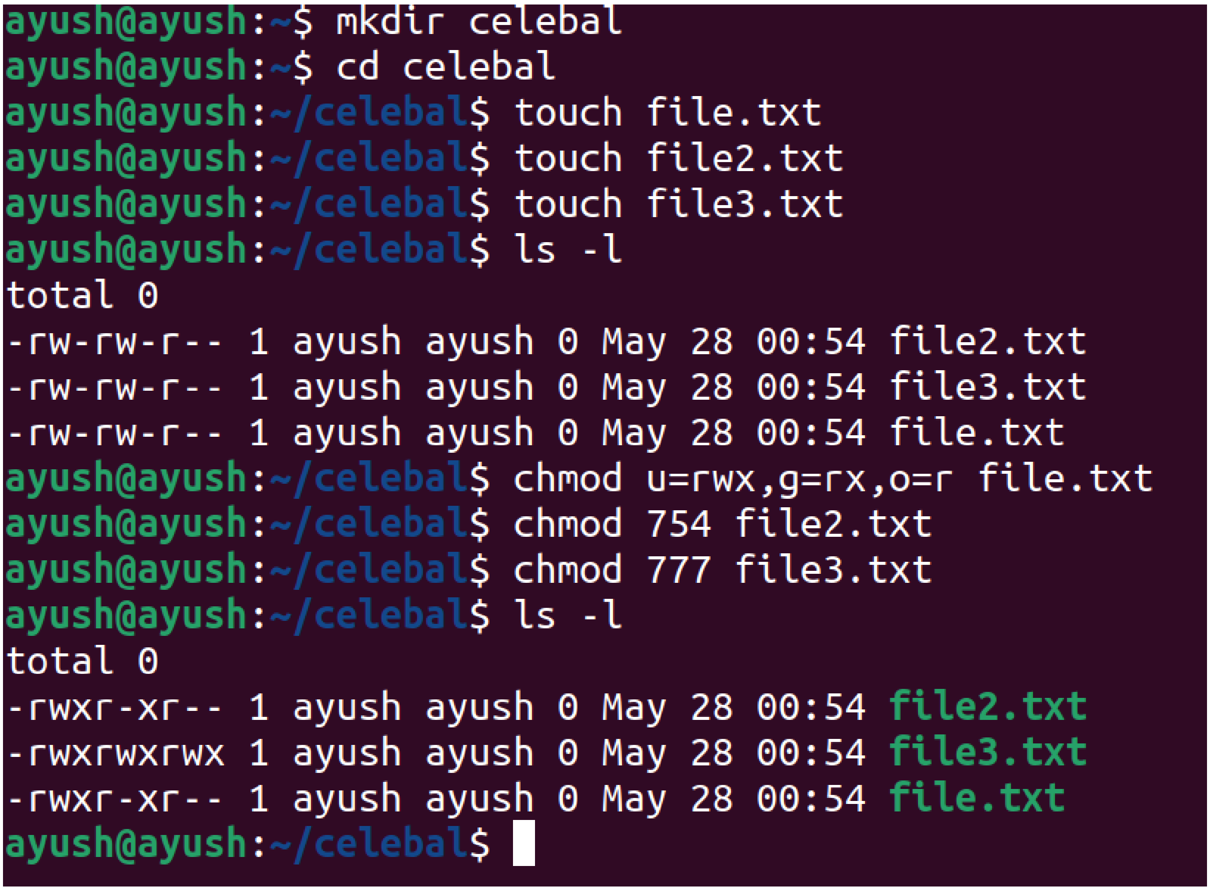
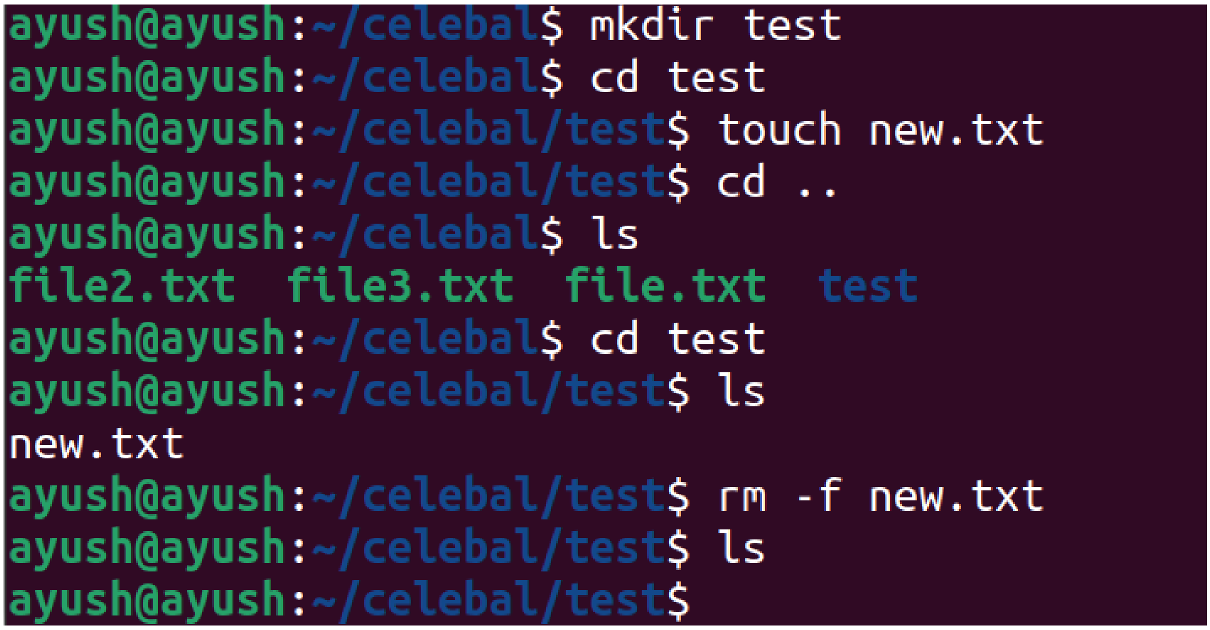
LINUX

1. Create a file, assign permissions (read, write, execute) to different user categories (owner, group, others), and practice changing permissions using chmod.



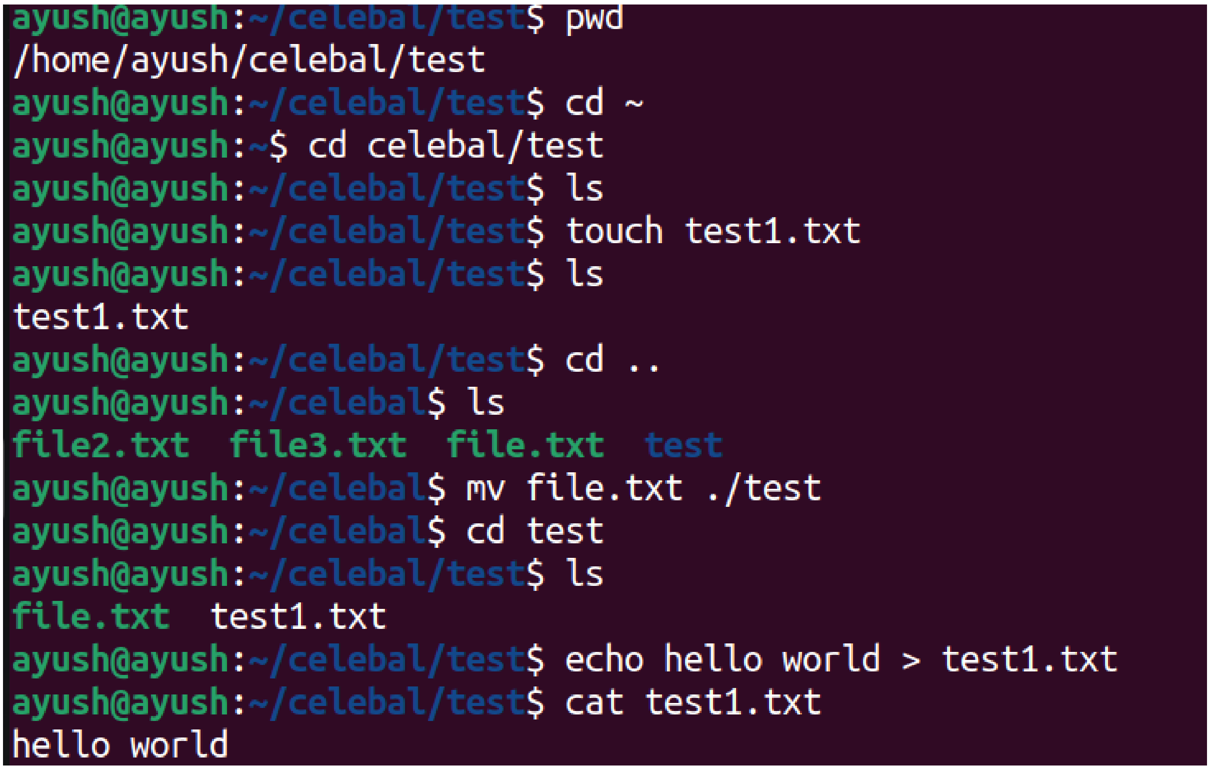
```
ayush@ayush:~$ mkdir celebal
ayush@ayush:~$ cd celebal
ayush@ayush:~/celebal$ touch file.txt
ayush@ayush:~/celebal$ touch file2.txt
ayush@ayush:~/celebal$ touch file3.txt
ayush@ayush:~/celebal$ ls -l
total 0
-rw-rw-r-- 1 ayush ayush 0 May 28 00:54 file2.txt
-rw-rw-r-- 1 ayush ayush 0 May 28 00:54 file3.txt
-rw-rw-r-- 1 ayush ayush 0 May 28 00:54 file.txt
ayush@ayush:~/celebal$ chmod u=rwx,g=rx,o=r file.txt
ayush@ayush:~/celebal$ chmod 754 file2.txt
ayush@ayush:~/celebal$ chmod 777 file3.txt
ayush@ayush:~/celebal$ ls -l
total 0
-rwxr-xr-- 1 ayush ayush 0 May 28 00:54 file2.txt
-rwxrwxrwx 1 ayush ayush 0 May 28 00:54 file3.txt
-rwxr-xr-- 1 ayush ayush 0 May 28 00:54 file.txt
ayush@ayush:~/celebal$ 
```

2. Execute basic Linux commands (e.g., ls, cd, mkdir, rm, touch) to manipulate files and directories, with an emphasis on understanding their usage.

