

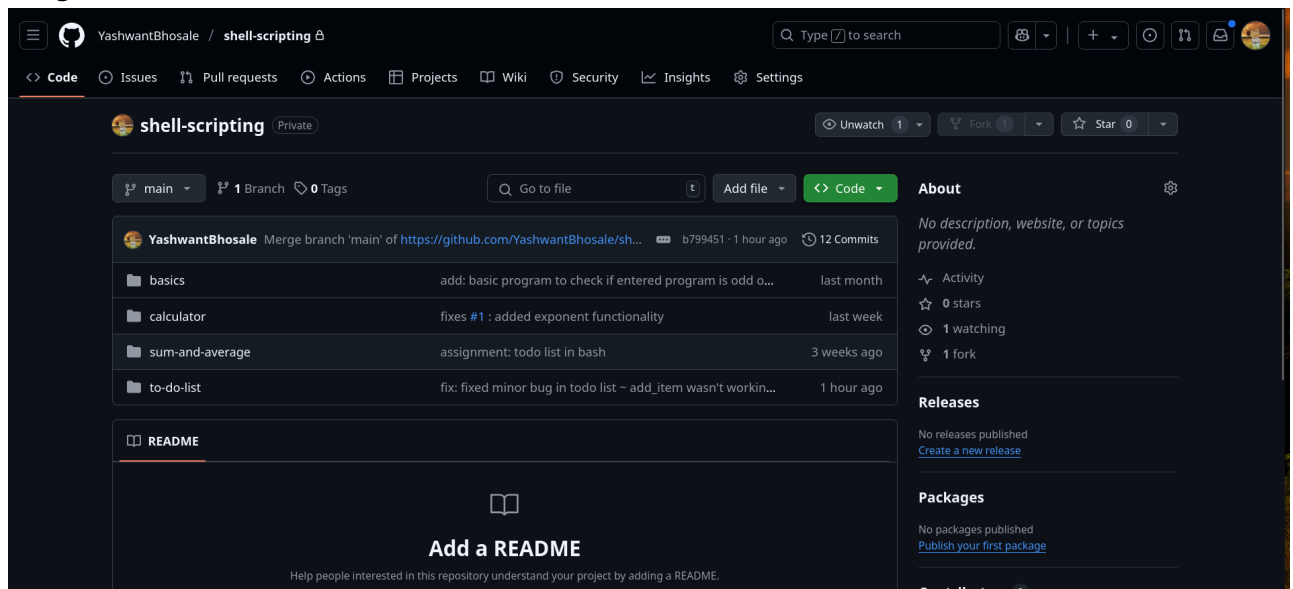
Overview

This assignment was designed to learn how to use Git and GitHub for version control and collaboration on software projects. The assignment involved creating a repository, opening issue, creating branches, making commits, and merging branches. The goal was to understand the basic concepts of Git and GitHub and how they can be used in software development.

