EXPERIMENT-03

AIM: Practice Source code management on GitHub. Experiment with the source code Written in exercise 1.

Objective: To practice source code management activities such as adding, committing, pushing, pulling, branching, merging, and collaboration using GitHub.

Software Requirements

- Git installed (https://git-scm.com/downloads)
- GitHub account (https://github.com/)
- Code editor (VS Code or any)

Procedure

1. Configure Git (First Time Setup)

```
git config --global user.name "Your Name"
git config --global user.email "your-email@example.com"
git config --list
```

---

2. Create or Clone Repository

- **Create a new repository on GitHub** or
- **Clone an existing repository:**

```
git clone https://github.com/your-username/your-repo-name.git
cd your-repo-name
```

---

3. Initialize Local Repository (If Not Cloned)

```
mkdir source-code-management
cd source-code-management
git init
```

---

4. Add Source Code Files

```
echo "print('Hello DevOps')" > main.py
git status
git add main.py
```

---

5. Commit Changes

```
git commit -m "Added main.py with Hello DevOps script"
```

---

6. Create and Manage Branches
git branch feature-branch
git switch feature-branch

---

7. Modify Files and Commit
echo "print('Feature work added')" >> main.py
git add main.py
git commit -m "Updated main.py in feature branch"

---

8. Merge Feature Branch into Main Branch
git switch main
git merge feature-branch

---

9. Connect to GitHub and Push Code
git remote add origin https://github.com/your-username/your-repo-name.git
git branch -M main
git push -u origin main

---

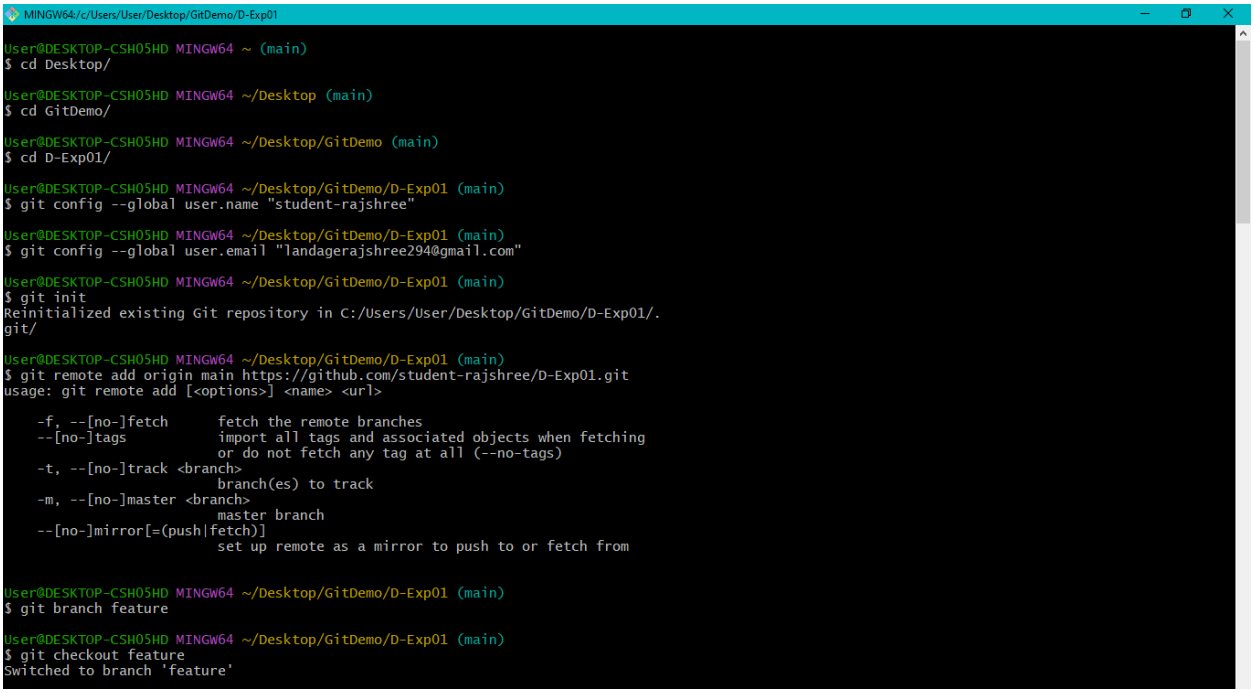
10. Pull Updates from GitHub
git pull origin main

