Experiment -1.4

Editing a file and committing changes on GitHub

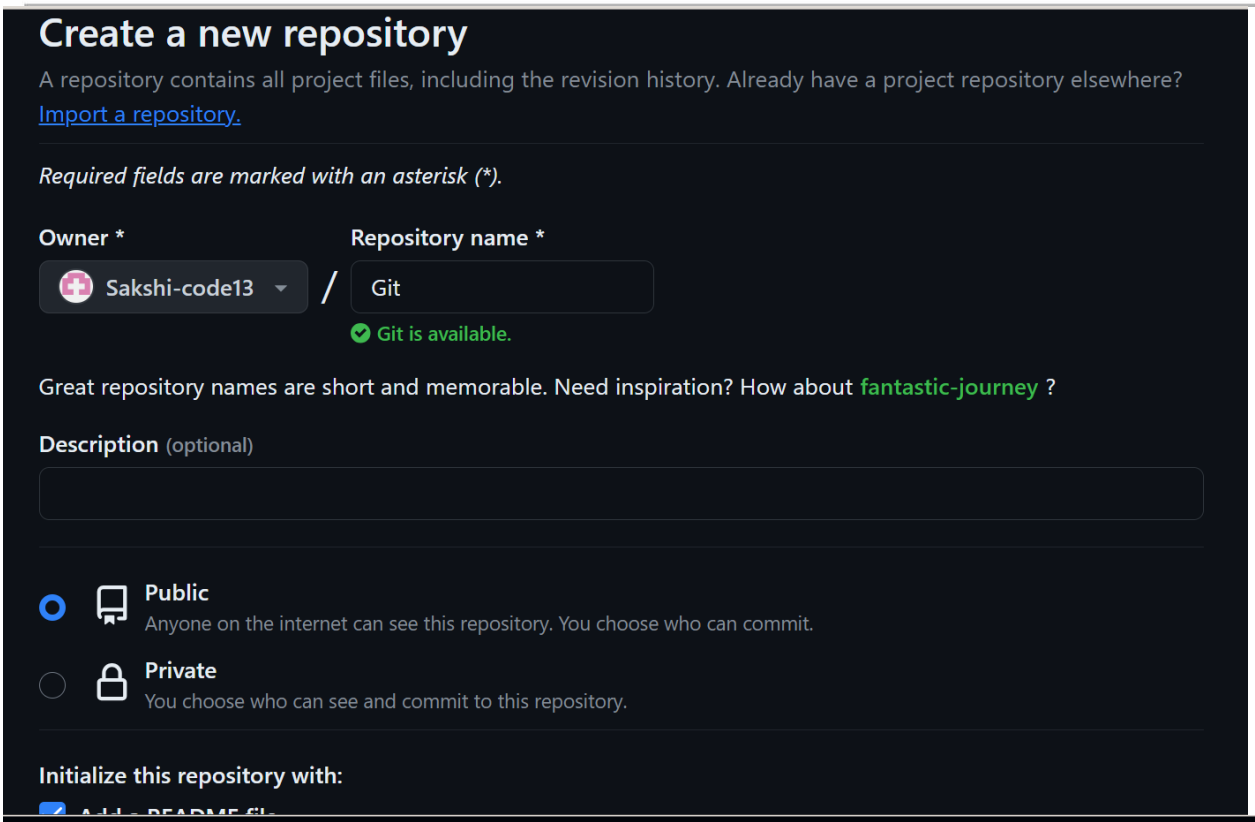
Student Name: Sakshi
UID: 22BDO10064
Branch: CSE(DevOps)
Section/Group: 22BCD-1/B
Semester: 4th
Date of Performance: 09-02-24
Subject Name: Git and GitHub
Subject Code: 22CSH-293

