

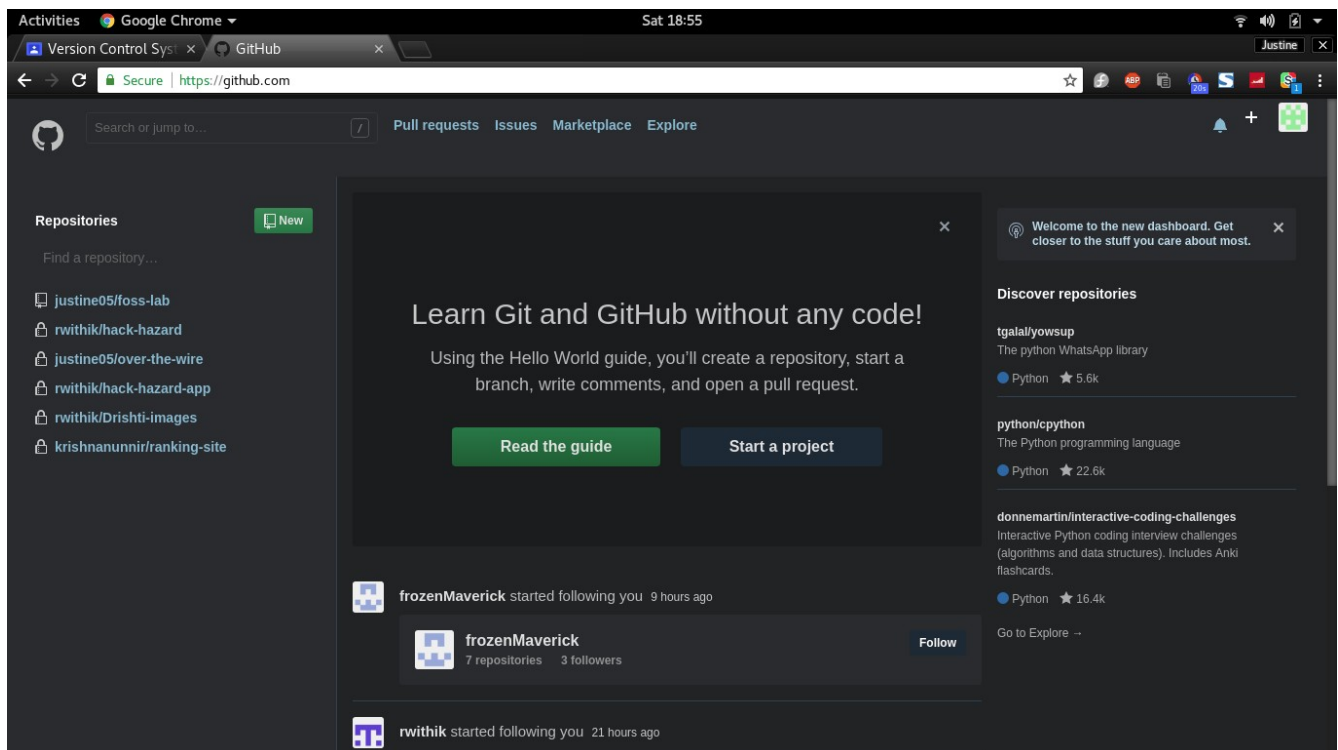
Experiment 4

23th February 2019

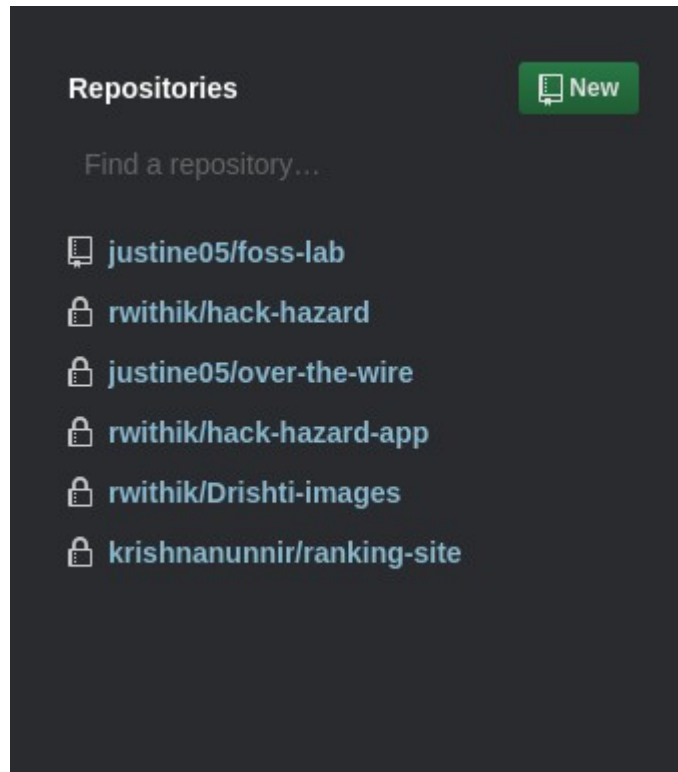
Aim: Version Control System setup and usage using GIT.