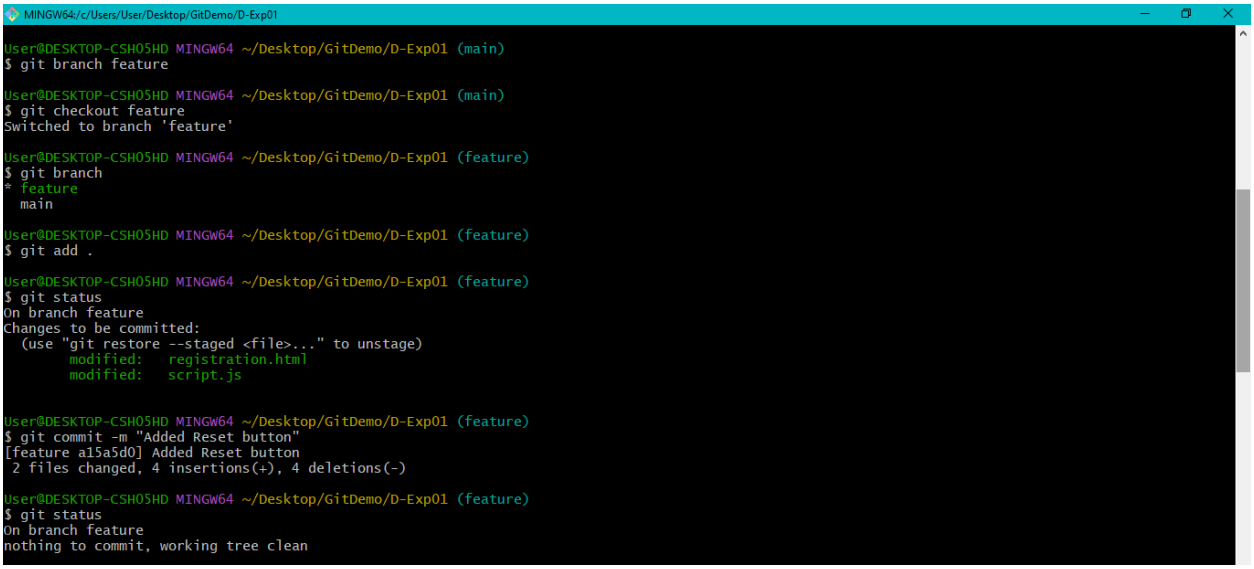
---

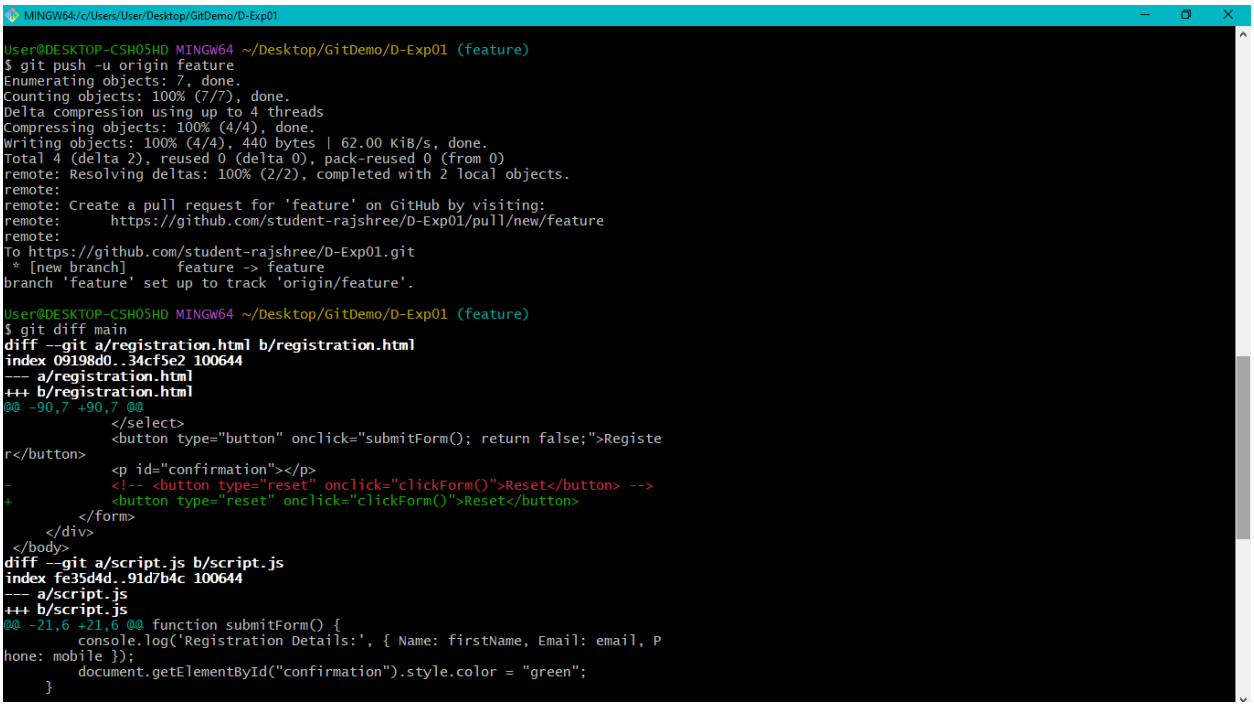
11. Collaboration Tasks

- Collaborators clone repository.
- Create new branches.
- Make code changes.
- Push to GitHub.
- Raise pull requests (PR).
- Review and merge PRs.

