

## Experiment-5

**Student Name: Moksh Gupta**

**Branch: Cse (DevOps)**

**Semester: 4**

**Subject Name: Git and hub**

**UID: 22BDO10061**

**Section/Group: 22BCD1/B**

**Date of Performance: 23/02/2024**

**Subject Code: 22CSH-293**

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

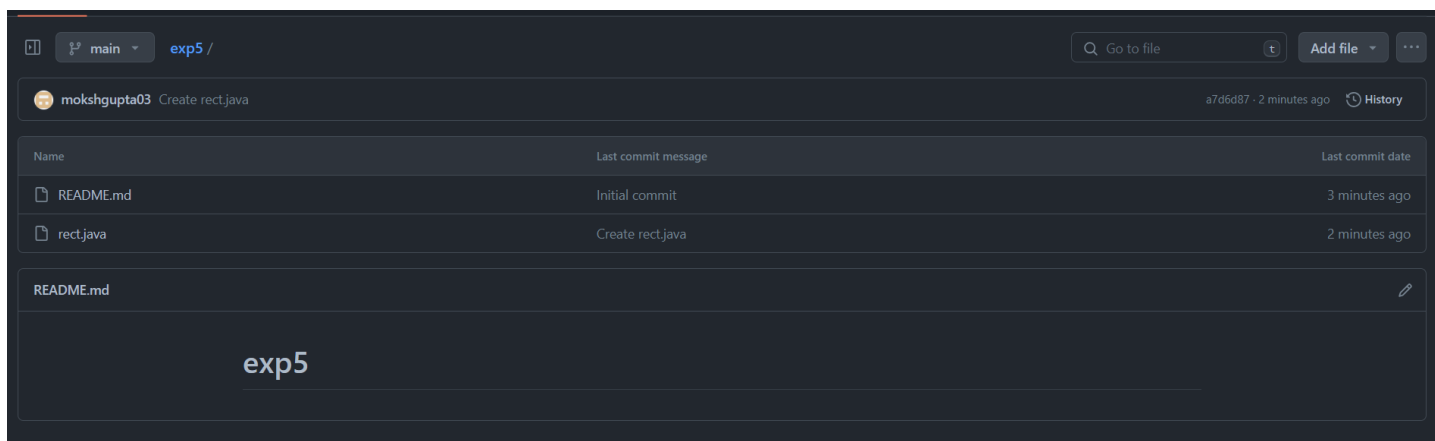
To merge Pull request and update local repository.

### 2.Apparatus :

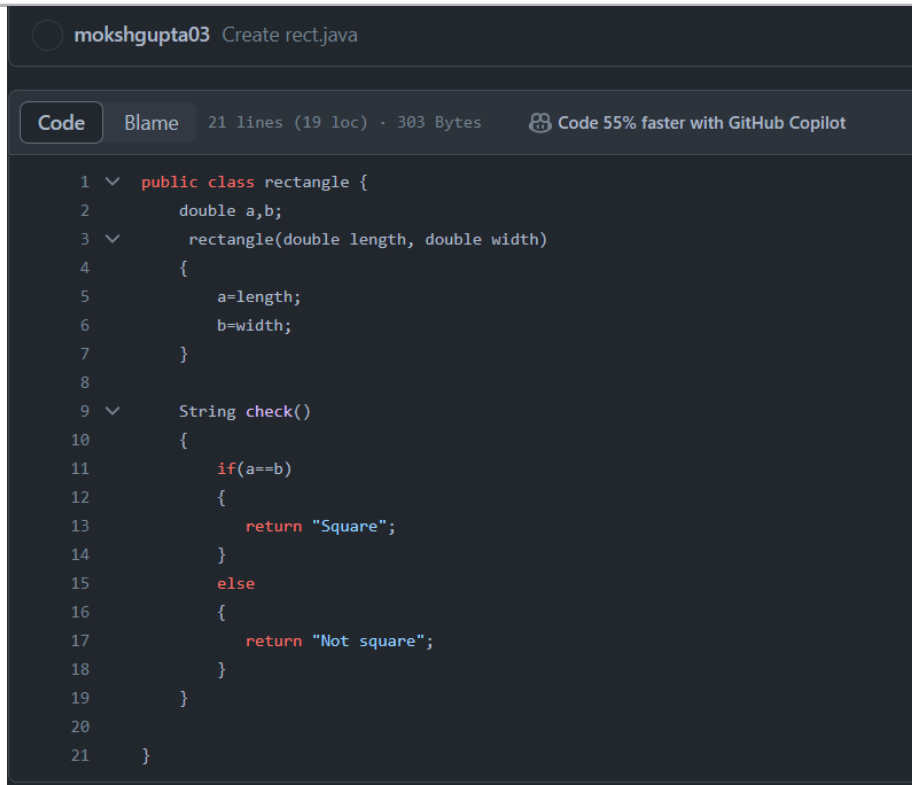
Laptop, Git software

### 3.Steps for experiment/practical:

- Create a repository on github with a file rect.java.