1. Creating a GIT repository.

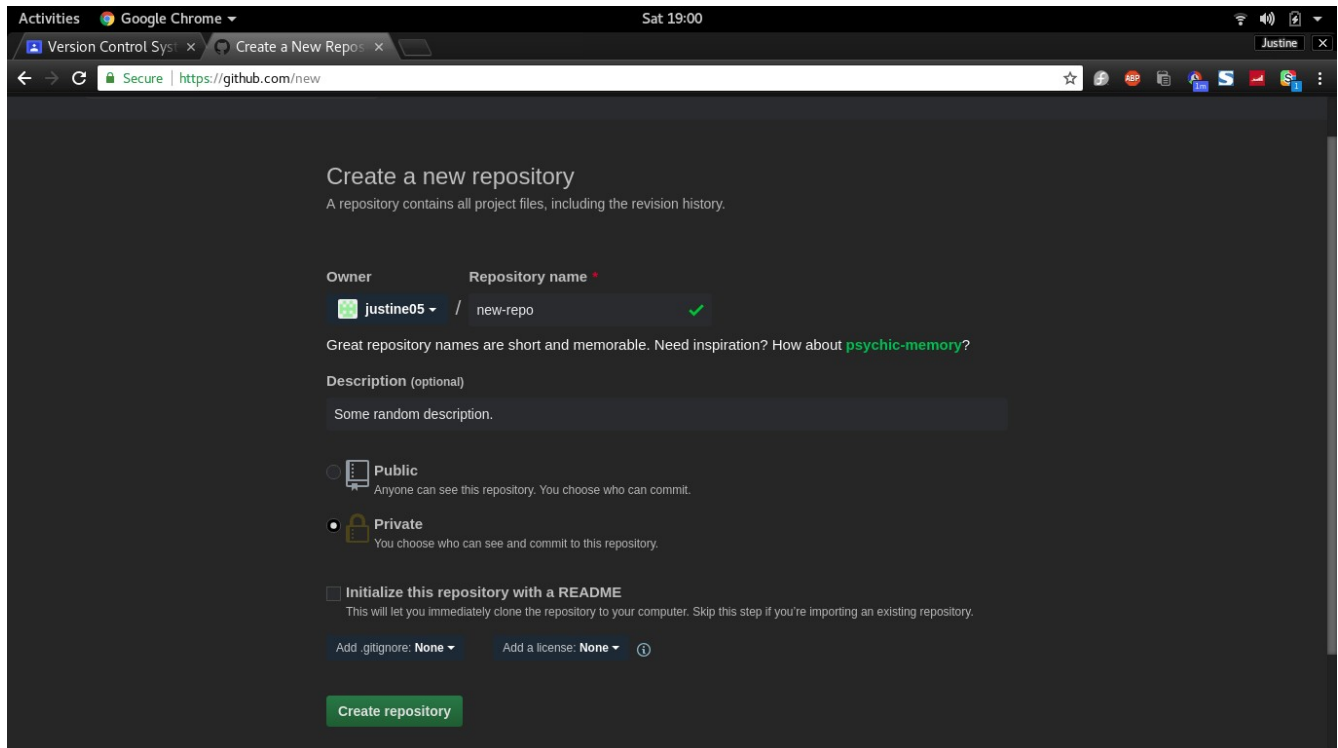
A git repository can be created by using we-based hosting services such as <https://www.github.com>. After signing into Github you come across such a homepage.



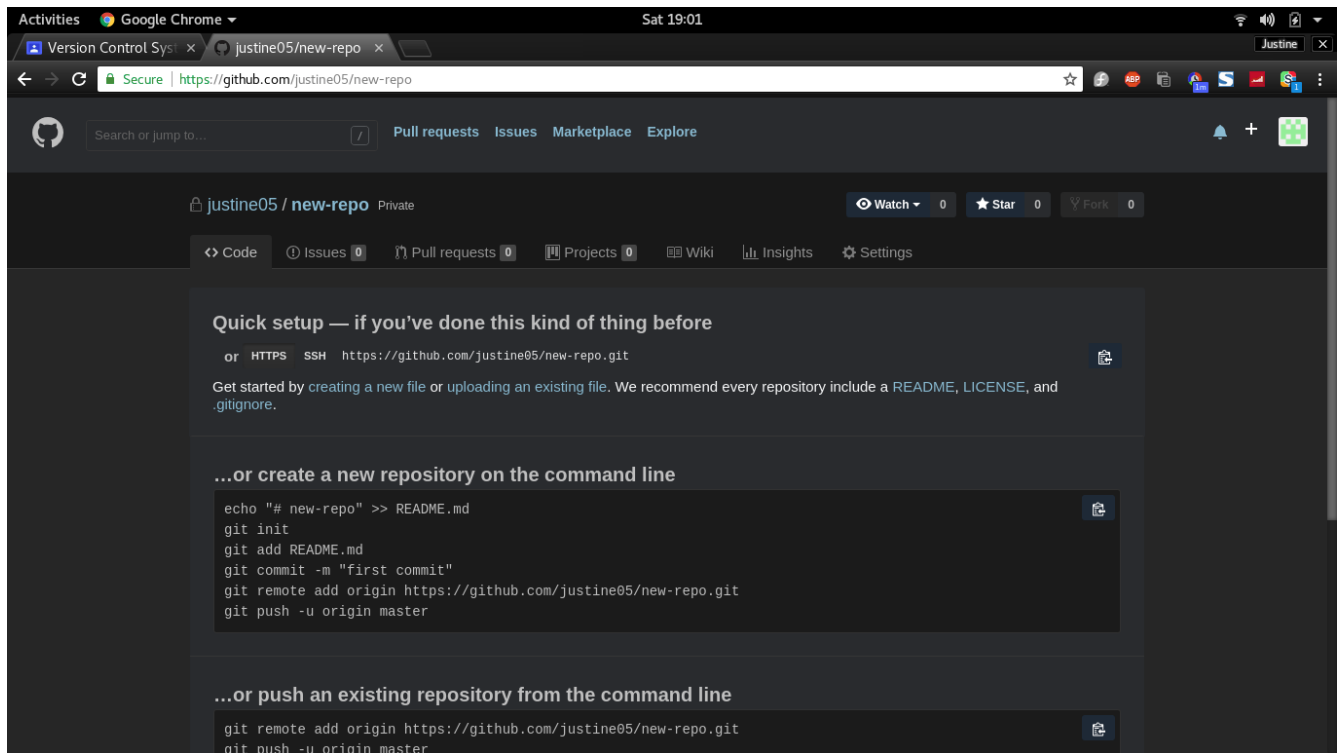
You can create a new GIT repository by clicking the 'New' button just beside the repository column on the left hand side



Now you can enter the repository name, its description and also select who are allowed to see your repository.



Now click the green 'Create Repository' button to finish the process.



Now you have set up an Online Version Control System.

