SECURE SOFTWARE DESIGN AND DEVELOPMENT – ASSIGNMENT # 1

SUBMITTED BY: FARWAH HAMID (FA22-BCT-007)

SUBMITTED TO: SIR MUHAMMAD AHMAD NAWAZ

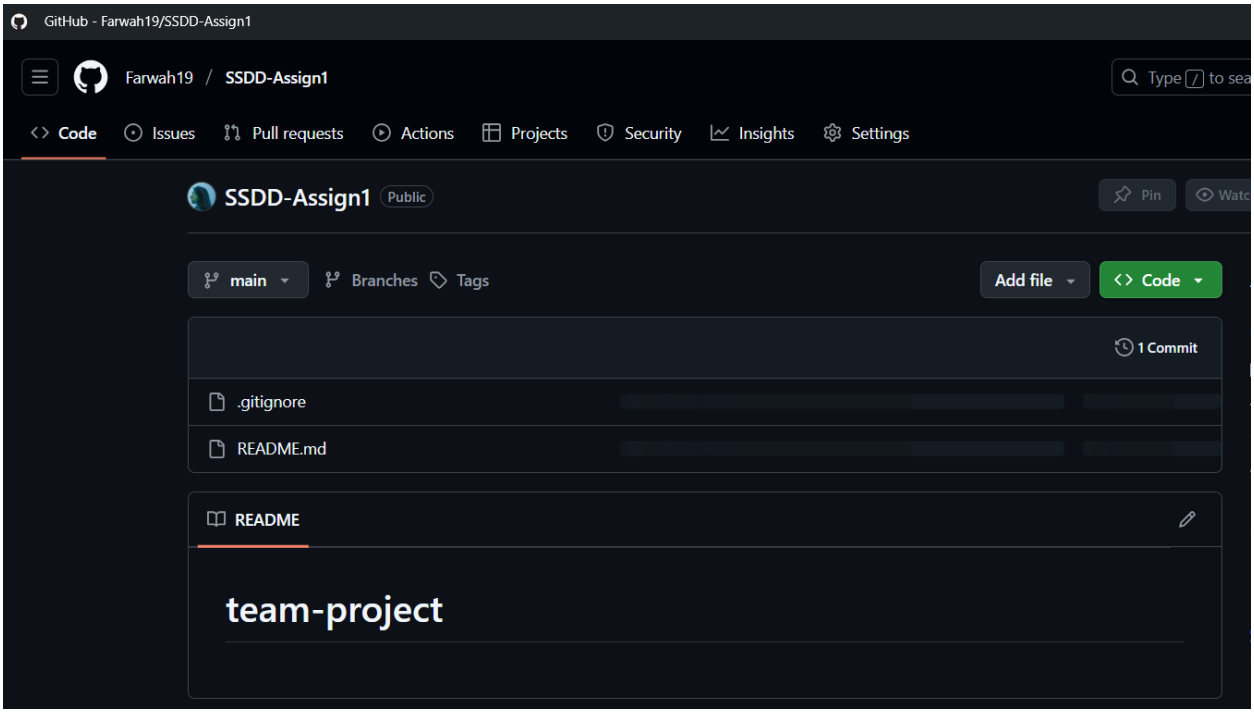
26TH SEPTEMBER 2025

Table of Contents

Step 1 – Setting up the repository.

1. Create a repo on GitHub.

Created a GitHub repository named **“SSDD-Assign1”.**



2. Clone the repository locally.

Now I cloned the repository and moved to the directory in which it was cloned.

```
Lenovo@Farwah MINGW64 ~
$ git clone https://github.com/Farwah19/SSDD-Assign1.git
Cloning into 'SSDD-Assign1'...
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (4/4), done.

Lenovo@Farwah MINGW64 ~
$ cd SSDD-Assign1/
```

3. Make your first commit.

Created a new file **“first.txt”,** added some text to it, and then staged and committed the changes. Then pushed it to GitHub.

```
Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (main)
$ echo "Hello This is Farwah Hamid, doing Assignment 1" >> first.txt

Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (main)
$ git add first.txt
warning: in the working copy of 'first.txt', LF will be replaced by CRLF the nex
t time Git touches it

Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (main)
$ git add .
warning: in the working copy of 'first.py', LF will be replaced by CRLF the next
 time Git touches it

Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (main)
$ git commit -m "first file commit"
[main d17c8e0] first file commit
 2 files changed, 2 insertions(+)
 create mode 100644 first.py
 create mode 100644 first.txt

Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (main)
$ git push origin main
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 365 bytes | 121.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Farwah19/SSDD-Assign1.git
   d8b2420..d17c8e0  main -> main
```

Step 2 – Collaborate using branches.

1. Create a new branch.

Created a branch for a new feature.

Created a new file named **“newFile.txt”**.

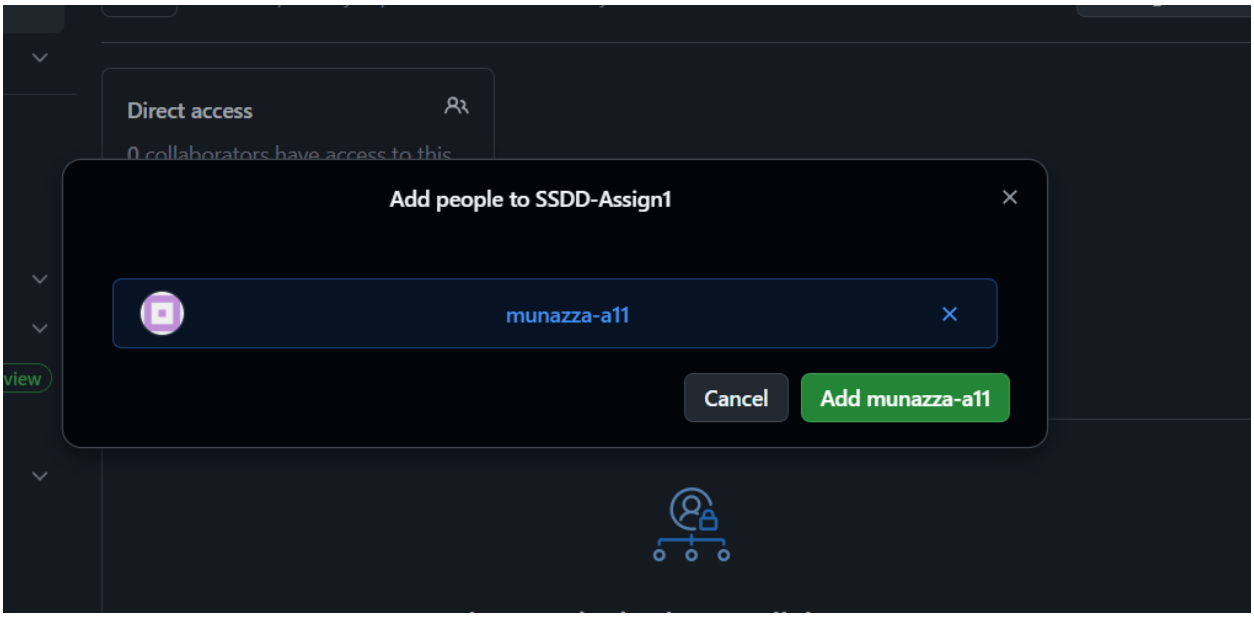
```
Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (main)
$ git checkout -b feature/additional-feature
Switched to a new branch 'feature/additional-feature'

Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (feature/additional-feature)
$ echo "This is a new file being added to check if branching is working correctl
y" >> newFile.txt
```