Flow

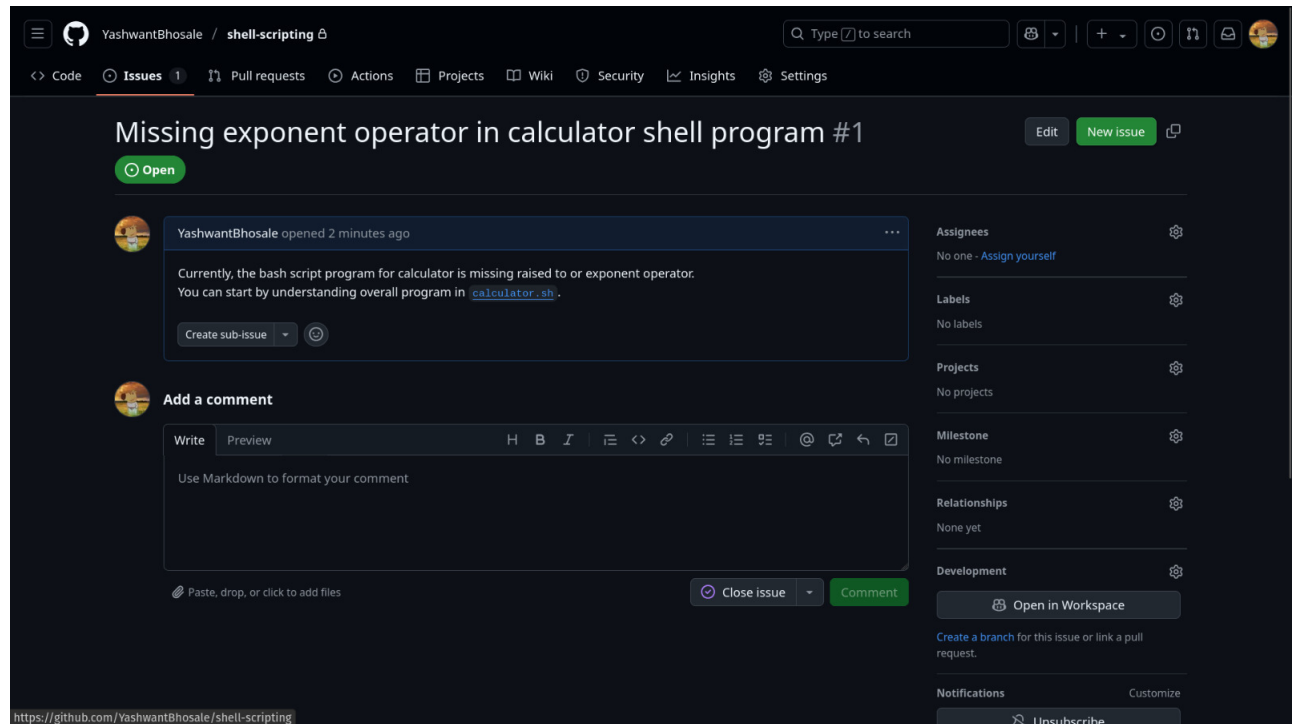
For context, I (username: **YashwantBhosale**) am playing the role of a **Maintainer** and my friend (username: **arymehta**) is playing the role of a **Contributor**. The following steps outline the process we followed to complete the assignment:

1. **Repository:** We could either create a new repository or use an existing one. In this case, we used an existing repository which contained **calculator.sh**- calculator program in shell scripting assignment.