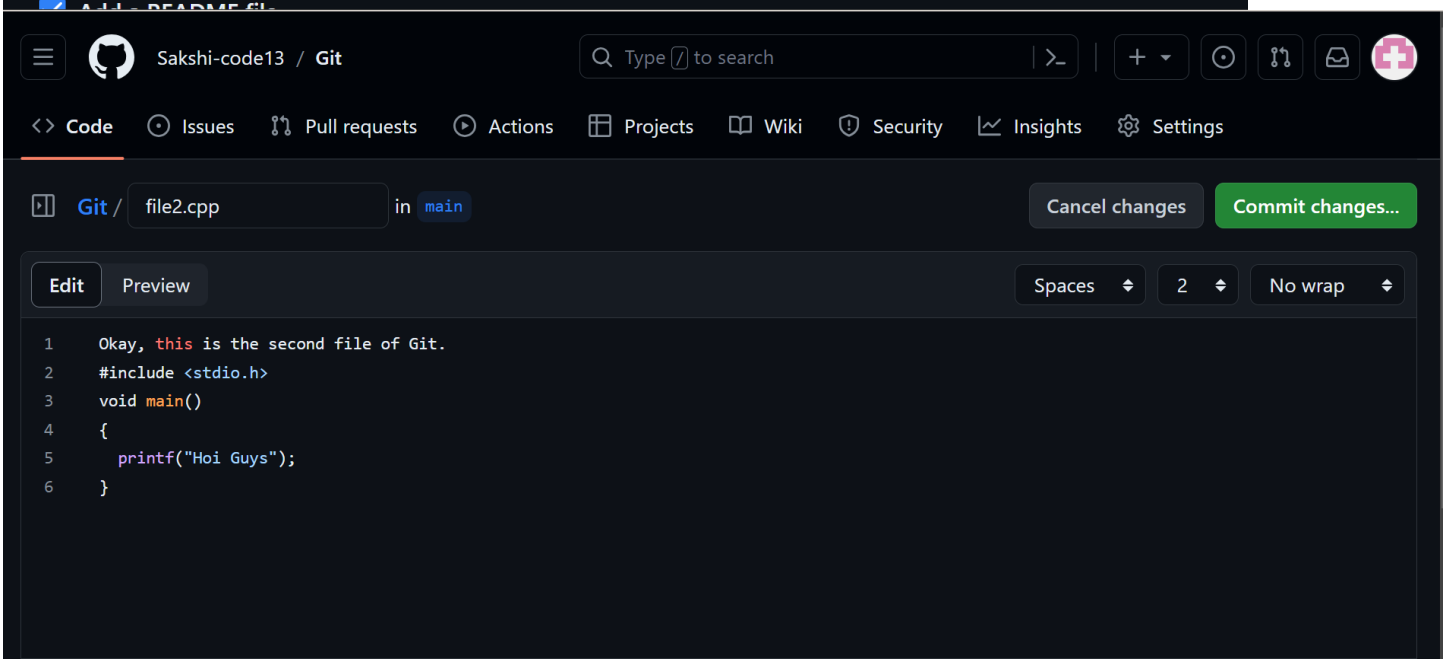
1. Aim of the practical: Editing a file and committing changes on GitHub

2. Task to be done: In this experiment, we have to edit a file and committing changes on GitHub.

3. Steps for experiment/practical:

1. In the first step, we will create the repository on the GitHub. And add 2 files into


Create a new repository
A repository contains all project files, including the revision history. Already have a project repository elsewhere?
Import a repository.
Required fields are marked with an asterisk (*).
Owner *
Repository name *
Sakshi-code13
Git
Git is available.
Great repository names are short and memorable. Need inspiration? How about fantastic-journey ?
Description (optional)
Public
Anyone on the internet can see this repository. You choose who can commit.
Private
You choose who can see and commit to this repository.
Initialize this repository with:


Sakshi-code13 / Git
Type / to search
Code
Issues
Pull requests
Actions
Projects
Wiki
Security
Insights
Settings
Git / file2.cpp in main
Cancel changes
Commit changes...
Edit
Preview
Spaces
2
No wrap
1 Okay, this is the second file of Git.
2 #include <stdio.h>
3 void main()
4 {
5 printf("Hoi Guys");
6 }

2. In the second step, initialize the git repository using the command **git init**

```
ADMIN@LAPTOP-RFULERMP MINGW64 ~/local/local (master)
$ git init Simsim
Initialized empty Git repository in C:/Users/ADMIN/local/local/Simsim/.git/
```

3. Change the current directory to the local repository, by using the command cd repository_name

```
ADMIN@LAPTOP-RFULERMP MINGW64 ~/local/local (master)
$ git init Simsim
Initialized empty Git repository in C:/Users/ADMIN/local/local/Simsim/.git/
```

4. Now, we will connect the local repository to the remote repository using the command **git remote add origin"**

```
ADMIN@LAPTOP-RFULERMP MINGW64 ~/local/local/Simsim (master)
$ git remote add origin https://github.com/Sakshi-code13/Git.git
```

5. Pull the main branch into the local machine

```
ADMIN@LAPTOP-RFULERMP MINGW64 ~/local/local/Simsim (master)
$ git pull origin main
remote: Enumerating objects: 11, done.
remote: Counting objects: 100% (11/11), done.
remote: Compressing objects: 100% (9/9), done.
remote: Total 11 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (11/11), 3.65 KiB | 73.00 KiB/s, done.
From https://github.com/Sakshi-code13/Git
 * branch            main       -> FETCH_HEAD
 * [new branch]      main       -> origin/main
```

6. Now, we have to bring the two files in our local repository from the remote repository

```
ADMIN@LAPTOP-RFULERMP MINGW64 ~/local/local/Simsim (master)
$ git checkout origin/main -- file1.cpp

ADMIN@LAPTOP-RFULERMP MINGW64 ~/local/local/Simsim (master)
$ git checkout origin/main -- file2.cpp
```

7. Now, we will view the content of the files we copied from the remote machine

```
ADMIN@LAPTOP-RFULERMP MINGW64 ~/local/local/Simsim (master)
$ cat file1.cpp
This is the file 1 of Git.
#include <stdio.h>
void main()
{
printf("Hello World");
}

ADMIN@LAPTOP-RFULERMP MINGW64 ~/local/local/Simsim (master)
$ cat file2.cpp
Okay, this is the second file of Git.
#include <stdio.h>
void main()
{
  printf("Hoi Guys");
}
```