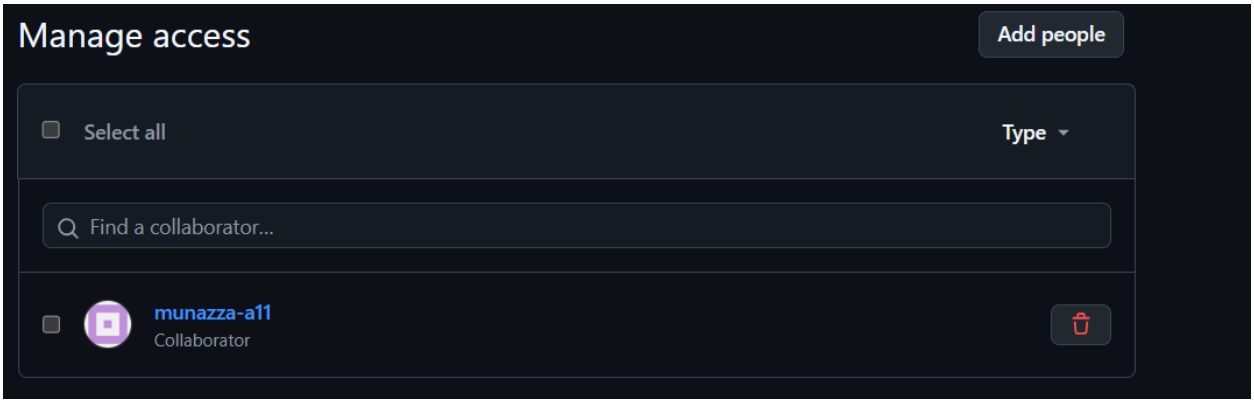
2. Commit and push the branch.

```
Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (feature/additional-feature)
$ git add .
warning: in the working copy of 'newFile.txt', LF will be replaced by CRLF the n
ext time Git touches it

Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (feature/additional-feature)
$ git commit -m "Added new feature"
[feature/additional-feature 9d9e5ac] Added new feature
 1 file changed, 1 insertion(+)
 create mode 100644 newFile.txt

Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (feature/additional-feature)
$ git push origin feature/additional-feature
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 336 bytes | 336.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
remote:
remote: Create a pull request for 'feature/additional-feature' on GitHub by visi
ting:
remote:      https://github.com/Farwah19/SSDD-Assign1/pull/new/feature/additiona
l-feature
remote:
To https://github.com/Farwah19/SSDD-Assign1.git
 * [new branch]      feature/additional-feature -> feature/additional-feature
```

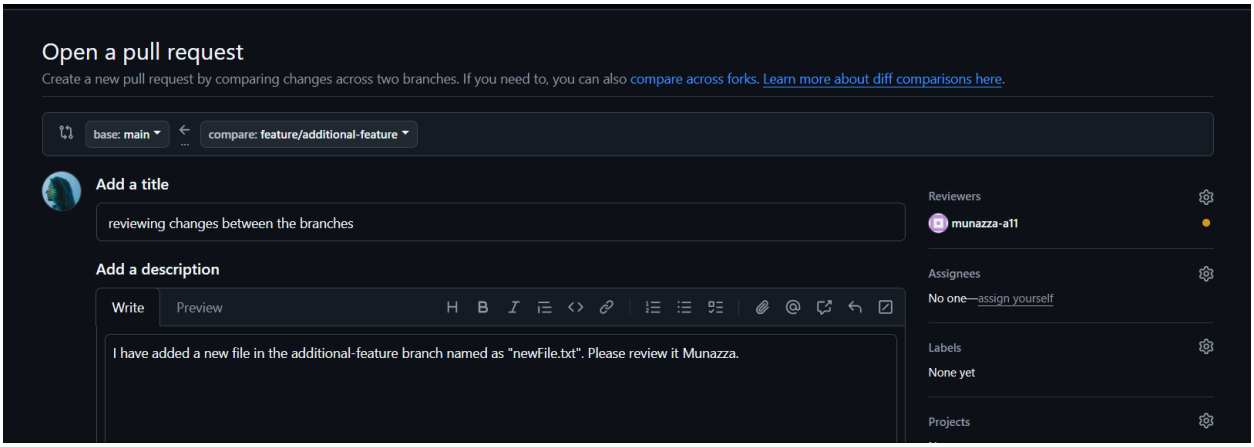
3. Create a pull request.

