



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



DEPARTMENT OF ACADEMIC AFFAIRS

Discover. Learn. Empower.



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 *

Sakshi-code13

Repository name *

/ 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:

☒ README file

Sakshi-code13 / Git

Q 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-RFULERP 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-RFULERP 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-RFULERP 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-RFULERP 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/Simstim (master)
$ git checkout origin/main -- file1.cpp

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

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

ADMIN@LAPTOP-RFULERP 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-RFULERP 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.



DEPARTMENT OF ACADEMIC AFFAIRS

Discover. Learn. Empower.

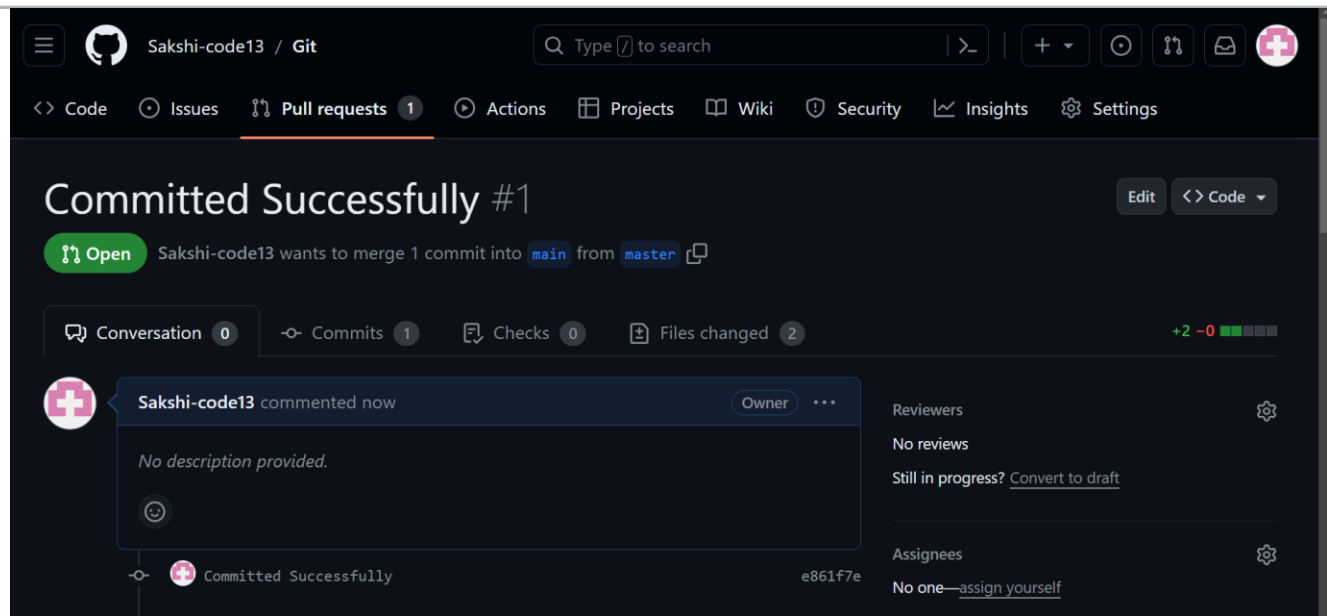


```
ADMIN@LAPTOP-RFULERP MINGW64 ~/local/local/Simsim (master)
$ git push origin master
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 495 bytes | 495.00 KiB/s, done.
Total 4 (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
```

The screenshot shows the GitHub interface for the repository 'Sakshi-code13 / Git'. The repository is public and has 1 unwatch, 0 forks, and 0 stars. The main branch is 'main'. The repository has 4 commits. The commit history shows:

Commit	Message	Time
Sakshi-code13	Create file2.cpp	8834921 · 10 minutes ago
	Initial commit	1 hour ago
	Rename file1 to file1.cpp	24 minutes ago
	Create file2.cpp	10 minutes ago

The right sidebar shows the 'About' section with the text 'No description, website, or topics provided.' and the 'Releases' section.



