



Experiment -2.3

Student Name:Sneha Mehrotra

Branch: CSE(DEVOPS)

Semester: 4TH

Subject Name:GIT AND GITHUB

UID: 22BDO10048

Section/Group:22BCD-1/A

Date of Performance: 28/02/2024

Subject Code: 22CSH-293

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

2. Software used: Git Bash and Github.

3. Hardware Used: Computer system.

4. Steps for experiment:

A: On Git Bash:-

1. Create a file on local device and open that folder with Git Bash.
2. Create a repository on Github with ExpGit name and copy the http link.
3. Now open the file using “vi filename” and edit the file.

```
MINGW64/c:/Users/sneha/OneDrive/Desktop/exp7
this is exp 7|
~
~
~
~
~
~
~
~
~
~
```

4. Now add and commit the file using “git add .” and “git commit -m”message””.

```
MINGW64/c/Users/sneha/OneDrive/Desktop/ex7
sneha@LAPTOP-2GFSQH9Q MINGW64 ~/OneDrive/Desktop/ex7 (main)
$ git clone https://github.com/Sneha2003m/ex7.git
Cloning into 'ex7'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.

sneha@LAPTOP-2GFSQH9Q MINGW64 ~/OneDrive/Desktop/ex7 (main)
$ vi file.txt

sneha@LAPTOP-2GFSQH9Q MINGW64 ~/OneDrive/Desktop/ex7 (main)
$ git add .
warning: adding embedded git repository: OneDrive/Desktop/ex7/ex7
```

5. Now create a branch and edit the file in that branch using “git checkout -b name ” and “vi file”.

6. Now add and commit this file and move to main .

```
singh@HARSH1695 MINGW64 /d/CU/GIT/gitbash/Exp5 (main)
$ git checkout -b nope
Switched to a new branch 'nope'

singh@HARSH1695 MINGW64 /d/CU/GIT/gitbash/Exp5 (nope)
$ vi na.c

singh@HARSH1695 MINGW64 /d/CU/GIT/gitbash/Exp5 (nope)
$ git add .
warning: in the working copy of 'na.c', LF will be replaced by CRLF the next time Git touches it

singh@HARSH1695 MINGW64 /d/CU/GIT/gitbash/Exp5 (nope)
$ git commit -m"branch commit "
[nope 6e9f547] branch commit
1 file changed, 3 insertions(+), 2 deletions(-)

singh@HARSH1695 MINGW64 /d/CU/GIT/gitbash/Exp5 (nope)
$ git checkout main
Switched to branch 'main'
Your branch is ahead of 'origin/main' by 4 commits.
(use "git push" to publish your local commits)

singh@HARSH1695 MINGW64 /d/CU/GIT/gitbash/Exp5 (main)
$ |
```

6. Now show the difference using “git diff main nope” and merge the file with fast forward using “git merge --no-ff branch_name”.

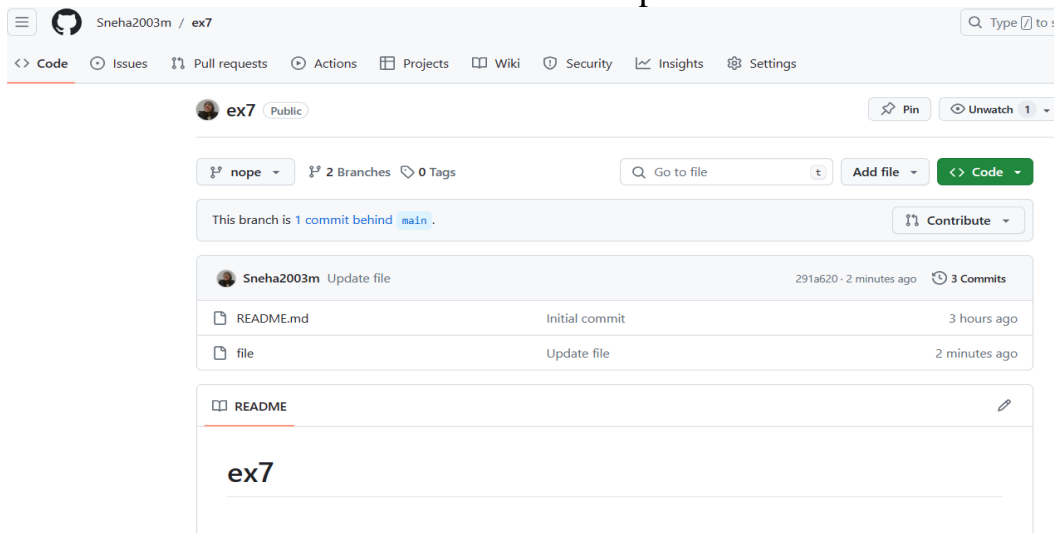
```
sneha@LAPTOP-2GFSQH9Q MINGW64 ~/OneDrive/Desktop/ex7 (main)
$ git diff main nope
diff --git a/OneDrive/Desktop/ex7/file.txt b/OneDrive/Desktop/ex7/file.txt
index d7dd287..5550a0c 100644
--- a/OneDrive/Desktop/ex7/file.txt
+++ b/OneDrive/Desktop/ex7/file.txt
@@ -1,1 @@
- this is exp 7
+ this is exp 7 of git
```

