**AMITY UNIVERSITY**

**SOURCE CODE MANAGEMENT**

**Name: MITHUN K**

**Course: Source code management**

**School: ASET B. tech CSE**

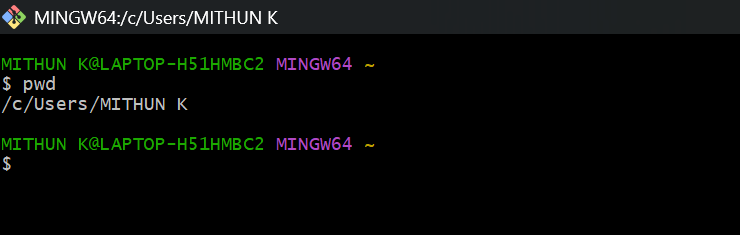
**Assignment: Lab manual including all the experiments**

**Course instructor: Dr. Monith Kapoor**

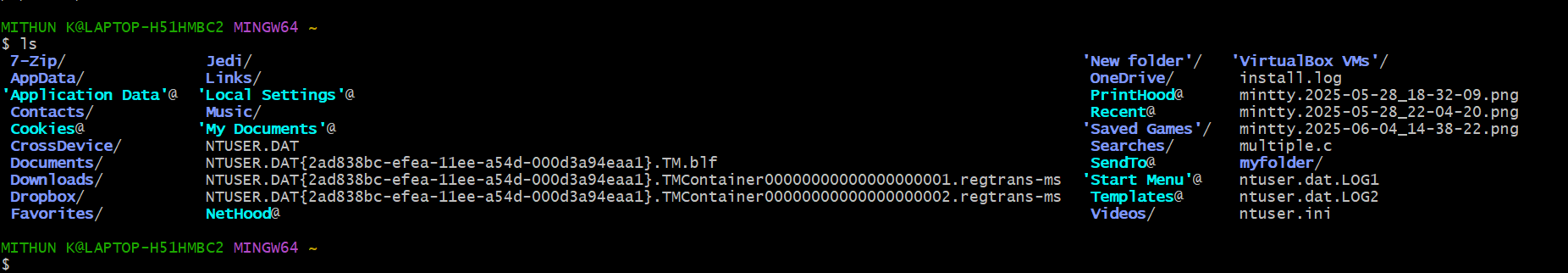
**Date: 05/06/2025**

**EXPERIMENT 1: BASICS OF LINUX**

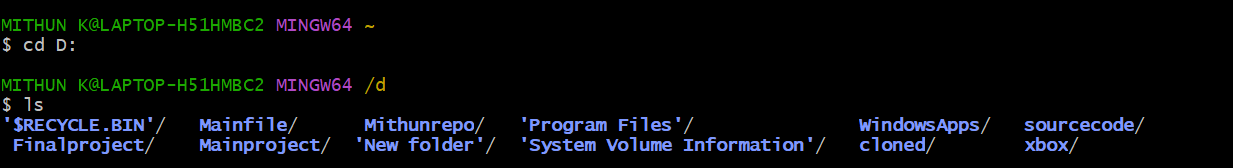
* **The command pwd stands for "print working directory" to display the absolute path of the current directory you are in**

****

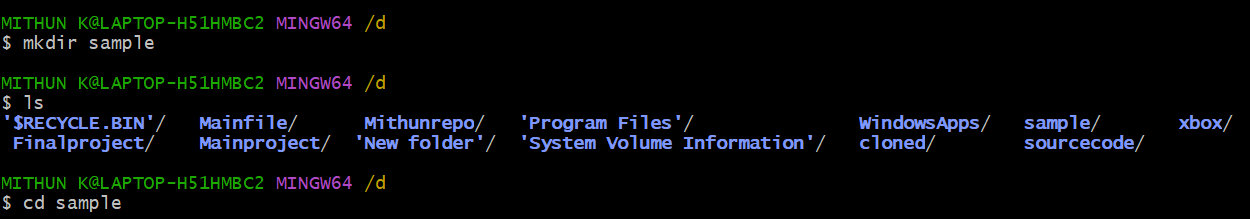
* **ls, which is short for "list" and is used to display the contents of a directory**

****

* **changing directory**

**now we have changed our location form c to d drive**

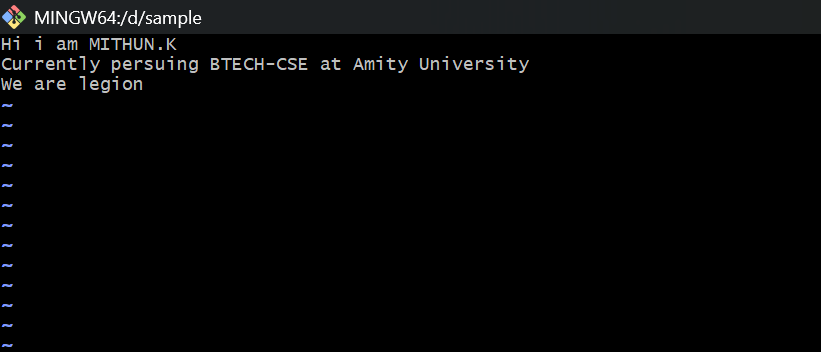
* **The mkdir command stands for "make directory" and is used to create new folders (directories) in your file system**

****

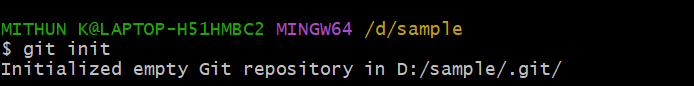
* **“vi”  used to create, view, and edit text files directly from the terminal.**

****

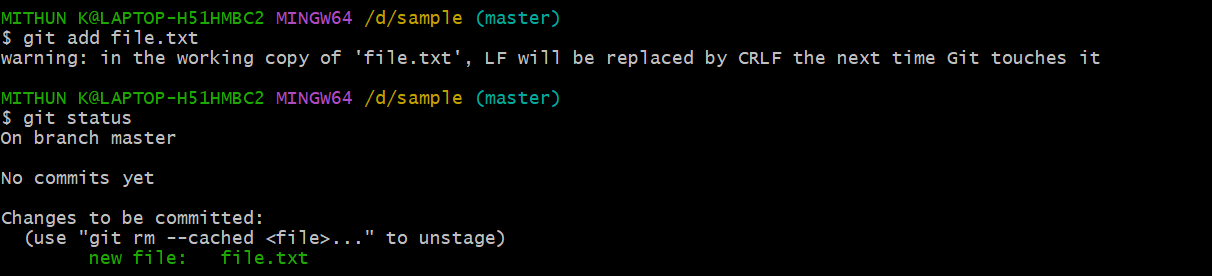
* **This is how the terminal looks like**