8. Now, we will open the file, edit it and reopening the file to see the changes made in the local repository.

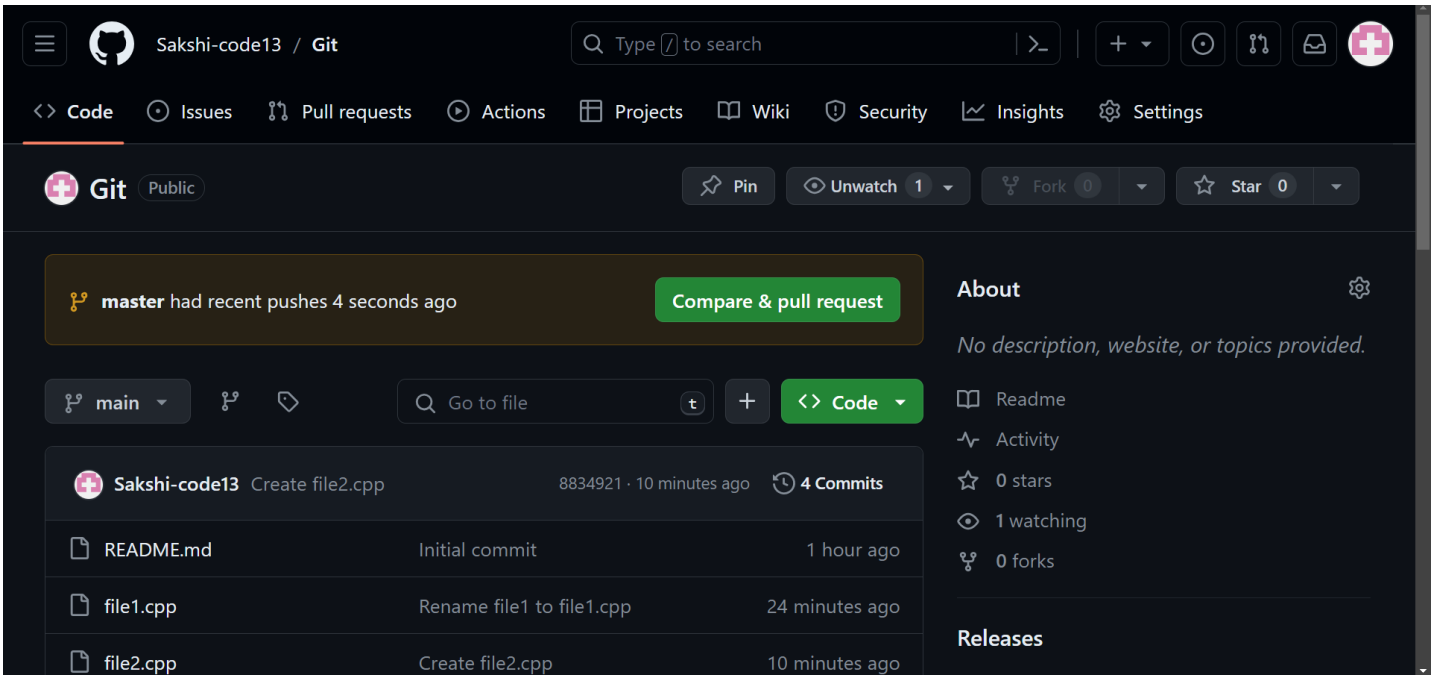
```
ADMIN@LAPTOP-RFULERMP MINGW64 ~/local/local/Simsim (master)
$ vi file1.cpp

ADMIN@LAPTOP-RFULERMP MINGW64 ~/local/local/Simsim (master)
$ cat file1.cpp
This is the file 1 of Git.
#include <stdio.h>
void main()
{
printf("Hello World");
printf("How are you?? Doing great ?");
}

ADMIN@LAPTOP-RFULERMP MINGW64 ~/local/local/Simsim (master)
$ vi file2.cpp

ADMIN@LAPTOP-RFULERMP MINGW64 ~/local/local/Simsim (master)
$ cat file2.cpp
Okay, this is the second file of Git.
#include <stdio.h>
void main()
{
  printf("Hoi Guys");
  printf("This is experiment 4");
}
```