2. Checking out a repository.

You can clone other GIT repositories that are either public or shared to you using your terminal. But before that **you need to set your identity.**

This can be done as follows:

Setting up user-name



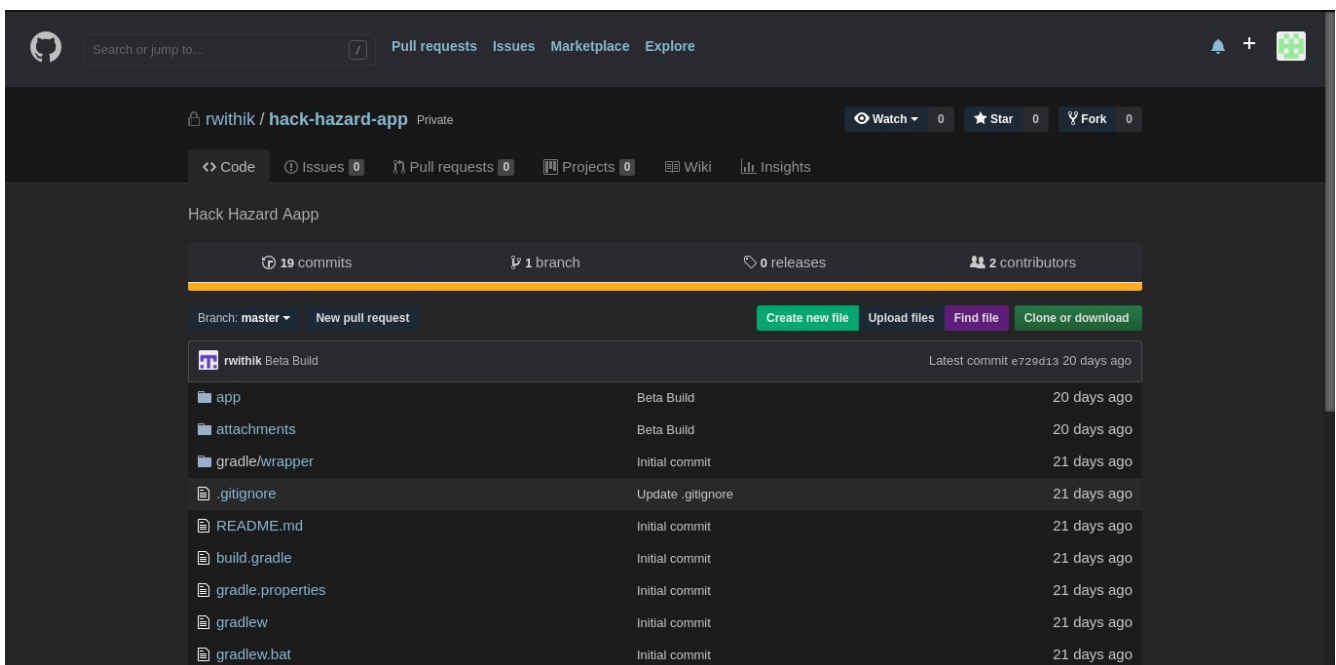
Setting up email-id



```
protonegative@fedora:~  
File Edit View Search Terminal Help  
[protonegative@fedora ~]$ git config --global user.email "justineb@cat.ac.in"
```

Now you have done the one time initial setup of git.

Suppose I want to clone the following git <https://github.com/rwithik/hack-hazard-app> .



You can clone this repository using the following steps:

Get the repository's web URL i.e <https://github.com/rwithik/hack-hazard-app.git> in this case.

```
protonegative@fedora:~  
File Edit View Search Terminal Help  
[protonegative@fedora ~]$ git clone https://github.com/rwithik/hack-hazard-app.git  
Cloning into 'hack-hazard-app'...  
Username for 'https://github.com': justine05  
Password for 'https://justine05@github.com':  
remote: Enumerating objects: 266, done.  
remote: Counting objects: 100% (266/266), done.  
remote: Compressing objects: 100% (146/146), done.  
remote: Total 266 (delta 83), reused 234 (delta 56), pack-reused 0  
Receiving objects: 100% (266/266), 145.68 KiB | 117.00 KiB/s, done.  
Resolving deltas: 100% (83/83), done.  
[protonegative@fedora ~]$ ls hack-hazard-app/  
app attachments build.gradle gradle gradle.properties gradlew gradlew.bat me README.md settings.gradle  
[protonegative@fedora ~]$
```

Type the above commands to make a local copy of that repository.

3. Adding content to the repository.

A local repository can be made git compatible using the command `git init`.

```
protonegative@fedora:~/new-repo  
File Edit View Search Terminal Help  
[protonegative@fedora ~]$ mkdir new-repo  
[protonegative@fedora ~]$ cd new-repo/  
[protonegative@fedora new-repo]$ git init  
Initialized empty Git repository in /home/protonegative/new-repo/.git/  
[protonegative@fedora new-repo]$
```

A locally made file (here `file.sh`) can be added to the repository using the `push` command. But before that the file should be made track able and put to the staging area

```
protonegative@fedora:~/new-repo
File Edit View Search Terminal Help
[protonegative@fedora new-repo]$ ls
file.sh
[protonegative@fedora new-repo]$ git add .
[protonegative@fedora new-repo]$ git status
On branch master

No commits yet

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

        new file:   file.sh
[protonegative@fedora new-repo]$
```

Now the file is tracked.

```
protonegative@fedora:~/new-repo
File Edit View Search Terminal Help
[protonegative@fedora new-repo]$ git commit -m "Added file.sh"
[master (root-commit) 59e7244] Added file.sh
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 file.sh
[protonegative@fedora new-repo]$ status
bash: status: command not found...
^[[A[protonegative@fedora new-repo]$ git status
On branch master
nothing to commit, working tree clean
[protonegative@fedora new-repo]$
```