****

* **Now we have to initialize an empty git repo in our local device.**

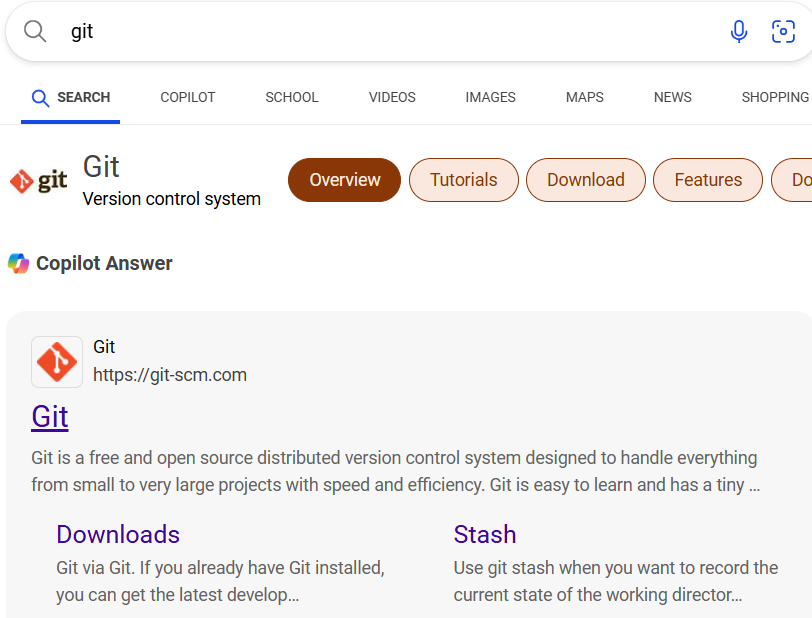
****

* **Now adding the file into it**
* **Also we will use the command “git status” to check the status of the file.**

****

**EXPERIMENT 2: INTRODUCTION-GITBASH**

**Step1: Search Git in your browser and click on the first link ‘Git’**

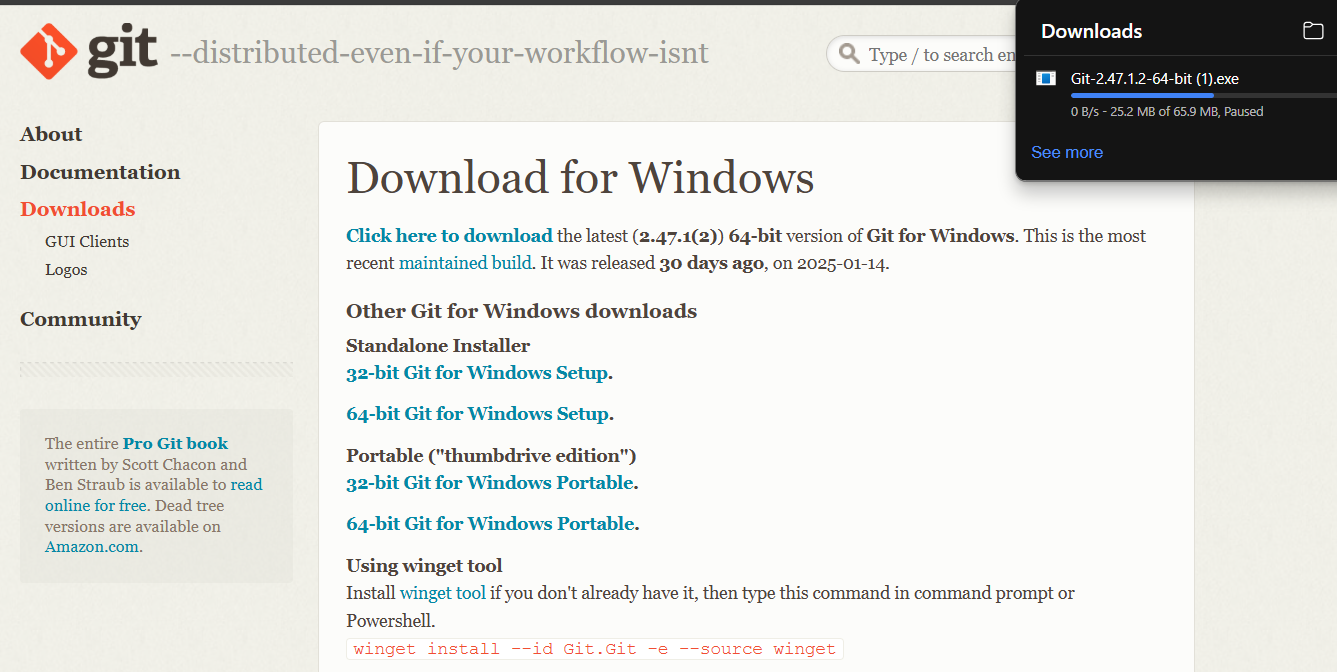
****



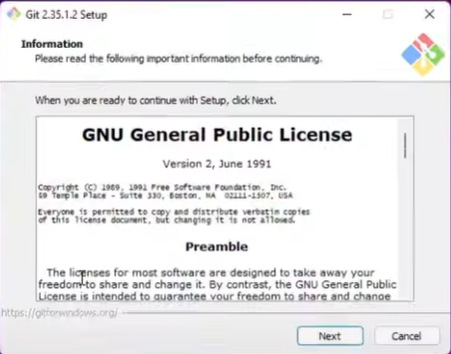
**Step2: Click on ‘Download for windows’**



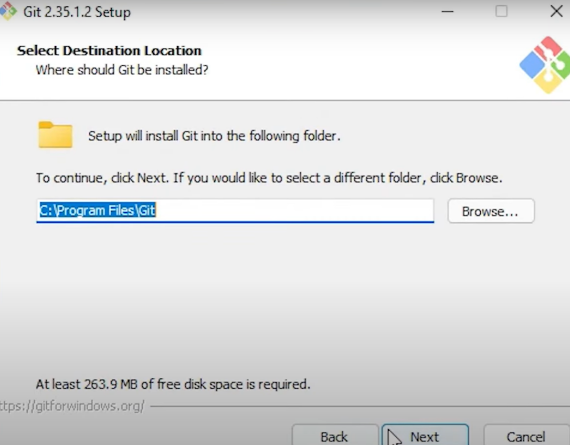
**Step3: Download git (according to your device specifications) click on ’64-bit Git foe windows setup’**

****

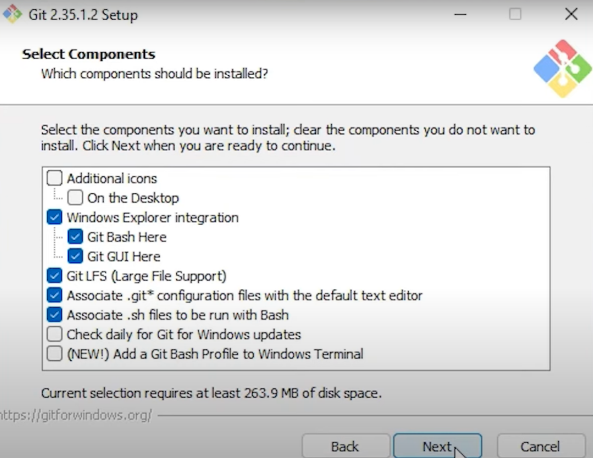
**Step4: Open git from files and click on Next**