Committed Successfully #1 Edit <> Code

Open Sakshi-code13 wants to merge 1 commit into `main` from `master`

Conversation 0 Commits 1 Checks 0 Files changed 2 +2 -0

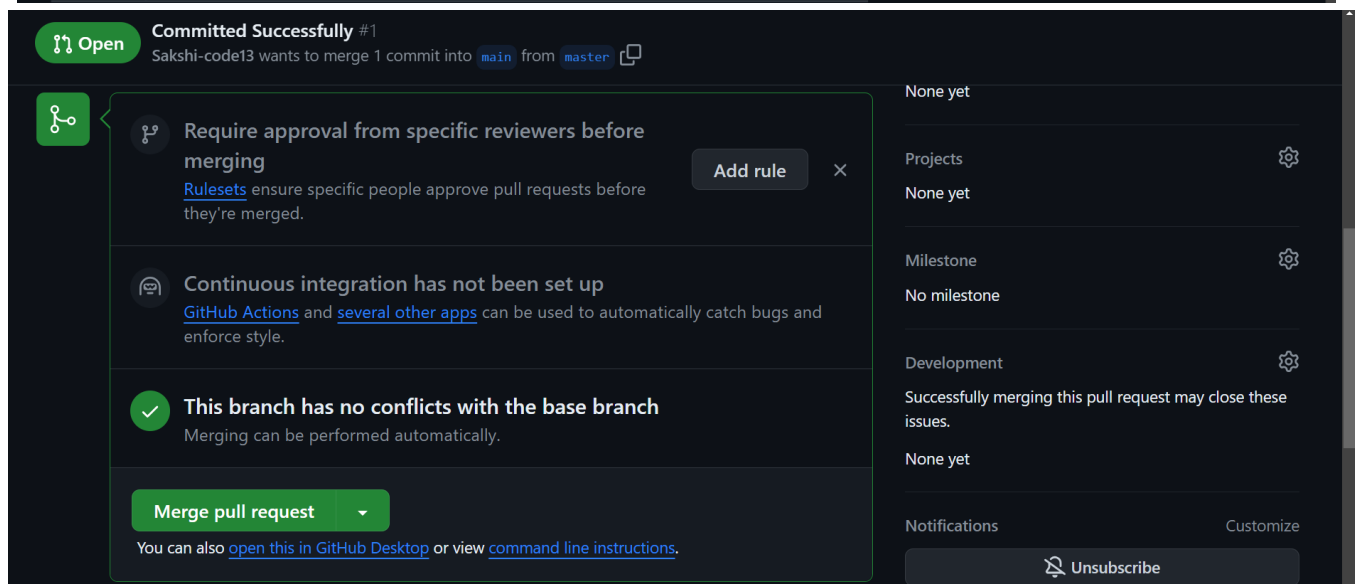
Sakshi-code13 commented now Owner

No description provided.

Reviewers
No reviews
Still in progress? [Convert to draft](#)

Assignees
No one—[assign yourself](#)

Committed Successfully e861f7e



Open **Committed Successfully #1**
Sakshi-code13 wants to merge 1 commit into `main` from `master`

Require approval from specific reviewers before merging
[Rulesets](#) ensure specific people approve pull requests before they're merged. Add rule

Continuous integration has not been set up
[GitHub Actions](#) and [several other apps](#) can be used to automatically catch bugs and enforce style.

This branch has no conflicts with the base branch
Merging can be performed automatically.

Merge pull request
You can also [open this in GitHub Desktop](#) or view [command line instructions](#).

None yet
Projects
None yet
Milestone
No milestone
Development
Successfully merging this pull request may close these issues.
None yet
Notifications Customize
Unsubscribe

11. We can see the changes in our remote repositories



DEPARTMENT OF ACADEMIC AFFAIRS

Discover. Learn. Empower.