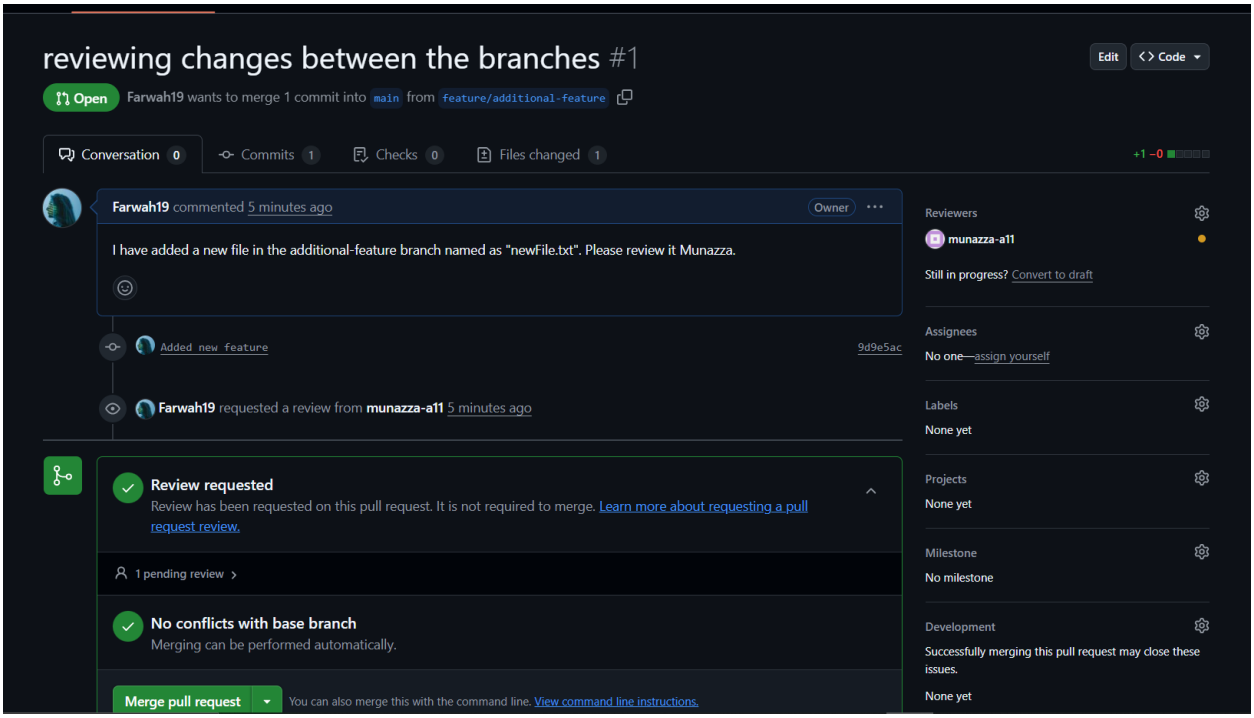
First, I invited my teammate as a collaborator on my repository.