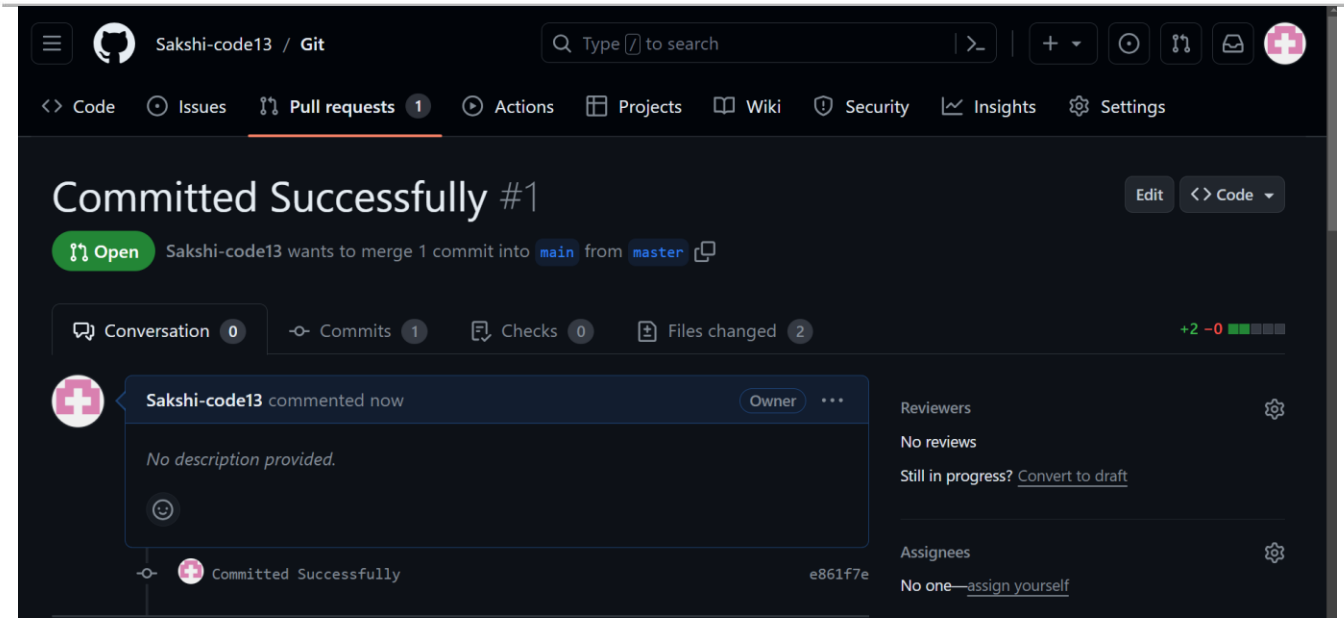
9. Now, add and commit both the files

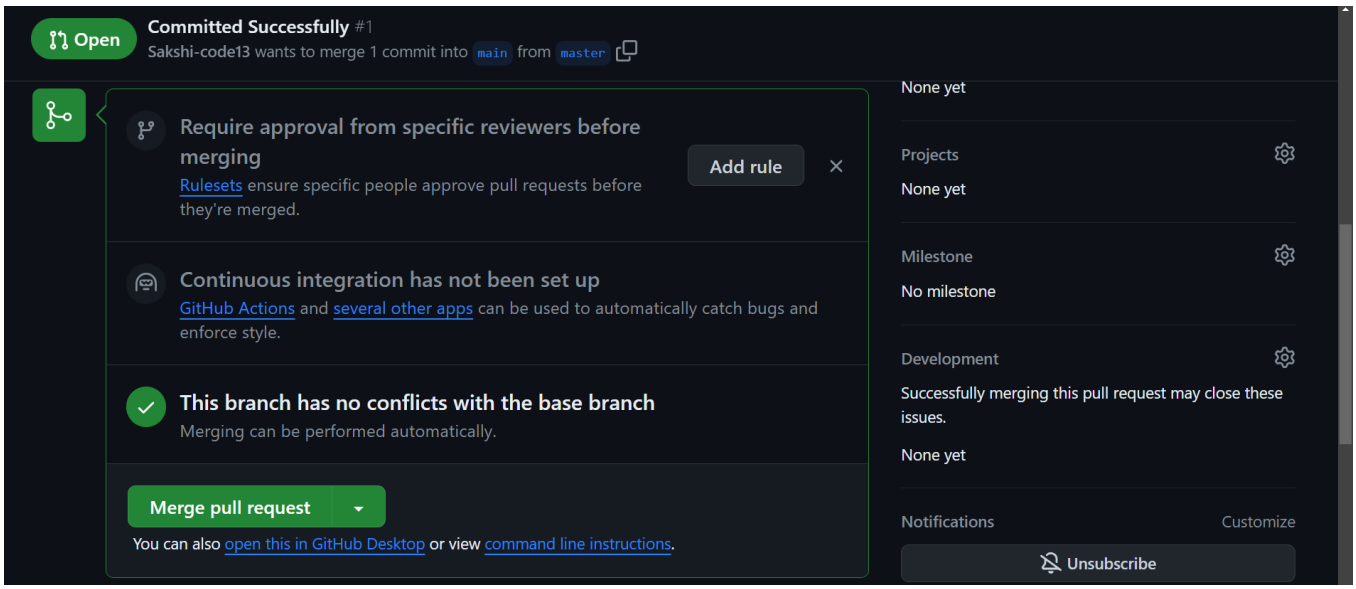
```
ADMIN@LAPTOP-RFULERMP MINGW64 ~/local/local/Simsim (master)
$ git add -A && git commit -m "Committed Successfully"
[master e861f7e] Committed Successfully
 2 files changed, 2 insertions(+)
```