Sakshi-code13 / Git

Q Type to search

Q Go to file

Sakshi-code13 Committed Successfully e861f7e · 4 minutes ago History

Code Blame 7 lines (7 loc) · 125 Bytes

Raw

```
1 This is the file 1 of Git.
2 #include <stdio.h>
3 void main()
4 {
5     printf("Hello World");
6     printf("How are you?? Doing great ?");
7 }
```

Symbols

Find definitions and references for functions and other symbols in this file by clicking a symbol below or in the code.

Filter symbols

func main

Sakshi-code13 / Git

Q Type to search

Q Go to file

Sakshi-code13 Committed Successfully e861f7e · 4 minutes ago History

Code Blame 7 lines (7 loc) · 129 Bytes

Raw

```
1 Okay, this is the second file of Git.
2 #include <stdio.h>
3 void main()
4 {
5     printf("Hoi Guys");
6     printf("This is experiment 4");
7 }
```

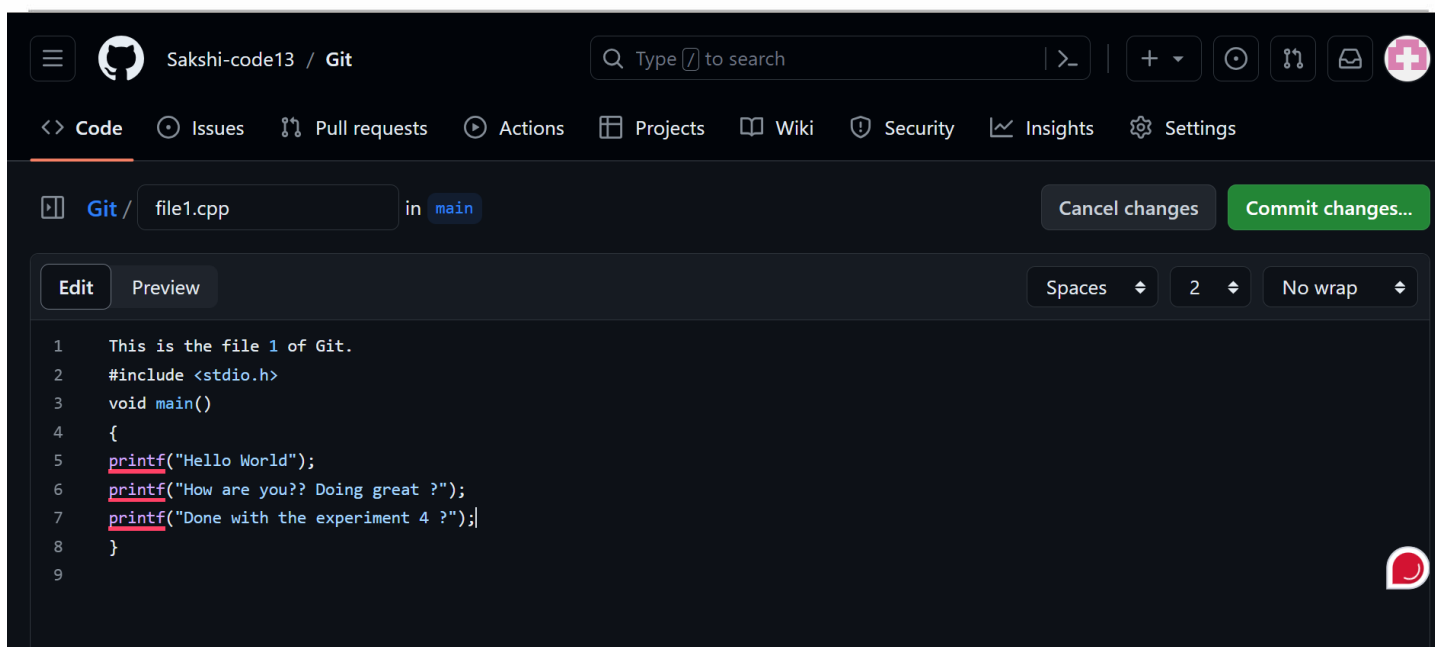
Symbols

Find definitions and references for functions and other symbols in this file by clicking a symbol below or in the code.