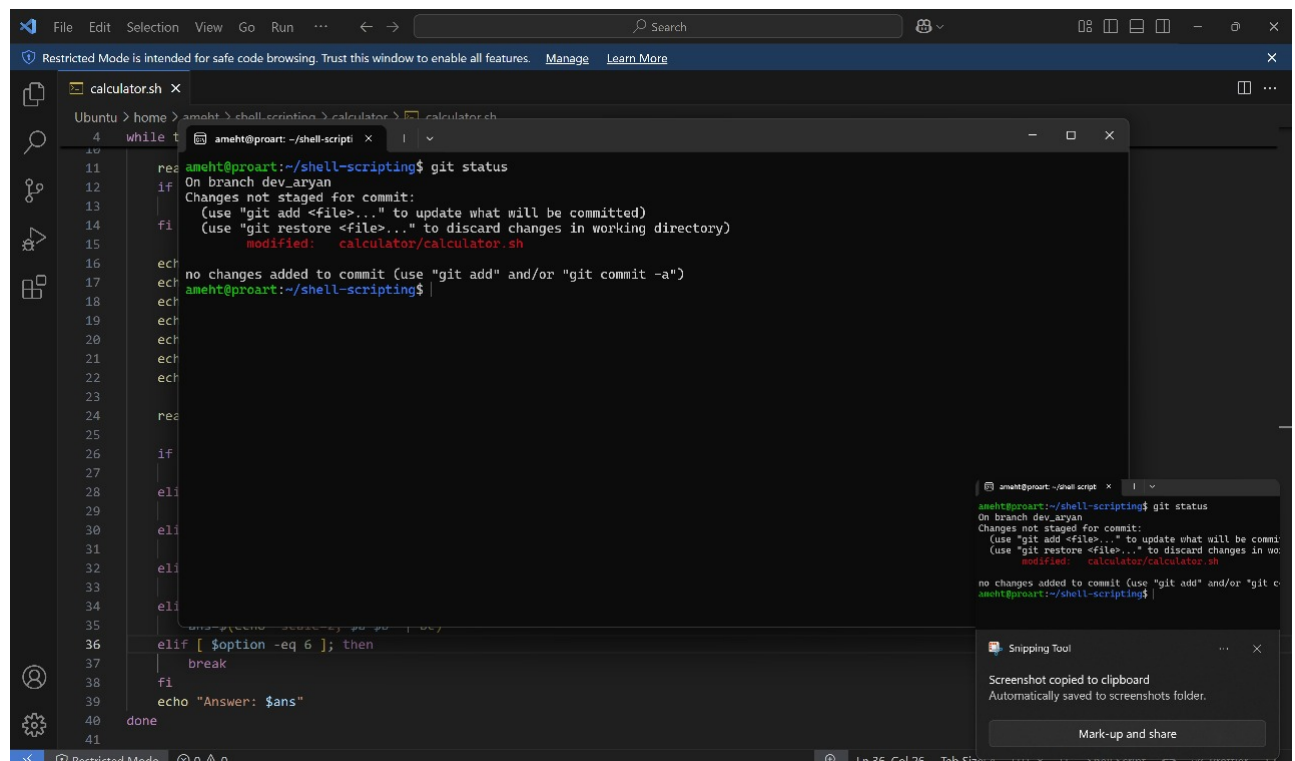


2. **Issue:** I opened an issue in the repository to discuss the changes I wanted to make. The issue was titled "Missing exponent operator in calculator shell program" and described the changes I wanted to

make.