OUTPUT

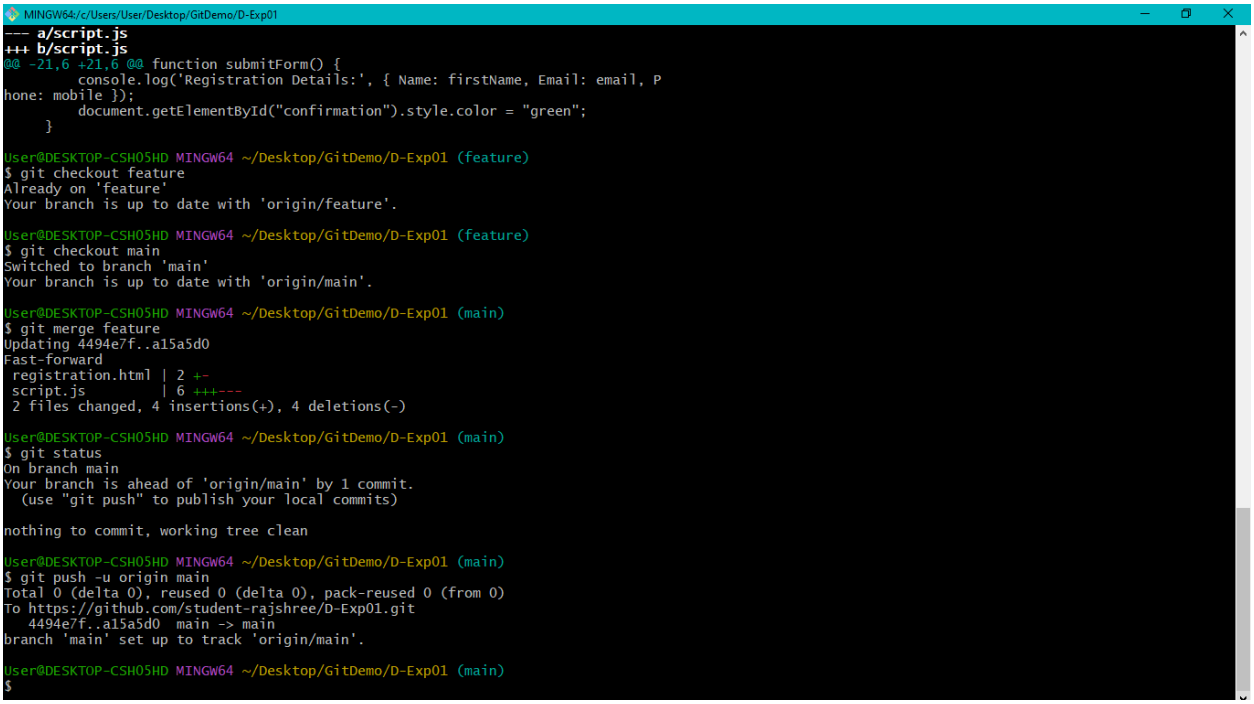

MINGW64:/c/Users/User/Desktop/GitDemo/D-Exp01
User@DESKTOP-CSHO5HD MINGW64 ~ (main)
$ cd Desktop/
User@DESKTOP-CSHO5HD MINGW64 ~/Desktop (main)
$ cd GitDemo/
User@DESKTOP-CSHO5HD MINGW64 ~/Desktop/GitDemo (main)
$ cd D-Exp01/
User@DESKTOP-CSHO5HD MINGW64 ~/Desktop/GitDemo/D-Exp01 (main)
$ git config --global user.name "student-rajshree"
User@DESKTOP-CSHO5HD MINGW64 ~/Desktop/GitDemo/D-Exp01 (main)
$ git config --global user.email "landagerajshree294@gmail.com"
User@DESKTOP-CSHO5HD MINGW64 ~/Desktop/GitDemo/D-Exp01 (main)
$ git init
Reinitialized existing Git repository in C:/Users/User/Desktop/GitDemo/D-Exp01/.
git/
User@DESKTOP-CSHO5HD MINGW64 ~/Desktop/GitDemo/D-Exp01 (main)
$ git remote add origin main https://github.com/student-rajshree/D-Exp01.git
usage: git remote add [<options>] <name> <url>
-f, --[no-]fetch fetch the remote branches
--[no-]tags import all tags and associated objects when fetching
or do not fetch any tag at all (--no-tags)
-t, --[no-]track <branch>
branch(es) to track
-m, --[no-]master <branch>
master branch
--[no-]mirror[=(push|fetch)]
set up remote as a mirror to push to or fetch from
User@DESKTOP-CSHO5HD MINGW64 ~/Desktop/GitDemo/D-Exp01 (main)
$ git branch feature
User@DESKTOP-CSHO5HD MINGW64 ~/Desktop/GitDemo/D-Exp01 (main)
$ git checkout feature
Switched to branch 'feature'