- Content inside rect.java.



```
1  public class rectangle {
2      double a,b;
3      rectangle(double length, double width)
4      {
5          a=length;
6          b=width;
7      }
8
9      String check()
10     {
11         if(a==b)
12         {
13             return "Square";
14         }
15         else
16         {
17             return "Not square";
18         }
19     }
20
21 }
```

- Create a local repository and initialize it.

```
new@DESKTOP-U33LA0K MINGW64 ~ (main)
$ git init local_repo1
Initialized empty Git repository in C:/Users/new/local_repo1/.git/
```

- Change directory to current repository.

```
new@DESKTOP-U33LA0K MINGW64 ~ (main)
$ cd local_repo1
```

- Add remote origin to local repository.

```
new@DESKTOP-U33LA0K MINGW64 ~/local_repo1 (main)
$ git remote add origin https://github.com/mokshgupta03/exp5.git
```

- Use git fetch command to fetch the remote repository and then pull the content of remote repository by pull command.

```
new@DESKTOP-U33LA0K MINGW64 ~/local_repo1 (main)
$ git fetch
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 6 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (6/6), 1.88 KiB | 137.00 KiB/s, done.
From https://github.com/mokshgupta03/exp5
* [new branch]      main       -> origin/main

new@DESKTOP-U33LA0K MINGW64 ~/local_repo1 (main)
$ git pull origin main
From https://github.com/mokshgupta03/exp5
* branch            main       -> FETCH_HEAD
```

- Create a new branch called firstbranch.

```
new@DESKTOP-U33LA0K MINGW64 ~/local_repo1 (main)
$ git checkout -b firstbranch
Switched to a new branch 'firstbranch'
```

- Open vi editor and make changes in rect.java file

```
new@DESKTOP-U33LA0K MINGW64 ~/local_repo1 (firstbranch)
$ vi rect.java

new@DESKTOP-U33LA0K MINGW64 ~/local_repo1 (firstbranch)
$ cat rect.java
public class rectangle {
    double a1,b1;
    rectangle(double length, double width)
    {
        a1=length;
        b1=width;
    }

    String check()
    {
        if(a1==b1)
        {
            return "Square";
        }
        else
        {
            return "Not square";
        }
    }
}
```

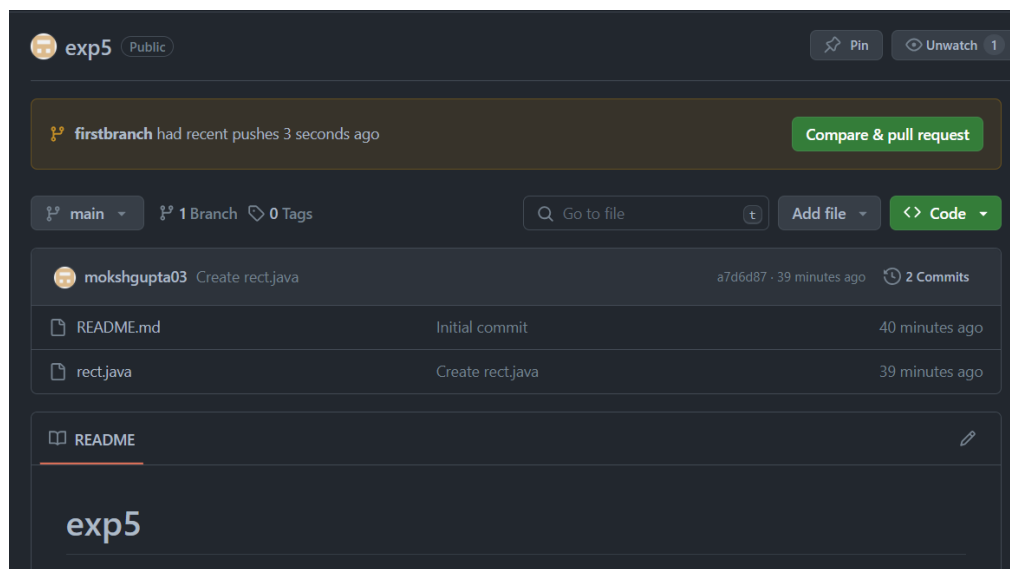
- Add and commit the changes in firstbranch.

```
new@DESKTOP-U33LA0K MINGW64 ~/local_repo1 (firstbranch)
$ git add -A && git commit -m "Committing in firstbranch"
[firstbranch a5abe68] Committing in firstbranch
1 file changed, 4 insertions(+), 4 deletions(-)
```

