

Experiment-3.3

Student Name: Adarsh Kumar Singh

UID: 22BDO10053

Branch: CSE(DEVOPS)

Section/Group: 22BCD-1/B

Semester: 4th

Date of Performance: 05-04-24

Subject Name: Git and GitHub

Subject Code: 22CSH-293

1. Aim/Overview of the practical: work with your history in git.

2. Task to be done: git histories etc.

3. Steps for Experiment: -

1). Create a new repository in your local machine i.e. (**mkdir <repo_name>**), or go to the already created repository by (**cd <repo_name>**).

```
adarsh@ASUS MINGW64 ~ (master)
$ mkdir expr10

adarsh@ASUS MINGW64 ~ (master)
$ cd expr10
```

2). Now initialize a empty repository by (**git init**).

```
adarsh@ASUS MINGW64 ~/expr10 (master)
$ git init
Initialized empty Git repository in C:/Users/adars/expr10/.git/
```

3). After that create a file and do 3 commits and add and commit the changes (**vi <file_name> & git add . & git commit -m "your_message"**).

```
adarsh@ASUS MINGW64 ~/expr10 (master)
$ vi exp10.txt

adarsh@ASUS MINGW64 ~/expr10 (master)
$ git add .
warning: in the working copy of 'exp10.txt', LF will be replaced by CRLF the
next time Git touches it

adarsh@ASUS MINGW64 ~/expr10 (master)
$ git commit -m "first commit"
[master (root-commit) e5eba7e] first commit
1 file changed, 2 insertions(+)
create mode 100644 exp10.txt
```

```

adarsh@ASUS MINGW64 ~/expr10 (master)
$ vi exp10.txt

adarsh@ASUS MINGW64 ~/expr10 (master)
$ git add .
warning: in the working copy of 'exp10.txt', LF will be replaced by CRLF the
next time Git touches it

adarsh@ASUS MINGW64 ~/expr10 (master)
$ git commit -m "second commit line 3 added"
[master ba94e6e] second commit line 3 added
1 file changed, 1 insertion(+)

adarsh@ASUS MINGW64 ~/expr10 (master)
$ vi exp10.txt

adarsh@ASUS MINGW64 ~/expr10 (master)
$ git add .
warning: in the working copy of 'exp10.txt', LF will be replaced by CRLF the
next time Git touches it

adarsh@ASUS MINGW64 ~/expr10 (master)
$ git commit -m "third commit 4th line added"
[master e377696] third commit 4th line added
1 file changed, 1 insertion(+)

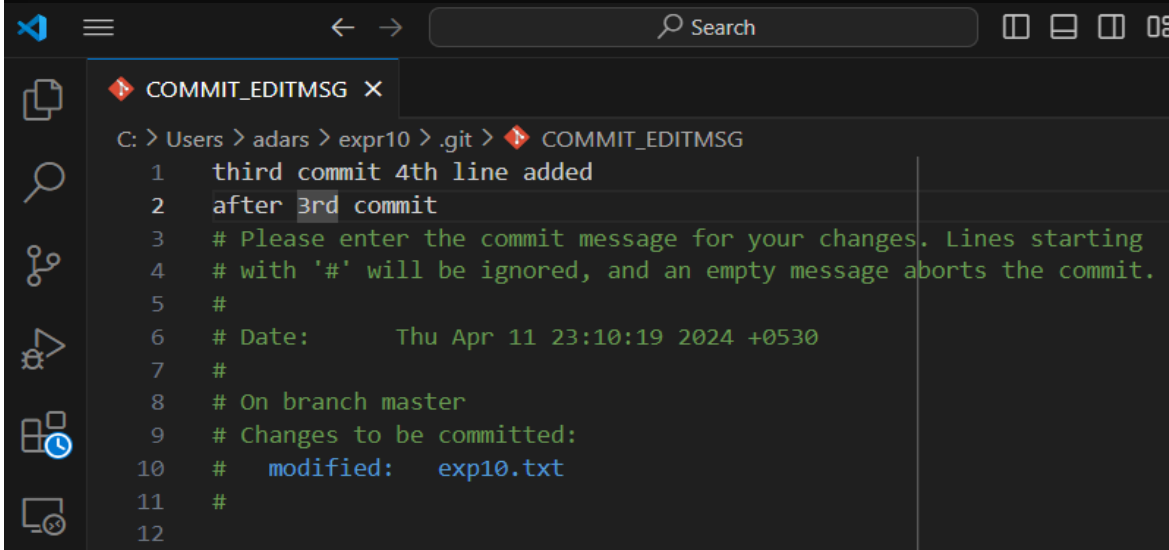
```

