Experiment -3.2

Student Name: Sushant jha **UID: 22BDO10052**
Branch: CSE (Devops) **Section/Group: 22BCD-1\A**
Semester: 4th **Date of Performance: 03/04/24**
Subject Name: Git and Git Hub **Subject Code: 22CSH-293**

1. **Aim/Overview of the practical:**

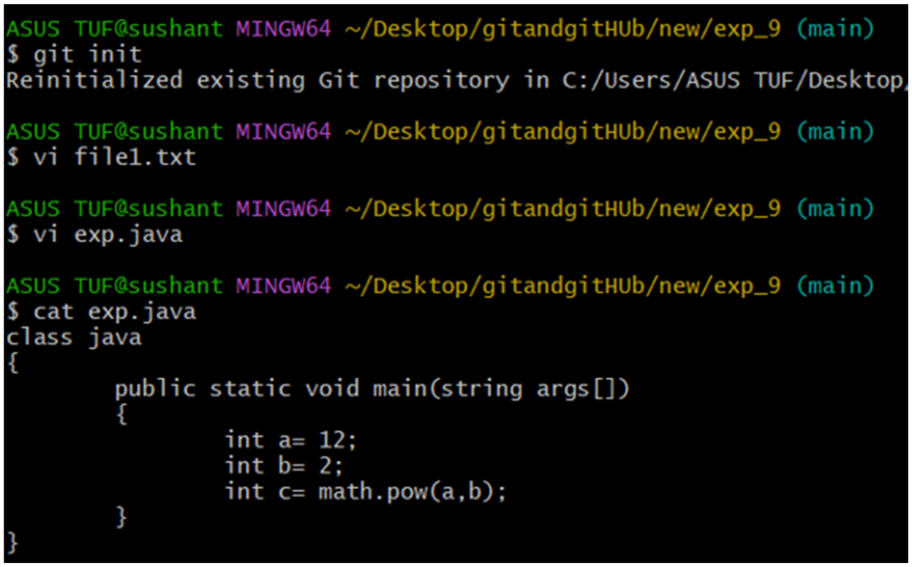
   Understanding the various reset modes in GitHub.

2. **Software Used:**

   Git Bash, Git-Hub.

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

1. Create a folder and open git bash .
2. Initialize git .
3. Open and edit the file "file.java" using the vi editor.
4. Edit the file as desired and save the changes.



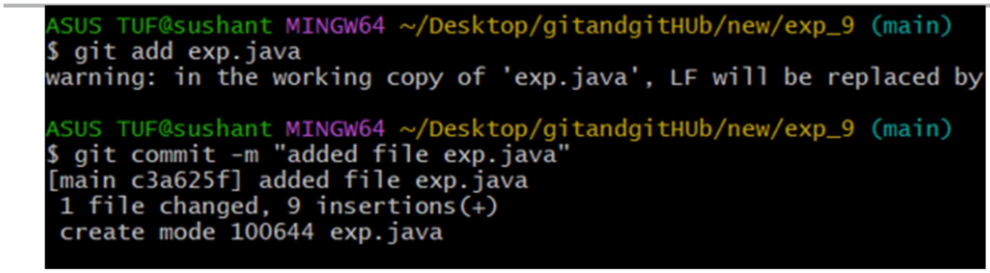
```
ASUS TUF@sushant MINGW64 ~/Desktop/gitandgitHUb/new/exp_9 (main)
$ git init
Reinitialized existing Git repository in C:/Users/ASUS TUF/Desktop/

ASUS TUF@sushant MINGW64 ~/Desktop/gitandgitHUb/new/exp_9 (main)
$ vi file1.txt

ASUS TUF@sushant MINGW64 ~/Desktop/gitandgitHUb/new/exp_9 (main)
$ vi exp.java

ASUS TUF@sushant MINGW64 ~/Desktop/gitandgitHUb/new/exp_9 (main)
$ cat exp.java
class java
{
        public static void main(string args[])
        {
                int a= 12;
                int b= 2;
                int c= math.pow(a,b);
        }
}
```