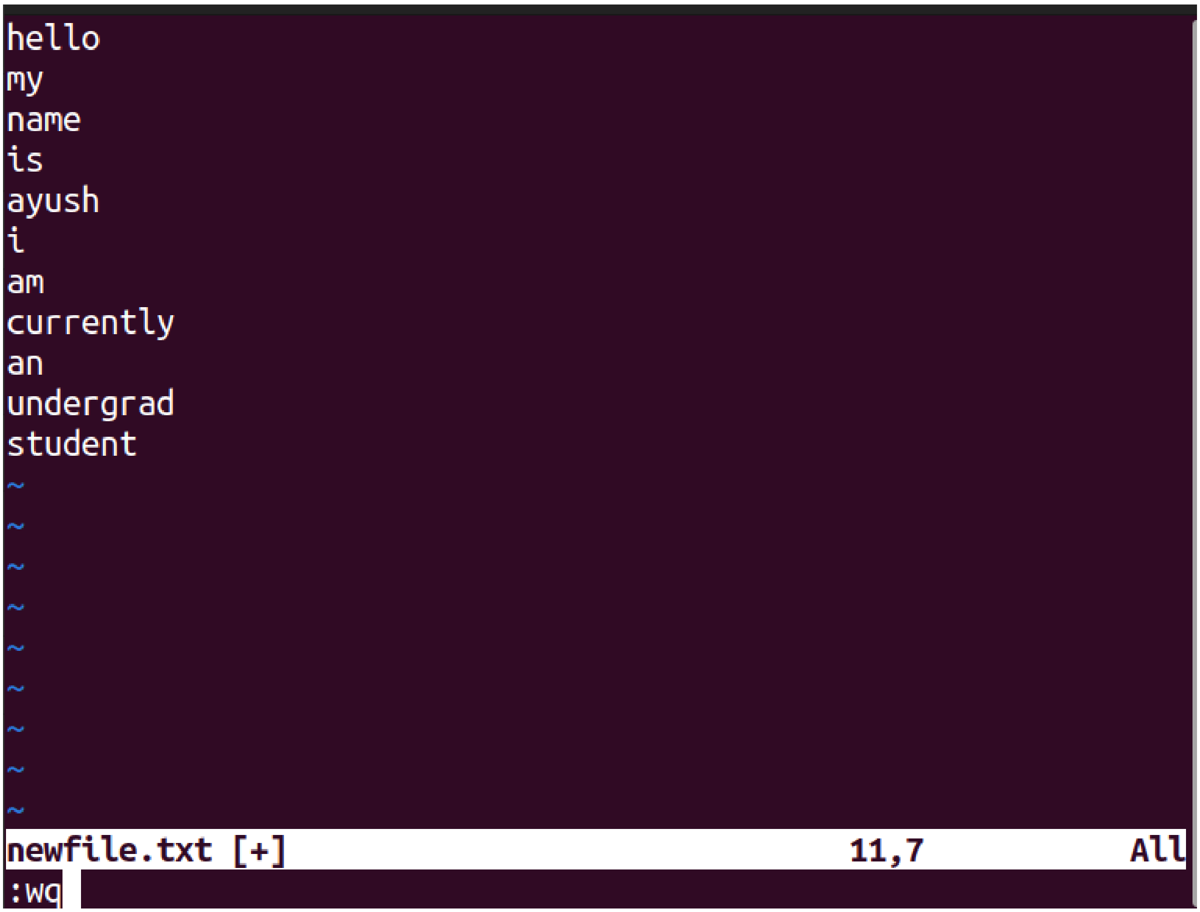


```
ayush@ayush:~/celebal$ mkdir test
ayush@ayush:~/celebal$ cd test
ayush@ayush:~/celebal/test$ touch new.txt
ayush@ayush:~/celebal/test$ cd ..
ayush@ayush:~/celebal$ ls
file2.txt  file3.txt  file.txt  test
ayush@ayush:~/celebal$ cd test
ayush@ayush:~/celebal/test$ ls
new.txt
ayush@ayush:~/celebal/test$ rm -f new.txt
ayush@ayush:~/celebal/test$ ls
ayush@ayush:~/celebal/test$
```

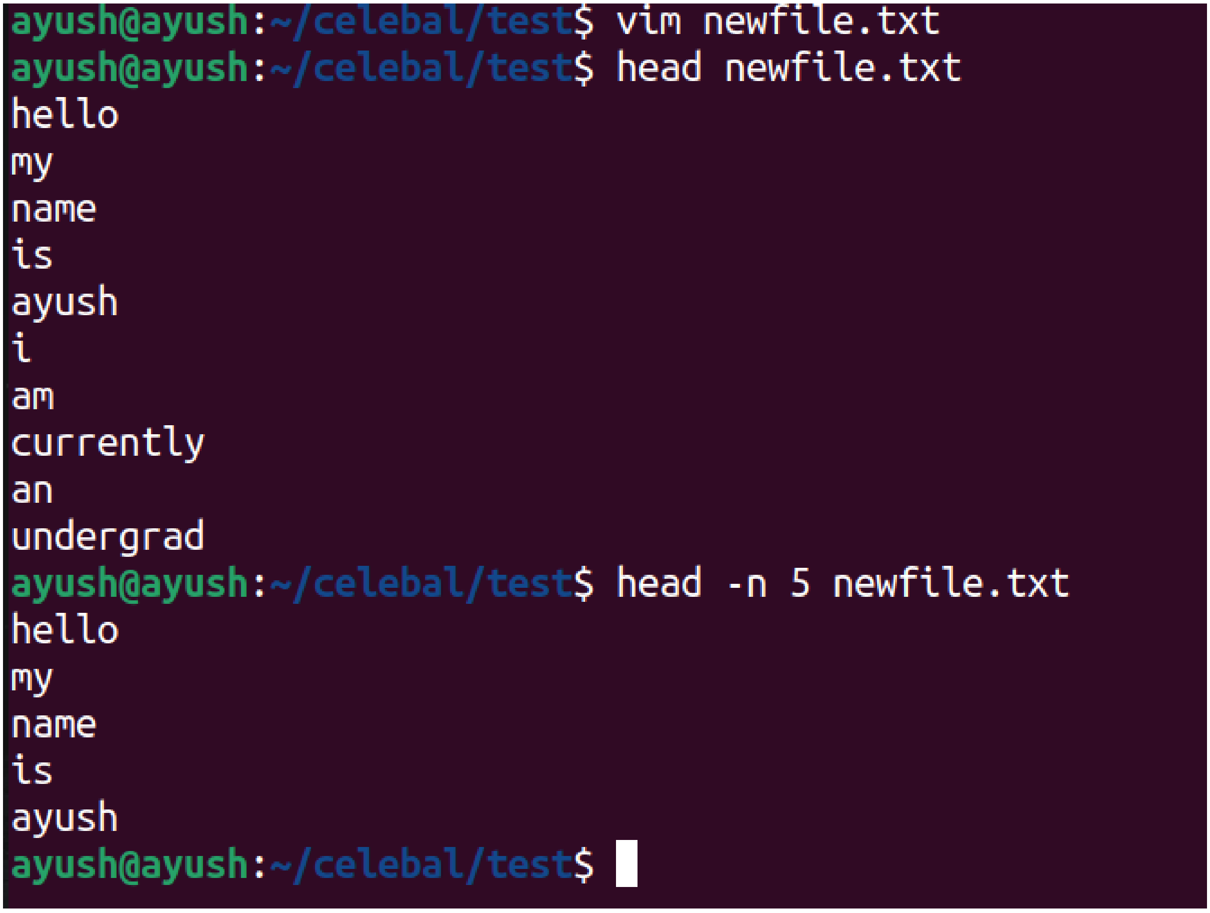
3. Using the terminal, practice navigating through directories, listing file contents, and moving files to different locations.



```
ayush@ayush:~/celebal/test$ pwd
/home/ayush/celebal/test
ayush@ayush:~/celebal/test$ cd ~
ayush@ayush:~$ cd celebal/test
ayush@ayush:~/celebal/test$ ls
ayush@ayush:~/celebal/test$ touch test1.txt
ayush@ayush:~/celebal/test$ ls
test1.txt
ayush@ayush:~/celebal/test$ cd ..
ayush@ayush:~/celebal$ ls
file2.txt  file3.txt  file.txt  test
ayush@ayush:~/celebal$ mv file.txt ./test
ayush@ayush:~/celebal$ cd test
ayush@ayush:~/celebal/test$ ls
file.txt  test1.txt
ayush@ayush:~/celebal/test$ echo hello world > test1.txt
ayush@ayush:~/celebal/test$ cat test1.txt
hello world
```