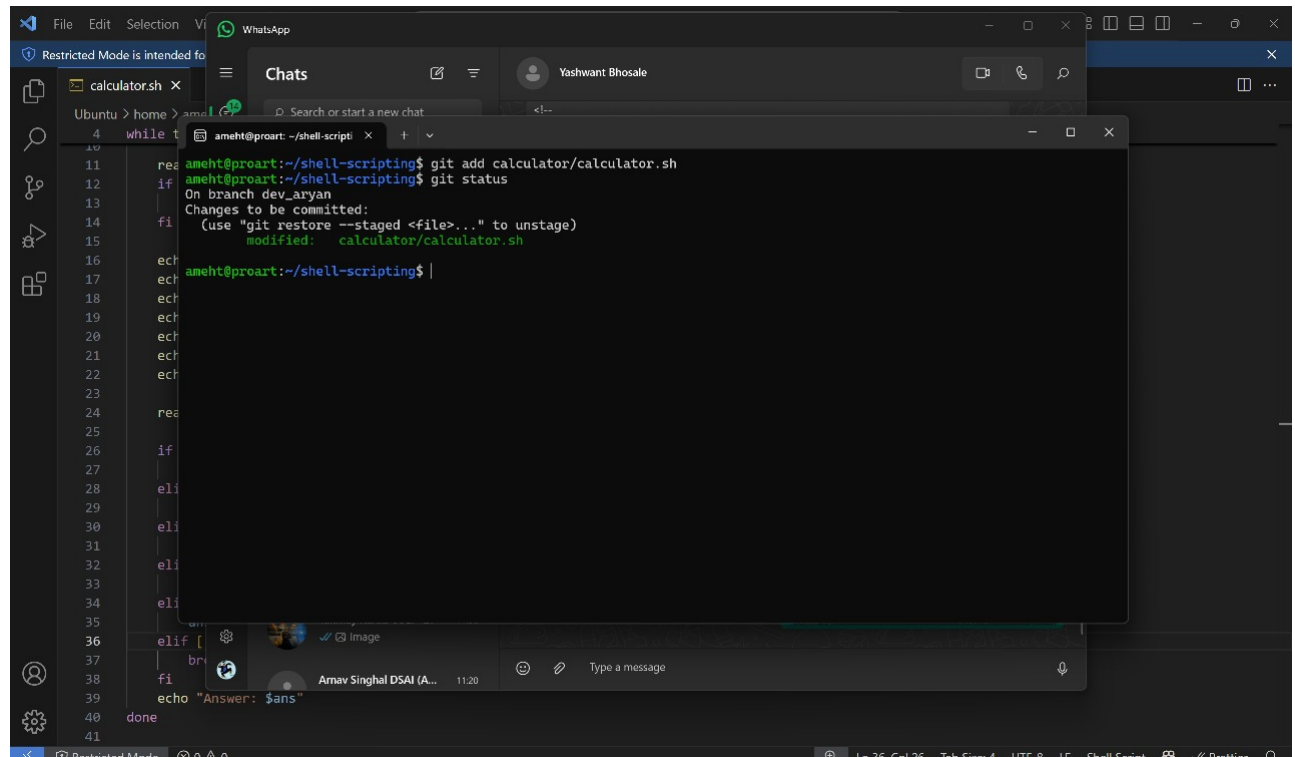


3. **Branch:** My friend **forked** (copied) the repository to their own GitHub account and created a new branch called **dev_aryan** to work on the changes. This allowed them to work on the changes without affecting the main branch of the original repository.
4. **Clone:** They cloned the forked repository to their local machine using the command **git clone <repository-url>**. This allowed them to work on the changes locally.
5. **Check Status:** They checked the status of the repository using the command **git status**. This showed them which files had been modified and which files were untracked.



6. **Add Changes:** They added the changes to the staging area using the command **git add <file-name>**. This allowed them to prepare the changes for commit. and then ran the command **git status** again to check the status of the repository. This showed them that the changes were now in

the staging area and ready to be committed.



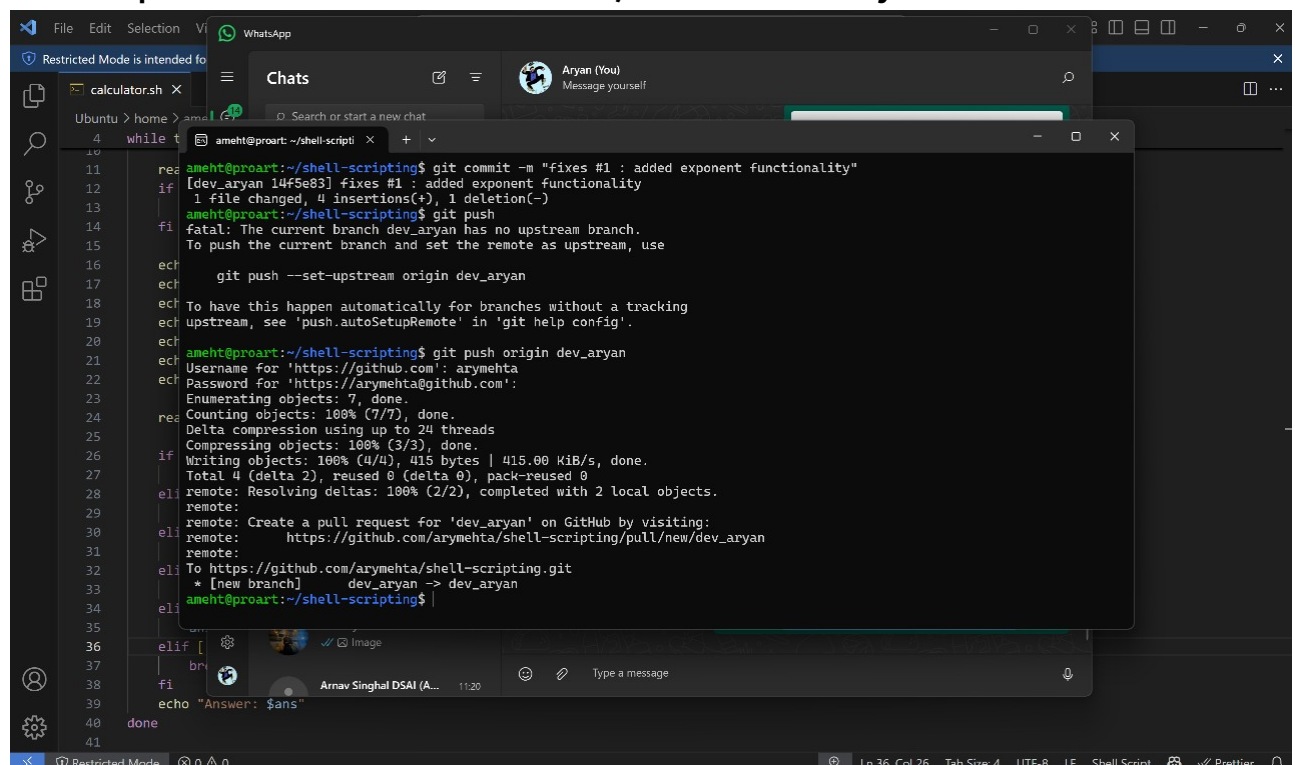
```

ameht@proart:~/shell-scripting$ git add calculator/calculator.sh
ameht@proart:~/shell-scripting$ git status
On branch dev_aryan
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   calculator/calculator.sh

ameht@proart:~/shell-scripting$