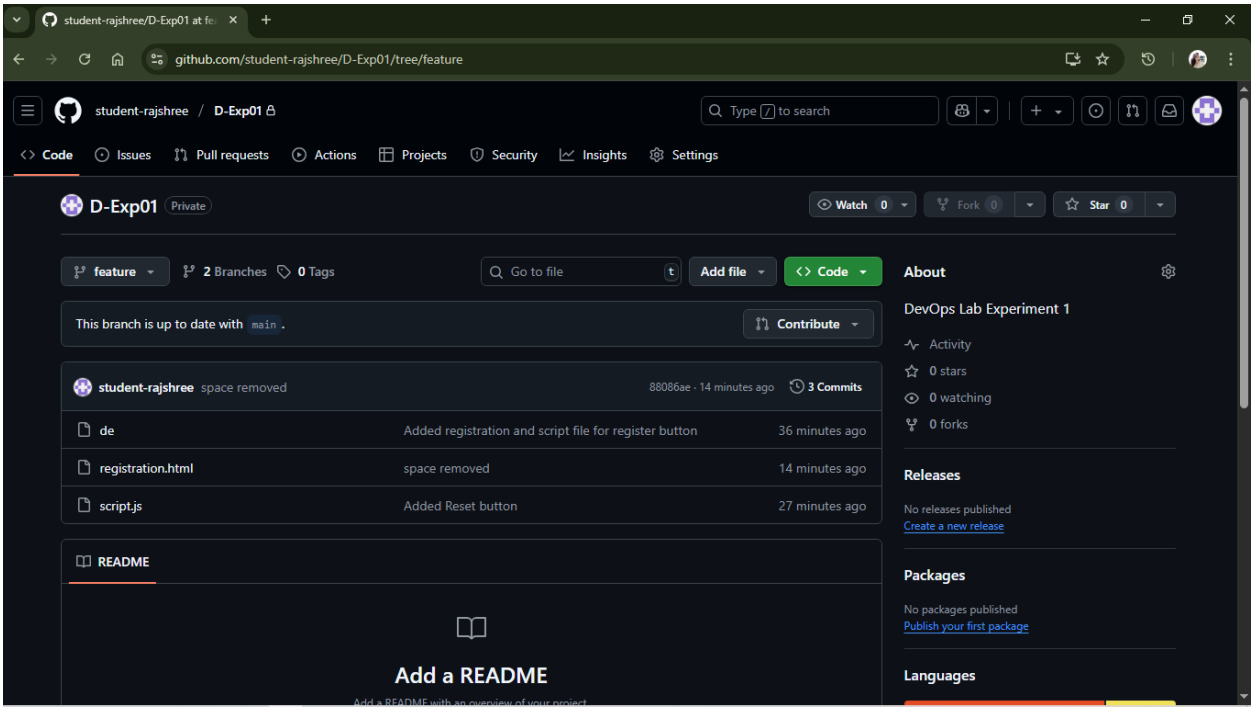

MINGW64:/c/Users/User/Desktop/GitDemo/D-Exp01
User@DESKTOP-CSHO5HD MINGW64 ~/Desktop/GitDemo/D-Exp01 (main)
$ git branch feature
User@DESKTOP-CSHO5HD MINGW64 ~/Desktop/GitDemo/D-Exp01 (main)
$ git checkout feature
Switched to branch 'feature'
User@DESKTOP-CSHO5HD MINGW64 ~/Desktop/GitDemo/D-Exp01 (feature)
$ git branch
* feature
main
User@DESKTOP-CSHO5HD MINGW64 ~/Desktop/GitDemo/D-Exp01 (feature)
$ git add .
User@DESKTOP-CSHO5HD MINGW64 ~/Desktop/GitDemo/D-Exp01 (feature)
$ git status
On branch feature
Changes to be committed:
(use "git restore --staged <file>..." to unstage)
modified: registration.html
modified: script.js
User@DESKTOP-CSHO5HD MINGW64 ~/Desktop/GitDemo/D-Exp01 (feature)
$ git commit -m "Added Reset button"
[feature a15a5d0] Added Reset button
2 files changed, 4 insertions(+), 4 deletions(-)
User@DESKTOP-CSHO5HD MINGW64 ~/Desktop/GitDemo/D-Exp01 (feature)
$ git status
On branch feature
nothing to commit, working tree clean

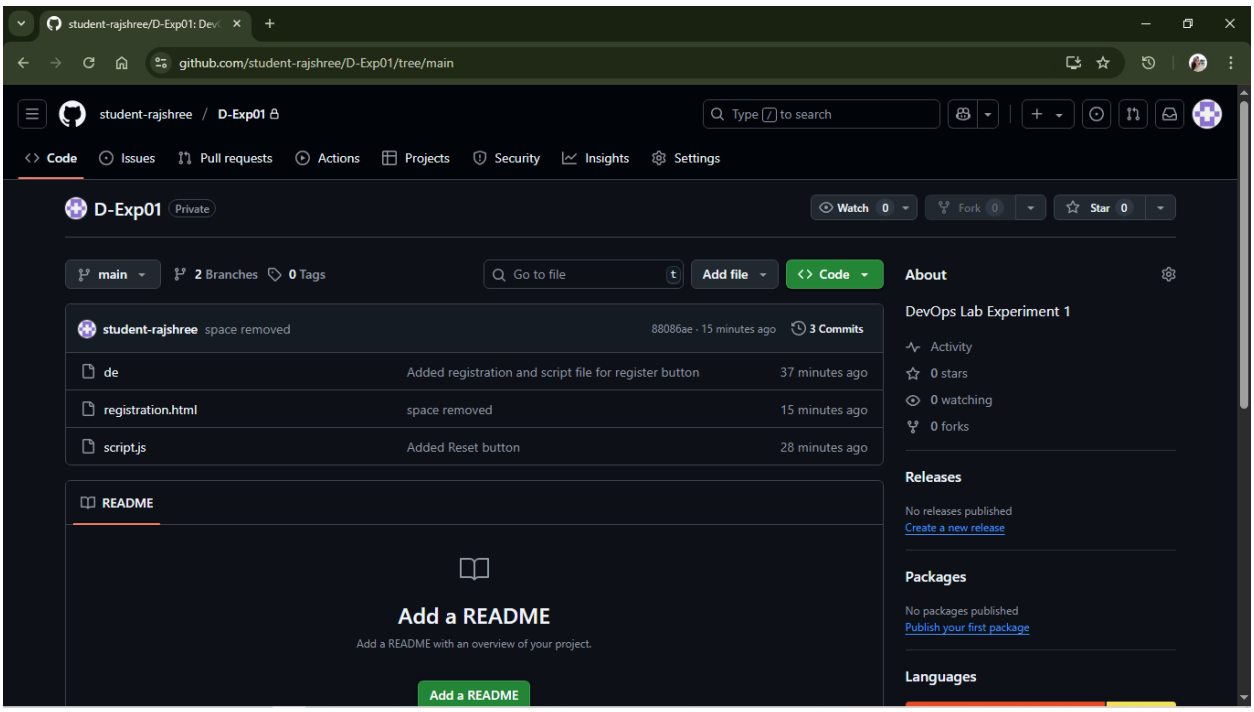

MINGW64:/c/Users/User/Desktop/GitDemo/D-Exp01
User@DESKTOP-CSHO5HD MINGW64 ~/Desktop/GitDemo/D-Exp01 (feature)
$ git push -u origin feature
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 4 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 440 bytes | 62.00 KiB/s, done.
Total 4 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
remote:
remote: Create a pull request for 'feature' on GitHub by visiting:
remote: https://github.com/student-rajshree/D-Exp01/pull/new/feature
remote:
To https://github.com/student-rajshree/D-Exp01.git
* [new branch] feature -> feature
branch 'feature' set up to track 'origin/feature'.
User@DESKTOP-CSHO5HD MINGW64 ~/Desktop/GitDemo/D-Exp01 (feature)
$ git diff main
diff --git a/registration.html b/registration.html
index 09198d0..34cf5e2 100644
--- a/registration.html
+++ b/registration.html
@@ -90,7 +90,7 @@
</select>
<button type="button" onclick="submitForm(); return false;">Registe
r</button>