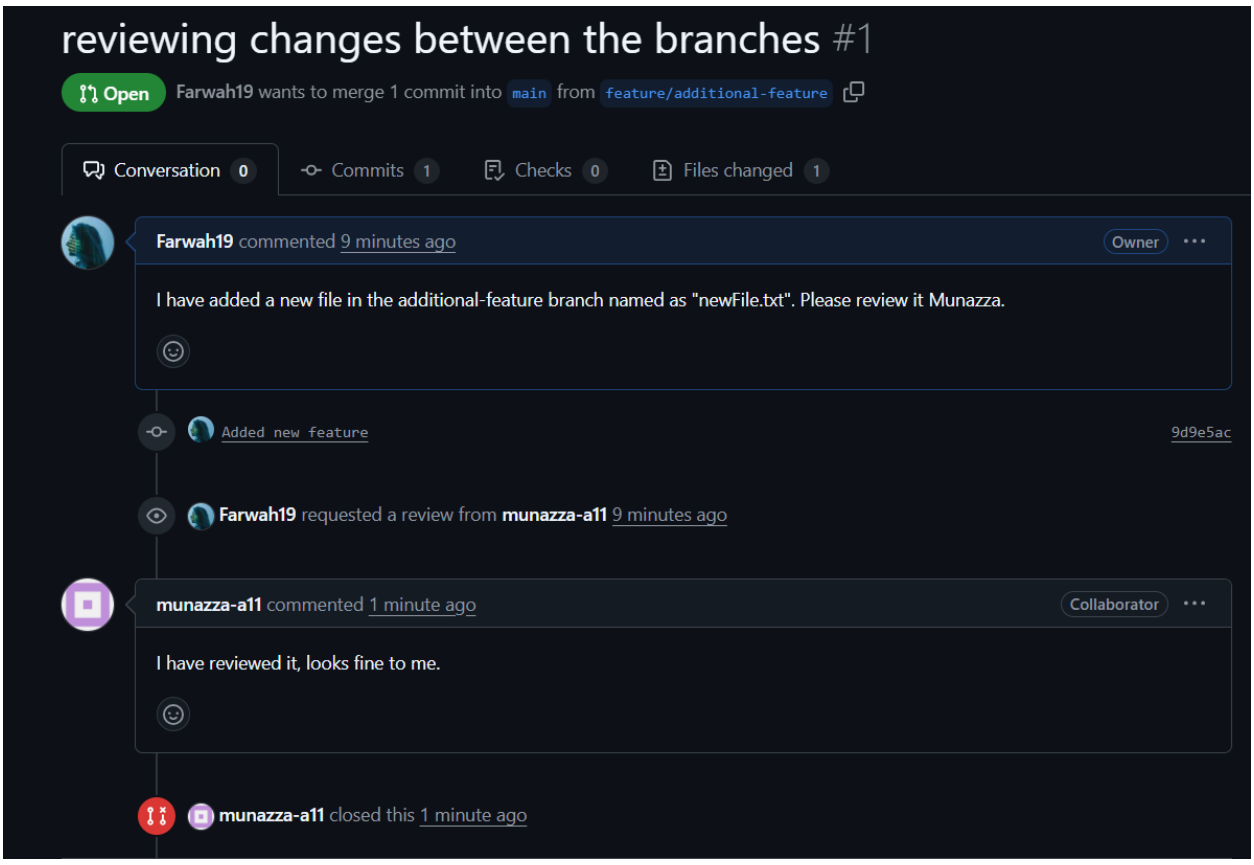




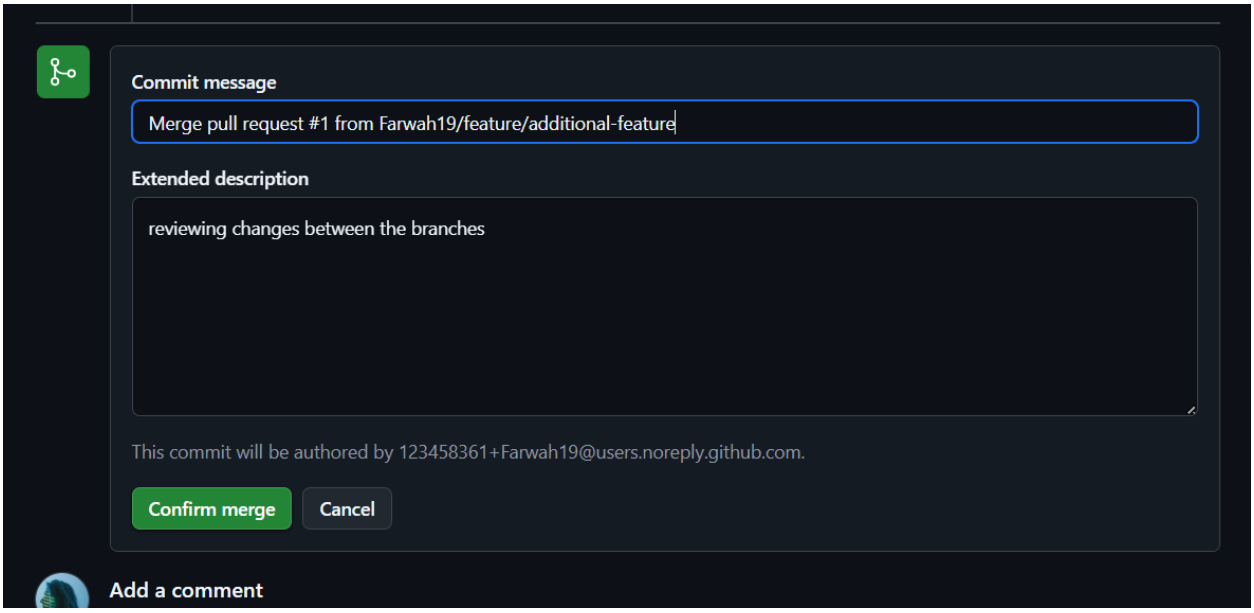
Created a pull request and added her for reviewing it.

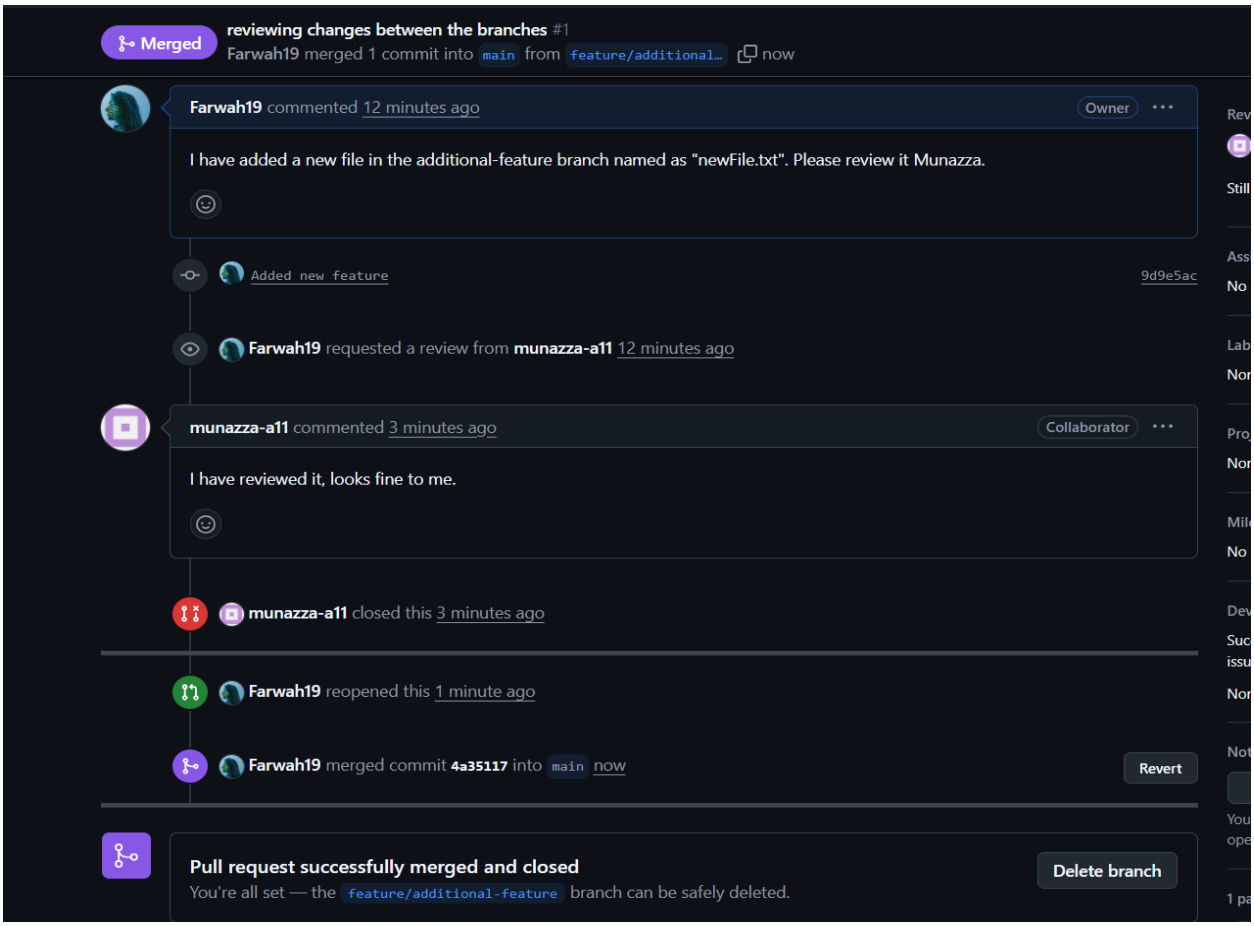



reviewing changes between the branches #1
Edit
Code
Open
Farwah19 wants to merge 1 commit into main from feature/additional-feature
Conversation 0
Commits 1
Checks 0
Files changed 1
+1 −0
Farwah19 commented 5 minutes ago
Owner
I have added a new file in the additional-feature branch named as "newFile.txt". Please review it Munazza.
Added new feature
9d9e5ac
Farwah19 requested a review from munazza-a11 5 minutes ago
Review requested
Review has been requested on this pull request. It is not required to merge. Learn more about requesting a pull request review.
1 pending review
No conflicts with base branch
Merging can be performed automatically.
Merge pull request
You can also merge this with the command line. View command line instructions.
Reviewers
munazza-a11
Still in progress? Convert to draft
Assignees
No one—assign yourself
Labels
None yet
Projects
None yet
Milestone
No milestone
Development
Successfully merging this pull request may close these issues.
None yet


reviewing changes between the branches #1
Open
Farwah19 wants to merge 1 commit into main from feature/additional-feature
Conversation 0
Commits 1
Checks 0
Files changed 1
Farwah19 commented 9 minutes ago
Owner
I have added a new file in the additional-feature branch named as "newFile.txt". Please review it Munazza.
Added new feature
9d9e5ac
Farwah19 requested a review from munazza-a11 9 minutes ago
munazza-a11 commented 1 minute ago
Collaborator
I have reviewed it, looks fine to me.
munazza-a11 closed this 1 minute ago

Now after the review, merged the request.