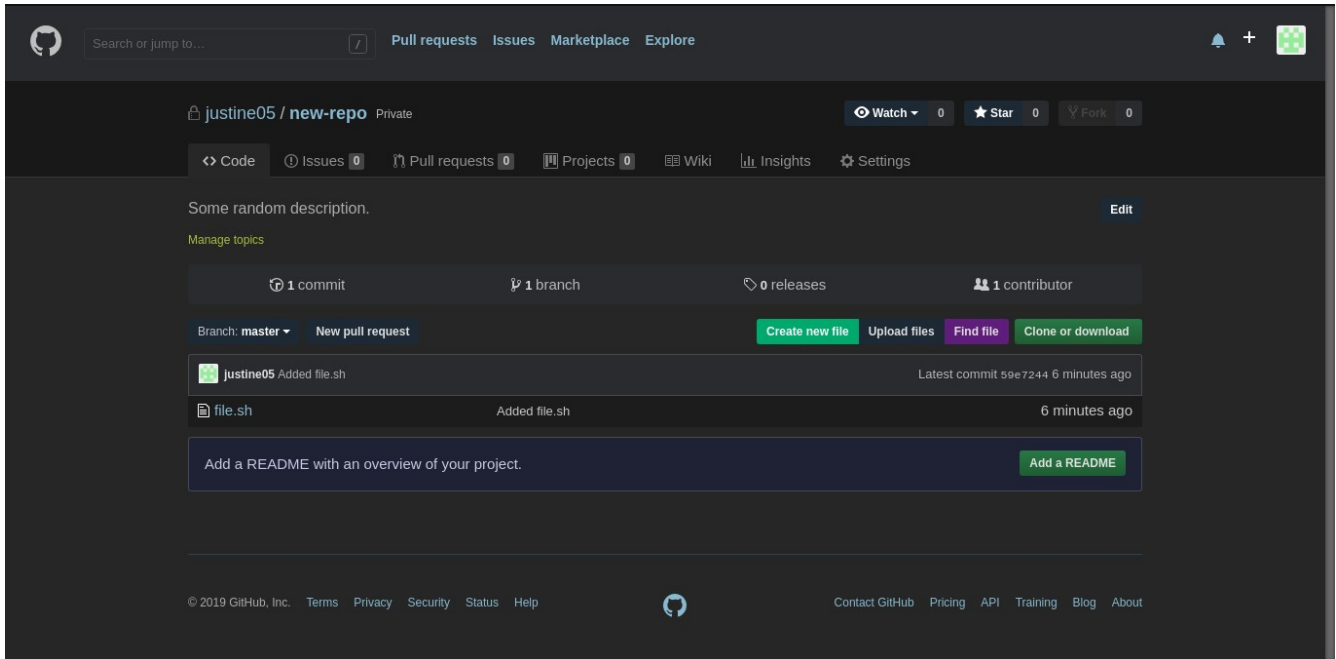
Now the file is in the staging area.

Now connecting the local and the remote repository.

```
protonegative@fedora:~/new-repo
File Edit View Search Terminal Help
[protonegative@fedora new-repo]$ git remote add origin https://github.com/justine05/new-repo.git
[protonegative@fedora new-repo]$ git remote -v
origin  https://github.com/justine05/new-repo.git (fetch)
origin  https://github.com/justine05/new-repo.git (push)
[protonegative@fedora new-repo]$
```

Now this file can be pushed into the local repository using the push command.

```
protonegative@fedora:~/new-repo
File Edit View Search Terminal Help
[protonegative@fedora new-repo]$ git push -u origin master
Username for 'https://github.com': justine05
Password for 'https://justine05@github.com':
Counting objects: 3, done.
Writing objects: 100% (3/3), 211 bytes | 211.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/justine05/new-repo.git
 * [new branch]      master -> master
Branch master set up to track remote branch master from origin.
[protonegative@fedora new-repo]$
```



Hence the file is successfully added to the remote server.

4. Committing the data to a repository.

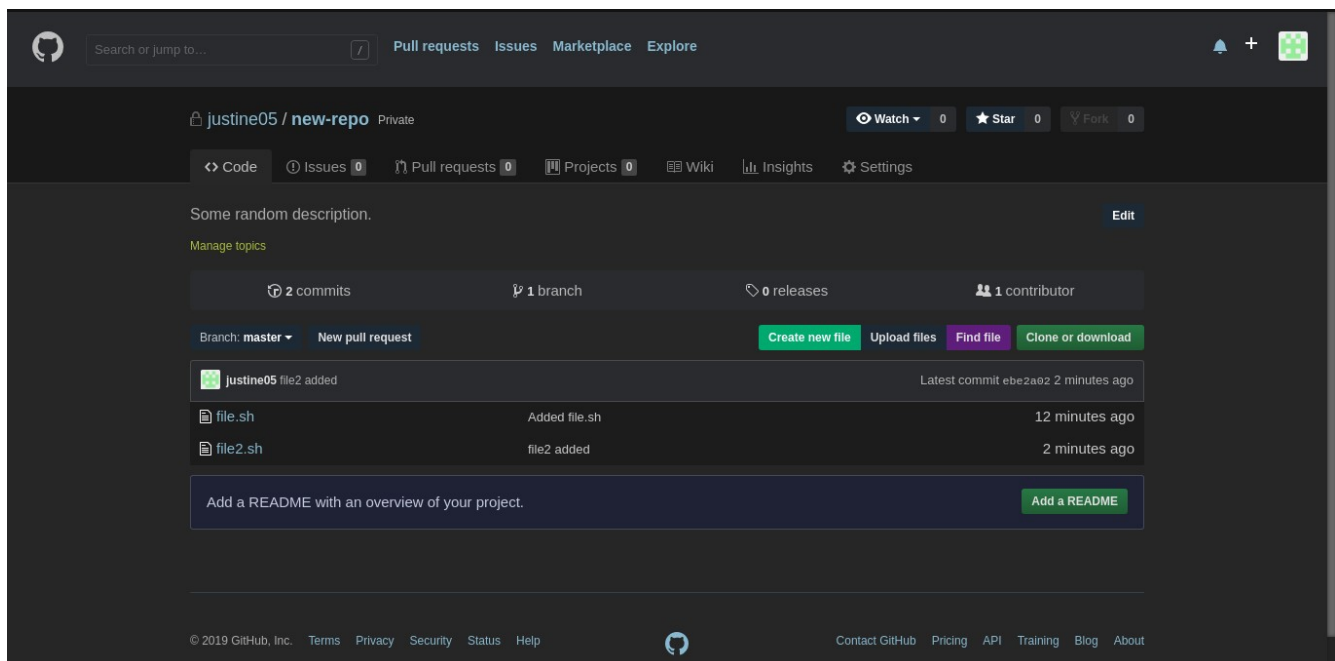
```
protonegative@fedora:~/new-repo
File Edit View Search Terminal Help
[protonegative@fedora new-repo]$ touch file2.sh
[protonegative@fedora new-repo]$ git add file2.sh
[protonegative@fedora new-repo]$ git commit -m "file2 added"
[master ebe2a02] file2 added
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 file2.sh
[protonegative@fedora new-repo]$ status
bash: status: command not found...
[protonegative@fedora new-repo]$ git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
[protonegative@fedora new-repo]$
```