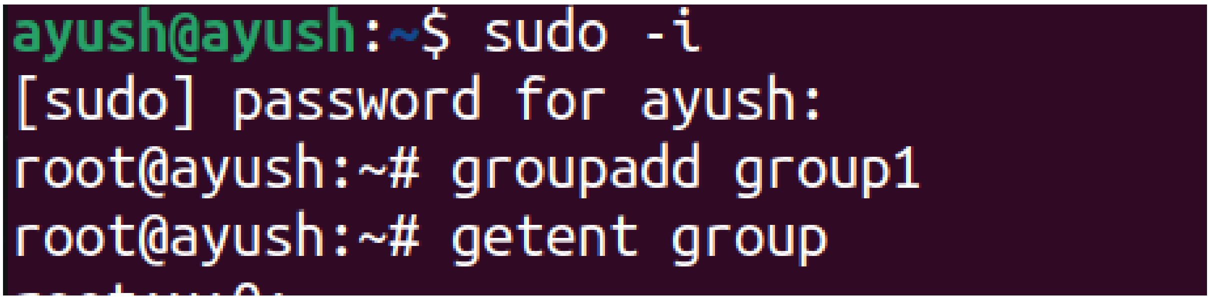


```
hello
my
name
is
ayush
i
am
currently
an
undergrad
student
~
~
~
~
~
~
~
~
~
newfile.txt [+]                                  11,7           All
:wq
```

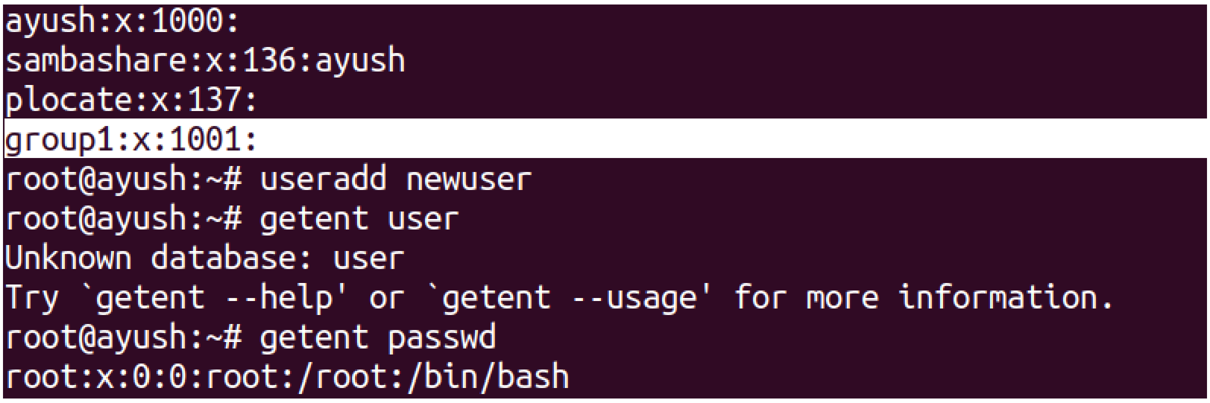


```
ayush@ayush:~/celebal/test$ vim newfile.txt
ayush@ayush:~/celebal/test$ head newfile.txt
hello
my
name
is
ayush
i
am
currently
an
undergrad
ayush@ayush:~/celebal/test$ head -n 5 newfile.txt
hello
my
name
is
ayush
ayush@ayush:~/celebal/test$
```

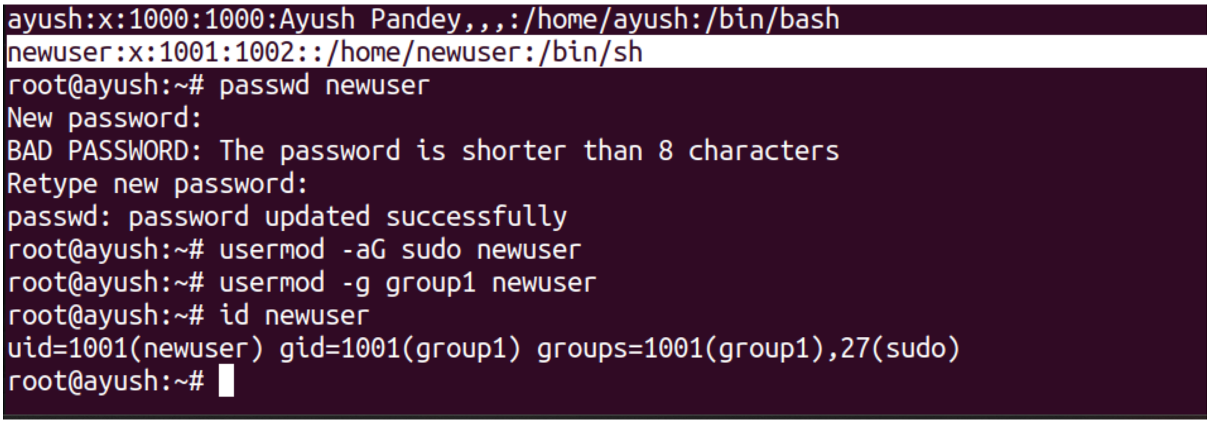
4. Create a new user and group, set their permissions, and explore user management commands like useradd, usermod, and userdel.



```
ayush@ayush:~$ sudo -i
[sudo] password for ayush:
root@ayush:~# groupadd group1
root@ayush:~# getent group
```



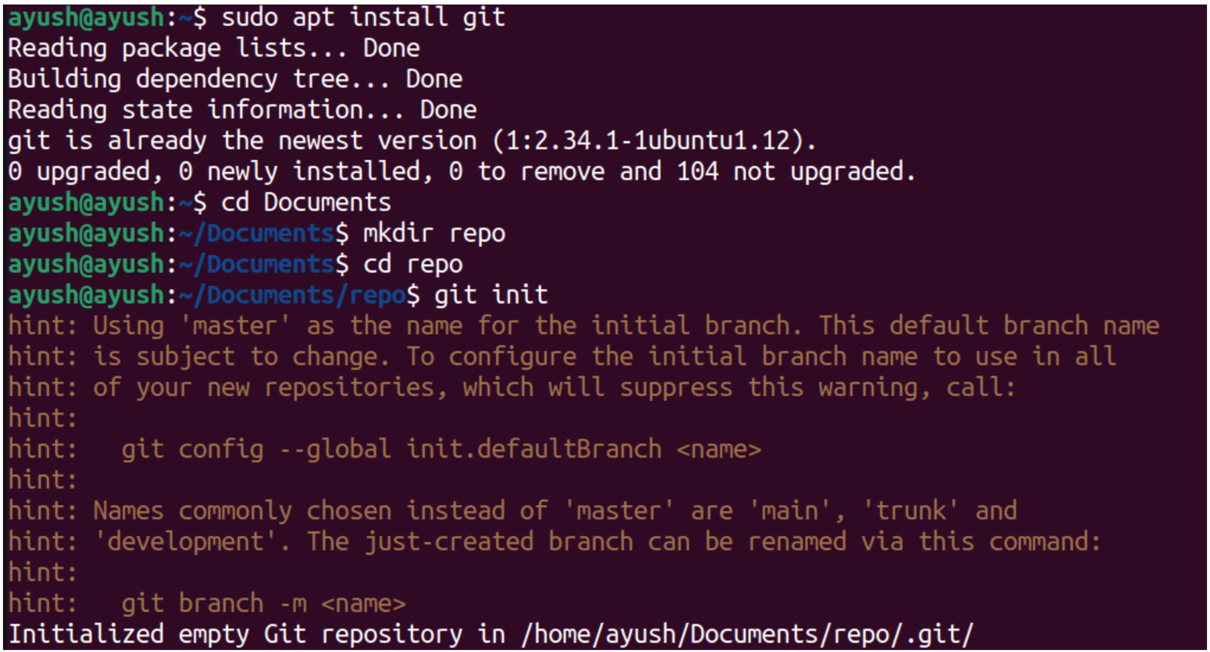
```
ayush:x:1000:
sambashare:x:136:ayush
plocate:x:137:
group1:x:1001:
root@ayush:~# useradd newuser
root@ayush:~# getent user
Unknown database: user
Try `getent --help' or `getent --usage' for more information.
root@ayush:~# getent passwd
root:x:0:0:root:/root:/bin/bash
```