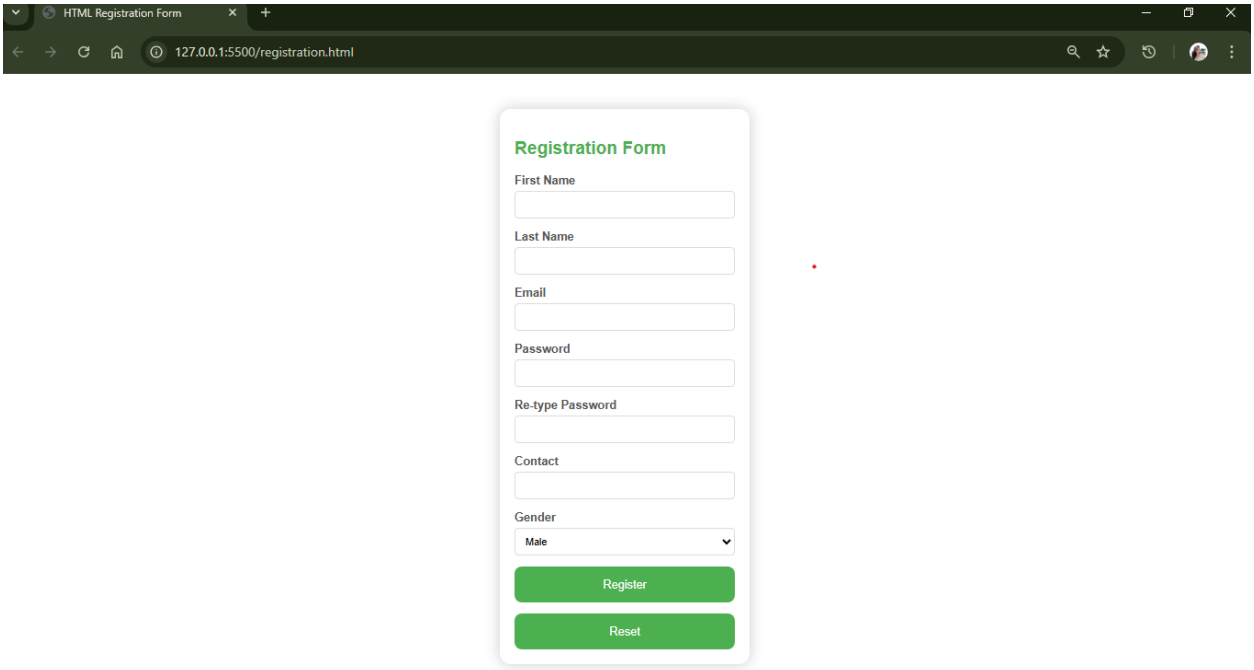
- <!-- <button type="reset" onclick="clickForm()">Reset</button> -->
+ <button type="reset" onclick="clickForm()">Reset</button>
</form>

</body>
diff --git a/script.js b/script.js
index fe35d4d..91d7b4c 100644
--- a/script.js
+++ b/script.js
@@ -21,6 +21,6 @@ function submitForm() {
console.log('Registration Details:', { Name: firstName, Email: email, P
hone: mobile });
document.getElementById("confirmation").style.color = "green";
}


MINGW64:/c/Users/User/Desktop/GitDemo/D-Exp01
--- a/script.js
+++ b/script.js
@@ -21,6 +21,6 @@ function submitForm() {
console.log('Registration Details:', { Name: firstName, Email: email, P
hone: mobile });
document.getElementById("confirmation").style.color = "green";
}
User@DESKTOP-CSHO5HD MINGW64 ~/Desktop/GitDemo/D-Exp01 (feature)
$ git checkout feature
Already on 'feature'
Your branch is up to date with 'origin/feature'.
User@DESKTOP-CSHO5HD MINGW64 ~/Desktop/GitDemo/D-Exp01 (feature)
$ git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
User@DESKTOP-CSHO5HD MINGW64 ~/Desktop/GitDemo/D-Exp01 (main)
$ git merge feature
Updating 4494e7f..a15a5d0
Fast-forward
registration.html | 2 +-
script.js | 6 +++---
2 files changed, 4 insertions(+), 4 deletions(-)
User@DESKTOP-CSHO5HD MINGW64 ~/Desktop/GitDemo/D-Exp01 (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
(use "git push" to publish your local commits)
nothing to commit, working tree clean
User@DESKTOP-CSHO5HD MINGW64 ~/Desktop/GitDemo/D-Exp01 (main)
$ git push -u origin main
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/student-rajshree/D-Exp01.git
4494e7f..a15a5d0 main -> main
branch 'main' set up to track 'origin/main'.
User@DESKTOP-CSHO5HD MINGW64 ~/Desktop/GitDemo/D-Exp01 (main)
$