A new file file2.sh is made and is committed. Now pushing it to remote repository.

```
protonegative@fedora:~/new-repo
File Edit View Search Terminal Help
[protonegative@fedora new-repo]$ touch file2.sh
[protonegative@fedora new-repo]$ git add file2.sh
[protonegative@fedora new-repo]$ git commit -m "file2 added"
[master ebe2a02] file2 added
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 file2.sh
[protonegative@fedora new-repo]$ status
bash: status: command not found...
[protonegative@fedora new-repo]$ git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
(use "git push" to publish your local commits)

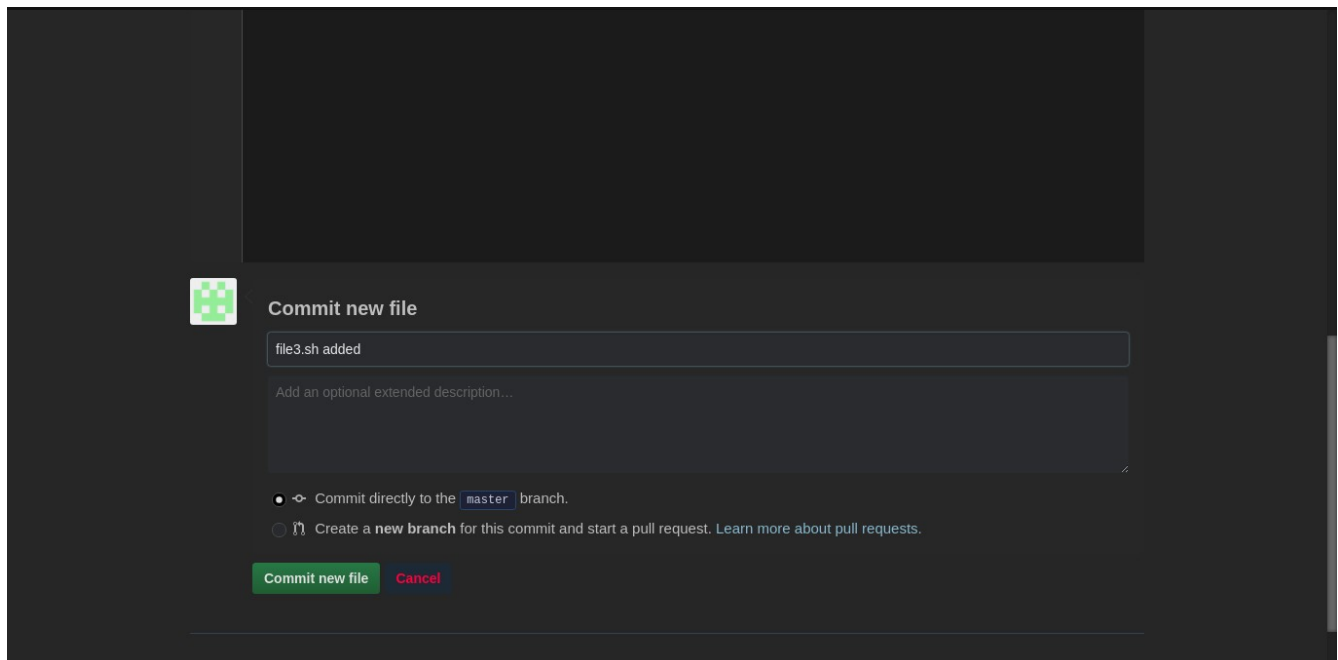
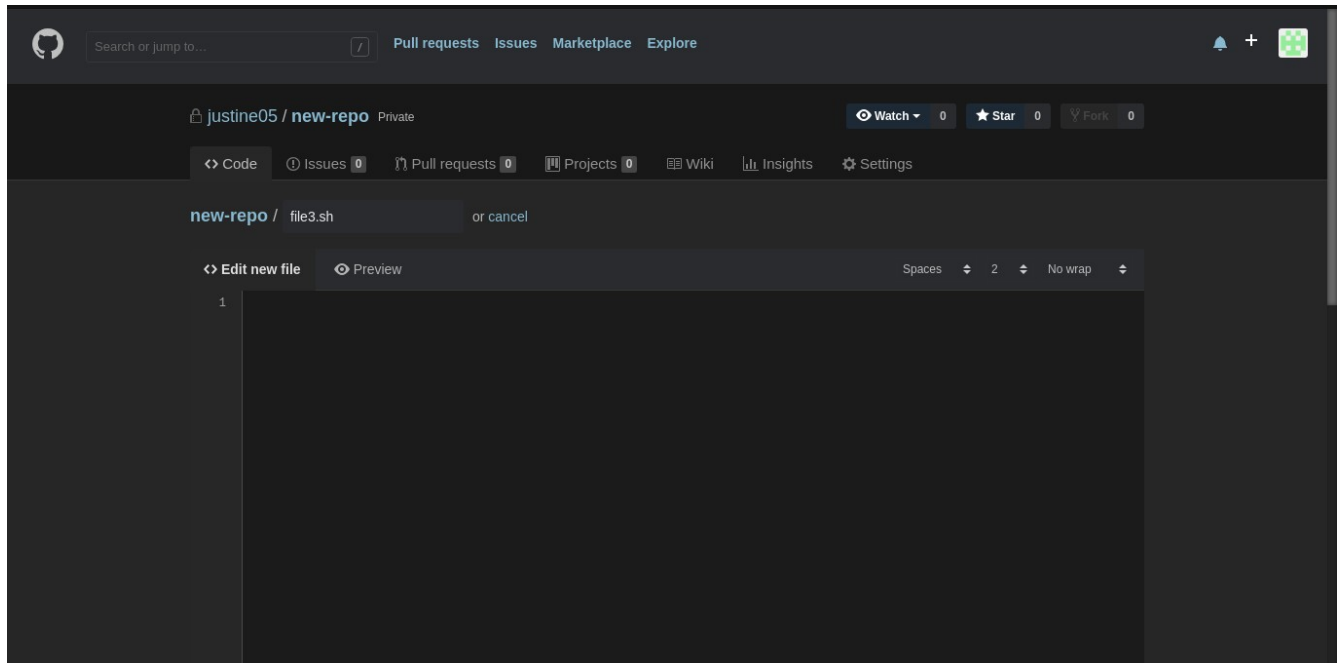
nothing to commit, working tree clean
[protonegative@fedora new-repo]$ git push
Username for 'https://github.com': justine05
Password for 'https://justine05@github.com':
Counting objects: 2, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 235 bytes | 235.00 KiB/s, done.
Total 2 (delta 0), reused 0 (delta 0)
To https://github.com/justine05/new-repo.git
59e7244..ebe2a02 master -> master
[protonegative@fedora new-repo]$
```



