# Experiment -1.3

**Student Name:** Vinay
**UID:** 22BDO10040
**Branch:** AIT-CSE(DevOps)
**Section/Group**: 22BCD-1/A
**Semester:** 4th
**Date of Performance:** 31/01/2024
**Subject Name**: Git and Hub
**Subject Code:** 22CSH-293

**1. Aim/Overview of the practical:** To create and explore pull requests.

**2. Software Used:** Git Bash, GitHub.

**3. Steps for experiment/practical:**

1. Create or clone a repository on your local machine and open GIT BASH.
2. Move to the directory using the **cd** command.

```
thaku@LAPTOP-D9AREPUF MINGW64 ~/OneDrive/Desktop/git (master)
$ git clone https://github.com/Vinay000001/first.git
Cloning into 'first'...
remote: Enumerating objects: 28, done.
remote: Counting objects: 100% (28/28), done.
remote: Compressing objects: 100% (25/25), done.
remote: Total 28 (delta 9), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (28/28), 10.86 KiB | 2.17 MiB/s, done.
Resolving deltas: 100% (9/9), done.
```

3. Create a file in the master or main branch , eg , **file1.c** and add some text to the file.

4. Add the file to the staging area using **git add** and then commit the changes using the **git commit** command.

```
thaku@LAPTOP-D9AREPUF MINGW64 ~/OneDrive/Desktop/git (master)
$ cd first

thaku@LAPTOP-D9AREPUF MINGW64 ~/OneDrive/Desktop/git/first (main)
$ touch file1.c

thaku@LAPTOP-D9AREPUF MINGW64 ~/OneDrive/Desktop/git/first (main)
$ vi file1.c

thaku@LAPTOP-D9AREPUF MINGW64 ~/OneDrive/Desktop/git/first (main)
$ cat file1.c
#include <stdio.h>
int main() {
   // printf() displays the string inside quotation
   printf("Hello, World!");
   return 0;
}

thaku@LAPTOP-D9AREPUF MINGW64 ~/OneDrive/Desktop/git/first (main)
$ git add file1.c

thaku@LAPTOP-D9AREPUF MINGW64 ~/OneDrive/Desktop/git/first (main)
$ git commit -m "wrote hello world"
```

5. Create a new branch and checkout to it using the **git checkout -b** command , eg , **test**.
6. Open the **file1.c** on the **vi** editor and make some changes in it.

```
$ cat file1.c
#include <stdio.h>
int main() {
   // printf() displays the string inside quotation
   printf("Hello, World!");
   return 0;
}
```

( master )

```
$ cat file1.c
#include <stdio.h>
int main() {
   // printf() displays the string inside quotation
   printf("Hello, World!");
   printf("22bdo10040");
   return 0;
}
```

( test)

7. Repeat step 4 again.

```
MINGW64:/c/Users/thaku/OneDrive/Desktop/git/first

thaku@LAPTOP-D9AREPUF MINGW64 ~/OneDrive/Desktop/git/first (main)
$ git commit -m "wrote hello world"
[main 4c575c6] wrote hello world
 1 file changed, 6 insertions(+)
 create mode 100644 file1.c

thaku@LAPTOP-D9AREPUF MINGW64 ~/OneDrive/Desktop/git/first (main)
$ git checkout -b test
Switched to a new branch 'test'

thaku@LAPTOP-D9AREPUF MINGW64 ~/OneDrive/Desktop/git/first (test)
$ vi file1.c

thaku@LAPTOP-D9AREPUF MINGW64 ~/OneDrive/Desktop/git/first (test)
$ cat file1.c
#include <stdio.h>
int main() {
   // printf() displays the string inside quotation
   printf("Hello, World!");
   printf("22bdo10040");
   return 0;
}

thaku@LAPTOP-D9AREPUF MINGW64 ~/OneDrive/Desktop/git/first (test)
$ git add file1.c

thaku@LAPTOP-D9AREPUF MINGW64 ~/OneDrive/Desktop/git/first (test)
$ git commit -m "v2"
[test 4ff5981] v2
 1 file changed, 1 insertion(+)

thaku@LAPTOP-D9AREPUF MINGW64 ~/OneDrive/Desktop/git/first (test)
$ git checkout main
```

8. Merge the **test** branch in the **master** branch using the **git merge <branch_name>** command and resolve the merge conflict if necessary.