****

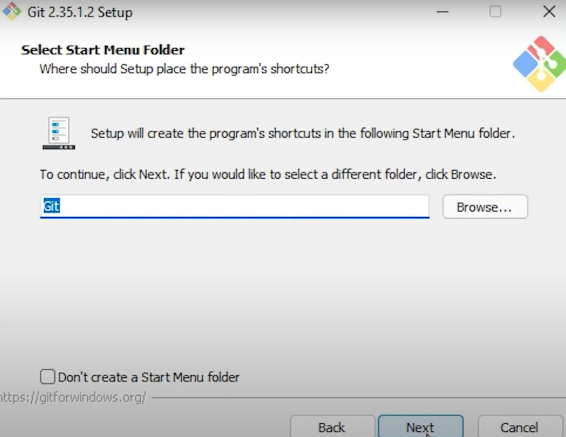
**Step5: click next**

****

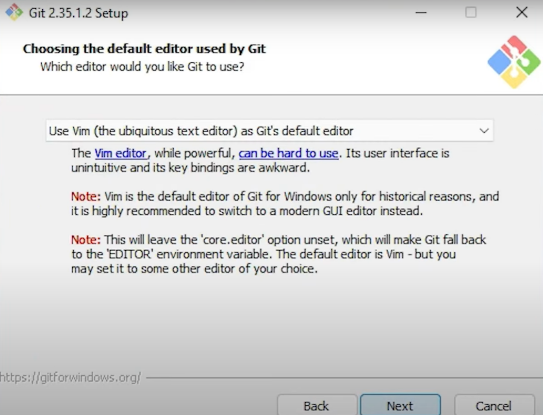
**Step6: click next**

****

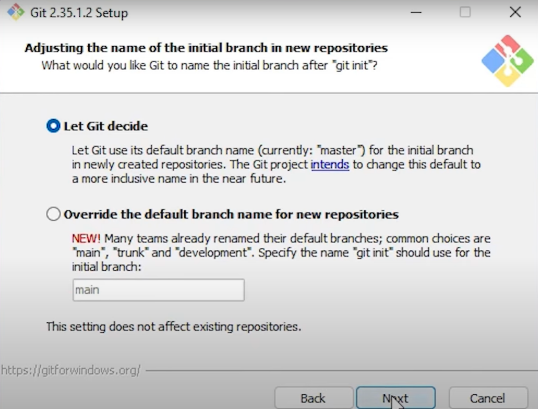
**Step7: Click next**

****

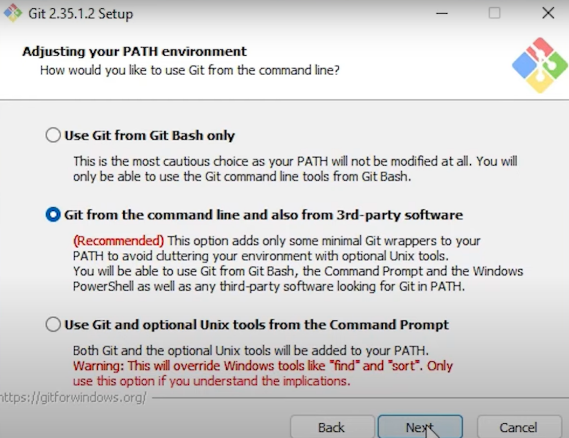
**Step8: click next**

****

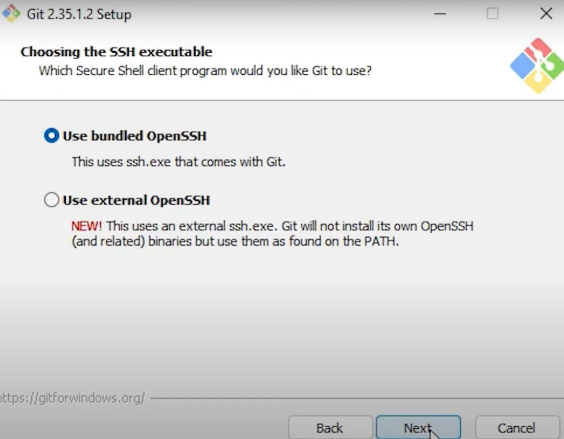
**Step9: Click next**

****

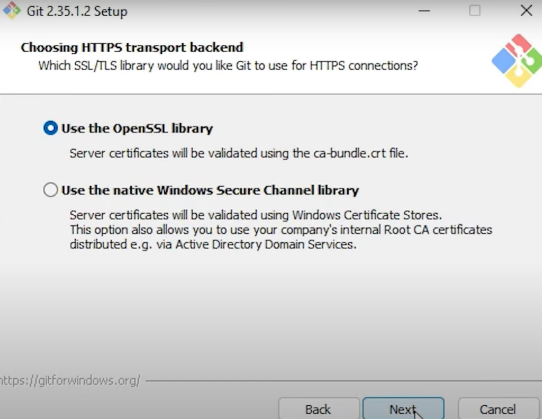
**Step10: Click next**

****

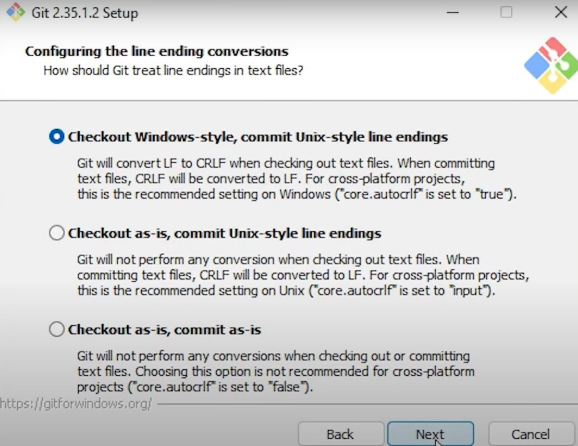
**Step11: Click next**

****

**Step12: Click next**

****

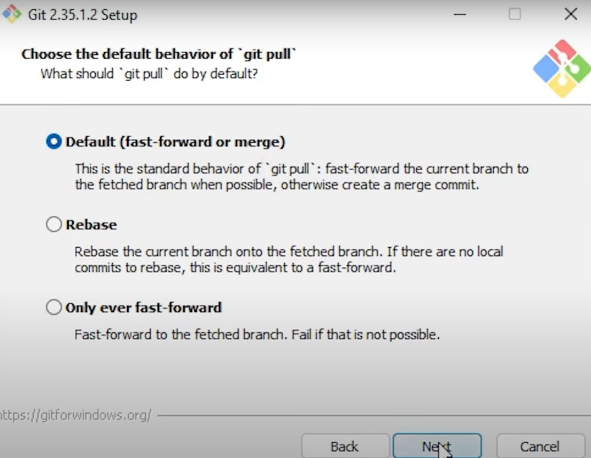
**Step13: Click next**

****

**Step14: Click next**

****

**Step15: Click next**

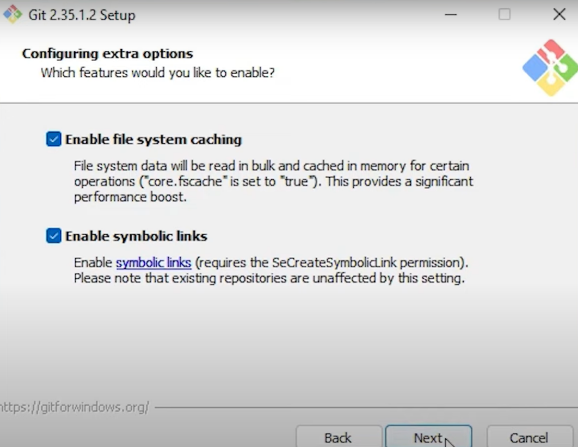
****

**Step16: Click next**

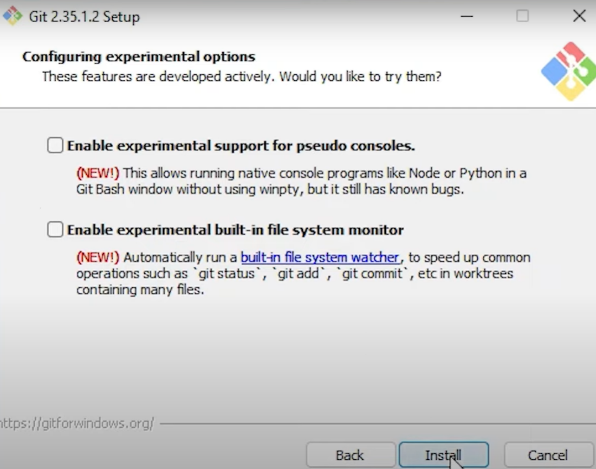
**A screenshot of a computer

AI-generated content may be incorrect.**