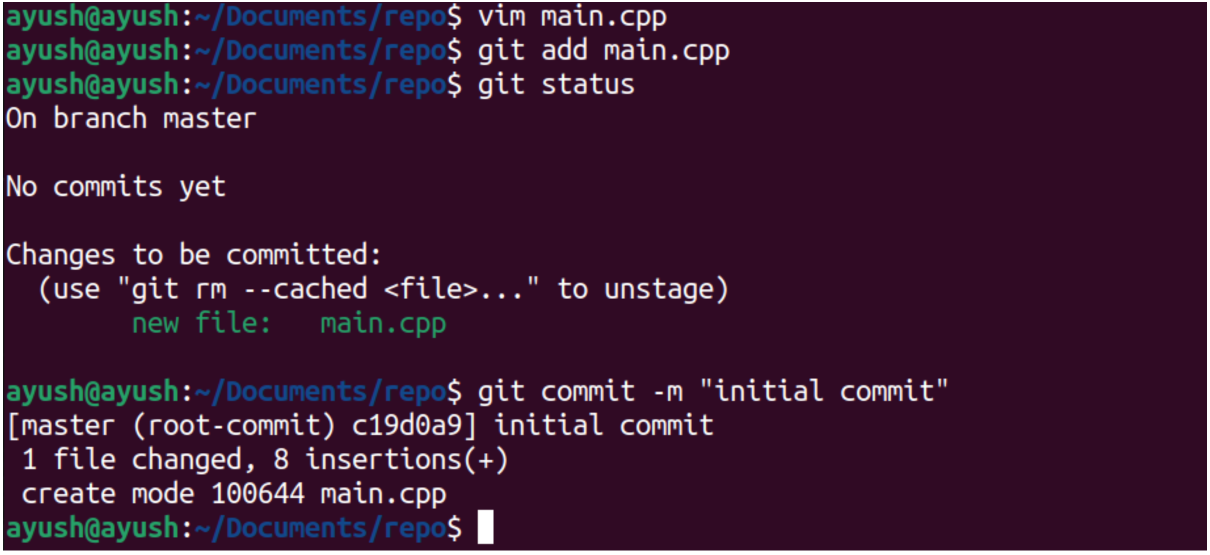
```
ayush:x:1000:1000:Ayush Pandey,,,:/home/ayush:/bin/bash
newuser:x:1001:1002::/home/newuser:/bin/sh
root@ayush:~# passwd newuser
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: password updated successfully
root@ayush:~# usermod -aG sudo newuser
root@ayush:~# usermod -g group1 newuser
root@ayush:~# id newuser
uid=1001(newuser) gid=1001(group1) groups=1001(group1),27(sudo)
root@ayush:~# 
```

GIT

5. Introduction to version control and Git basics, Git installation and configuration, Basic Git commands (init, add, commit, push, pull)

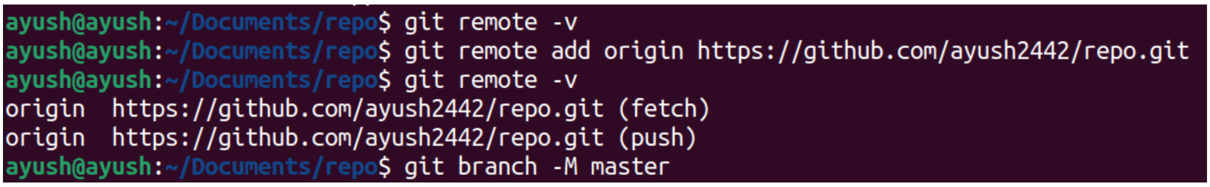


```
ayush@ayush:~$ sudo apt install git
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
git is already the newest version (1:2.34.1-1ubuntu1.12).
0 upgraded, 0 newly installed, 0 to remove and 104 not upgraded.
ayush@ayush:~$ cd Documents
ayush@ayush:~/Documents$ mkdir repo
ayush@ayush:~/Documents$ cd repo
ayush@ayush:~/Documents/repo$ git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /home/ayush/Documents/repo/.git/
```

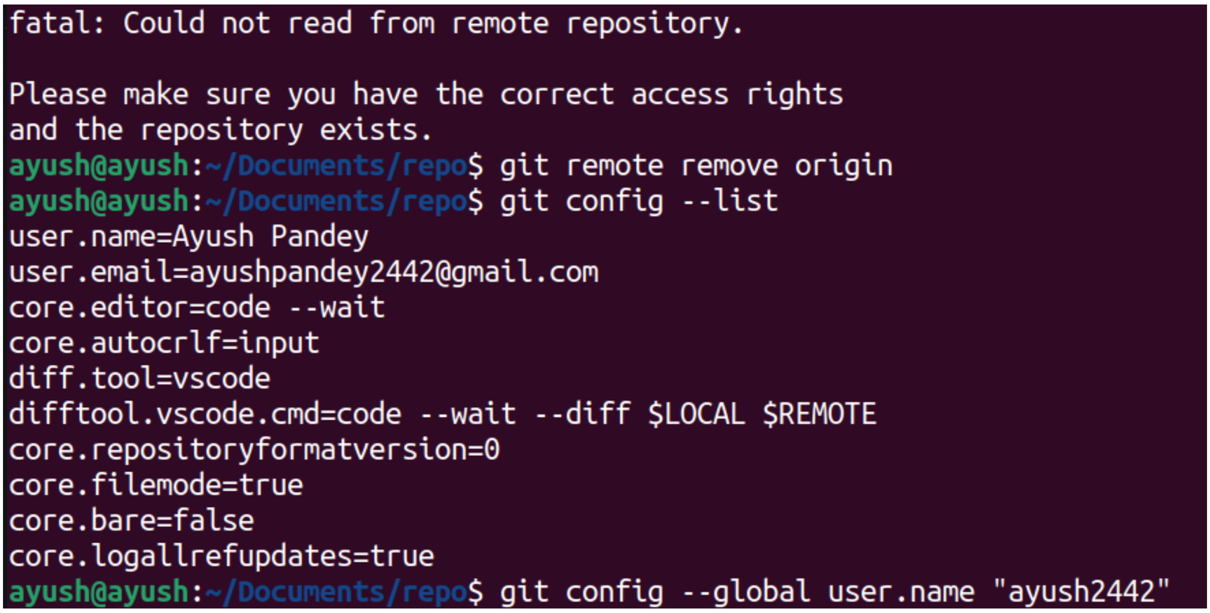


```
ayush@ayush:~/Documents/repo$ vim main.cpp
ayush@ayush:~/Documents/repo$ git add main.cpp
ayush@ayush:~/Documents/repo$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   main.cpp

ayush@ayush:~/Documents/repo$ git commit -m "initial commit"
[master (root-commit) c19d0a9] initial commit
 1 file changed, 8 insertions(+)
 create mode 100644 main.cpp
ayush@ayush:~/Documents/repo$ 
```

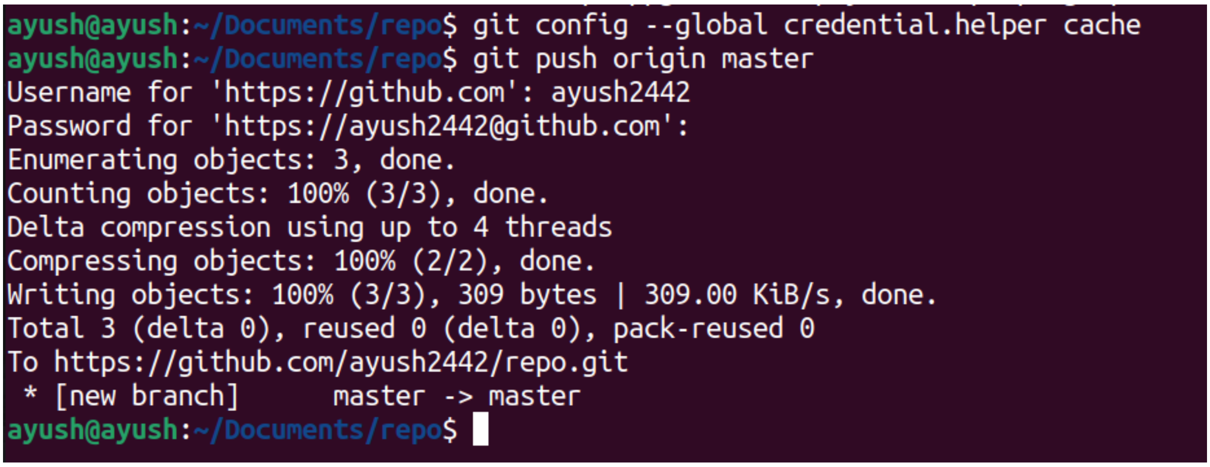
6. Setup a remote repository in Local , Add a file and commit or save the changes and push to master branch



```
ayush@ayush:~/Documents/repo$ git remote -v
ayush@ayush:~/Documents/repo$ git remote add origin https://github.com/ayush2442/repo.git
ayush@ayush:~/Documents/repo$ git remote -v
origin  https://github.com/ayush2442/repo.git (fetch)
origin  https://github.com/ayush2442/repo.git (push)
ayush@ayush:~/Documents/repo$ git branch -M master
```