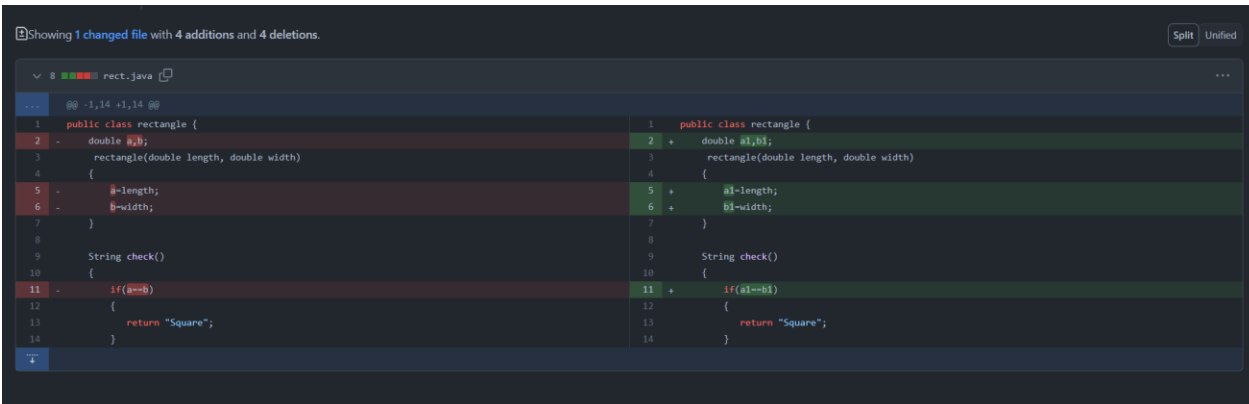
- Push the firstbranch on github.

```
new@DESKTOP-U33LA0K MINGW64 ~/local_repo1 (firstbranch)
$ git push origin firstbranch
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 345 bytes | 345.00 KiB/s, done.
Total 3 (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 'firstbranch' on GitHub by visiting:
remote:   https://github.com/mokshgupta03/exp5/pull/new/firstbranch
remote:
To https://github.com/mokshgupta03/exp5.git
 * [new branch]      firstbranch -> firstbranch
```

- We see compare and pull request on github.



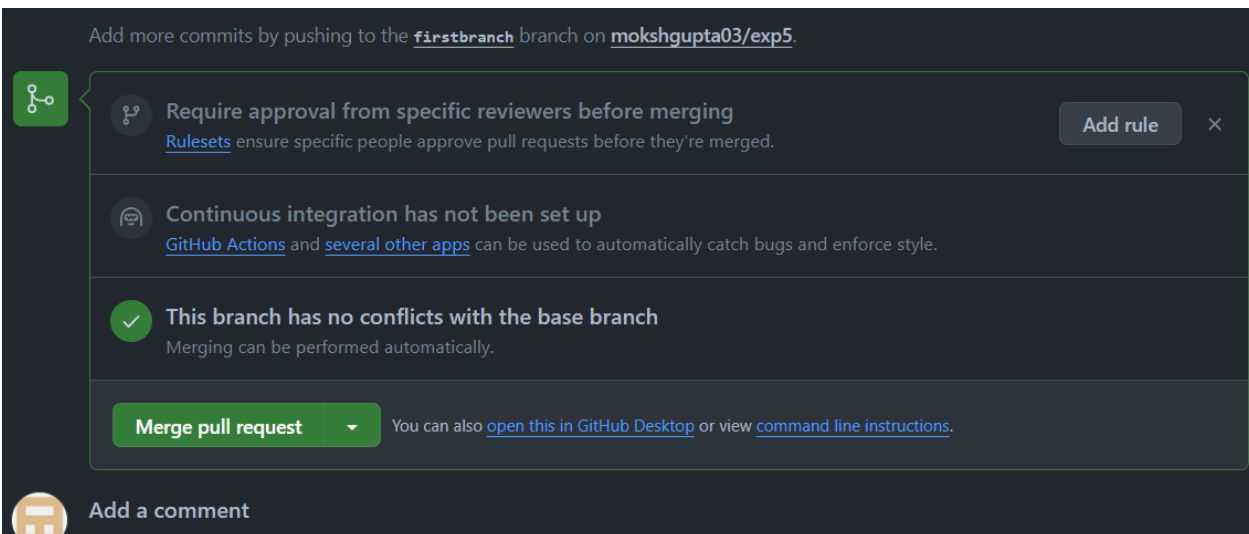
- We can see the difference in the merging files in present in different branches.