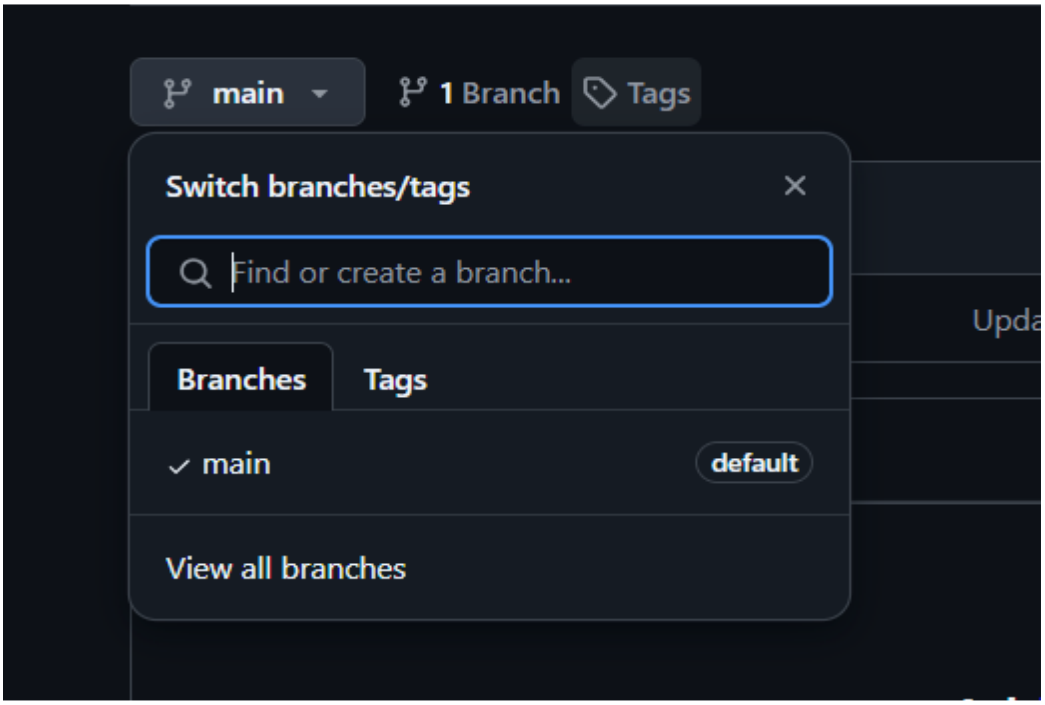

student-rajshree/D-Exp01 at fe
github.com/student-rajshree/D-Exp01/tree/feature
student-rajshree / D-Exp01
Type / to search
Code
Issues
Pull requests
Actions
Projects
Security
Insights
Settings
D-Exp01
Private
Watch 0
Fork 0
Star 0
feature
2 Branches
0 Tags
Go to file
t
Add file
Code
This branch is up to date with main.
Contribute
student-rajshree space removed
88086ae · 14 minutes ago
3 Commits
de
Added registration and script file for register button
36 minutes ago
registration.html
space removed
14 minutes ago
script.js
Added Reset button
27 minutes ago
README
Add a README
About
DevOps Lab Experiment 1
Activity
0 stars
0 watching
0 forks
Releases
No releases published
Create a new release
Packages
No packages published
Publish your first package
Languages


student-rajshree/D-Exp01: Dev
github.com/student-rajshree/D-Exp01/tree/main
student-rajshree / D-Exp01
Type / to search
Code
Issues
Pull requests
Actions
Projects
Security
Insights
Settings
D-Exp01
Private
Watch 0
Fork 0
Star 0
main
2 Branches
0 Tags
Go to file
t
Add file
Code
student-rajshree space removed
88086ae · 15 minutes ago
3 Commits
de
Added registration and script file for register button
37 minutes ago
registration.html
space removed
15 minutes ago
script.js
Added Reset button
28 minutes ago
README
Add a README
Add a README with an overview of your project.
Add a README
About
DevOps Lab Experiment 1
Activity
0 stars
0 watching
0 forks
Releases
No releases published
Create a new release
Packages
No packages published
Publish your first package
Languages

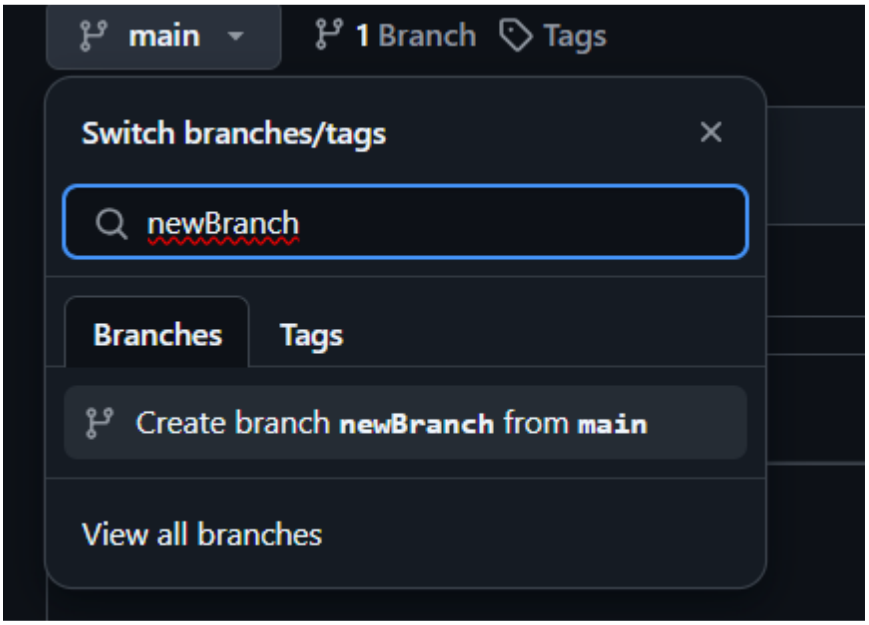


PERFORMING BRANCHING DIRECTLY THROUGH GIT HUB

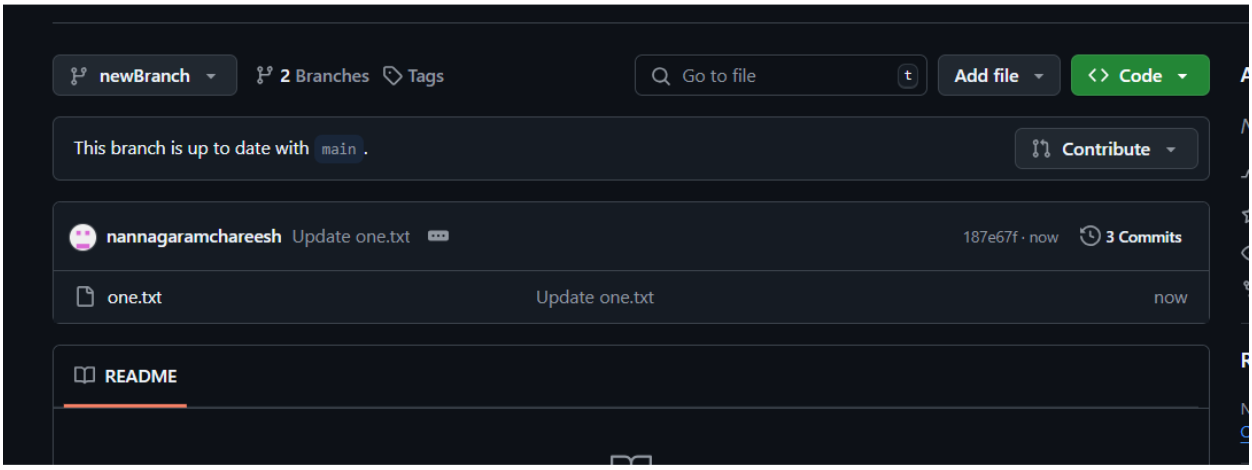
1. First go to your repository where you want to create a new branch
2. Now on the top left you will find the current branch, select it