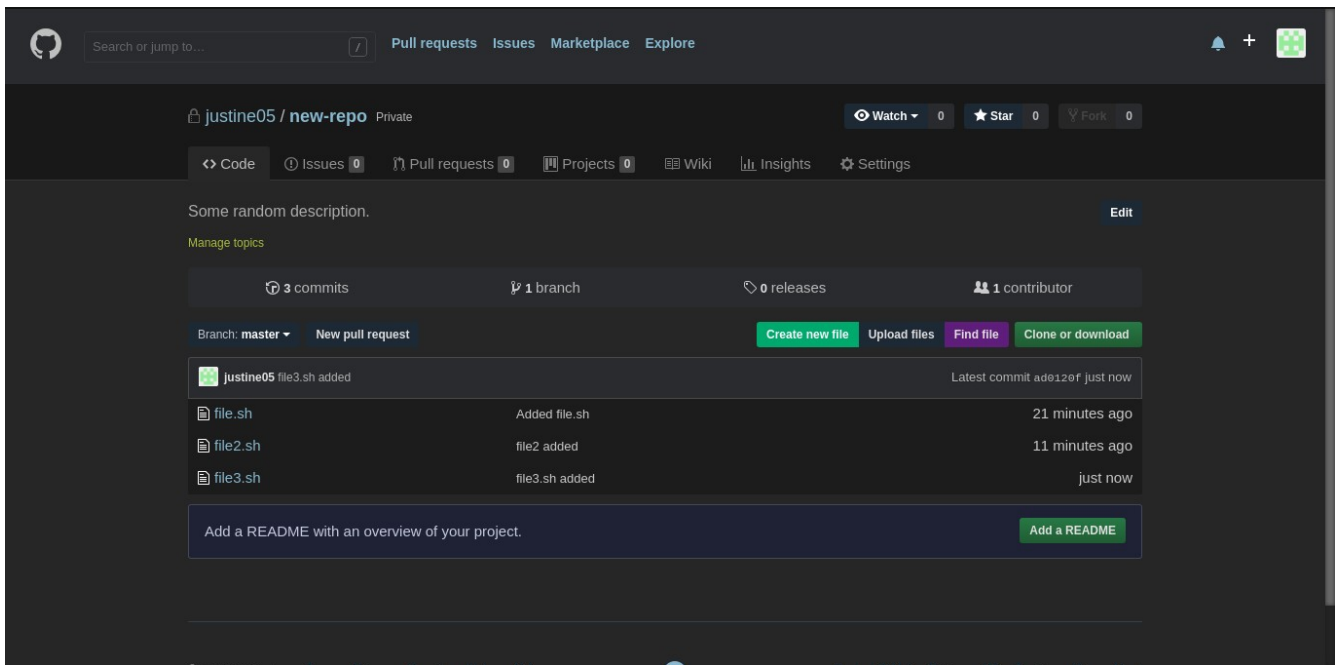
Hence the file has been successfully pushed with the commit message.

5. Updating the local copy.

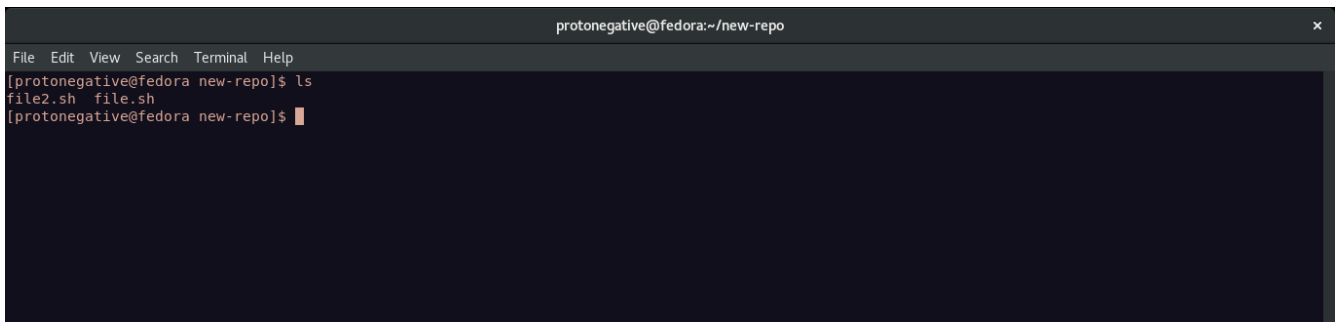
When a new file is added to the remote repository using github or another contributor, the local repo should be updated using the pull command.



Creating file3.sh online using git.



But our local repo is not updated.



Hence we use the git pull command.

```
protonegative@fedora:~/new-repo
File Edit View Search Terminal Help
[protonegative@fedora new-repo]$ ls
file2.sh  file.sh
[protonegative@fedora new-repo]$ git pull
Username for 'https://github.com': justine05
Password for 'https://justine05@github.com':
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
From https://github.com/justine05/new-repo
   ebe2a02..ad0120f  master    -> origin/master
Updating ebe2a02..ad0120f
Fast-forward
   file3.sh | 1 +
   1 file changed, 1 insertion(+)
   create mode 100644 file3.sh
[protonegative@fedora new-repo]$ ls
file2.sh  file3.sh  file.sh
[protonegative@fedora new-repo]$
```

Now file3.sh is available in our local repository.