Filter symbols

func main

12. Now, change the files on remote repository.

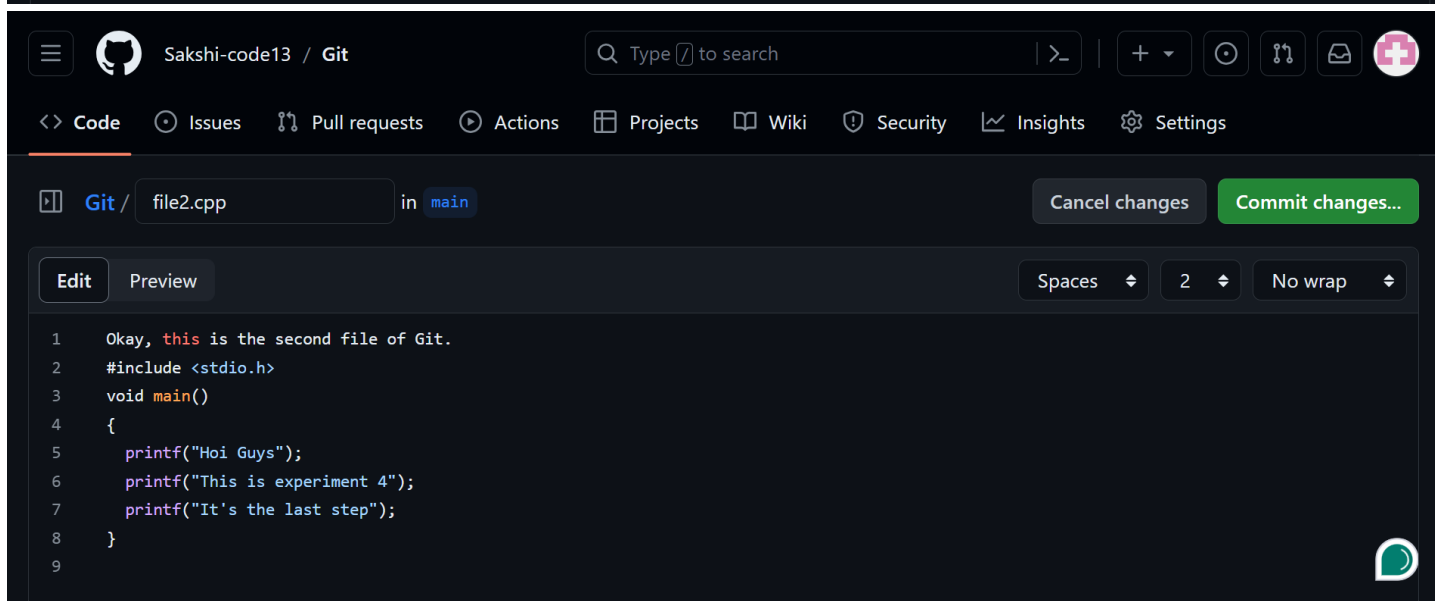


The screenshot shows the GitHub web interface for the repository 'Sakshi-code13 / Git'. The file 'file1.cpp' is selected in the 'main' branch. The interface includes a search bar, navigation tabs (Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, Settings), and buttons for 'Cancel changes' and 'Commit changes...'. The code editor shows the following C++ code:

```

1  This is the file 1 of Git.
2  #include <stdio.h>
3  void main()
4  {
5      printf("Hello World");
6      printf("How are you?? Doing great ?");
7      printf("Done with the experiment 4 ?");
8  }
9

```



The screenshot shows the GitHub web interface for the repository 'Sakshi-code13 / Git'. The file 'file2.cpp' is selected in the 'main' branch. The interface includes a search bar, navigation tabs (Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, Settings), and buttons for 'Cancel changes' and 'Commit changes...'. The code editor shows the following C++ code:

```

1  Okay, this is the second file of Git.
2  #include <stdio.h>
3  void main()
4  {
5      printf("Hoi Guys");
6      printf("This is experiment 4");
7      printf("It's the last step");
8  }
9

