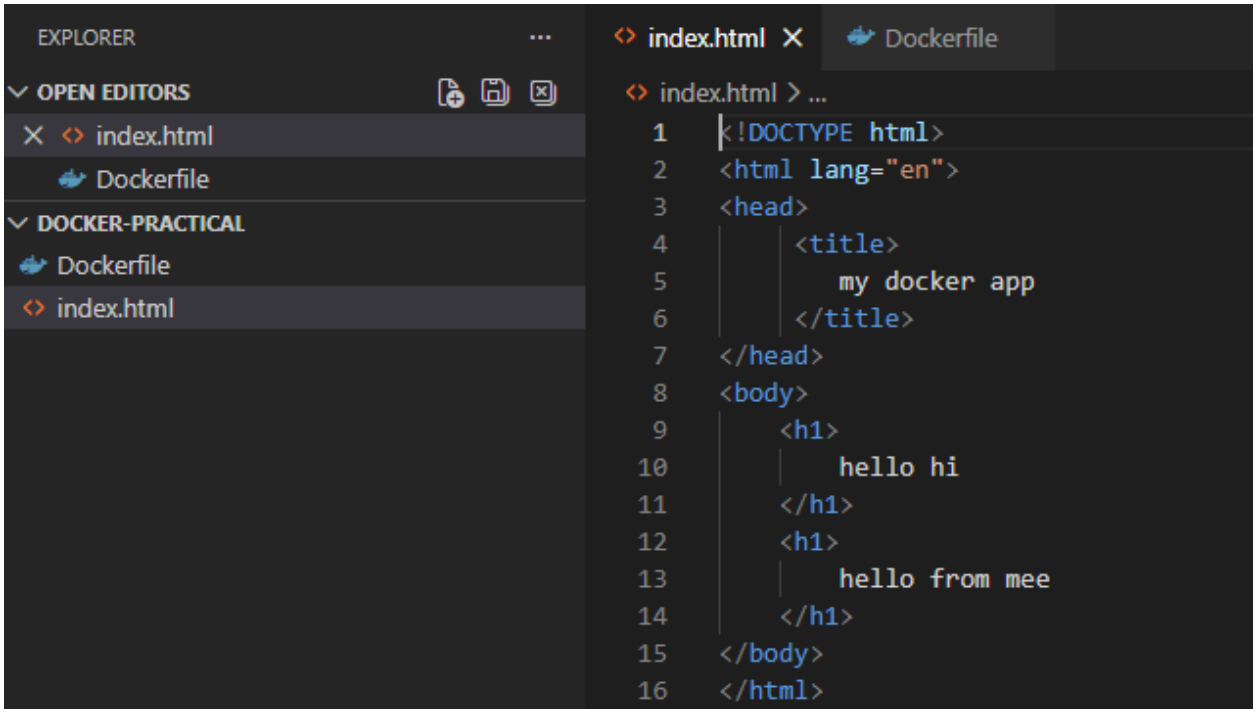
Output



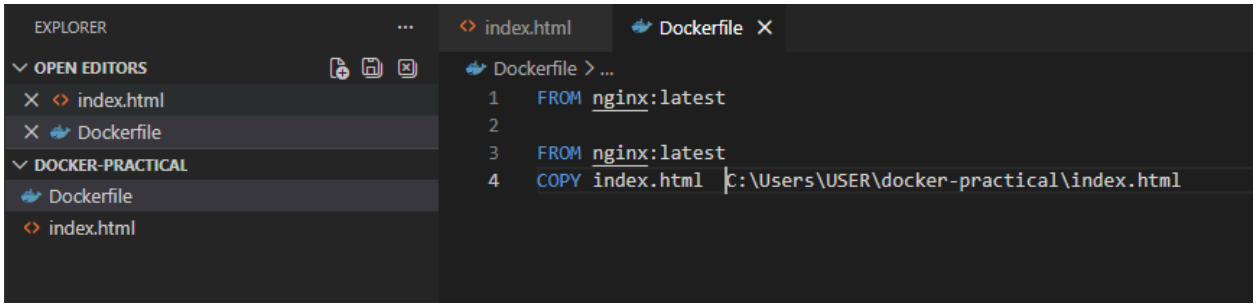


Practical – 10

Aim: Develop a Simple Containerized Application using Docker



Dokerfile



cd docker-practical

docker build -t my-docker-webapp .