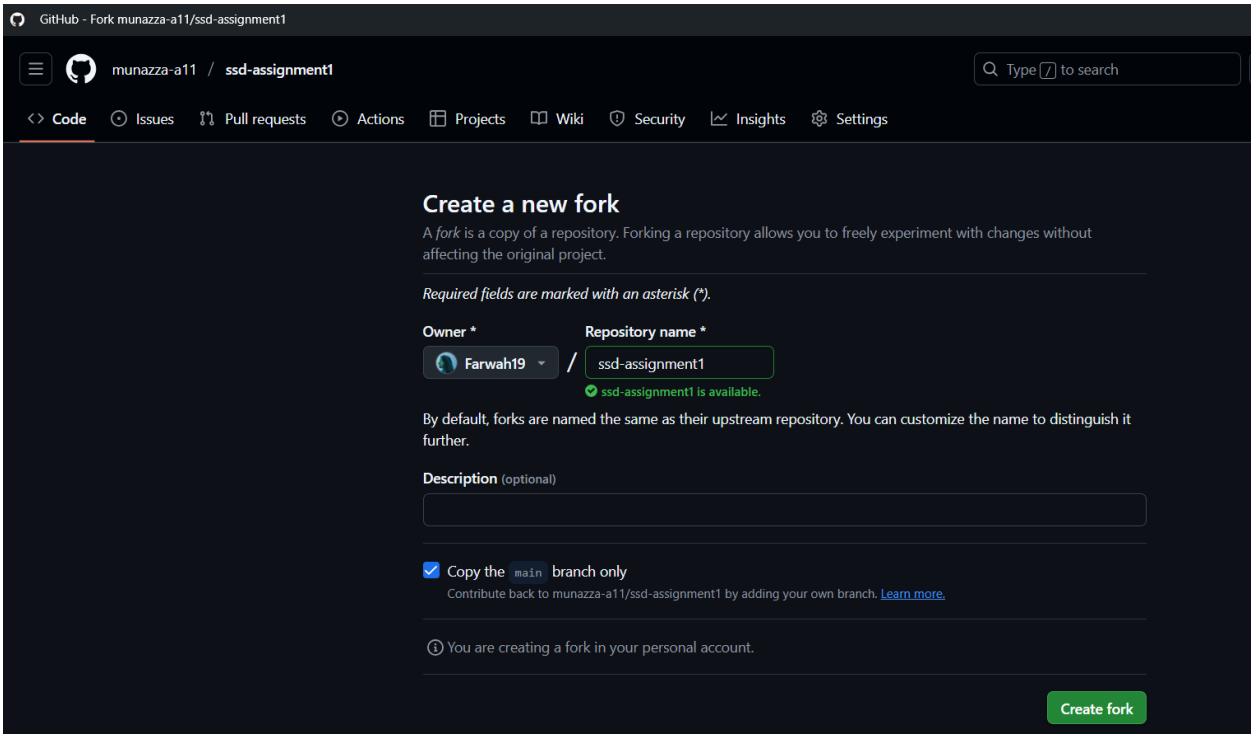


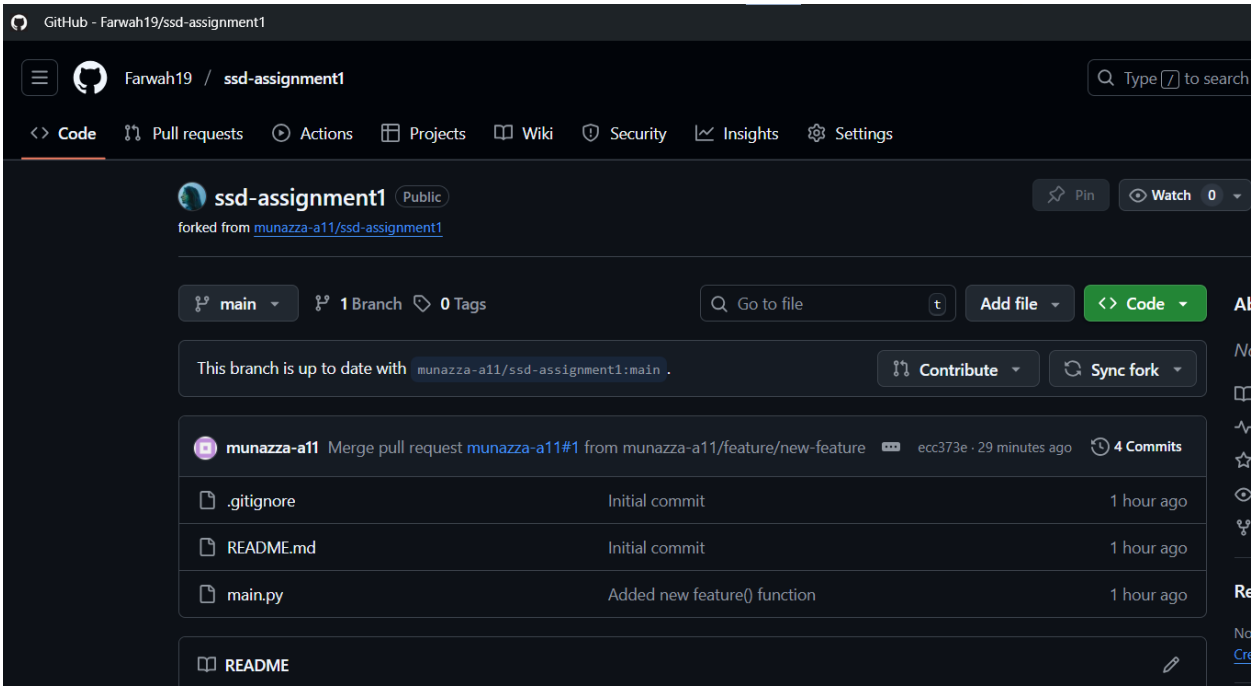


Step 3 – Fork a repository.

1. Fork a repository.

Moving to my teammate's GitHub and forking her public repository.





2. Clone the forked repository.

Now cloning the forked repo and changing the contents of a file.