Note: ort is 3way commit and specify the commit message

```
sneha@LAPTOP-2GFSQH9Q MINGW64 ~/OneDrive/Desktop/ex7 (main)
$ git merge --no-ff nope
Merge made by the 'ort' strategy.
OneDrive/Desktop/ex7/file.txt | 2 +-
1 file changed, 1 insertion(+), 1 deletion(-)
```

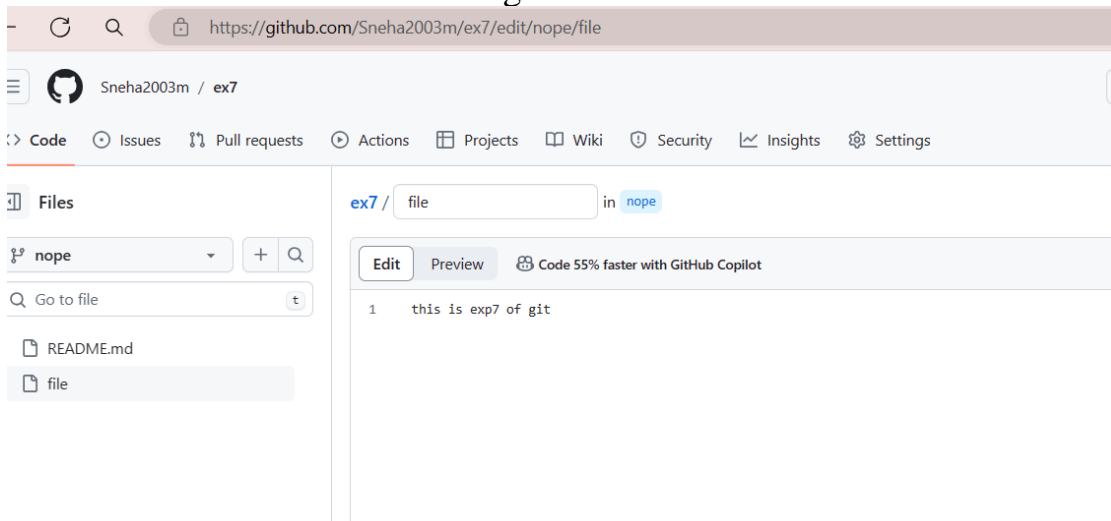
B: On Git Hub:-

1. Go to Github and create a branch name nope.



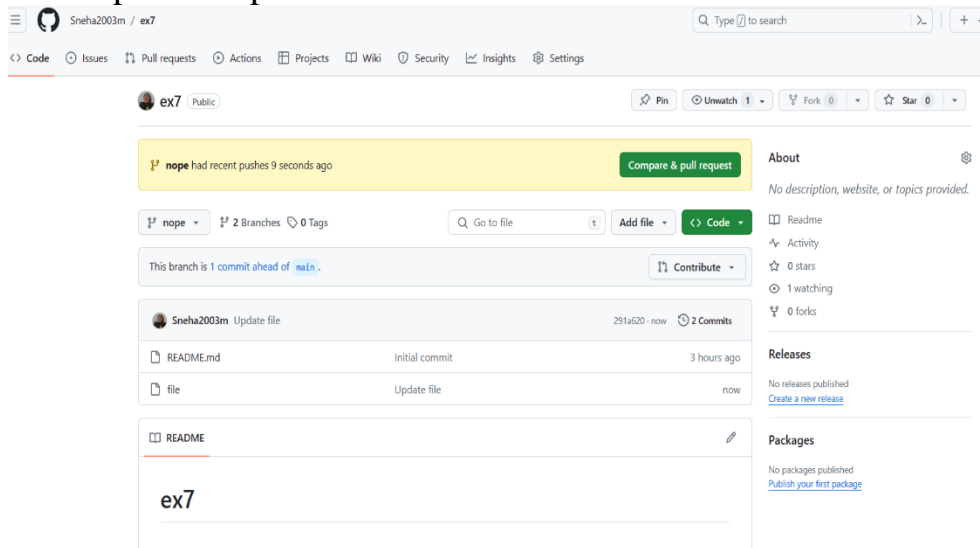
The screenshot shows a new GitHub repository named 'ex7' by user 'Sneha2003m'. The repository is public and has a branch named 'nope'. The 'nope' branch is 1 commit behind the 'main' branch. The repository contains two files: 'README.md' (Initial commit, 3 hours ago) and 'file' (Update file, 2 minutes ago). The 'README.md' file is open, showing the text 'ex7'.