**Step17: Click next**

****

**Stpe18: Click Install**

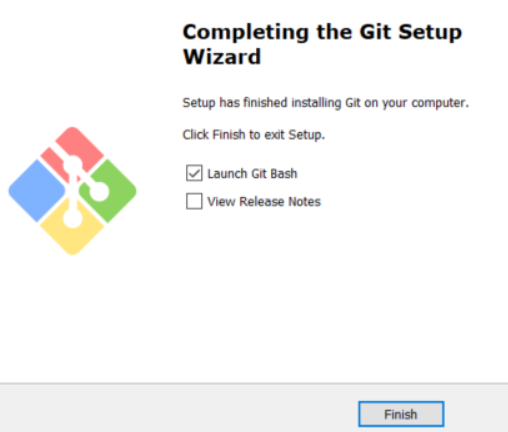
****

**Installing**

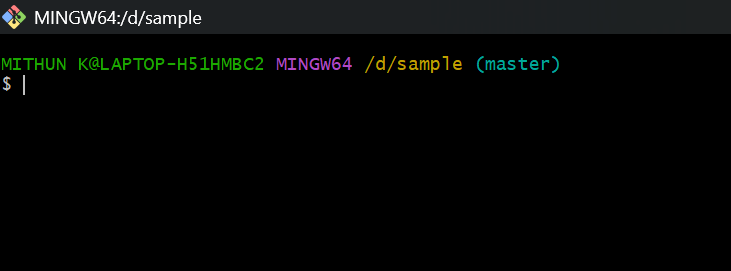
**A screenshot of a computer

AI-generated content may be incorrect.**

**Step19: Select ‘Launch Git Bash’ and click ‘Finish’**

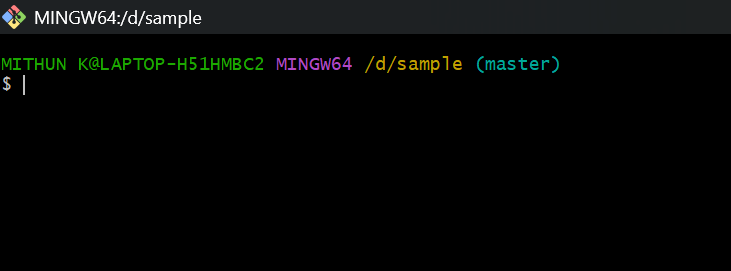
****

**Step20: Interface**

****

**EXPERIMENT 3: GITBASH AND GITHUB**

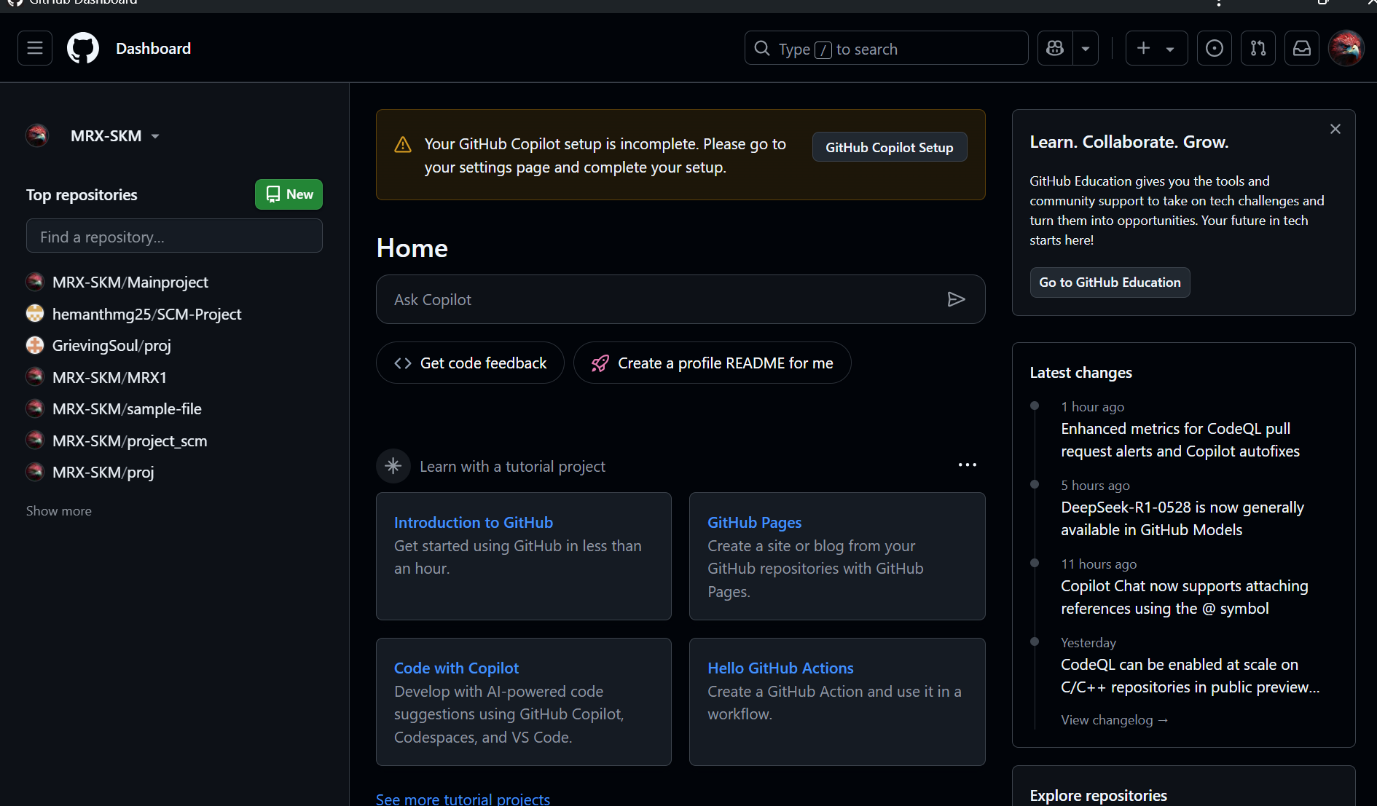
* **Terminal of gitbash**

****

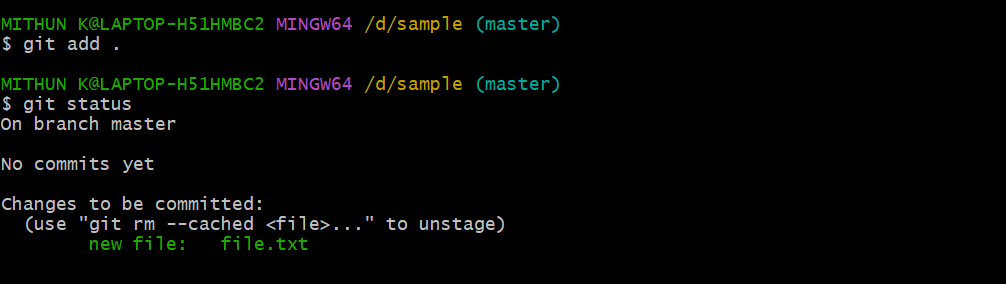
* **This is the terminal where we use our linux commands to create, red, edit, add, commit, and push our files into github.**

**Github interface**

**To the left top corner we see a “+” symbol, from there we can create a repository in github. To this repo only we will add our files and projects from git bash**