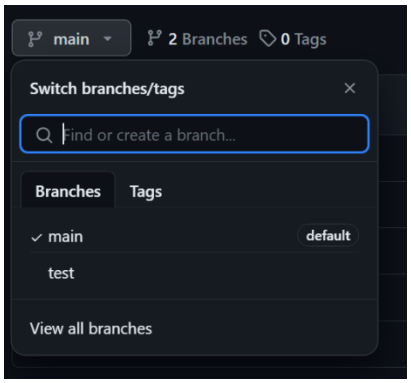
```
thaku@LAPTOP-D9AREPUF MINGW64 ~/OneDrive/Desktop/git/first (main)
$ git merge test
Updating 4c575c6..4ff5981
Fast-forward
 file1.c | 1 +
 1 file changed, 1 insertion(+)
```

9. Now, push your changes in the **master** and **test** branch to the remote repository.

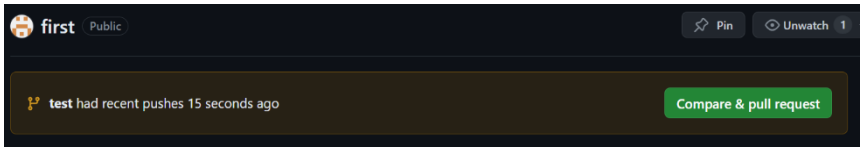
```
thaku@LAPTOP-D9AREPUF MINGW64 ~/OneDrive/Desktop/git/first (main)
$ git push origin main
info: please complete authentication in your browser...
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 12 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 622 bytes | 311.00 KiB/s, done.
Total 6 (delta 3), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (3/3), completed with 1 local object.
To https://github.com/Vinay000001/first.git
   bf66104..4ff5981  main -> main

thaku@LAPTOP-D9AREPUF MINGW64 ~/OneDrive/Desktop/git/first (main)
$ git checkout test
Switched to branch 'test'
```

```
thaku@LAPTOP-D9AREPUF MINGW64 ~/OneDrive/Desktop/git/first (test)
$ git push origin test
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'test' on GitHub by visiting:
remote:      https://github.com/Vinay000001/first/pull/new/test
remote:
To https://github.com/Vinay000001/first.git
 * [new branch]      test -> test
```

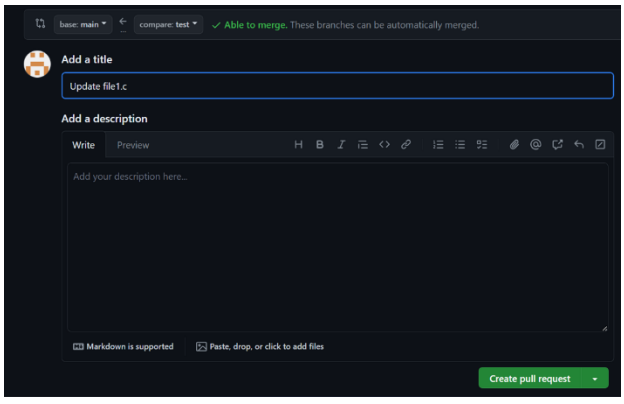
10. Now, Go to github, open the repository and move to the **test** branch and make some changes in a file.