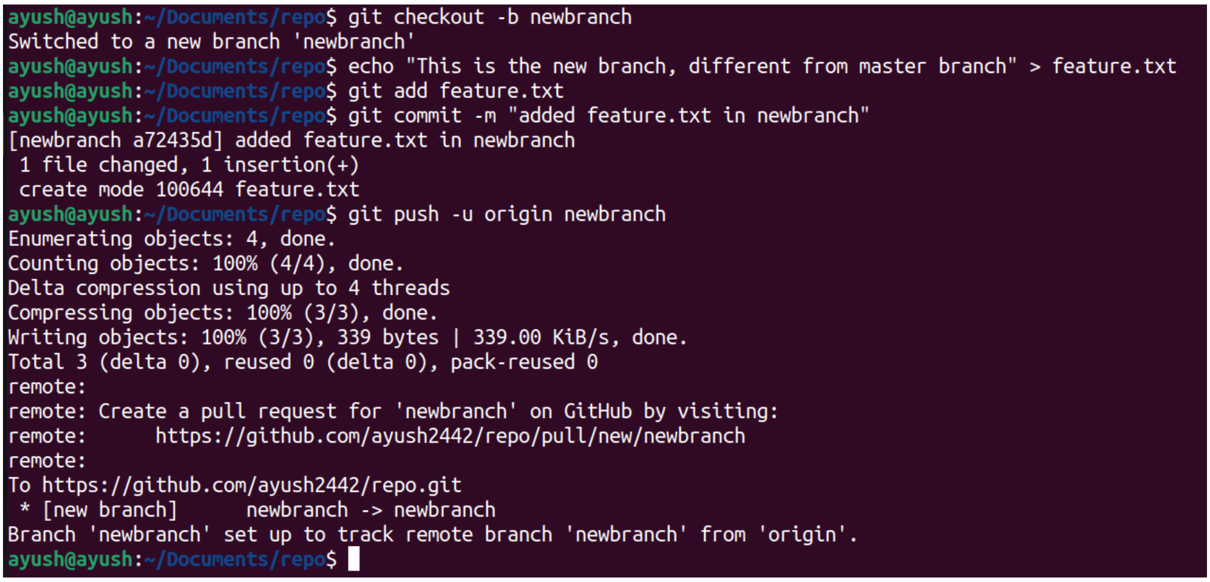


```
fatal: Could not read from remote repository.

Please make sure you have the correct access rights
and the repository exists.
ayush@ayush:~/Documents/repo$ git remote remove origin
ayush@ayush:~/Documents/repo$ git config --list
user.name=Ayush Pandey
user.email=ayushpandey2442@gmail.com
core.editor=code --wait
core.autocrlf=input
diff.tool=vscode
difftool.vscode.cmd=code --wait --diff $LOCAL $REMOTE
core.repositoryformatversion=0
core.filemode=true
core.bare=false
core.logallrefupdates=true
ayush@ayush:~/Documents/repo$ git config --global user.name "ayush2442"
```

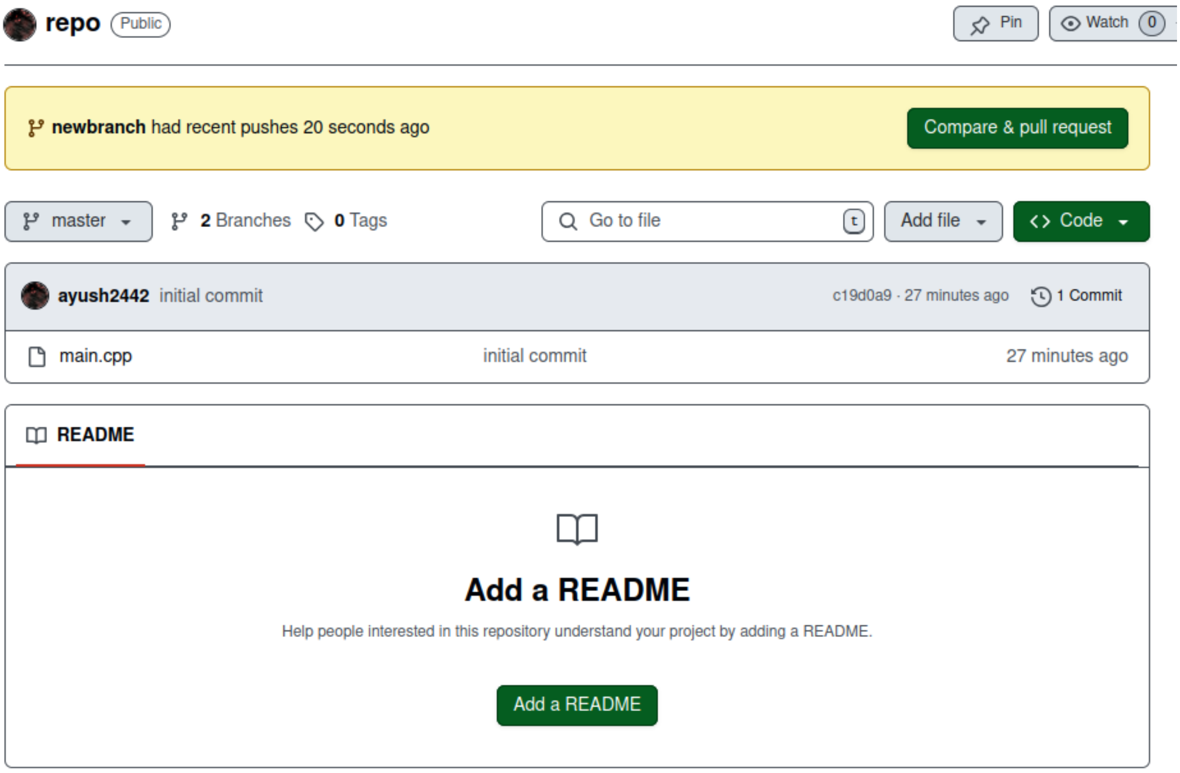


```
ayush@ayush:~/Documents/repo$ git config --global credential.helper cache
ayush@ayush:~/Documents/repo$ git push origin master
Username for 'https://github.com': ayush2442
Password for 'https://ayush2442@github.com':
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 309 bytes | 309.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/ayush2442/repo.git
 * [new branch]      master -> master
ayush@ayush:~/Documents/repo$ 
```

7. Merge types, Create a new branch then commit and push the changes to new branch and merge it with the master branch using pull request.



```
ayush@ayush:~/Documents/repo$ git checkout -b newbranch
Switched to a new branch 'newbranch'
ayush@ayush:~/Documents/repo$ echo "This is the new branch, different from master branch" > feature.txt
ayush@ayush:~/Documents/repo$ git add feature.txt
ayush@ayush:~/Documents/repo$ git commit -m "added feature.txt in newbranch"
[newbranch a72435d] added feature.txt in newbranch
 1 file changed, 1 insertion(+)
 create mode 100644 feature.txt
ayush@ayush:~/Documents/repo$ git push -u origin newbranch
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 339 bytes | 339.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'newbranch' on GitHub by visiting:
remote:      https://github.com/ayush2442/repo/pull/new/newbranch
remote:
To https://github.com/ayush2442/repo.git
 * [new branch]      newbranch -> newbranch
Branch 'newbranch' set up to track remote branch 'newbranch' from 'origin'.
ayush@ayush:~/Documents/repo$ 
```