****

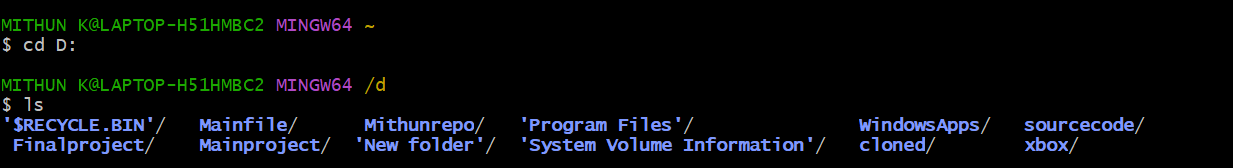
* **Connecting our local repository to github repositories.**

****

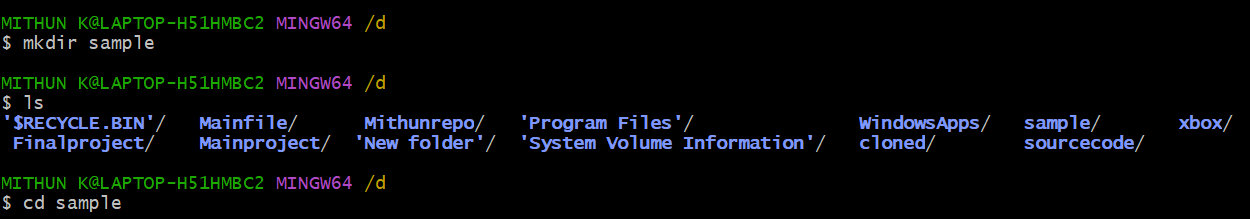
* **We use “git remote add origin <repo\_url>” to connect our local repo to github repo.**

**EXPERIMENT 4: FILE CREATION WITH COMMIT AND PUSH COMMAND.**

* **we will first open our directory and create a local repo.**
* **Mkdir(make directory) to create a local folder.**
* **Then use cd(change directory command) and get into it.**

****

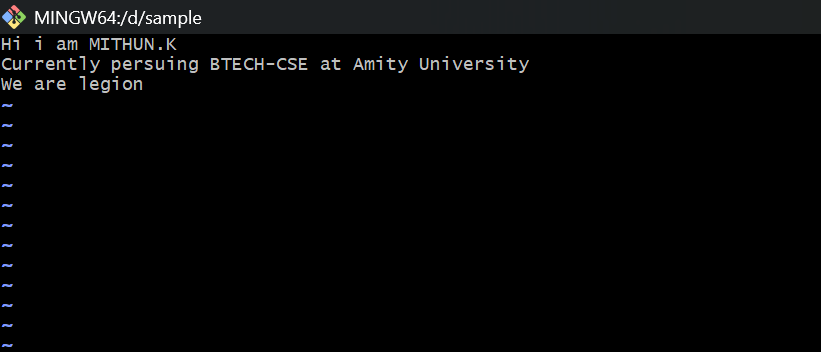
* **“vi” is used to create a file in the folder that we have created.**

****

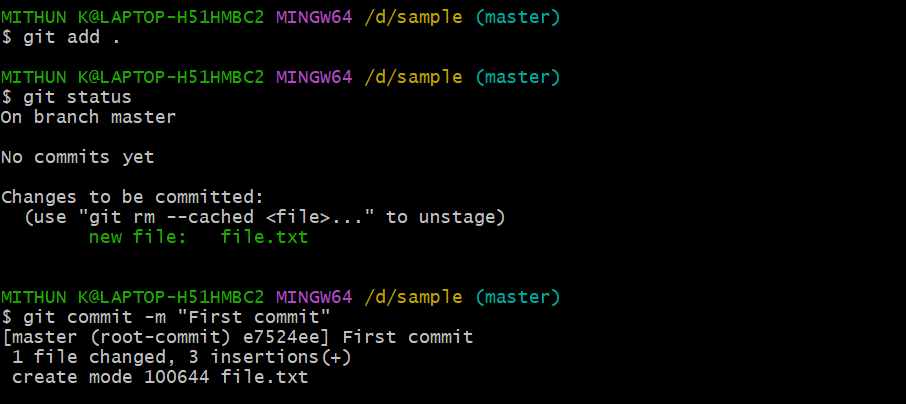
**A computer screen shot of a black screen

AI-generated content may be incorrect.**

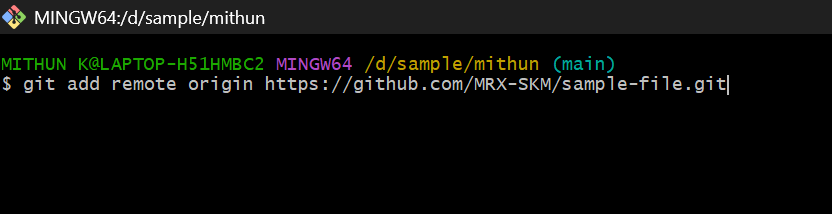
* **In this terminal we will include all the info that needs to be put into the file.**

****

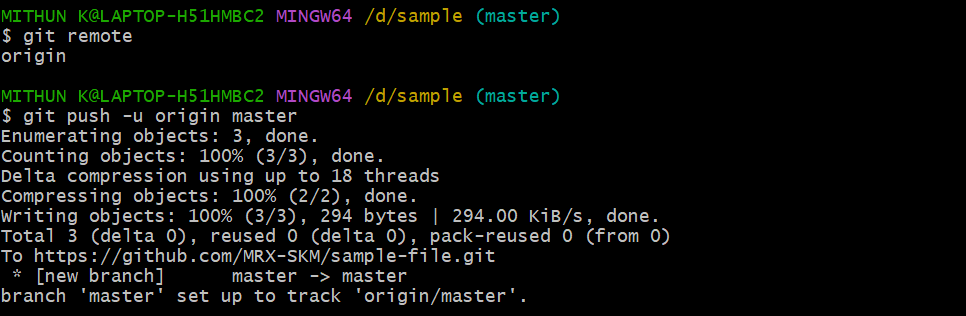
* **Git init is used to initialize an empty git repo**
* **After creating and writing into the file we will add it to empty repo using “git add “file\_name” command.**
* **Then we will check the status of the file using “git status” command.**

****

* **After adding the file we will add a commit using “git commit -m “comment” command.**

****

* **Now we will connect our local repo with our github using “git remote add origin “repo-url” command.**

****

* **Now we will push our files into github repo using “ git push -u origin “branch\_name” command.**

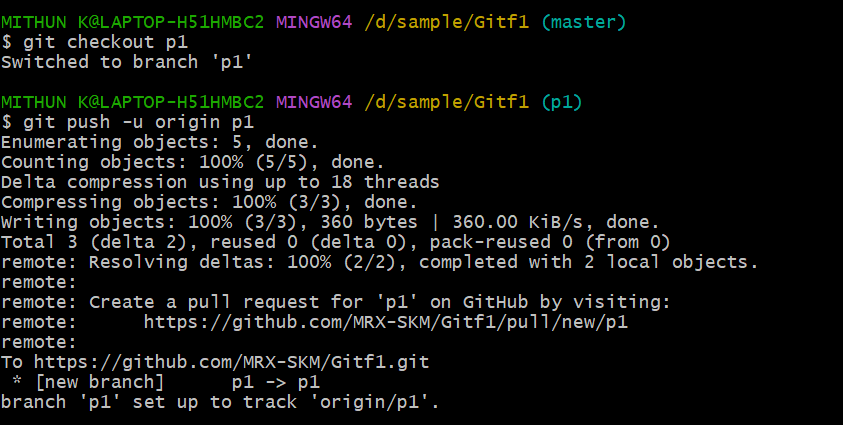
**EXPERIMENT 5: BRANCH CREATION**

* **To create a new branch using git we will use the command**

**# git branch “new branch name”**

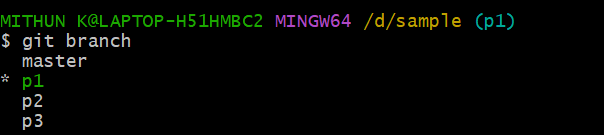
* **To move from one branch to another we use**