10. Push the changes to the remote repository.

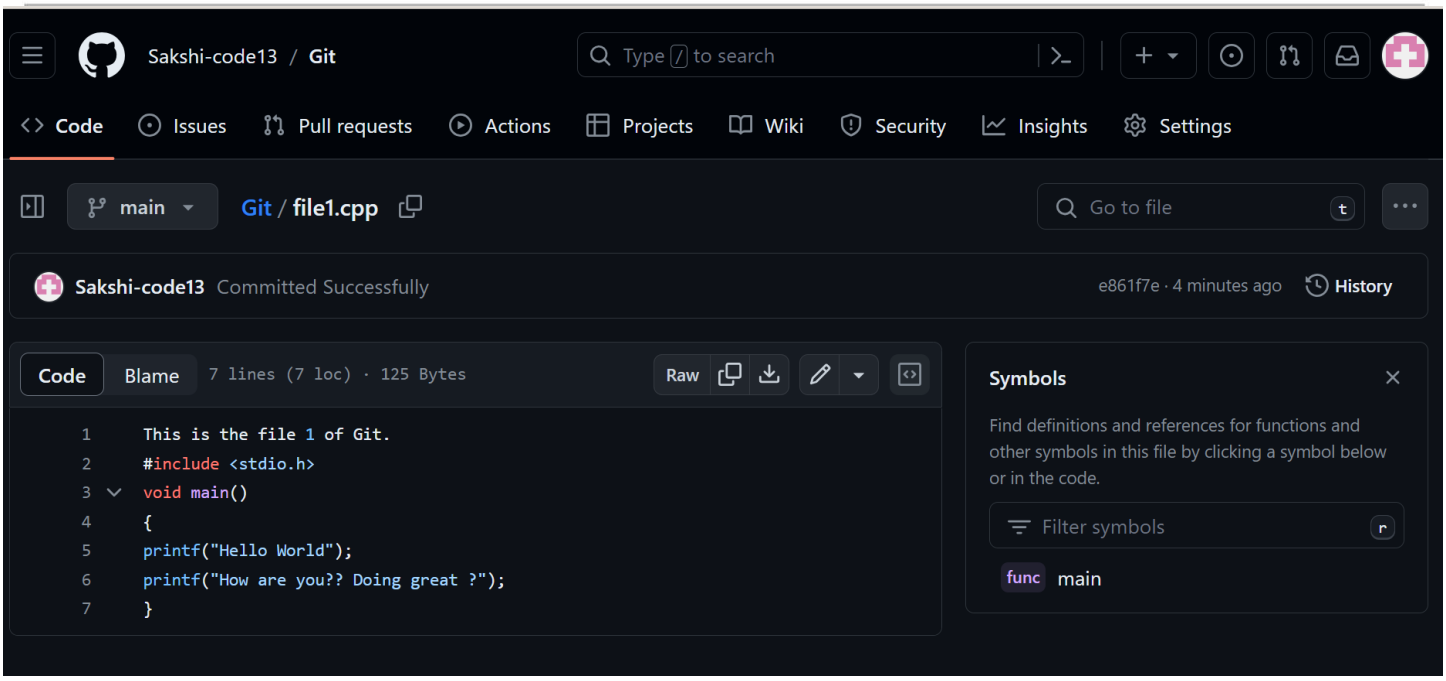
```
ADMIN@LAPTOP-RFULERMP MINGW64 ~/local/local/Simsim (master)
$ git push origin master
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 495 bytes | 495.00 KiB/s, done.
Total 4 (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 'master' on GitHub by visiting:
remote:      https://github.com/Sakshi-code13/Git/pull/new/master
remote:
To https://github.com/Sakshi-code13/Git.git
 * [new branch]      master -> master
```