4). Now let's execute amend by (**git commit --amend**).

```

adarsh@ASUS MINGW64 ~/expr10 (master)
$ git commit --amend
[master a9cdc57] third commit 4th line added after 3rd commit
Date: Thu Apr 11 23:10:19 2024 +0530
1 file changed, 1 insertion(+)

```



The screenshot shows the `COMMIT_EDITMSG` editor window. The commit message is:

```

1  third commit 4th line added
2  after 3rd commit
3  # Please enter the commit message for your changes. Lines starting
4  # with '#' will be ignored, and an empty message aborts the commit.
5  #
6  # Date:      Thu Apr 11 23:10:19 2024 +0530
7  #
8  # On branch master
9  # Changes to be committed:
10 #   modified:   exp10.txt
11 #
12

```

5). Now execute git rebase to change the multiple commits by (**git rebase -i HEAD~2**).

```
adarsh@ASUS MINGW64 ~/expr10 (master)
$ git rebase -i HEAD~2
Successfully rebased and updated refs/heads/master.

adarsh@ASUS MINGW64 ~/expr10 (master)
$ git rebase -i HEAD~2
error: invalid command 'adding'
error: invalid line 3: adding some para
You can fix this with 'git rebase --edit-todo' and then run 'git rebase --co
ntinue'.
Or you can abort the rebase with 'git rebase --abort'.
```

```
adarsh@ASUS MINGW64 ~/expr10 (master|REBASE)
$ git rebase -i HEAD~1
fatal: It seems that there is already a rebase-merge directory, and
I wonder if you are in the middle of another rebase. If that is the
case, please try
    git rebase (--continue | --abort | --skip)
If that is not the case, please
    rm -fr ".git/rebase-merge"
and run me again. I am stopping in case you still have something
valuable there.

adarsh@ASUS MINGW64 ~/expr10 (master|REBASE)
$ git rebase --abort
```

6). Now reset the head by (**git reset HEAD^**).

```
adarsh@ASUS MINGW64 ~/expr10 (master)
$ git reset HEAD^
Unstaged changes after reset:
M    expr10.txt

adarsh@ASUS MINGW64 ~/expr10 (master)
$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   expr10.txt

no changes added to commit (use "git add" and/or "git commit -a")
```

7). Now add and commit your all the changes that you have made.

```
adarsh@ASUS MINGW64 ~/expr10 (master)
$ cat expr10.txt
this file is for experiment 10
adding some content in the file.
adding some line in the file i.e. line 3
adding another line i.e. line 4

adarsh@ASUS MINGW64 ~/expr10 (master)
$ git add .

adarsh@ASUS MINGW64 ~/expr10 (master)
$ git commit -m "final commit"
[master 7b387d2] final commit
1 file changed, 1 insertion(+)
```

8). Now execute git log by (git log --pretty=format: %h -%s).

```
adarsh@ASUS MINGW64 ~/expr10 (master)
$ git log --pretty=format:"%h - %s"
7b387d2 - final commit
ba94e6e - second commit line 3 added
e5eba7e - first commit
```

Learning outcomes (What I have learnt):

1. Learnt about Git.
2. Learnt about GitHub.
3. Learnt about various git histories commands that can be applied on Git Bash.
4. Learnt about git reset, amend, log, pretty.

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

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