**# git checkout “branch\_name”**



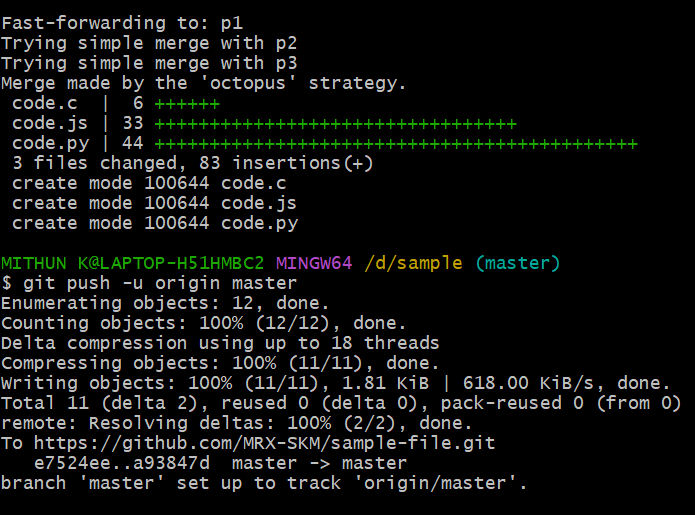
* **To check number of branches we use**

**# git branch command**

****

**EXPERIMENT 6: MERGE REQUEST**

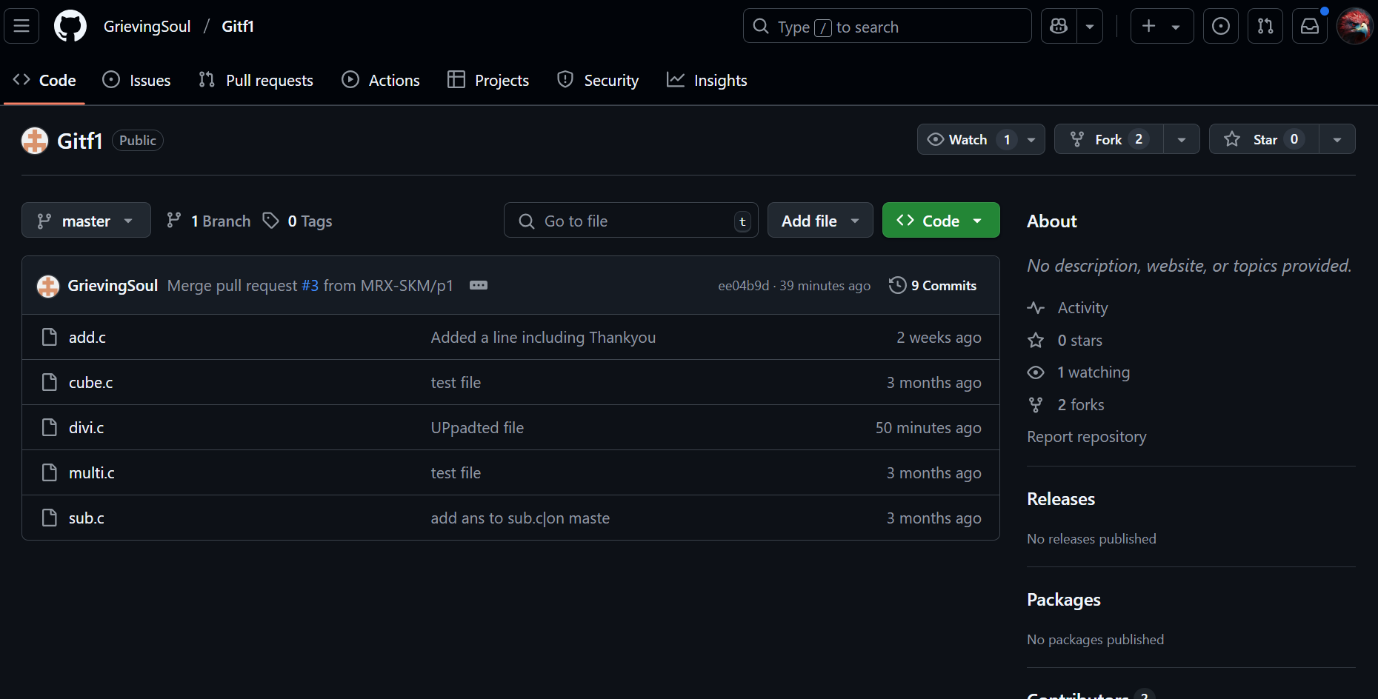
* **For using merge we will first get into the branch to which we have to merge the other branch**

****

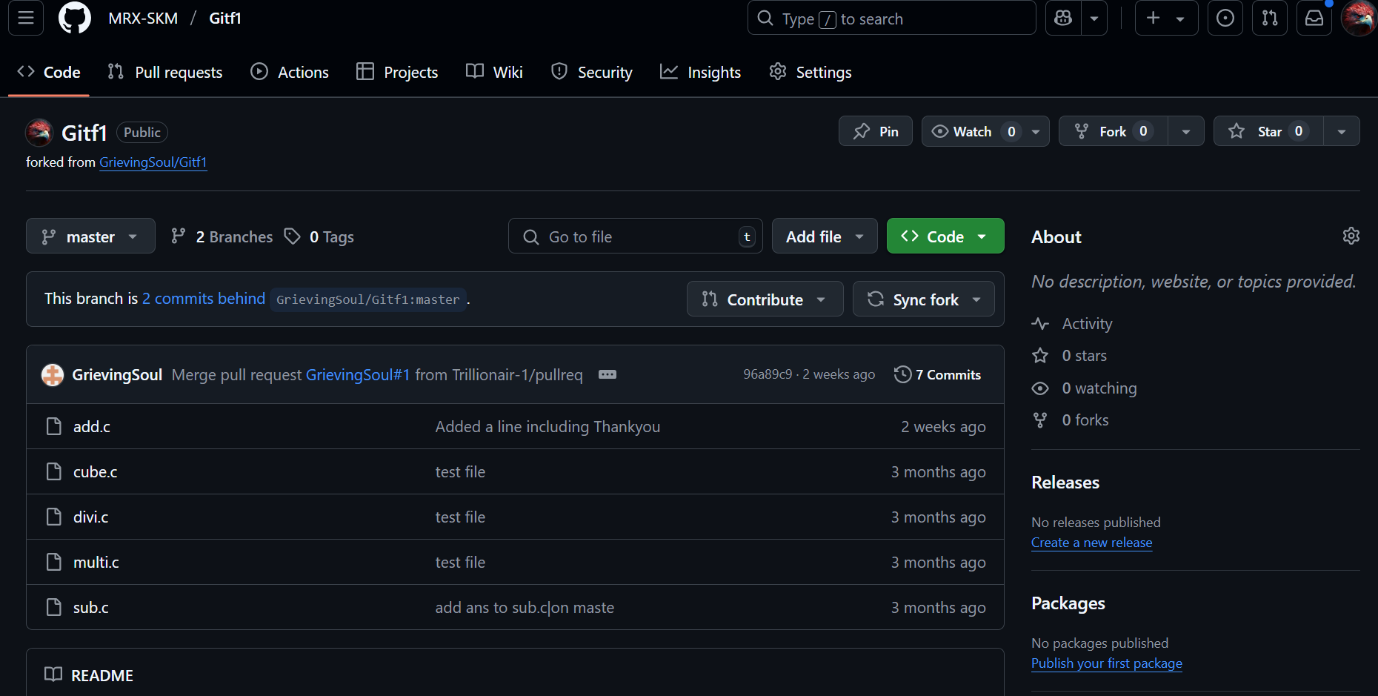
* **Then use the command “git merger <branch\_name>”. Enter the name of the branch that needs to be merged to the master branch.**
* **Then use git push command to complete the merge request.**