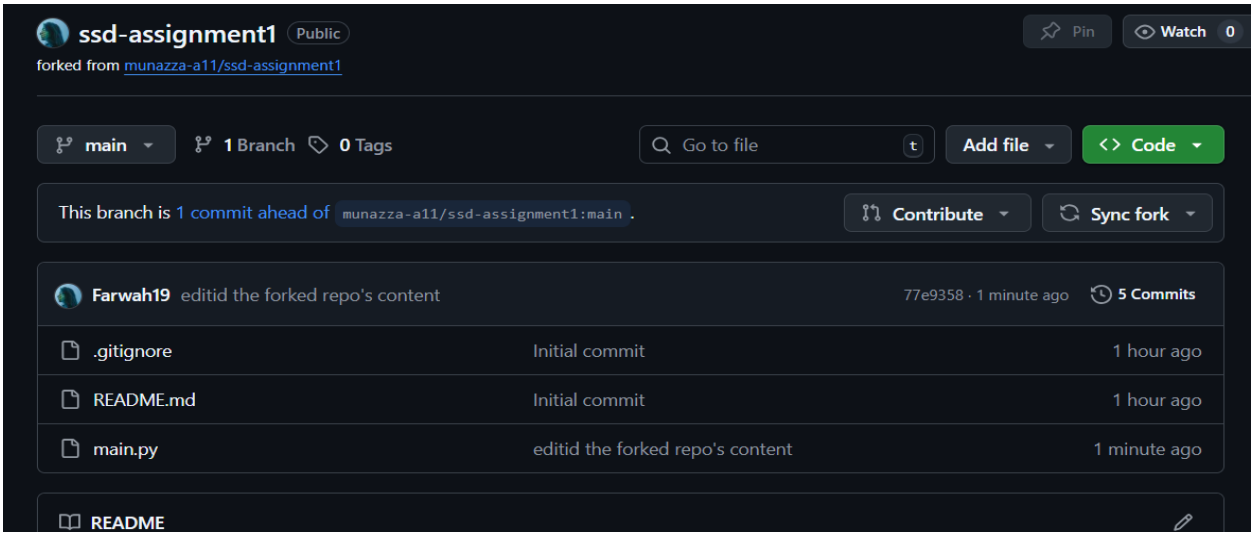
```
Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (feature/additional-feature)
$ git clone https://github.com/Farwah19/ssd-assignment1.git
Cloning into 'ssd-assignment1'...
remote: Enumerating objects: 11, done.
remote: Counting objects: 100% (11/11), done.
remote: Compressing objects: 100% (8/8), done.
remote: Total 11 (delta 3), reused 6 (delta 2), pack-reused 0 (from 0)
Receiving objects: 100% (11/11), 4.26 KiB | 4.26 MiB/s, done.
Resolving deltas: 100% (3/3), done.

Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (feature/additional-feature)
$ cd ssd-assignment1/

Lenovo@Farwah MINGW64 ~/SSDD-Assign1/ssd-assignment1 (main)
$ ls
README.md  main.py
```

```
Lenovo@Farwah MINGW64 ~/SSDD-Assign1/ssd-assignment1 (main)
$ cat main.py
# main.py - initial file for team project
def hello():
    print("Hello, team project!")

if __name__ == "__main__":
    hello()

def feature():
    print("New feature is working!")

Lenovo@Farwah MINGW64 ~/SSDD-Assign1/ssd-assignment1 (main)
$ echo "I am editing this file" >> main.py

Lenovo@Farwah MINGW64 ~/SSDD-Assign1/ssd-assignment1 (main)
$ cat main.py
# main.py - initial file for team project
def hello():
    print("Hello, team project!")

if __name__ == "__main__":
    hello()

def feature():
    print("New feature is working!")
I am editing this file
```

3. Make changes and push.

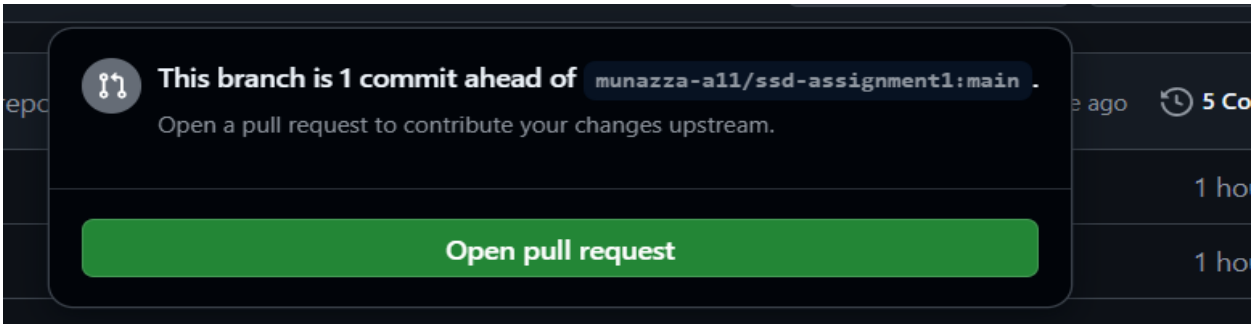
Committing the changes and pushing it.

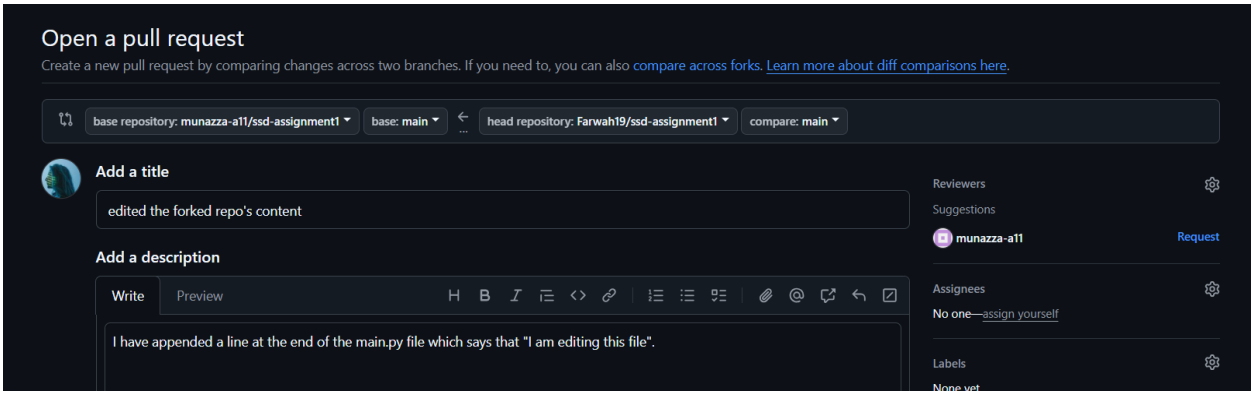
```
Lenovo@Farwah MINGW64 ~/SSDD-Assign1/ssd-assignment1 (main)
$ git add .
warning: in the working copy of 'main.py', LF will be replaced by CRLF the next time Git touches it

Lenovo@Farwah MINGW64 ~/SSDD-Assign1/ssd-assignment1 (main)
$ git commit -m "editid the forked repo's content"
[main 77e9358] editid the forked repo's content
 1 file changed, 1 insertion(+)

Lenovo@Farwah MINGW64 ~/SSDD-Assign1/ssd-assignment1 (main)
$ git push origin main
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 318 bytes | 318.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/Farwah19/ssd-assignment1.git
   ecc373e..77e9358  main -> main
```