```

7. **Commit Changes:** They committed the changes using the command `git commit -m "<commit-message>"`. This created a new commit with the changes they made and added a message describing the changes. then they pushed the changes to the remote repository using the command `git push origin <branch-name>`. This uploaded the changes to the forked repository on GitHub. **Note: Here, we carefully used fixes #1 in the commit message this is gitHub feature. When we push the commit to the main branch, it will automatically close the issue.**



```

ameht@proart:~/shell-scripting$ git commit -m "fixes #1: added exponent functionality"
[dev_aryan 14f5e83] fixes #1: added exponent functionality
1 file changed, 4 insertions(+), 1 deletion(-)
ameht@proart:~/shell-scripting$ git push
fatal: The current branch dev_aryan has no upstream branch.
To push the current branch and set the remote as upstream, use

    git push --set-upstream origin dev_aryan

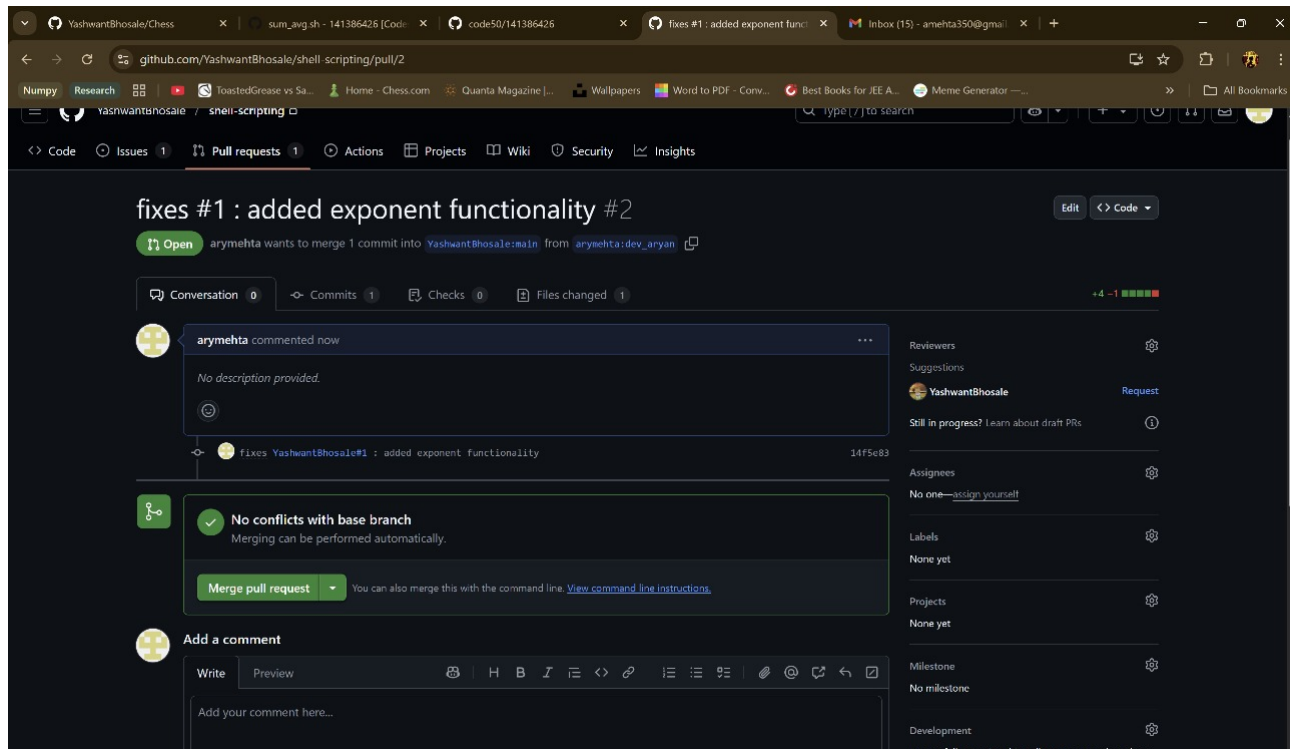
To have this happen automatically for branches without a tracking
upstream, see 'push.autoSetUpRemote' in 'git help config'.

ameht@proart:~/shell-scripting$ git push origin dev_aryan
Username for 'https://github.com': arymehta
Password for 'https://arymehta@github.com':
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 24 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 415 bytes | 415.00 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.
remote: Create a pull request for 'dev_aryan' on GitHub by visiting:
remote:   https://github.com/arymehta/shell-scripting/pull/new/dev_aryan
remote:
To https://github.com/arymehta/shell-scripting.git
 * [new branch] dev_aryan -> dev_aryan
ameht@proart:~/shell-scripting$

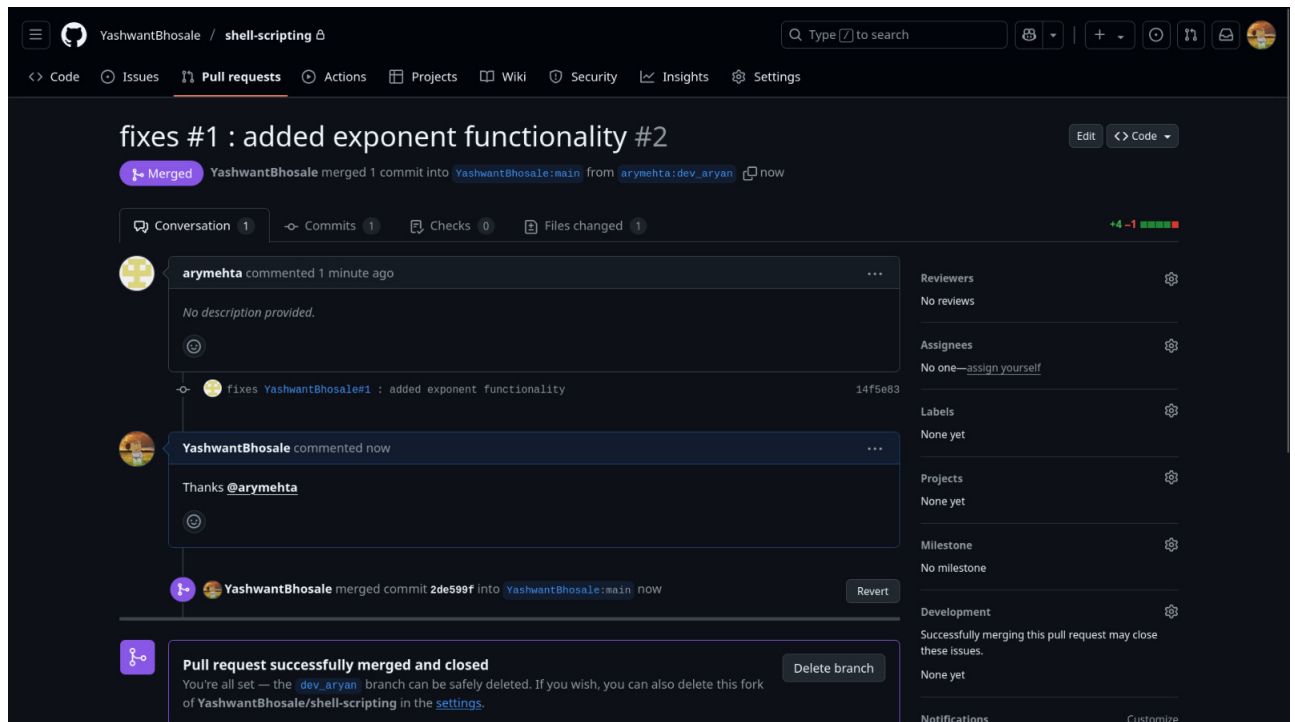
```