**EXPERIMENT 7: OPEN AND CLOSE PULL REQUEST**

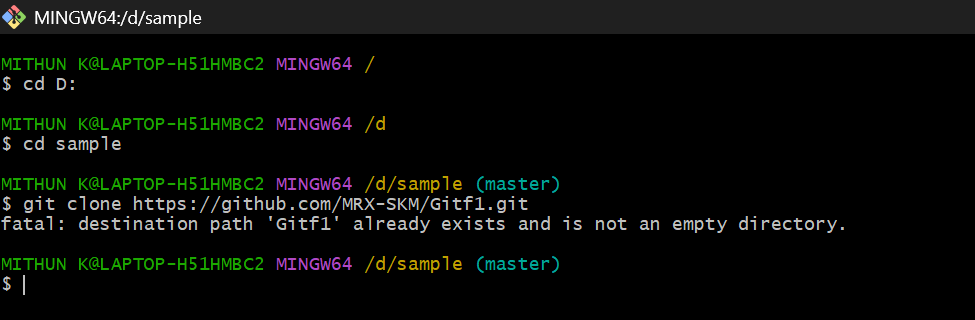
* **Search for the repo for which you need to send pull request on github**

****

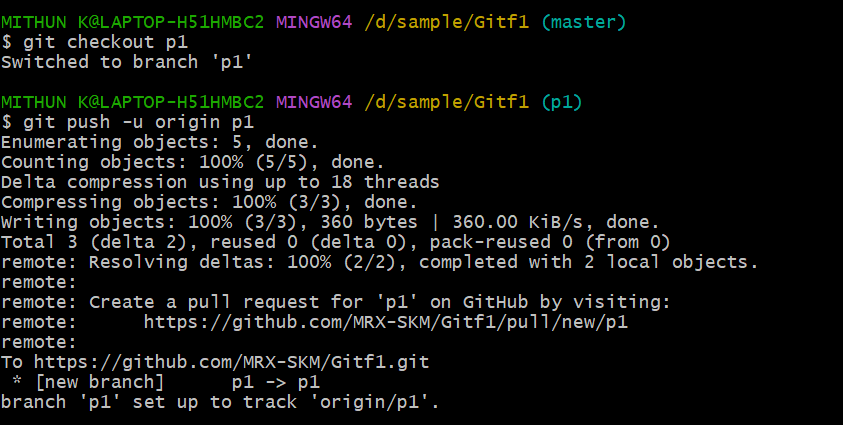
* **Then look for fork option on top left corner, and fork the repo.**

****

* **Now copy the URL of this forked repo.**
* **We will use the command “git clone” to clone the git repo into local repo.**

****

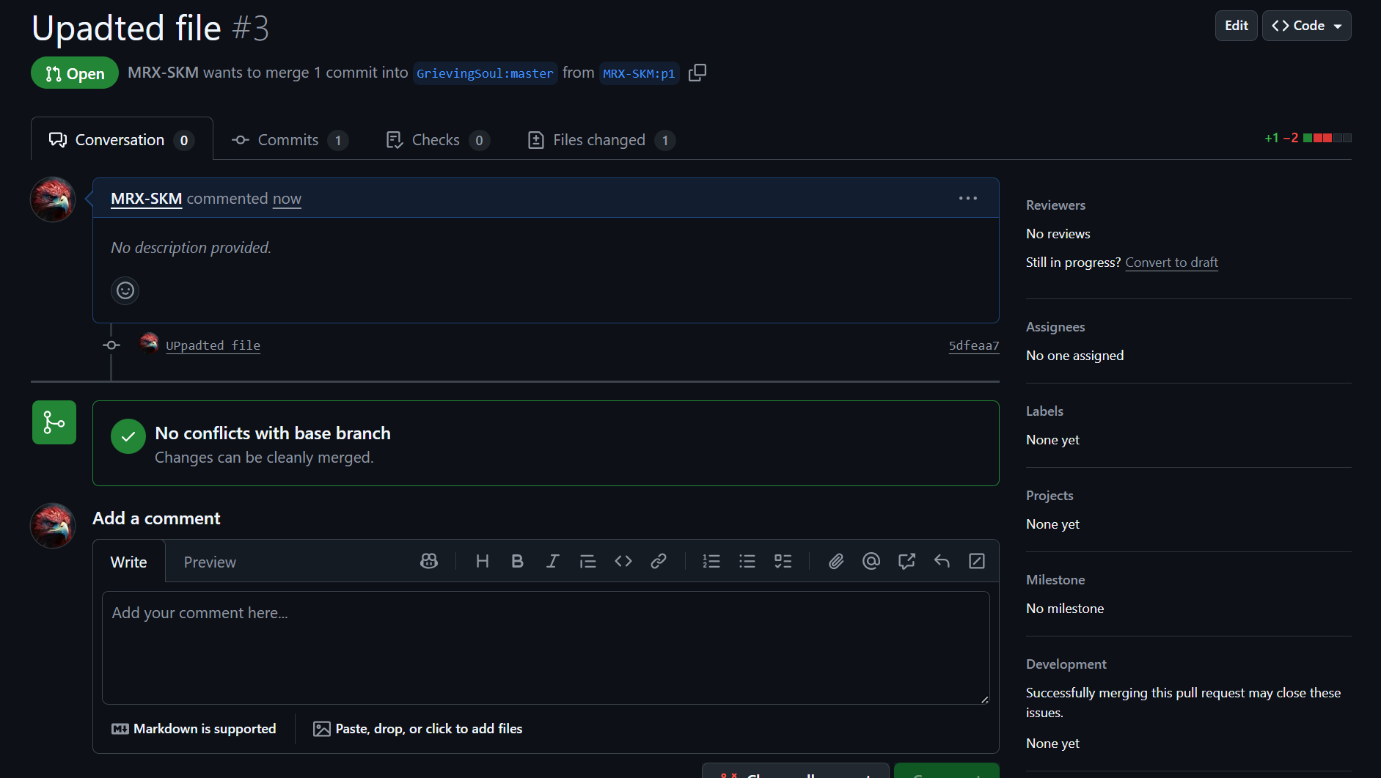
* **Then we will navigate into the forked repo, create a new branch , edit the file , commit and push it.**