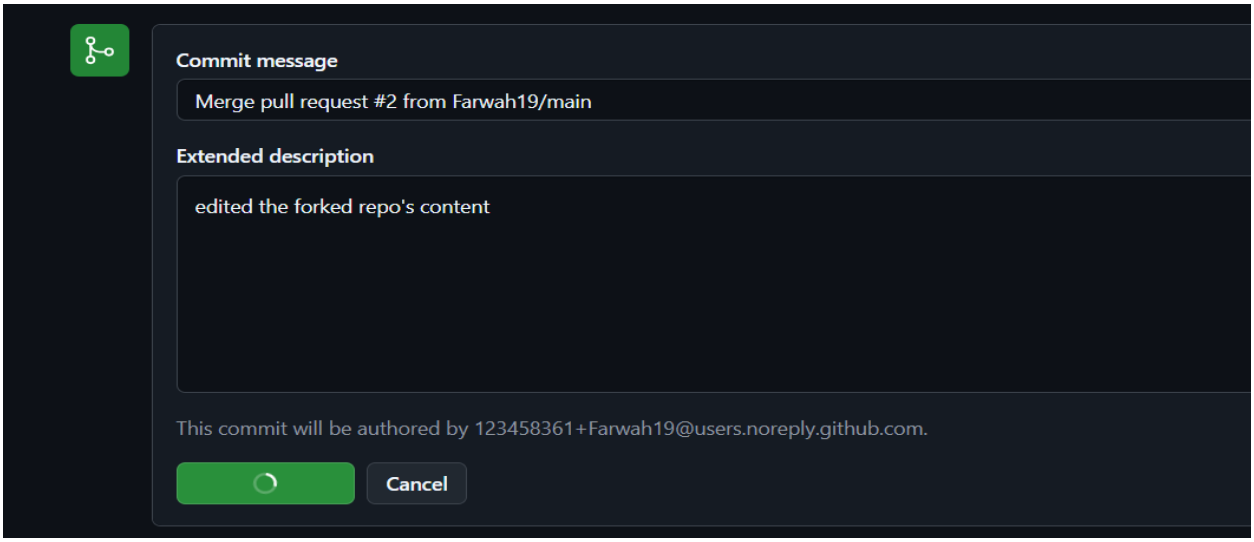


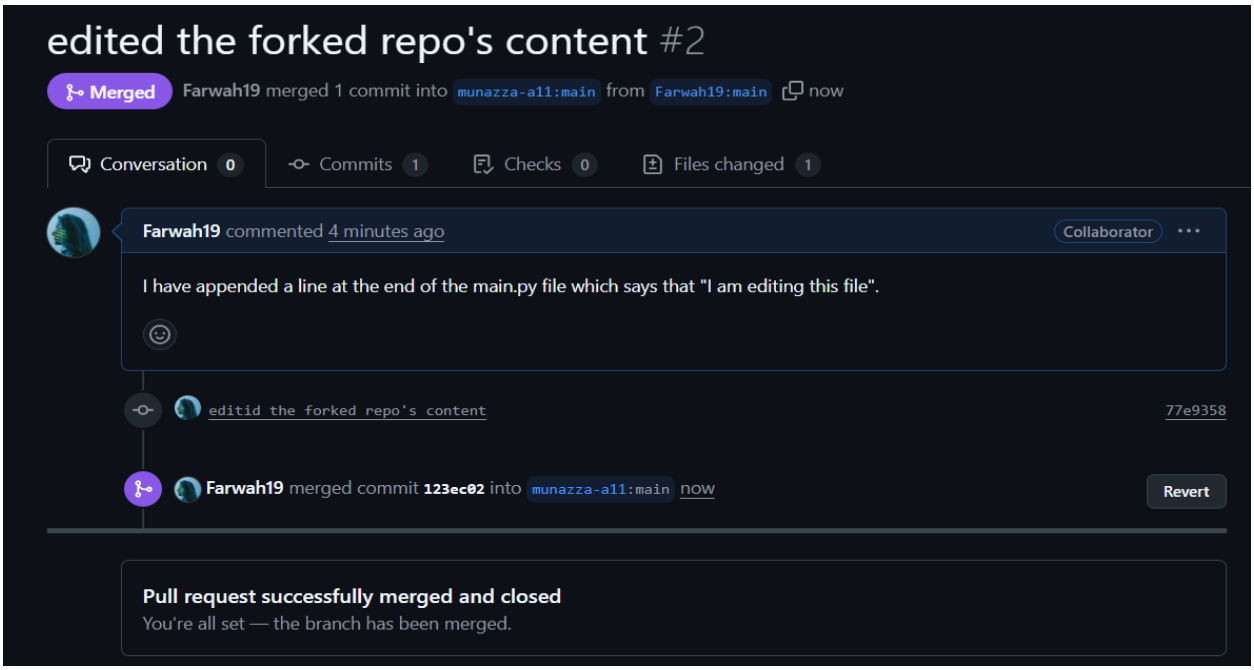
4. Create a PR to the original repository.

Now moving to contribute and creating a pull request to submit changes to the original repo.