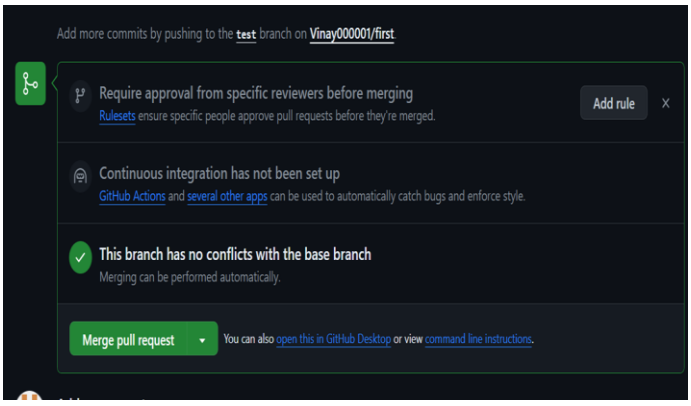



11. Commit the changes and move to the **master** branch. Click on the **Compare & Pull request**.

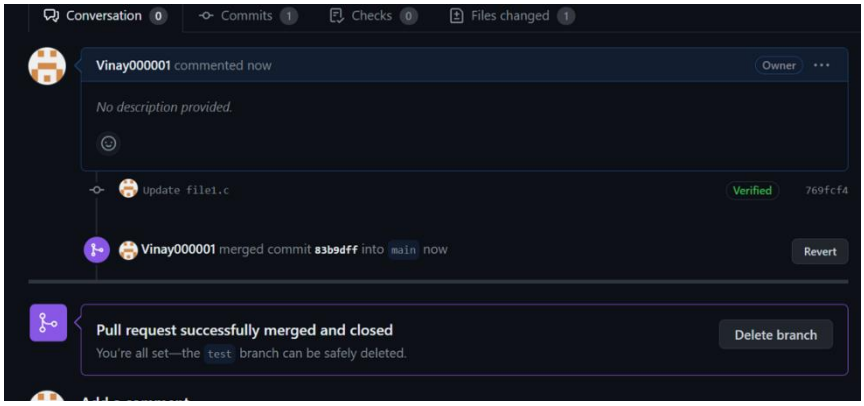



12. **Create the pull request,** resolve the merge conflicts (if any) and then **merge pull request**.

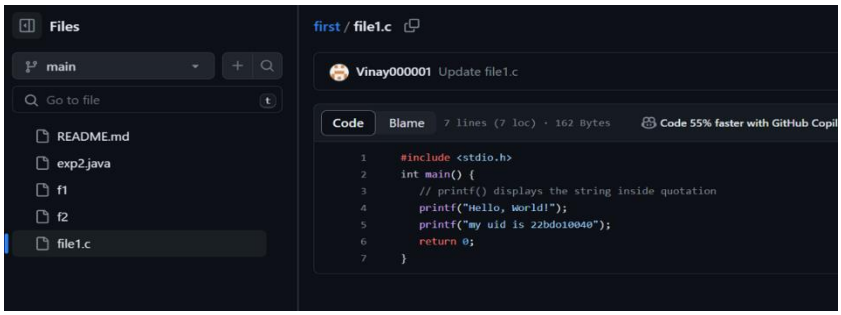



**13.** After the merging, you may choose to delete your branch , i.e , **test**




14. The master branch will now be reflecting the changes.




**15.** In the Git Bash, you may get the changes in your local repository using the **git pull** command and if you want the references of the commits, use **git fetch**.

**16.** Now, after **git pull**, we will be seeing the changes in **file1.c**

```
thaku@LAPTOP-D9AREPUF MINGW64 ~/OneDrive/Desktop/git/first (main)
$ git fetch
```

```
thaku@LAPTOP-D9AREPUF MINGW64 ~/OneDrive/Desktop/git/first (main)
$ git pull origin main
From https://github.com/vinay000001/first
 * branch            main       -> FETCH_HEAD
Updating 4ff5981..83b9dff
Fast-forward
 file1.c | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)
```

```
chaku@LAPTOP-D9AREPUF MINGW64 ~/OneDrive/Desktop/git/first (main)
$ cat file1.c
#include <stdio.h>
int main() {
   // printf() displays the string inside quotation
   printf("Hello, World!");
   printf("my uid is 22bdo10040");
   return 0;
}
```

**4. Result/Output/Writing Summary:**

In this experiment, we have created and explored the pull requests. We have created a new branch, made some changes in the files in that new branch and then merged the changes with the main branch by resolving merge conflicts by using both GitHub and Git Bash.

**Learning outcomes (What I have learnt):**

**1.** Learnt how to create a branch.

**2.** Learnt how to push the changes to the remote repository.

**3.** Learnt how to pull the changes from the remote repository.

**4.** Learnt to merge two branches.

**5.** Learnt how to resolve merge conflicts.

**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. | | | |
| | | | |