****

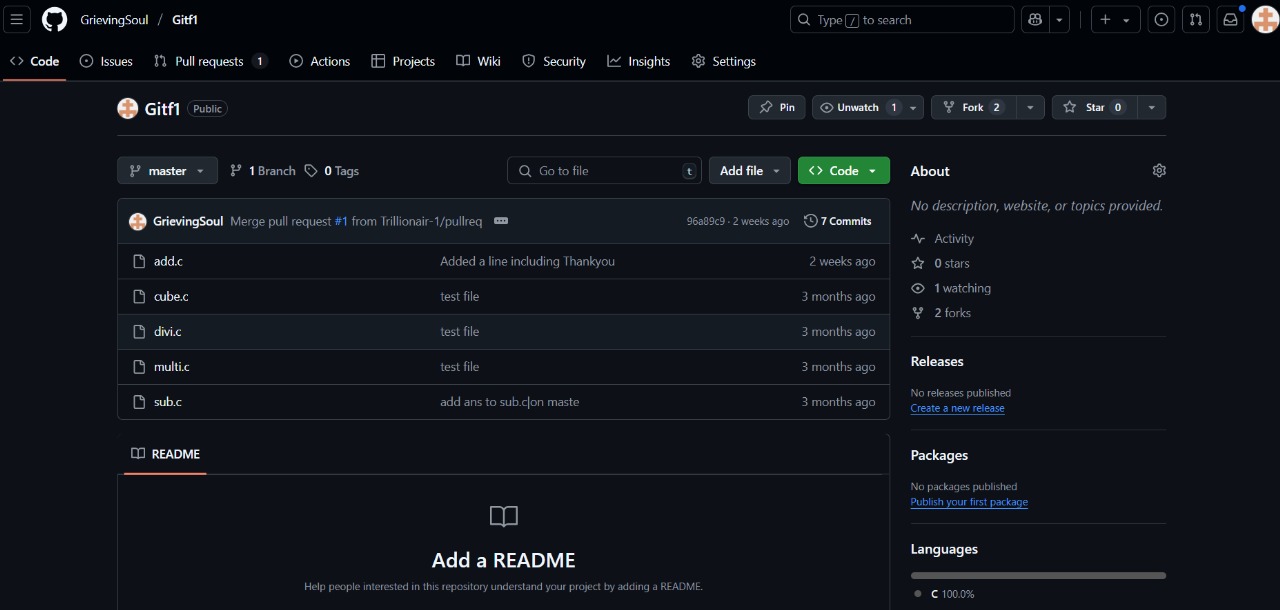
* **After pushing , go to the github and on top psnel there will be a “pull request” option. Click on it.**



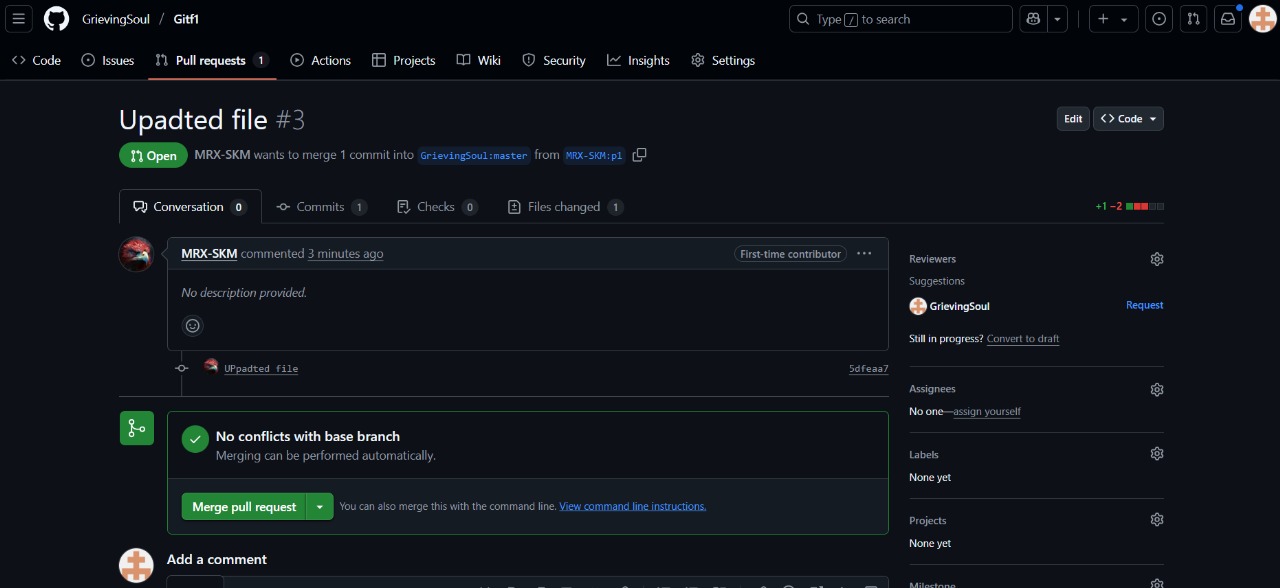
* **The above page will open. Now select your base and head-repos, add comments if necessary and click on create pull request.**
* **After creating pull request this page will appear.**



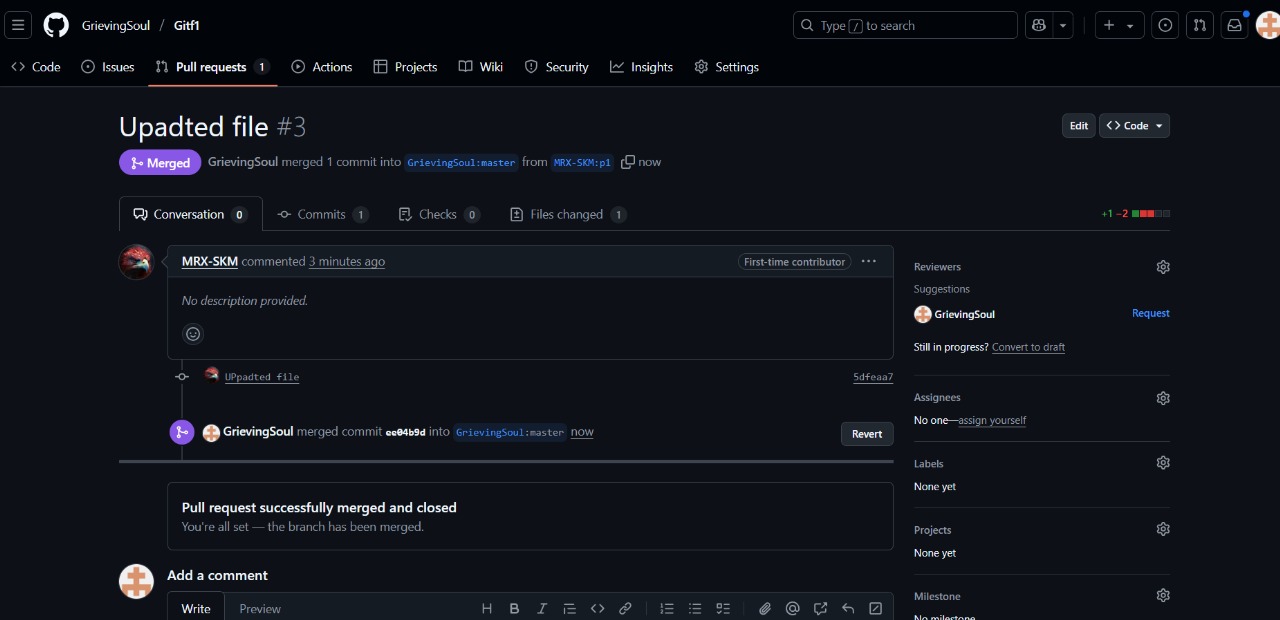
* **The receiver will have got a pull request notification.**



* **After clicking on it…**
* **After scrolling down the receiver will have an option of “merge and pull”.**



* **Click on that and close the pull request**



* **After clicking of merge and pull , edited code will get merged into the receivers repo.**
* **This is all about pull request.**

**THANK YOU**