



Experiment -2.4

Student Name: Chayan Gope <u>UID</u>: 22BDO10036

Branch: AIT-CSE(DevOps) Section/Group: 22BCD-1/A

Semester: 4th Date of Performance: 28/01/2024

Subject Name: Git and Hub Subject Code: 22CSH-293

1. Aim/Overview of the practical: Git merge conflicts and resolving Git merge conflicts

2. Software Used: Git Bash, GitHub.

3. Steps for experiment/practical:

❖ Clone a repo from the remote to the local system and move inside it.

```
chaya@Chayan MINGW64 /
$ git clone https://github.com/Chayan-12/new-github.git
Cloning into 'new-github'...
remote: Enumerating objects: 70, done.
remote: Counting objects: 100% (70/70), done.
remote: Compressing objects: 100% (48/48), done.
remote: Total 70 (delta 28), reused 38 (delta 13), pack-reused 0
Receiving objects: 100% (70/70), 26.94 KiB | 26.94 MiB/s, done.
Resolving deltas: 100% (28/28), done.

chaya@Chayan MINGW64 /
$ cd new-github
```

❖ Create a file in the local system, add it to the staging area and commit the changes.

```
chaya@Chayan MINGW64 /new-github (main)
$ vi file.c

chaya@Chayan MINGW64 /new-github (main)
$ git add file.c

chaya@Chayan MINGW64 /new-github (main)
$ git commit -m "Committed"
[main b9d9174] Committed
1 file changed, 24 insertions(+)
create mode 100644 file.c
```







❖ Create a new branch named **branch1**, checkout to it, made some changes in the file, add it to the staging area and commit the changes.

```
chaya@Chayan MINGW64 /new-github (main)
$ git checkout -b branch1
Switched to a new branch 'branch1'
chaya@Chayan MINGW64 /new-github (branch1)
$ vi file.c

chaya@Chayan MINGW64 /new-github (branch1)
$ git add file.c

chaya@Chayan MINGW64 /new-github (branch1)
$ git commit -m "Edited and Committed"
[branch1 613d709] Edited and Committed
1 file changed, 1 insertion(+)
```

Checkout to the main branch and merge the changes using the git merge command and
 - no - ff operator.

```
chaya@Chayan MINGW64 /new-github (branch1)
$ git checkout main
Switched to branch 'main'
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)
chaya@Chayan MINGW64 /new-github (main)
$ git merge --no-ff branch1
Merge made by the 'ort' strategy.
file.c | 1 +
1 file changed, 1 insertion(+)
chaya@Chayan MINGW64 /new-github (main)
$ cat file.c
#include <stdio.h>
// Function to calculate the factorial of a number
unsigned long long factorial(int n) {
    if (n == 0)
        return 1;
```







❖ Push the changes to the remote repository.

```
chaya@Chayan MINGW64 /new-github (main)
$ git push origin main
Enumerating objects: 8, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 12 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (7/7), 885 bytes | 885.00 KiB/s, done.
Total 7 (delta 4), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (4/4), completed with 1 local object.
To https://github.com/Chayan-12/new-github.git
    elba5a3..3867b30 main -> main

chaya@Chayan MINGW64 /new-github (main)
$ git push origin branch1
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/Chayan-12/new-github.git
    45483cb..613d709 branch1 -> branch1
```

Now, move to the **branch1** branch on remote repo, make some changes and commit them.

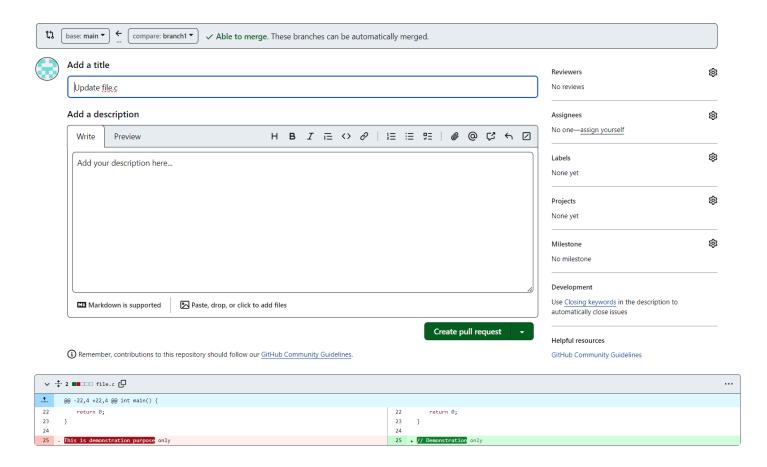


❖ Move to the **main** branch, click on **compare & pull request**, and create a pull request.

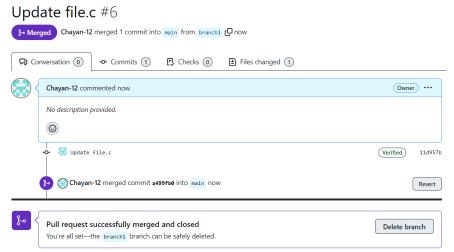








Now, merge the pull request and confirm merge.

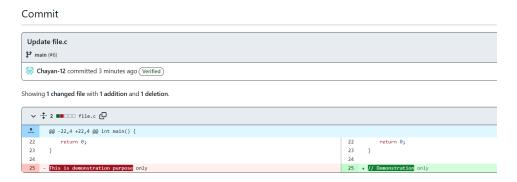








❖ You can see the changes in the **main** branch.



4. Result/Output/Writing Summary:

In this experiment, we have merge the contents of a branch to the main branch using pull request on both git bash and github.

Learning outcomes (What I have learnt):

- 1. Learnt how to create a branch.
- **2.** Learnt how to clone a remote repo to our local system.
- 3. Learnt how to create a pull request and handle their merging.
- **4.** Learnt to merge two branches.

Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty):

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.			
2.			
3.			