6. Comparing different revisions.

Different revisions can be compared using the log command.

```
protonegative@fedora:~/new-repo
File Edit View Search Terminal Help
[protonegative@fedora new-repo]$ git log
commit ad0120fd118ba78ebfe035a4544fb09bd4cc974 (HEAD -> master, origin/master)
Author: Justine Biju <36364765+justine05@users.noreply.github.com>
Date:   Sat Feb 23 19:50:53 2019 +0530

    file3.sh added

commit ebe2a02da81a526e072d0cd591b2d0fa8ceab31f
Author: justine05 <justinebiju05@gmail.com>
Date:   Sat Feb 23 19:39:58 2019 +0530

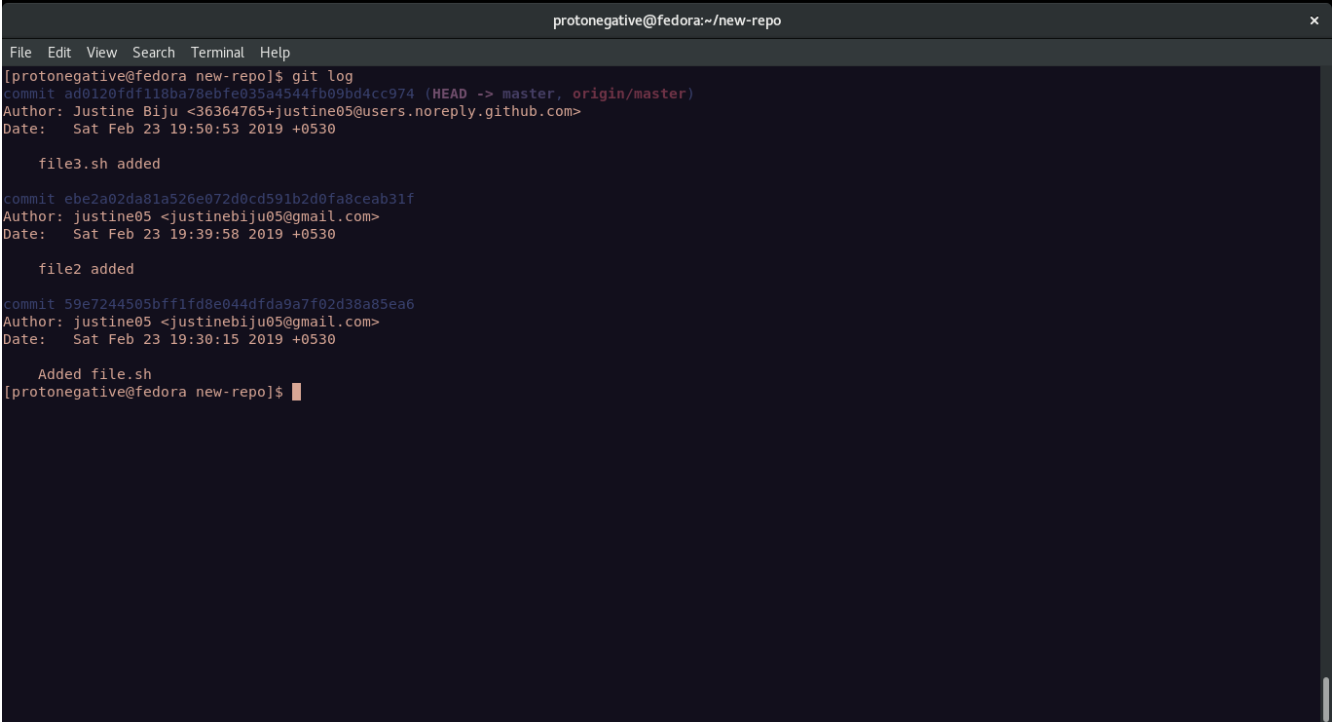
    file2 added

commit 59e7244505bfff1fd8e044dfda9a7f02d38a85ea6
Author: justine05 <justinebiju05@gmail.com>
Date:   Sat Feb 23 19:30:15 2019 +0530

    Added file.sh
[protonegative@fedora new-repo]$
```

Here list all the three commits I've done to 'new-repo'.

7. Revert

A terminal window titled 'protonegative@fedora:~/new-repo' with a menu bar (File, Edit, View, Search, Terminal, Help). The terminal shows the output of 'git log' for a repository named 'new-repo'. It lists three commits in reverse chronological order. The first commit (top) is 'ad0120fdf118ba78ebfe035a4544fb09bd4cc974' by Justine Biju, adding 'file3.sh'. The second commit is 'ebe2a02da81a526e072d0cd591b2d0fa8ceab31f' by justine05, adding 'file2'. The third commit (bottom) is '59e7244505bfff1fd8e044dfa9a7f02d38a85ea6' by justine05, adding 'file.sh'.

```
protonegative@fedora:~/new-repo
File Edit View Search Terminal Help
[protonegative@fedora new-repo]$ git log
commit ad0120fdf118ba78ebfe035a4544fb09bd4cc974 (HEAD -> master, origin/master)
Author: Justine Biju <36364765+justine05@users.noreply.github.com>
Date: Sat Feb 23 19:50:53 2019 +0530

    file3.sh added

commit ebe2a02da81a526e072d0cd591b2d0fa8ceab31f
Author: justine05 <justinebiju05@gmail.com>
Date: Sat Feb 23 19:39:58 2019 +0530

    file2 added

commit 59e7244505bfff1fd8e044dfa9a7f02d38a85ea6
Author: justine05 <justinebiju05@gmail.com>
Date: Sat Feb 23 19:30:15 2019 +0530

    Added file.sh
[protonegative@fedora new-repo]$
```

Now I want to revert to the second commit where file3.sh was not present.