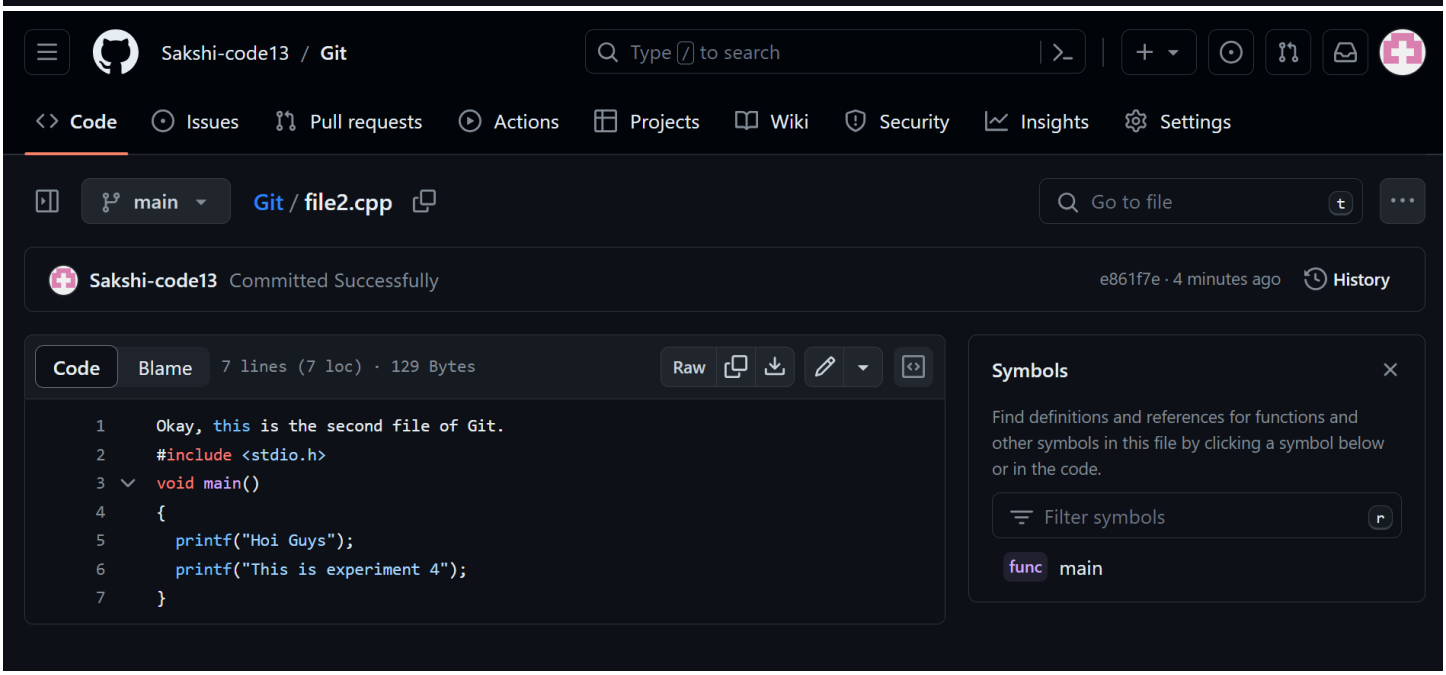




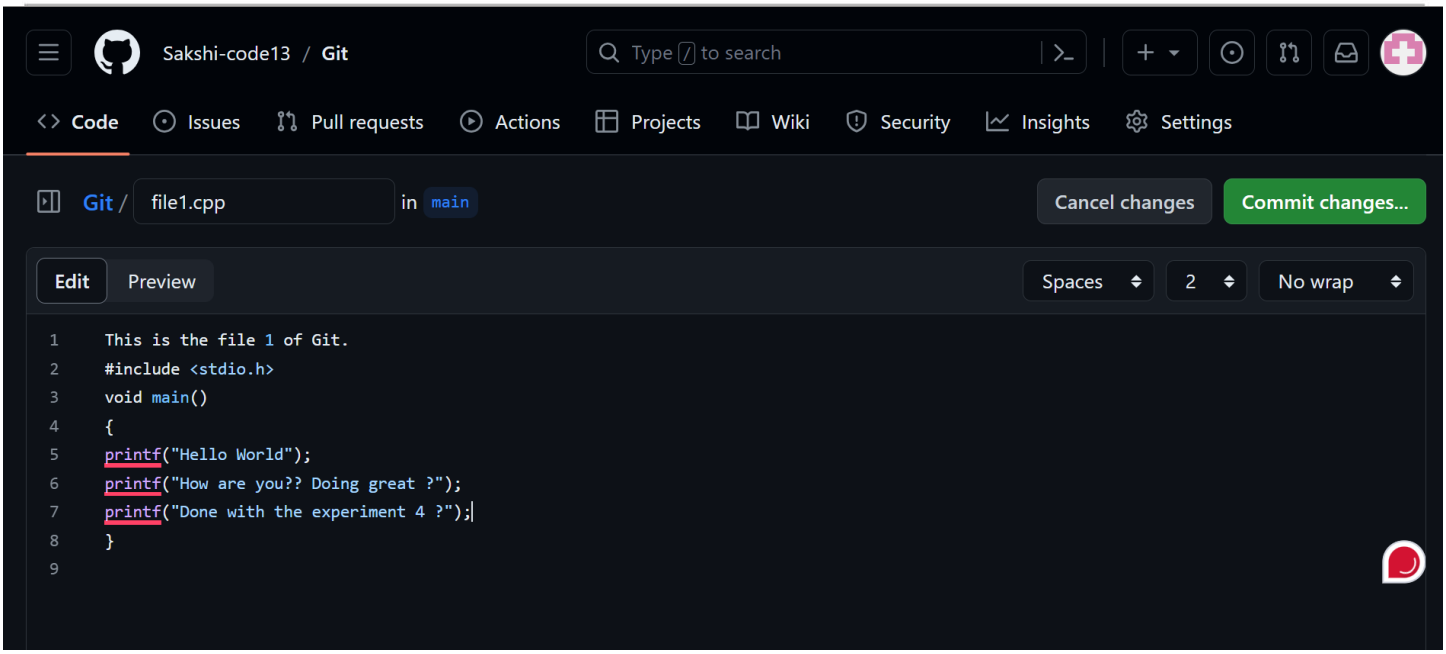


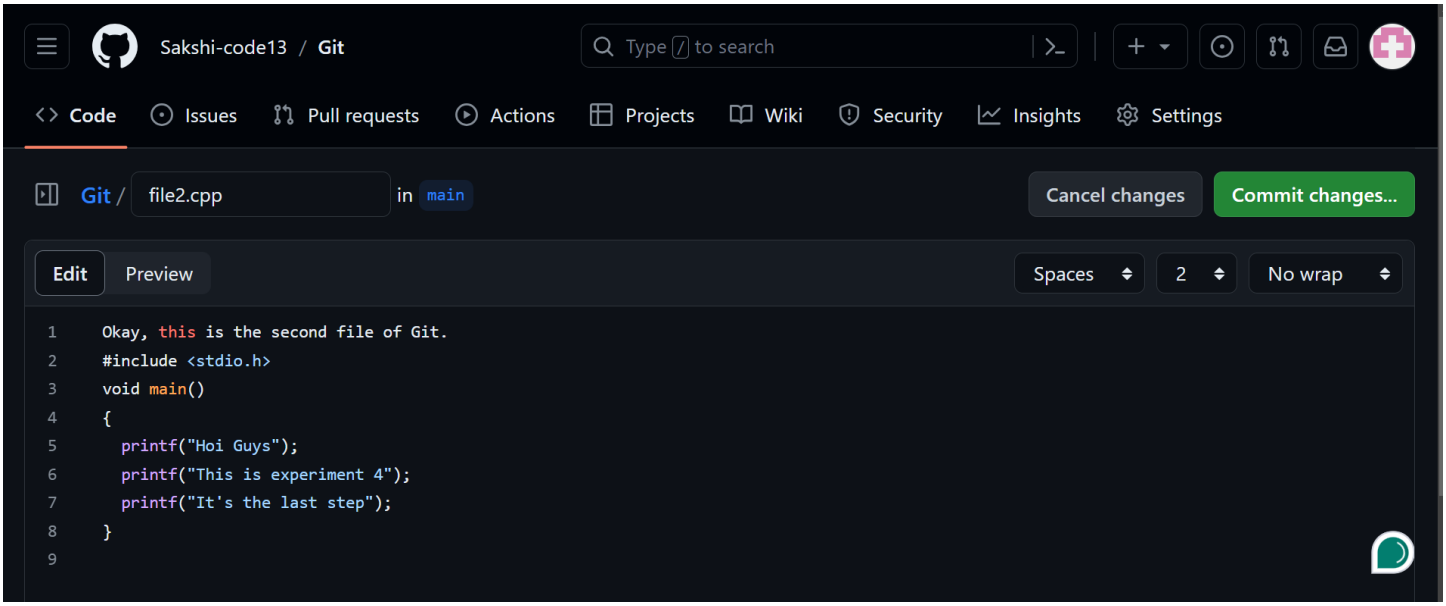
11. We can see the changes in our remote repositories





12. Now, change the files on remote repository.





13. Now, pull the changes into the remote repository.

```
ADMIN@LAPTOP-RFULERMP MINGW64 ~/local/local/Simsim (master)
$ git pull origin main
remote: Enumerating objects: 13, done.
remote: Counting objects: 100% (13/13), done.
remote: Compressing objects: 100% (7/7), done.
remote: Total 7 (delta 3), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (7/7), 2.79 KiB | 86.00 KiB/s, done.
From https://github.com/Sakshi-code13/Git
 * branch            main       -> FETCH_HEAD
   8834921..aa11aec  main       -> origin/main
Updating e861f7e..aa11aec
Fast-forward
 file1.cpp | 1 +
 file2.cpp | 1 +
 2 files changed, 2 insertions(+)
```

14. Now, we will observe the changes that are made on to the remote repository into the local repository.

```
ADMIN@LAPTOP-RFULERMP MINGW64 ~/local/local/Simsim (master)
$ cat file1.cpp
This is the file 1 of Git.
#include <stdio.h>
void main()
{
printf("Hello World");
printf("How are you?? Doing great ?");
printf("Done with the experiment 4 ?");
}

ADMIN@LAPTOP-RFULERMP MINGW64 ~/local/local/Simsim (master)
$ cat file2.cpp
Okay, this is the second file of Git.