repo Public

Pin Watch 0

newbranch had recent pushes 20 seconds ago

Compare & pull request

master 2 Branches 0 Tags Go to file t Add file Code

ayush2442 initial commit c19d0a9 · 27 minutes ago 1 Commit

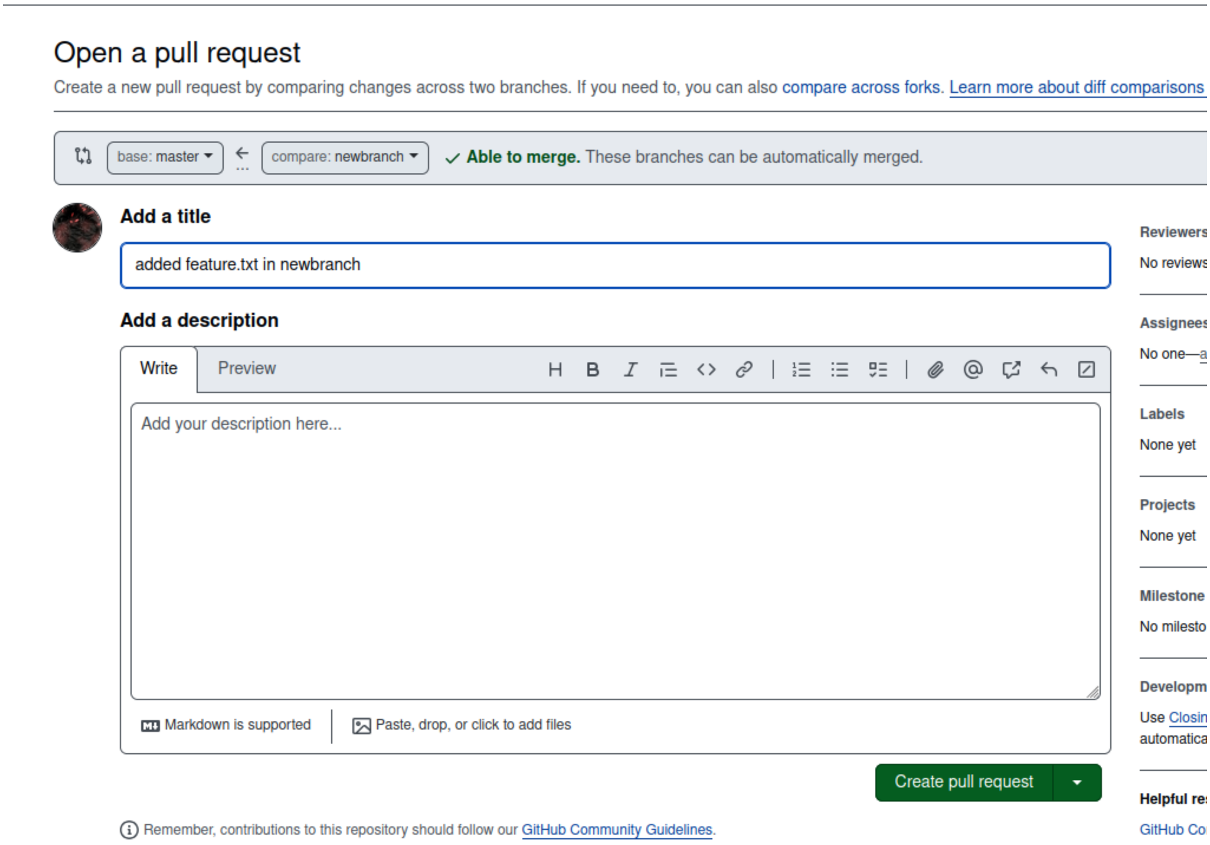
main.cpp	initial commit	27 minutes ago

README

Add a README

Help people interested in this repository understand your project by adding a README.

Add a README



Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also compare across forks. Learn more about diff comparisons

base: master ← compare: newbranch ✓ Able to merge. These branches can be automatically merged.

Add a title

added feature.txt in newbranch

Add a description

Write Preview

Add your description here...

Markdown is supported Paste, drop, or click to add files

Create pull request

Remember, contributions to this repository should follow our GitHub Community Guidelines.

Reviewers
No reviews

Assignees
No one—a

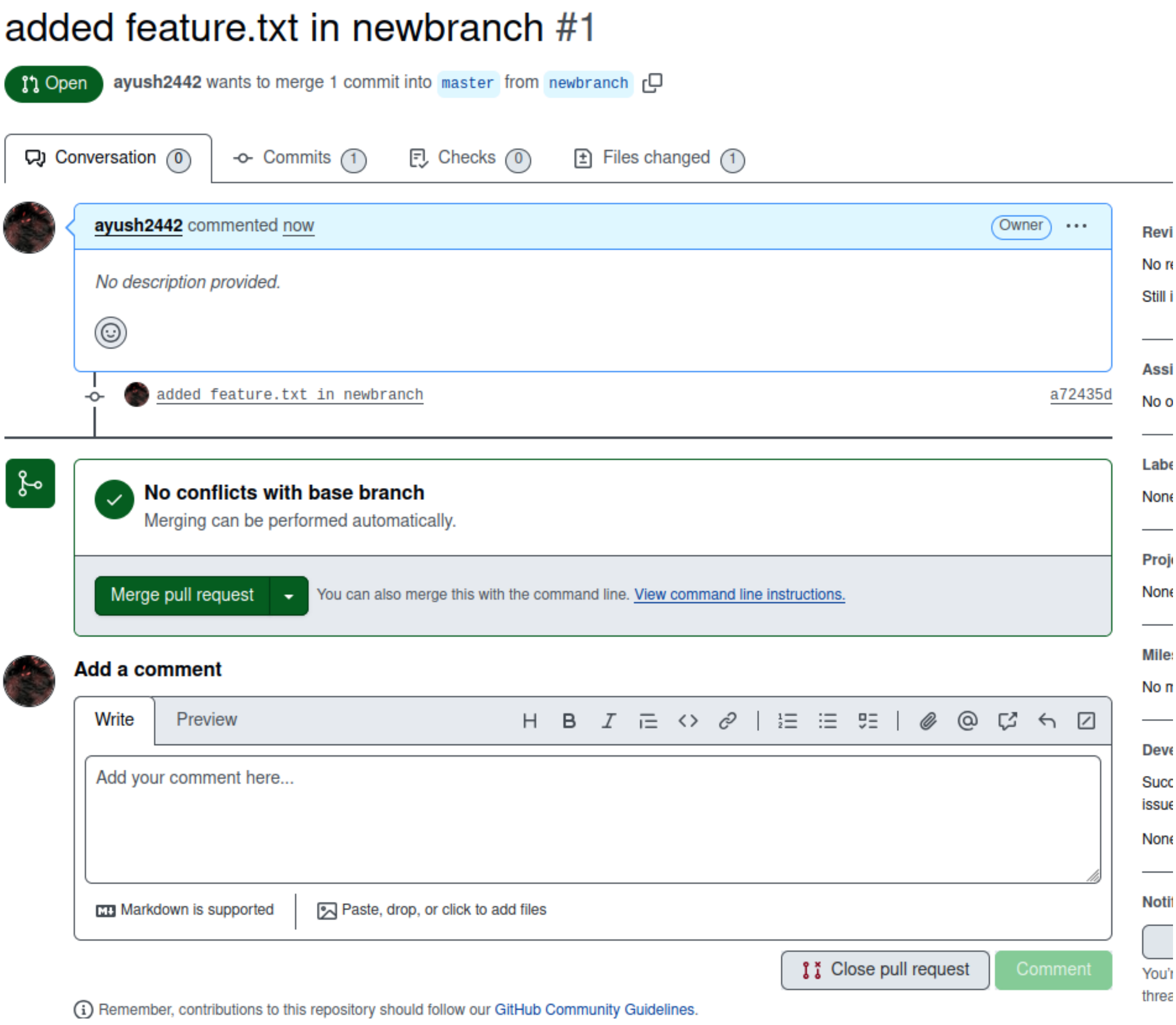
Labels
None yet

Projects
None yet

Milestone
No milesto

Developm
Use Closin
automatica

Helpful re
GitHub Co



added feature.txt in newbranch #1

Open ayush2442 wants to merge 1 commit into master from newbranch

Conversation 0 Commits 1 Checks 0 Files changed 1

ayush2442 commented now Owner

No description provided.

added feature.txt in newbranch a72435d

No conflicts with base branch

Merging can be performed automatically.

Merge pull request You can also merge this with the command line. View command line instructions.

Add a comment

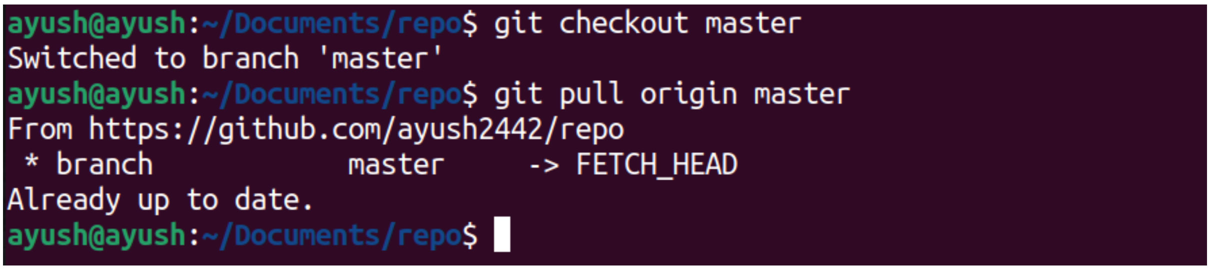
Write Preview

Add your comment here...

Markdown is supported Paste, drop, or click to add files

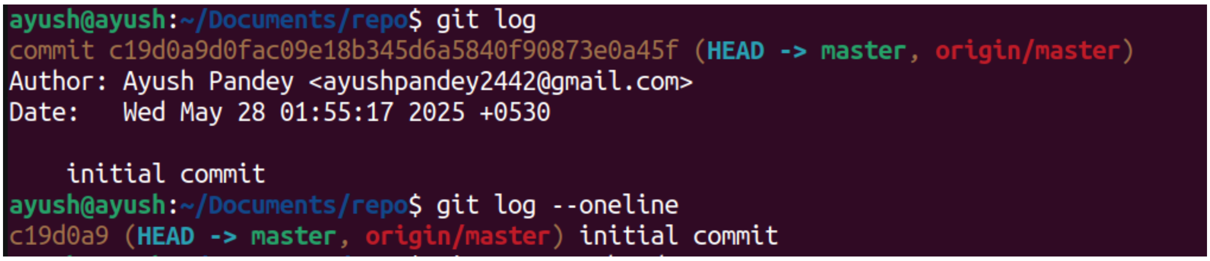
Close pull request Comment

Remember, contributions to this repository should follow our GitHub Community Guidelines.