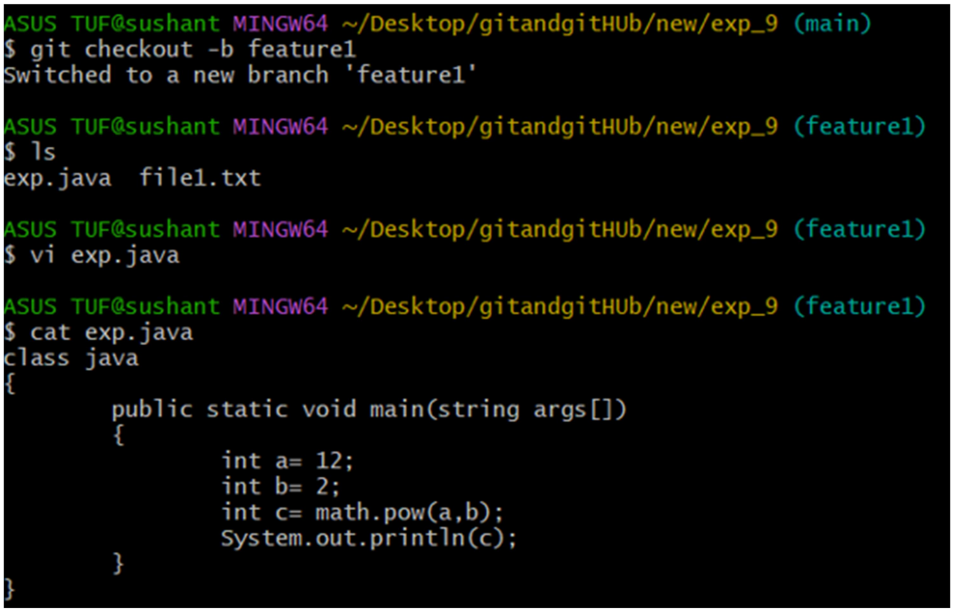
5.Add and commit the changes to the file using Git.



```
ASUS TUF@sushant MINGW64 ~/Desktop/gitandgitHUb/new/exp_9 (main)
$ git add exp.java
warning: in the working copy of 'exp.java', LF will be replaced by

ASUS TUF@sushant MINGW64 ~/Desktop/gitandgitHUb/new/exp_9 (main)
$ git commit -m "added file exp.java"
[main c3a625f] added file exp.java
 1 file changed, 9 insertions(+)
 create mode 100644 exp.java
```

6. Create and switch to feature branch .
7. Make changes in the file using vi file.java command.



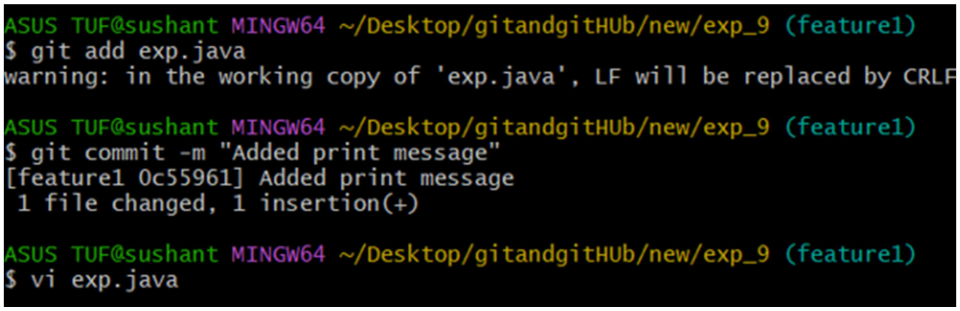
```
ASUS TUF@sushant MINGW64 ~/Desktop/gitandgitHUb/new/exp_9 (main)
$ git checkout -b feature1
Switched to a new branch 'feature1'

ASUS TUF@sushant MINGW64 ~/Desktop/gitandgitHUb/new/exp_9 (feature1)
$ ls
exp.java  file1.txt

ASUS TUF@sushant MINGW64 ~/Desktop/gitandgitHUb/new/exp_9 (feature1)
$ vi exp.java

ASUS TUF@sushant MINGW64 ~/Desktop/gitandgitHUb/new/exp_9 (feature1)
$ cat exp.java
class java
{
        public static void main(string args[])
        {
                int a= 12;
                int b= 2;
                int c= math.pow(a,b);
                System.out.println(c);
        }
}
```

8. Add the changes to the staging area using git add ..
9. Commit the changes with a descriptive message using git commit -m "message".
10. Repeat steps 3 and 4 multiple times to create multiple commits.