```

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

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

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

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

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

18. Checkout to the main branch

```
ADMIN@LAPTOP-RFULERP MINGW64 ~/local/local/Sim (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.

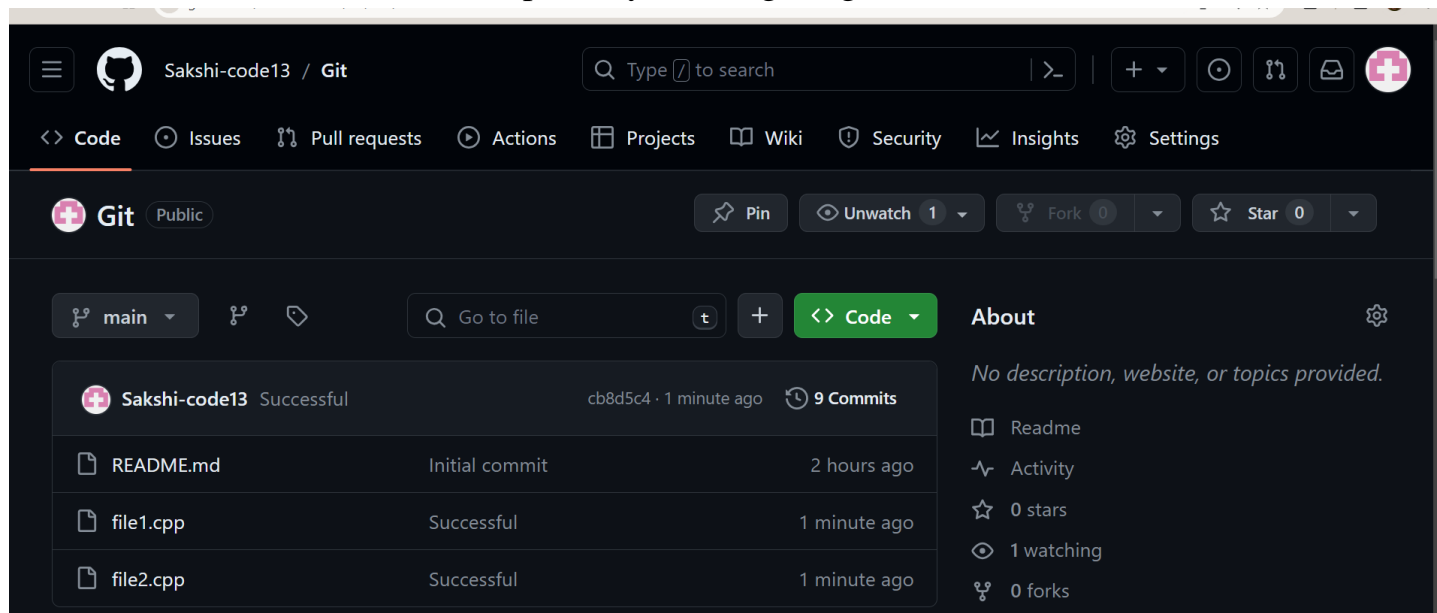
```
ADMIN@LAPTOP-RFULERP MINGW64 ~/local/local/Sim (main)
$ git diff Sim
diff --git a/file1.cpp b/file1.cpp
index 911a45e..030ce7f 100644
--- a/file1.cpp
+++ b/file1.cpp
@@ -5,4 +5,5 @@ void main()
    printf("Hello World");
    printf("How are you?? Doing great ?");
    printf("Done with the experiment 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 a11aec..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
 a11aec..cb8d5c4  main -> main
```

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



The screenshot shows the GitHub repository page for 'Sakshi-code13 / Git'. The repository is public and has 9 commits. The commit history shows three commits: 'Initial commit' (2 hours ago), 'file1.cpp' (1 minute ago), and 'file2.cpp' (1 minute ago). The repository has 0 stars, 1 watching, and 0 forks. The 'About' section is empty.

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.			