8. **Pull Request:** After pushing the changes, they opened a pull request in the original repository. This allowed them to propose the changes to the original repository and request that the changes be merged into the main branch. They provided a description of the changes and linked to the issue that

was fixed by the changes.



9. **Merge:** After the pull request was reviewed and approved, the changes were merged into the main branch of the original repository. This allowed the changes to be incorporated into the main codebase.



10. **Check status:** After merging, changes were pulled from the original repository to the local machine using the command `git pull origin <branch-name>`. This updated the local repository with

the latest changes from the original repository.

```

ameht@proart:~/shell-scripting$ git pull
Username for 'https://github.com': arymehta
Password for 'https://arymehta@github.com':
remote: Enumerating objects: 1, done.
remote: Counting objects: 100% (1/1), done.
remote: Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (1/1), 927 bytes | 927.00 KiB/s, done.
From https://github.com/arymehta/shell-scripting
  1867b4b..2de599f  main       -> origin/main
Updating 1867b4b..2de599f
Fast-forward
 calculator/calculator.sh | 5 ++++
 1 file changed, 4 insertions(+), 1 deletion(-)
ameht@proart:~/shell-scripting$

```

11. **Confirm commits:** Finally, they checked the commit history using the command `git log --oneLine` to confirm that the changes were successfully merged into the main branch. This showed them a list of all commits in the repository, including the new commit that was just merged.

```

ameht@proart:~/shell-scripting$ git log --oneLine
2de599f (HEAD -> main, origin/main, origin/HEAD) Merge pull request #2 from arymehta/dev_aryan
14f5e83 (origin/dev_aryan, dev_aryan) fixes #1 : added exponent functionality
1867b4b fix: fixed calculator
61f9884 assignment: todo list in bash
ed8a89d add: Sum and average program in bash
ff64baa add: calculator program in bash
dadcaa9 add: basic program to check if entered program is odd or even
ab986e2 add: loops
e1a4117 add: basic echo command script
908d4d6 init: initialized repository
ameht@proart:~/shell-scripting$

```

12. **git diff:** For simple demonstration, we added a comment in the `calculator.sh` file and then used the command `git diff` to see the changes made in the file. This showed them a side-by-side comparison of the changes made to the file, highlighting the differences between the original and modified versions.

```

ameht@proart:~/shell-scripting$ git diff
diff --git a/calculator/calculator.sh b/calculator/calculator.sh
index 317b12e..fb87d73 180755
--- a/calculator/calculator.sh
+++ b/calculator/calculator.sh
@@ -1,6 +1,7 @@
#!/usr/bin/bash
ans=0

+# This is a cool project
while true; do
  echo "Enter two numbers"
  read a
ameht@proart:~/shell-scripting$

```

Student Information

- **Name:** Bhosale Yashwant Chandrakant
- **MIS:** 612303039
- **Batch:** S2 (Division 1)