2. Now edit the file and do the changes.

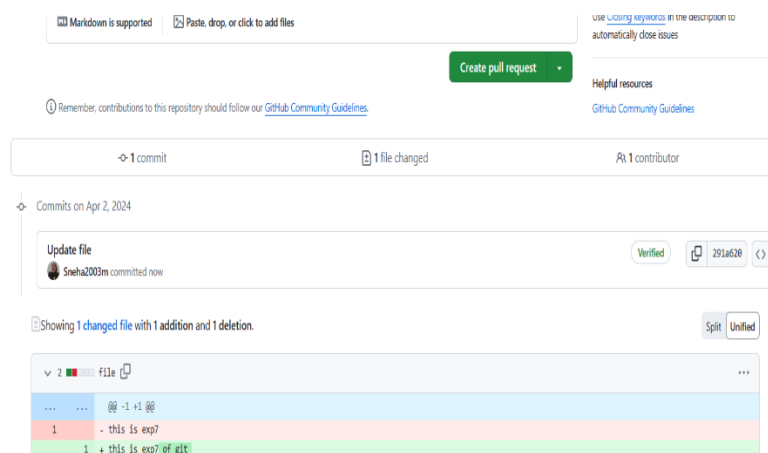


The screenshot shows the 'file' in the 'nope' branch being edited. The URL is 'https://github.com/Sneha2003m/ex7/edit/nope/file'. The file content is '1 this is exp7 of git'. The 'Edit' button is highlighted.

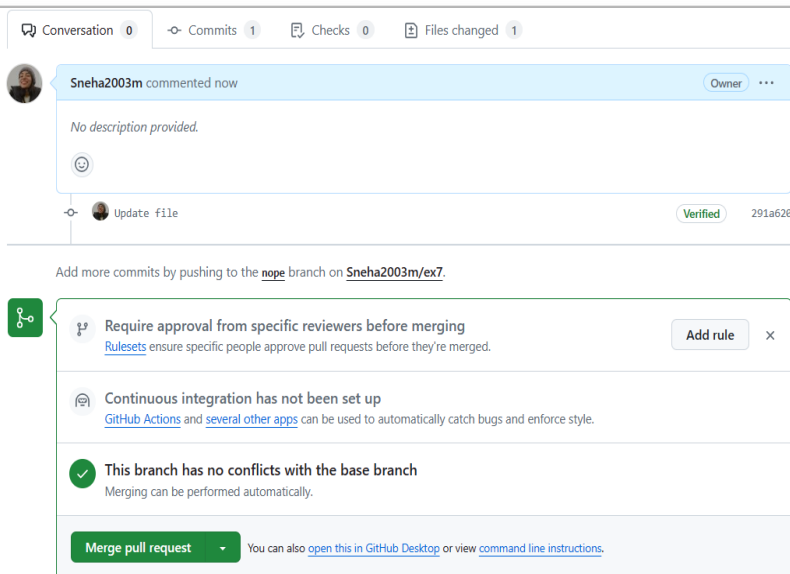
3. Commit the changes on the github.
4. Click on compare and pull.



5. Click on create pull request.



6. Click on merge pull request and after merging delete the Branch by click on delete branch option.



7. Verify the changes on github.

5. Result/Output/Writing Summary:

In this experiment we have learn about merge and merge conflict arises due to merging.

Learning outcomes (What I have learnt):

1. Learnt the difference between git merge and pull request.
2. Learnt how to make the merge
3. Learnt about type of merging.
4. Learnt about merge conflict.
5. Also learnt about ort merging.



DEPARTMENT OF ACADEMIC AFFAIRS

Discover. Learn. Empower.



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.			