Showing 1 changed file with 4 additions and 4 deletions.

File	Line	Content
rect.java	1	@@ -1,14 +1,14 @@
rect.java	2	- double a,b;
rect.java	3	rectangle(double length, double width)
rect.java	4	{
rect.java	5	- a=length;
rect.java	6	- b=width;
rect.java	7	}
rect.java	8	String check()
rect.java	9	{
rect.java	10	{
rect.java	11	- if(a==b)
rect.java	12	{
rect.java	13	return "Square";
rect.java	14	}

- Click on merge pull reuest.



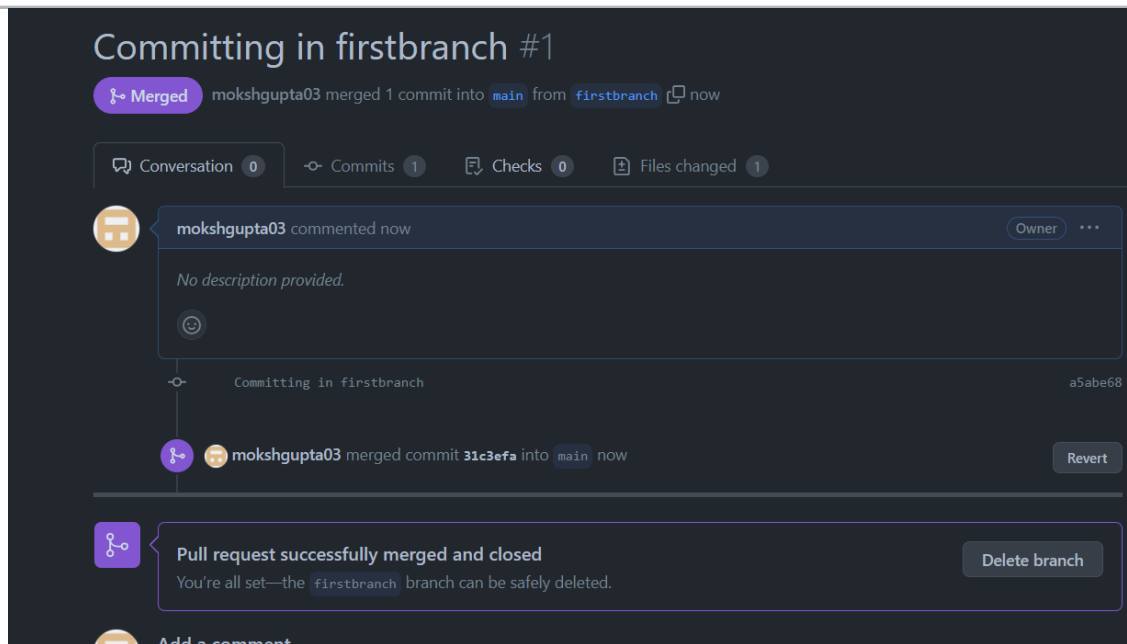
Add more commits by pushing to the [firstbranch](#) branch on [mokshgupta03/exp5](#).

- 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.

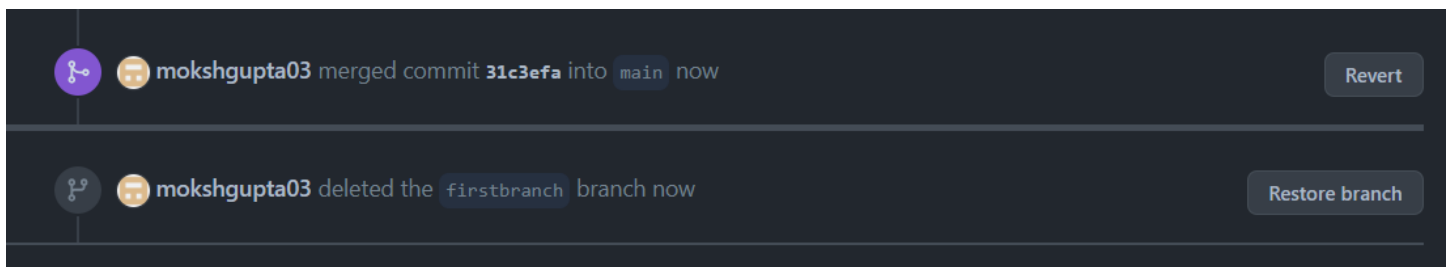
[Merge pull request](#) You can also [open this in GitHub Desktop](#) or view [command line instructions](#).