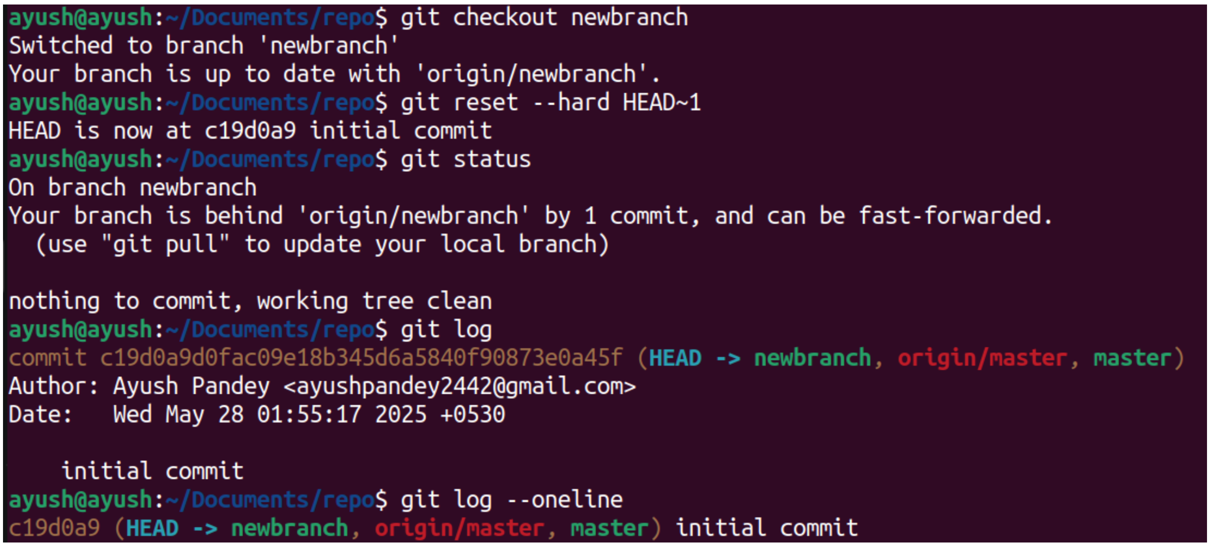


```
ayush@ayush:~/Documents/repo$ git checkout master
Switched to branch 'master'
ayush@ayush:~/Documents/repo$ git pull origin master
From https://github.com/ayush2442/repo
 * branch            master     -> FETCH_HEAD
Already up to date.
ayush@ayush:~/Documents/repo$ 
```

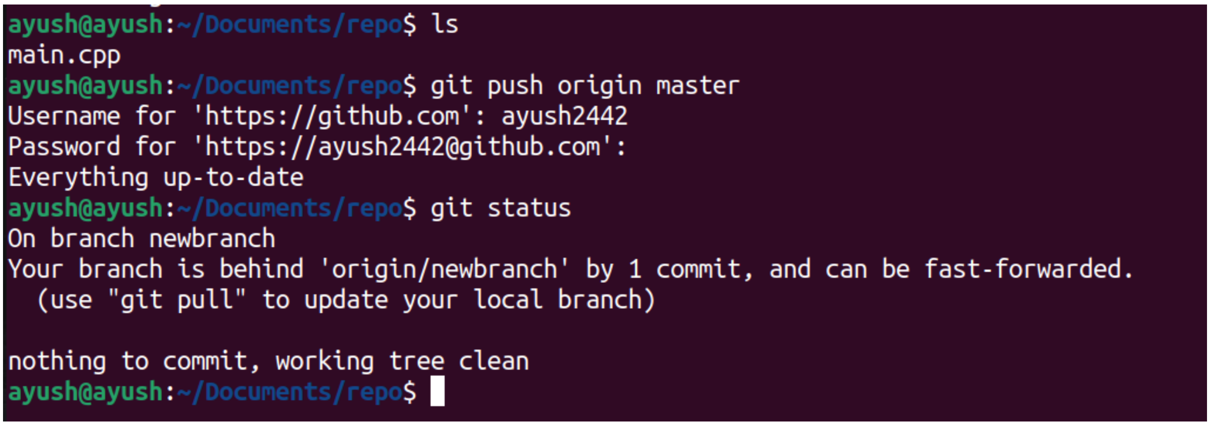
8. Undo the last commit or remove the last created file from remote repo using CLI



```
ayush@ayush:~/Documents/repo$ git log
commit c19d0a9d0fac09e18b345d6a5840f90873e0a45f (HEAD -> master, origin/master)
Author: Ayush Pandey <ayushpandey2442@gmail.com>
Date:   Wed May 28 01:55:17 2025 +0530

    initial commit
ayush@ayush:~/Documents/repo$ git log --oneline
c19d0a9 (HEAD -> master, origin/master) initial commit
```

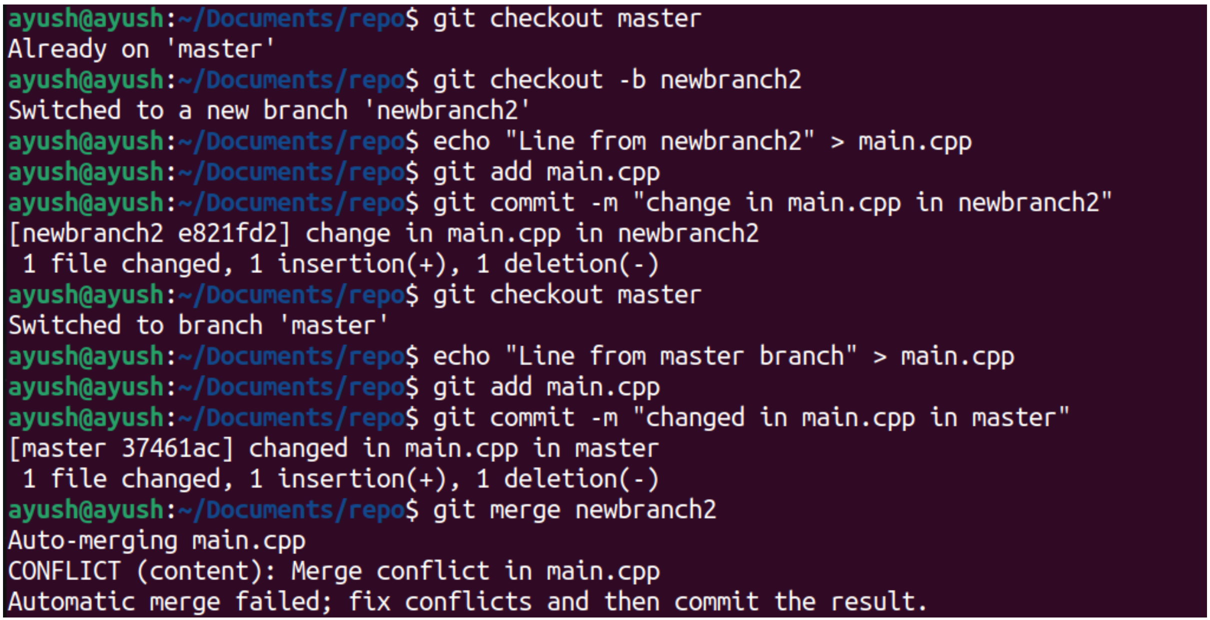


```
ayush@ayush:~/Documents/repo$ git checkout newbranch
Switched to branch 'newbranch'
Your branch is up to date with 'origin/newbranch'.
ayush@ayush:~/Documents/repo$ git reset --hard HEAD~1
HEAD is now at c19d0a9 initial commit
ayush@ayush:~/Documents/repo$ git status
On branch newbranch
Your branch is behind 'origin/newbranch' by 1 commit, and can be fast-forwarded.
  (use "git pull" to update your local branch)

nothing to commit, working tree clean
ayush@ayush:~/Documents/repo$ git log
commit c19d0a9d0fac09e18b345d6a5840f90873e0a45f (HEAD -> newbranch, origin/master, master)
Author: Ayush Pandey <ayushpandey2442@gmail.com>
Date:   Wed May 28 01:55:17 2025 +0530

    initial commit
ayush@ayush:~/Documents/repo$ git log --oneline
c19d0a9 (HEAD -> newbranch, origin/master, master) initial commit
```