3. Now write the branch name that you want to create in the search bar. If it does not exist then you will see an option saying Create branch newBranch from main.

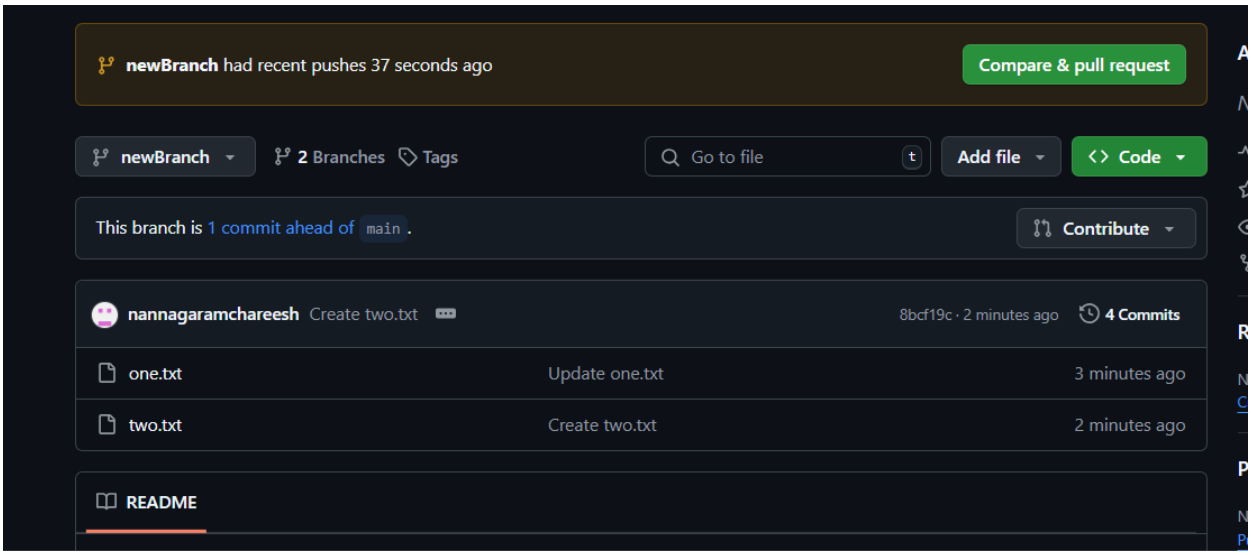


4. After clicking on Create branch, the newBranch will be created and you will automatically get switched to the new branch and all the files in the main branch will be available as a copy in this newBranch