[Add a comment](#)

- We see that we successfully merged the branches.



- We delete the firstbranch to reduce redundancy.



- We observe that our content is changed in our main branch after merging.

```
mokshgupta03 Committing in firstbranch

Code Blame 21 lines (19 loc) · 309 Bytes Code 55% faster with GitHub Copilot Raw Download Edit View

1 public class rectangle {
2     double a1,b1;
3     rectangle(double length, double width)
4     {
5         a1=length;
6         b1=width;
7     }
8
9     String check()
10    {
11        if(a1==b1)
12        {
13            return "Square";
14        }
15        else
16        {
17            return "Not square";
18        }
19    }
20
21 }
```

- Now pull the main branch in local repository.

```
new@DESKTOP-U33LA0K MINGW64 ~/local_repo1 (main)
$ git pull origin main
From https://github.com/mokshgupta03/exp5
 * branch            main            -> FETCH_HEAD
Updating a7d6d87..31c3efa
Fast-forward
 rect.java | 8 ++++----
 1 file changed, 4 insertions(+), 4 deletions(-)
```

- Create a new branch named branch

```
new@DESKTOP-U33LA0K MINGW64 ~/local_repo1 (main)
$ git checkout -b branch
Switched to a new branch 'branch'
```

- Open the vi editor edit the content and view the content using cat command.

```
new@DESKTOP-U33LA0K MINGW64 ~/local_repo1 (branch)
$ vi rect.java
```

```
new@DESKTOP-U33LA0K MINGW64 ~/local_repo1 (branch)
$ cat rect.java
public class rectangle {
    double a,b;
    rectangle(double length, double width)
    {
        a=length;
        b=width;
    }

    String check()
    {
        if(a==b)
        {
            return "Square";
        }
        else
        {
            return "Not square";
        }
    }
}
```

- Add and commit the changes in branch .

```
new@DESKTOP-U33LA0K MINGW64 ~/local_repo1 (branch)
$ git add rect.java

new@DESKTOP-U33LA0K MINGW64 ~/local_repo1 (branch)
$ git commit -m "committing in branch"
[branch 1140d17] committing in branch
1 file changed, 4 insertions(+), 4 deletions(-)
```

- Checkout to the main branch and use git diff command to observe the differences between both files.



```
new@DESKTOP-U33LA0K MINGW64 ~/local_repo1 (branch)
$ git checkout main
Switched to branch 'main'

new@DESKTOP-U33LA0K MINGW64 ~/local_repo1 (main)
$ git diff branch
diff --git a/rect.java b/rect.java
index e032afe..5d88b83 100644
--- a/rect.java
+++ b/rect.java
@@ -1,14 +1,14 @@
 public class rectangle {
-    double a,b;
+    double a1,b1;
     rectangle(double length, double width)
     {
-        a=length;
-        b=width;
+        a1=length;
+        b1=width;
     }

     String check()
     {
-        if(a==b)
+        if(a1==b1)
         {
             return "Square";
         }
     }
 }
```

- Use git merge two merge two branches.

```
new@DESKTOP-U33LA0K MINGW64 ~/local_repo1 (main)
$ git merge branch
Updating 31c3efa..1140d17
Fast-forward
 rect.java | 8 ++++----
 1 file changed, 4 insertions(+), 4 deletions(-)
```

- Push the main branch on github.

```
new@DESKTOP-U33LA0K MINGW64 ~/local_repo1 (main)
$ git push origin main
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 336 bytes | 336.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/mokshgupta03/exp5.git
 31c3efa..1140d17 main -> main
```

- We can see changes in the main branch.

```

1  public class rectangle {
2      double a,b;
3      rectangle(double length, double width)
4      {
5          a=length;
6          b=width;
7      }
8
9      String check()
10     {
11         if(a==b)
12         {
13             return "Square";
14         }
15         else
16         {
17             return "Not square";
18         }
19     }
20
21 }

```

## Learning outcomes (What I have learnt):

1. Learnt about merging in local and remote repositories.
2. Learnt about push and pull.
3. Learnt about merging.
4. Learnt about using vi command.

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.			