Open a pull request
Create a new pull request by comparing changes across two branches. If you need to, you can also compare across forks. Learn more about diff comparisons here.
base repository: munazza-a11/ssd-assignment1
base: main
head repository: Farwah19/ssd-assignment1
compare: main
Add a title
edited the forked repo's content
Add a description
Write
Preview
I have appended a line at the end of the main.py file which says that "I am editing this file".
Reviewers
Suggestions
munazza-a11
Request
Assignees
No one—assign yourself
Labels
None yet


Commit message
Merge pull request #2 from Farwah19/main
Extended description
edited the forked repo's content
This commit will be authored by 123458361+Farwah19@users.noreply.github.com.
Cancel


edited the forked repo's content #2
Merged
Farwah19 merged 1 commit into munazza-a11:main from Farwah19:main now
Conversation 0
Commits 1
Checks 0
Files changed 1
Farwah19 commented 4 minutes ago
Collaborator
I have appended a line at the end of the main.py file which says that "I am editing this file".
editid the forked repo's content
77e9358
Farwah19 merged commit 123ec02 into munazza-a11:main now
Revert
Pull request successfully merged and closed
You're all set — the branch has been merged.

Step 4 – Revert and reset.

1. Revert a commit.

First, I made a new file and committed the changes to it.

```
Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (feature/additional-feature)
$ echo "line one" >> demo.txt

Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (feature/additional-feature)
$ git add demo.txt
warning: in the working copy of 'demo.txt', LF will be replaced by CRLF the next
 time Git touches it

Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (feature/additional-feature)
$ git commit -m "first commit"
[feature/additional-feature a17d366] first commit
 1 file changed, 1 insertion(+)
 create mode 100644 demo.txt

Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (feature/additional-feature)
$ echo "line two" >> demo.txt

Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (feature/additional-feature)
$ git add demo.txt
warning: in the working copy of 'demo.txt', LF will be replaced by CRLF the next
 time Git touches it

Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (feature/additional-feature)
$ git commit -m "second commit"
[feature/additional-feature b758656] second commit
 1 file changed, 1 insertion(+)
```

Next, I reverted the most recent commit.

```
Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (feature/additional-feature)
$ git revert 5f884c3
[feature/additional-feature d09b330] Revert "Added practice line"
 1 file changed, 1 deletion(-)
 delete mode 100644 demo.txt~

Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (feature/additional-feature)
$ git log --oneline
d09b330 (HEAD -> feature/additional-feature) Revert "Added practice line"
5f884c3 Added practice line
d4f48e8 New commit
b758656 second commit
a17d366 first commit
bd174ed Reapply "new changes"
a5ba4be Revert "new changes"
6b90fec new changes
0d61772 new changes again
928db7b new change again
fbfeb73 New changes
8789321 New changes
9d9e5ac (origin/feature/additional-feature) Added new feature
d17c8e0 (origin/main, origin/HEAD, main) first file commit
d8b2420 Initial commit
```

2. Reset a commit.

Creating a new file and making some changes to it.

```
Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (feature/additional-feature)
$ echo "Hello, this is the file to be deleted" >> del.txt

Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (feature/additional-feature)
$ git add .
warning: in the working copy of 'del.txt', LF will be replaced by CRLF the next
time Git touches it

Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (feature/additional-feature)
$ git commit -m "Created test file"
[feature/additional-feature 9dff553] Created test file
 1 file changed, 1 insertion(+)
 create mode 100644 del.txt
```

```
Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (feature/additional-feature)
$ git log --oneline
9dff553 (HEAD -> feature/additional-feature) Created test file
d09b330 Revert "Added practice line"
5f884c3 Added practice line
d4f48e8 New commit
b758656 second commit
a17d366 first commit
bd174ed Reapply "new changes"
a5ba4be Revert "new changes"
6b90fec new changes
0d61772 new changes again
928db7b new change again
fbfeb73 New changes
8789321 New changes
9d9e5ac (origin/feature/additional-feature) Added new feature
d17c8e0 (origin/main, origin/HEAD, main) first file commit
d8b2420 Initial commit

Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (feature/additional-feature)
$ git reset --hard 5f884c3
HEAD is now at 5f884c3 Added practice line

Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (feature/additional-feature)
$ git log --oneline
5f884c3 (HEAD -> feature/additional-feature) Added practice line
d4f48e8 New commit
b758656 second commit
a17d366 first commit
bd174ed Reapply "new changes"
a5ba4be Revert "new changes"
6b90fec new changes
0d61772 new changes again
928db7b new change again
fbfeb73 New changes
8789321 New changes
9d9e5ac (origin/feature/additional-feature) Added new feature
d17c8e0 (origin/main, origin/HEAD, main) first file commit
d8b2420 Initial commit
```

Step 5 – Collaboration Workflow.

1. Create merge conflicts.

i. Firstly, I created a new branch, named as **“conflict-branch”.**

```
Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (main)
$ git checkout -b conflict-branch
Switched to a new branch 'conflict-branch'
```

ii. After moving to that branch, I edited the file **“demo.txt”** and committed the changes.

```
Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (conflict-branch)
$ echo "line changes in conflict-branch" >> demo.txt

Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (conflict-branch)
$ git add demo.txt
warning: in the working copy of 'demo.txt', LF will be replaced by CRLF the next
 time Git touches it

Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (conflict-branch)
$ git commit -m "changed line in conflit-branch"
[conflict-branch 91f2da9] changed line in conflit-branch
 1 file changed, 1 insertion(+)
 create mode 100644 demo.txt
```

iii. Now I moved back to main and edited **“demo.txt”.** Then I also committed those changes.

```
Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (conflict-branch)
$ git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.

Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (main)
$ echo "line changed in main" >> demo.txt

Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (main)
$ git add demo.txt
warning: in the working copy of 'demo.txt', LF will be replaced by CRLF the next
 time Git touches it

Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (main)
$ git commit -m "changed line in main"
[main 8a7ef6a] changed line in main
 1 file changed, 1 insertion(+)
 create mode 100644 demo.txt
```

iv. Now I tried to merge the branch and git raised a conflict.

```
Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (main)
$ git merge conflict-branch
Auto-merging demo.txt
CONFLICT (add/add): Merge conflict in demo.txt
Automatic merge failed; fix conflicts and then commit the result.
```

2. Resolve merge conflicts.

i. Now to resolve the conflict, I edited the file manually.

```
Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (main|MERGING)
$ echo "resolving the conflict" >> demo.txt
```

ii. Committed the changes.

```
Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (main|MERGING)
$ git add demo.txt
warning: in the working copy of 'demo.txt', LF will be replaced by CRLF the next
 time Git touches it

Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (main|MERGING)
$ git commit -m "resolved the conflict"
[main fe2bc8d] resolved the conflict
```

iii. Merged the branch.

```
Lenovo@Farwah MINGW64 ~/SSDD-Assign1 (main)
$ git merge conflict-branch
Already up to date.
```