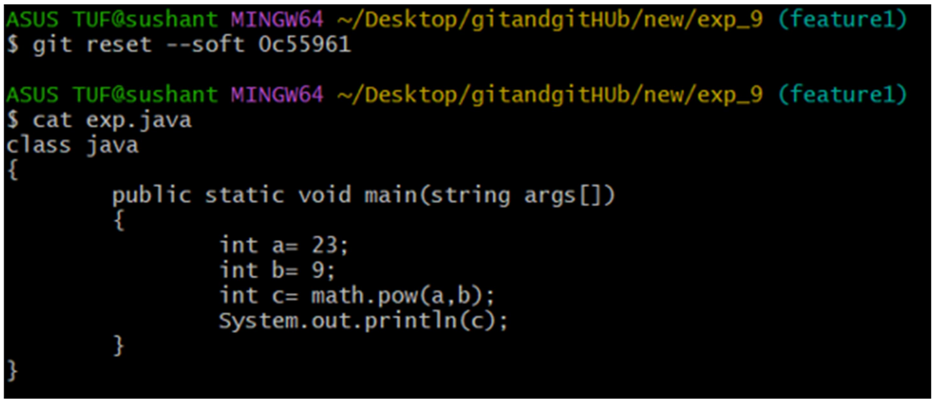
```
ASUS TUF@sushant MINGW64 ~/Desktop/gitandgitHUb/new/exp_9 (feature1)
$ git add exp.java
warning: in the working copy of 'exp.java', LF will be replaced by CRLF

ASUS TUF@sushant MINGW64 ~/Desktop/gitandgitHUb/new/exp_9 (feature1)
$ git commit -m "Added print message"
[feature1 0c55961] Added print message
 1 file changed, 1 insertion(+)

ASUS TUF@sushant MINGW64 ~/Desktop/gitandgitHUb/new/exp_9 (feature1)
$ vi exp.java
```

11. Perform a soft reset using Git to move the HEAD to the previous commit while keeping changes in the staging area.

12. Use the command git reset --soft HEAD~1.

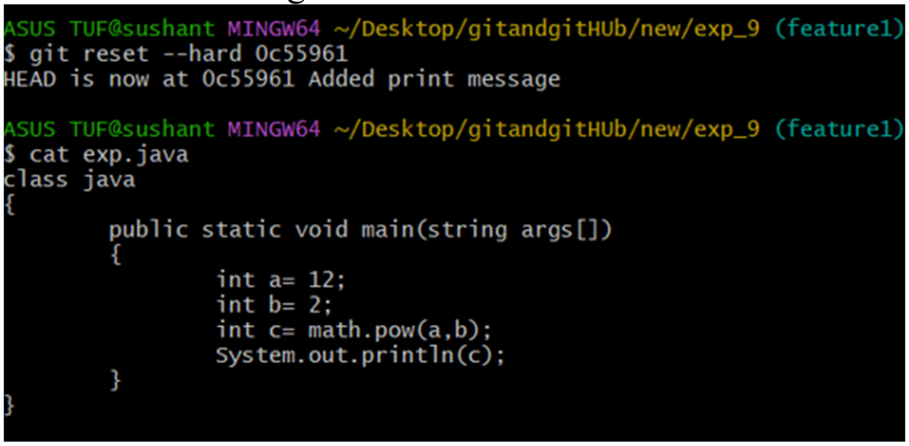


```
ASUS TUF@sushant MINGW64 ~/Desktop/gitandgitHUb/new/exp_9 (feature1)
$ git reset --soft 0c55961

ASUS TUF@sushant MINGW64 ~/Desktop/gitandgitHUb/new/exp_9 (feature1)
$ cat exp.java
class java
{
        public static void main(string args[])
        {
                int a= 23;
                int b= 9;
                int c= math.pow(a,b);
                System.out.println(c);
        }
}
```

13. Perform a hard reset using Git to move the HEAD to the previous commit, discarding all changes made since then.

14. Run the command git reset --hard HEAD~1.

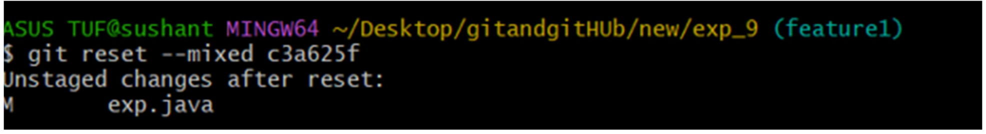


```
ASUS TUF@sushant MINGW64 ~/Desktop/gitandgitHUb/new/exp_9 (feature1)
$ git reset --hard 0c55961
HEAD is now at 0c55961 Added print message

ASUS TUF@sushant MINGW64 ~/Desktop/gitandgitHUb/new/exp_9 (feature1)
$ cat exp.java
class java
{
        public static void main(string args[])
        {
                int a= 12;
                int b= 2;
                int c= math.pow(a,b);
                System.out.println(c);
        }
}
```

15. Perform a mixed reset using Git to move the HEAD to the previous commit and unstage the changes, resulting in untracked files.

16. Execute the command git reset --mixed HEAD~1.



```
ASUS TUF@sushant MINGW64 ~/Desktop/gitandgitHUb/new/exp_9 (feature1)
$ git reset --mixed c3a625f
Unstaged changes after reset:
M       exp.java
```

Learning outcomes (What I have learnt):

1. Git Reset Variants
2. Staging Area Manipulation
3. File Tracking and Untracking
4. Commit Management

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.			