#include <stdio.h>
void main()
{
  printf("Hoi Guys");
  printf("This is experiment 4");
  printf("It's the last step");
}
```

15. Now, we will make the new branch and checkout.

```
ADMIN@LAPTOP-RFULERMP MINGW64 ~/local/local/Simsim (master)
$ git checkout -b Sim
Switched to a new branch 'Sim'
```

16. Use vi command to edit the content and use cat command to see the changes

```
ADMIN@LAPTOP-RFULERMP MINGW64 ~/local/local/Simsim (Sim)
$ vi file1.cpp

ADMIN@LAPTOP-RFULERMP MINGW64 ~/local/local/Simsim (Sim)
$ cat file1.cpp
This is the file 1 of Git.
#include <stdio.h>
void main()
{
printf("Hello World");
printf("How are you?? Doing great ?");
printf("Done with the experiment 4 ?");
printf("Now, let's move to the next one");
}

ADMIN@LAPTOP-RFULERMP MINGW64 ~/local/local/Simsim (Sim)
$ vi file2.cpp

ADMIN@LAPTOP-RFULERMP MINGW64 ~/local/local/Simsim (Sim)
$ cat file2.cpp
Okay, this is the second file of Git.
#include <stdio.h>
void main()
{
  printf("Hoi Guys");
  printf("This is experiment 4");
  printf("It's the last step");
  printf("Moving...");
}
```

17. Now, add and commit both the files.

```
ADMIN@LAPTOP-RFULERMP MINGW64 ~/local/local/Simsim (Sim)
$ git add .

ADMIN@LAPTOP-RFULERMP MINGW64 ~/local/local/Simsim (Sim)
$ git commit -m "Successful"
[Sim cb8d5c4] Successful
 2 files changed, 2 insertions(+)
```