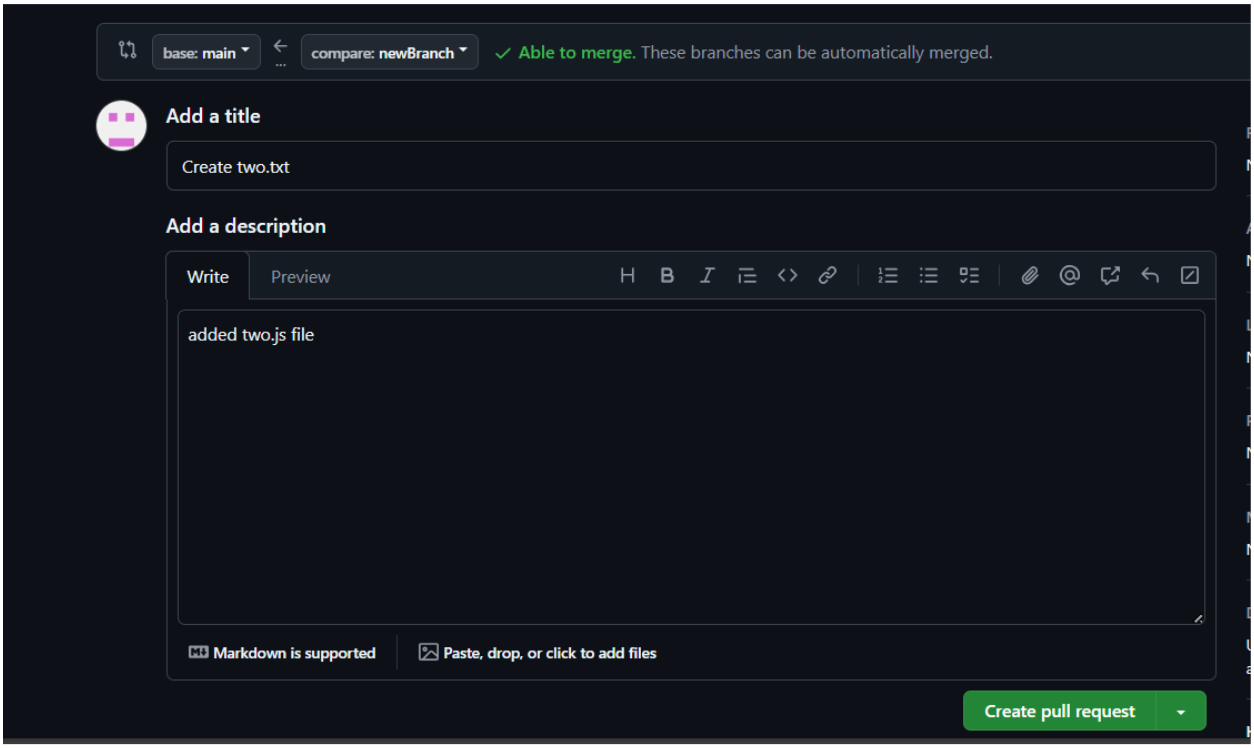


5. Create a new file in this branch write something in it and commit changes

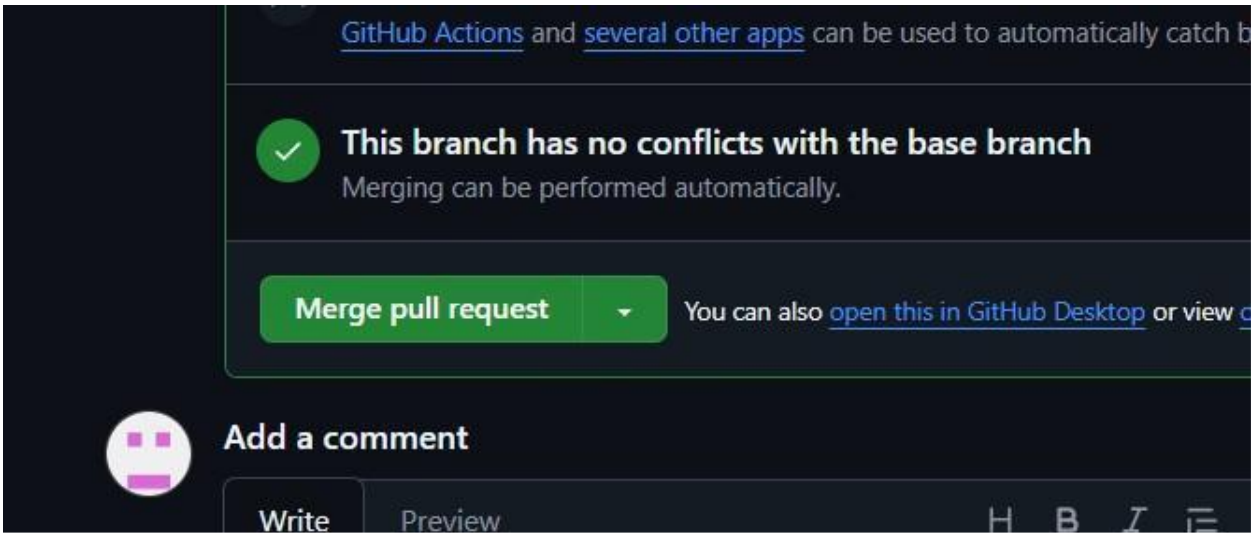
6. Now to merge the changes in this newBranch to the main branch, click on compare & pull request



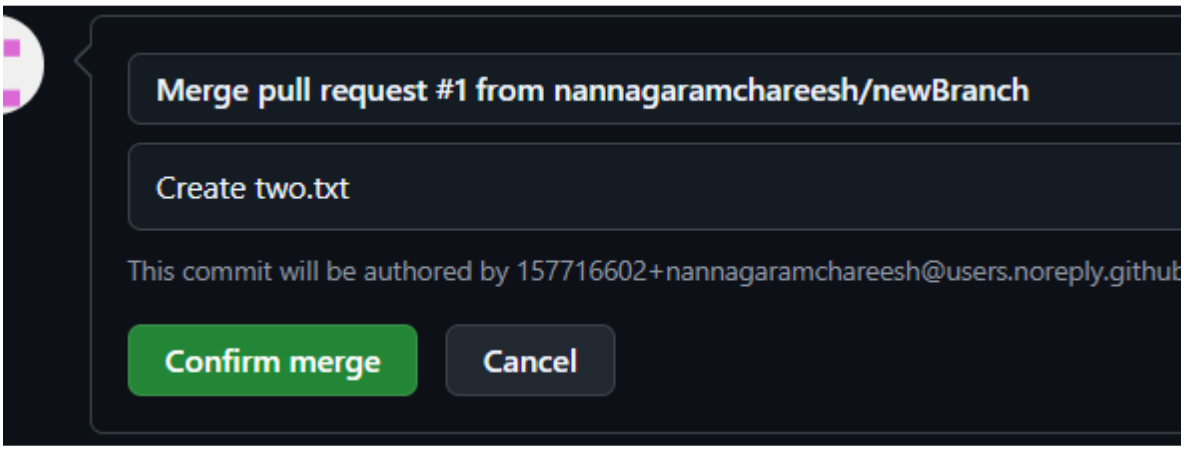
7. Now add a commit message and click on create pull request



8. Then click on Merge pull request in order to reflect the changes made in newBranch to main branch



9. Then click on confirm merge



10. After this if you go to main branch you will see that the changes made in newBranch will be reflected in the main branch

So this is how branching can be done in git hub.

CLONING A REPOSITORY

Inorder to clone a repository use the git clone command.

1. Go ahead and copy the repository of the project that you want to clone
2. Now create a new folder and open it in vs code
3. Now open terminal in vs code and execute the command: git clone "repository_url"

FORKING

Forking in GitHub is the process of cloning a repository of someone else's to your own GitHub account where you can modify it freely.

Steps to Fork and then make changes in your local repo-

1. Go to the repository: Visit the GitHub page of the repository you want to fork.

2. **Click "Fork"**: On the top-right corner of the repository page, you will see a "Fork" button. Click on it.