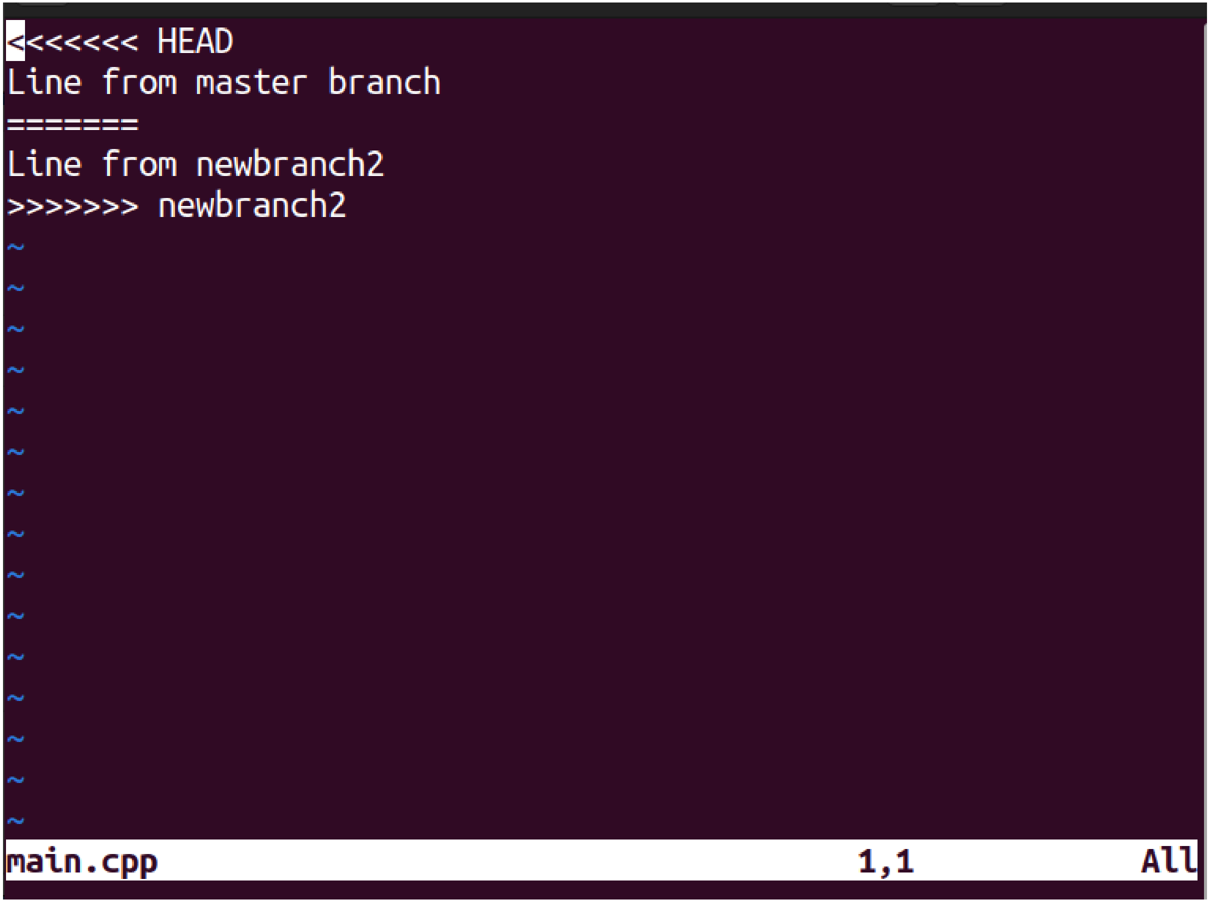


```
ayush@ayush:~/Documents/repo$ ls
main.cpp
ayush@ayush:~/Documents/repo$ git push origin master
Username for 'https://github.com': ayush2442
Password for 'https://ayush2442@github.com':
Everything up-to-date
ayush@ayush:~/Documents/repo$ git status
On branch newbranch
Your branch is behind 'origin/newbranch' by 1 commit, and can be fast-forwarded.
  (use "git pull" to update your local branch)

nothing to commit, working tree clean
ayush@ayush:~/Documents/repo$ 
```

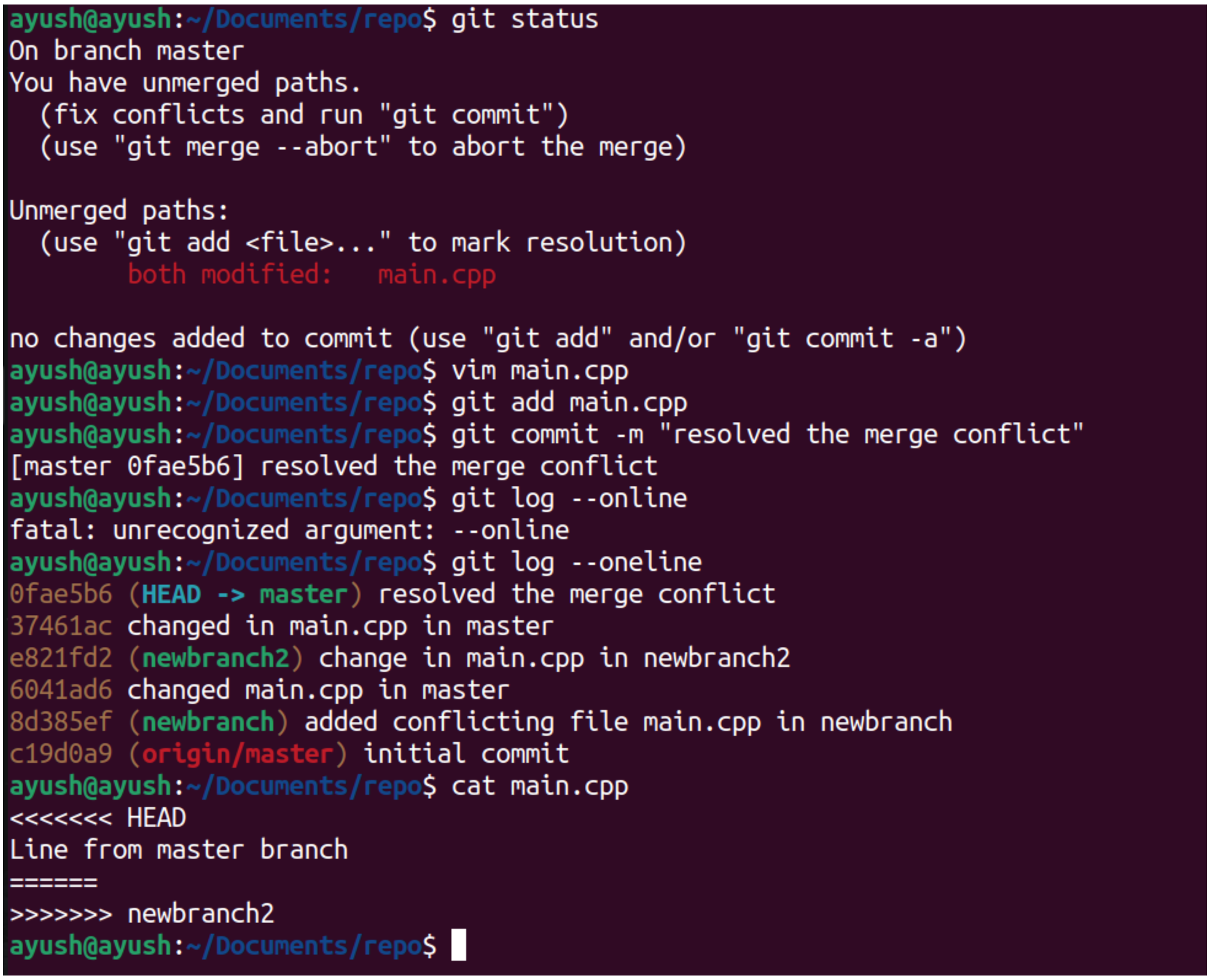
9. Branching and merging in Git, Resolve a merge conflicts



```
ayush@ayush:~/Documents/repo$ git checkout master
Already on 'master'
ayush@ayush:~/Documents/repo$ git checkout -b newbranch2
Switched to a new branch 'newbranch2'
ayush@ayush:~/Documents/repo$ echo "Line from newbranch2" > main.cpp
ayush@ayush:~/Documents/repo$ git add main.cpp
ayush@ayush:~/Documents/repo$ git commit -m "change in main.cpp in newbranch2"
[newbranch2 e821fd2] change in main.cpp in newbranch2
 1 file changed, 1 insertion(+), 1 deletion(-)
ayush@ayush:~/Documents/repo$ git checkout master
Switched to branch 'master'
ayush@ayush:~/Documents/repo$ echo "Line from master branch" > main.cpp
ayush@ayush:~/Documents/repo$ git add main.cpp
ayush@ayush:~/Documents/repo$ git commit -m "changed in main.cpp in master"
[master 37461ac] changed in main.cpp in master
 1 file changed, 1 insertion(+), 1 deletion(-)
ayush@ayush:~/Documents/repo$ git merge newbranch2
Auto-merging main.cpp
CONFLICT (content): Merge conflict in main.cpp
Automatic merge failed; fix conflicts and then commit the result.
```



```
<<<<<<< HEAD
Line from master branch
=======
Line from newbranch2
>>>>>>> newbranch2
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
main.cpp                                          1,1           All
```



```
ayush@ayush:~/Documents/repo$ git status
On branch master
You have unmerged paths.
  (fix conflicts and run "git commit")
  (use "git merge --abort" to abort the merge)

Unmerged paths:
  (use "git add <file>..." to mark resolution)
        both modified:   main.cpp

no changes added to commit (use "git add" and/or "git commit -a")
ayush@ayush:~/Documents/repo$ vim main.cpp
ayush@ayush:~/Documents/repo$ git add main.cpp
ayush@ayush:~/Documents/repo$ git commit -m "resolved the merge conflict"
[master 0fae5b6] resolved the merge conflict
ayush@ayush:~/Documents/repo$ git log --online
fatal: unrecognized argument: --online
ayush@ayush:~/Documents/repo$ git log --oneline
0fae5b6 (HEAD -> master) resolved the merge conflict
37461ac changed in main.cpp in master
e821fd2 (newbranch2) change in main.cpp in newbranch2
6041ad6 changed main.cpp in master
8d385ef (newbranch) added conflicting file main.cpp in newbranch
c19d0a9 (origin/master) initial commit
ayush@ayush:~/Documents/repo$ cat main.cpp
<<<<<<< HEAD
Line from master branch
======
>>>>>>> newbranch2
ayush@ayush:~/Documents/repo$ 
```