18. Checkout to the main branch

```
ADMIN@LAPTOP-RFULERMP MINGW64 ~/local/local/Simsim (Sim)
$ git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
```

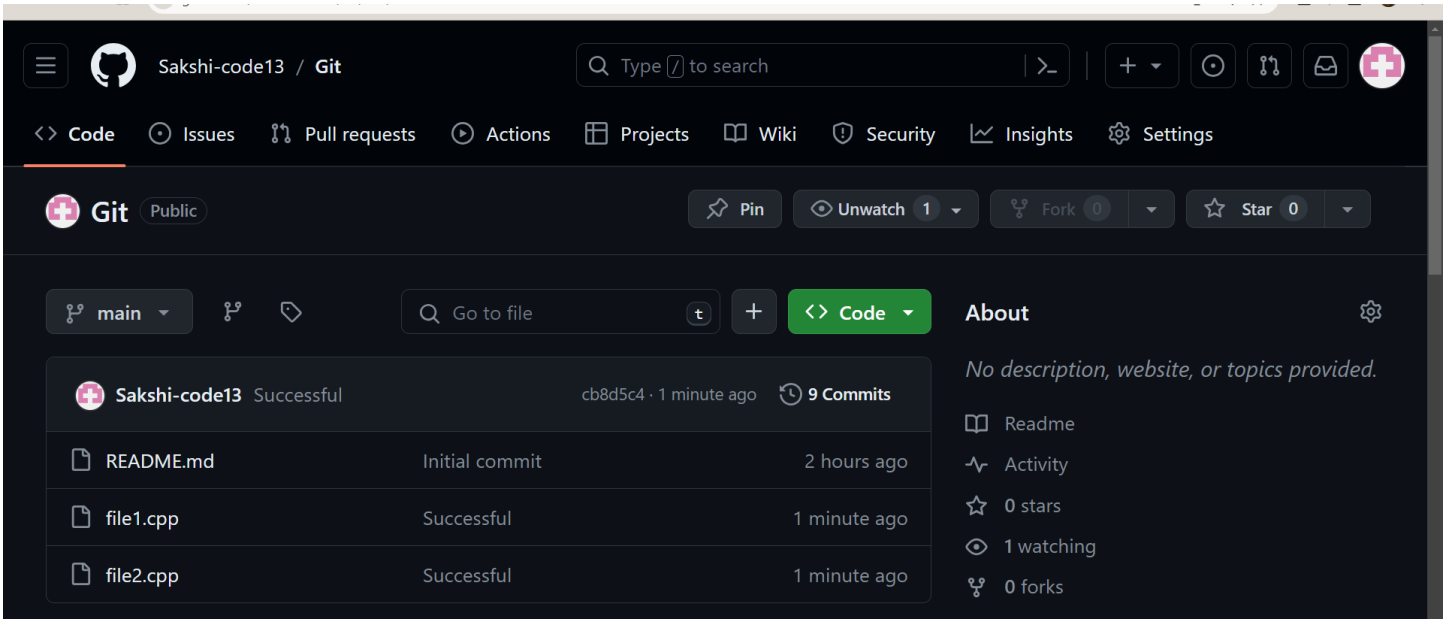
19. Use git diff command to see the differences.

```
ADMIN@LAPTOP-RFULERMP MINGW64 ~/local/local/Simsim (main)
$ git diff Sim
diff --git a/file1.cpp b/file1.cpp
index 911a45e..030ce7f 100644
--- a/file1.cpp
+++ b/file1.cpp
@@ -5,4 +5,5 @@ void main()
 printf("Hello World");
 printf("How are you?? Doing great ?");
 printf("Done with the experiment 4 ?");
+printf("Now, let's move to the next one");
 }
diff --git a/file2.cpp b/file2.cpp
index 362bde7..436592a 100644
--- a/file2.cpp
+++ b/file2.cpp
@@ -5,4 +5,5 @@ void main()
    printf("Hoi Guys");
    printf("This is experiment 4");
    printf("It's the last step");
+   printf("Moving...");
 }
```

20. Use git merge to merge two branches and push the merged branch on the github.

```
ADMIN@LAPTOP-RFULERMP MINGW64 ~/local/local/Simsim (main)
$ git merge Sim
Updating aa11aec..cb8d5c4
Fast-forward
 file1.cpp | 1 +
 file2.cpp | 1 +
 2 files changed, 2 insertions(+)

ADMIN@LAPTOP-RFULERMP MINGW64 ~/local/local/Simsim (main)
$ git push origin main
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 439 bytes | 439.00 KiB/s, done.
Total 4 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/Sakshi-code13/Git.git
   aa11aec..cb8d5c4  main -> main
```

21. Now, we can see the remote repository is changed again.



4. Result: In this experiment, we have created one repository and added two files with some contents into it. After that, we have pulled and edited the files into the Git Bash, made some changes, then again pushed it to the remote repository, and observed the changes that are now merged.

Learning outcomes (What I have learnt):

1. Understanding Git Workflow
2. Understanding the difference between local and remote repositories.
3. Pull and push commands
4. Committing changes
5. Working with staging area.

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.			