docker run -d -p 8081:80 --name webapp-container my-docker-webapp

docker ps

docker stop webapp-container

docker rm webapp-container

docker rmi webapp-container

Name-Bhargavi-89

```
PS C:\Users\USER\docker-practical> docker build -t my-docker-webapp .
[+] Building 1.3s (7/7) FINISHED                                                                                    docker:desktop-linux
 => [internal] load build definition from Dockerfile                                                                              0.0s
 => => transferring dockerfile: 135B                                                                                              0.0s
 => [internal] load metadata for docker.io/library/nginx:latest                                                                   0.0s
 => [internal] load .dockerignore                                                                                                 0.0s
 => => transferring context: 2B                                                                                                   0.0s
 => [internal] load build context                                                                                                 0.0s
 => => transferring context: 32B                                                                                                  0.0s
 => [stage-1 1/2] FROM docker.io/library/nginx:latest@sha256:6e23479198b998e5e25921dff8455837c7636a67111a04a635cf1bb363d199dc    0.0s
 => => resolve docker.io/library/nginx:latest@sha256:6e23479198b998e5e25921dff8455837c7636a67111a04a635cf1bb363d199dc             0.0s
 => CACHED [stage-1 2/2] COPY index.html  C:UsersUSERdocker-practicalindex.html                                                   0.0s
 => exporting to image                                                                                                            4.5s
 => => exporting layers                                                                                                           0.0s
 => => exporting manifest sha256:bd4b7fc2fa6de26ac8d356f063072c55fe1f88b5efdd60924e5788937f8c74b1                                 0.0s
 => => exporting config sha256:c08a3a2bffcd58a80e9827fbf4e0ed3fc6c64d411e3e747cb6bc7699cd7f5d8e                                   0.0s
 => => exporting attestation manifest sha256:d0fc8bb3145f5b0009c7383d92aaf3eff56c16881cf2b160c0cacea80d63718f                     0.0s
 => => exporting manifest list sha256:215755742cc58bd69c0c7a535f8613b8cd28ccbe83573d712ada5e2f65c85e01                            0.0s
 => => naming to docker.io/library/my-docker-webapp:latest                                                                        0.0s
 => => unpacking to docker.io/library/my-docker-webapp:latest                                                                     0.0s

View build details: docker-desktop:///dashboard/build/desktop-linux/desktop-linux/pxta4vabdxd67cqr5k723khjn
PS C:\Users\USER\docker-practical> docker run -d -p 8081:80 --name webapp-container my-docker-webapp
2340ef4f2119412b0f5d87f22518483eedc96098891555d2c878d7c4e7205803
PS C:\Users\USER\docker-practical> docker ps
CONTAINER ID   IMAGE              COMMAND                  CREATED         STATUS         PORTS                                       NAMES
2340ef4f2119   my-docker-webapp   "/docker-entrypoint.…"   8 seconds ago   Up 7 seconds   0.0.0.0:8081->80/tcp, [::]:8081->80/tcp   webapp-container
PS C:\Users\USER\docker-practical> docker stop webapp-container
webapp-container
PS C:\Users\USER\docker-practical> docker rm webapp-container
webapp-container
PS C:\Users\USER\docker-practical> docker rmi my-docker-webapp
Untagged: my-docker-webapp:latest
Deleted: sha256:215755742cc58bd69c0c7a535f8613b8cd28ccbe83573d712ada5e2f65c85e01
PS C:\Users\USER\docker-practical>
```