So we take the commit id and use git checkout.

```
protonegative@fedora:~/new-repo
File Edit View Search Terminal Help
[protonegative@fedora new-repo]$ git checkout ebe2a02da81a526e072d0cd591b2d0fa8ceab31f
Note: checking out 'ebe2a02da81a526e072d0cd591b2d0fa8ceab31f'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by performing another checkout.

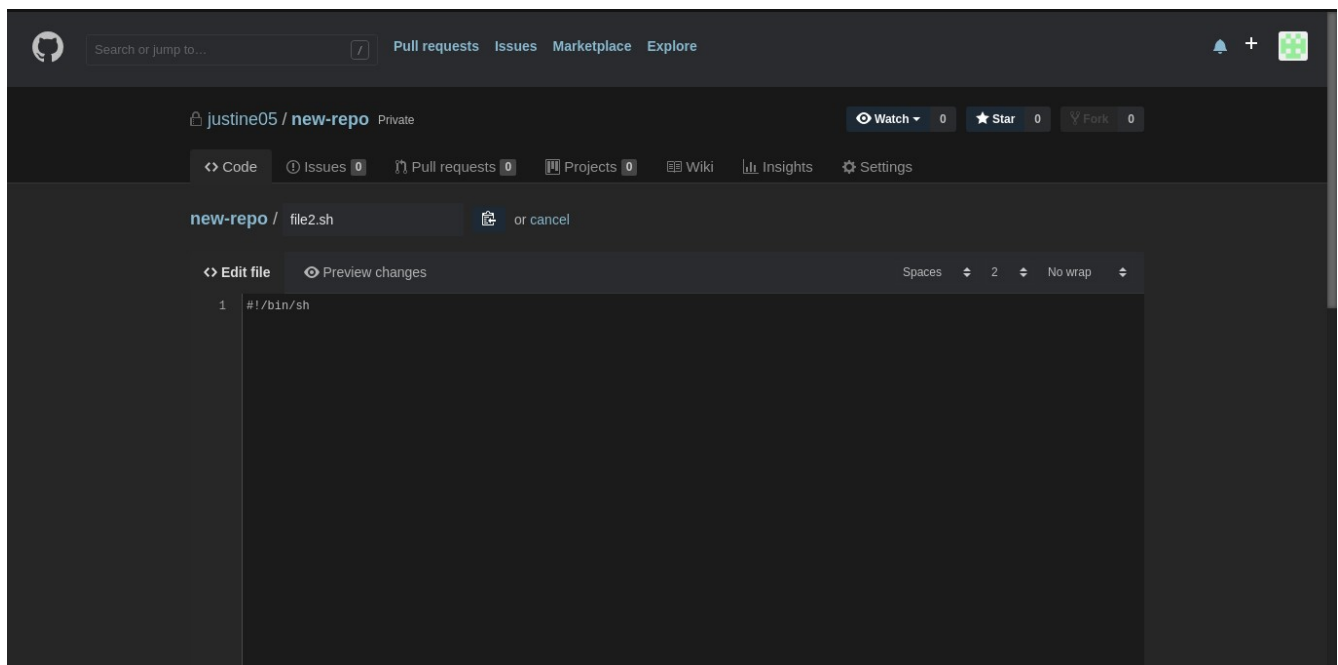
If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -b with the checkout command again. Example:

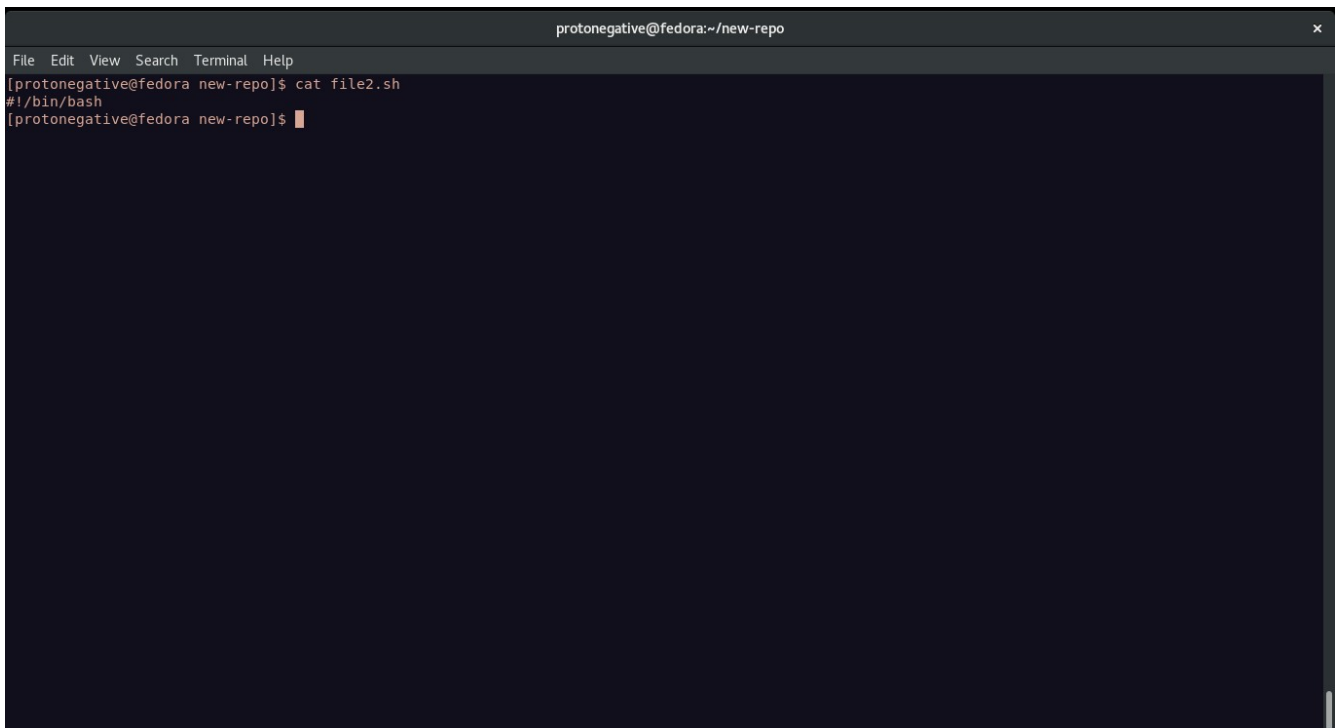
    git checkout -b <new-branch-name>

HEAD is now at ebe2a02... file2 added
[protonegative@fedora new-repo]$ ls
file2.sh  file.sh
[protonegative@fedora new-repo]$
```

Now we can see only two files are in the local repository.

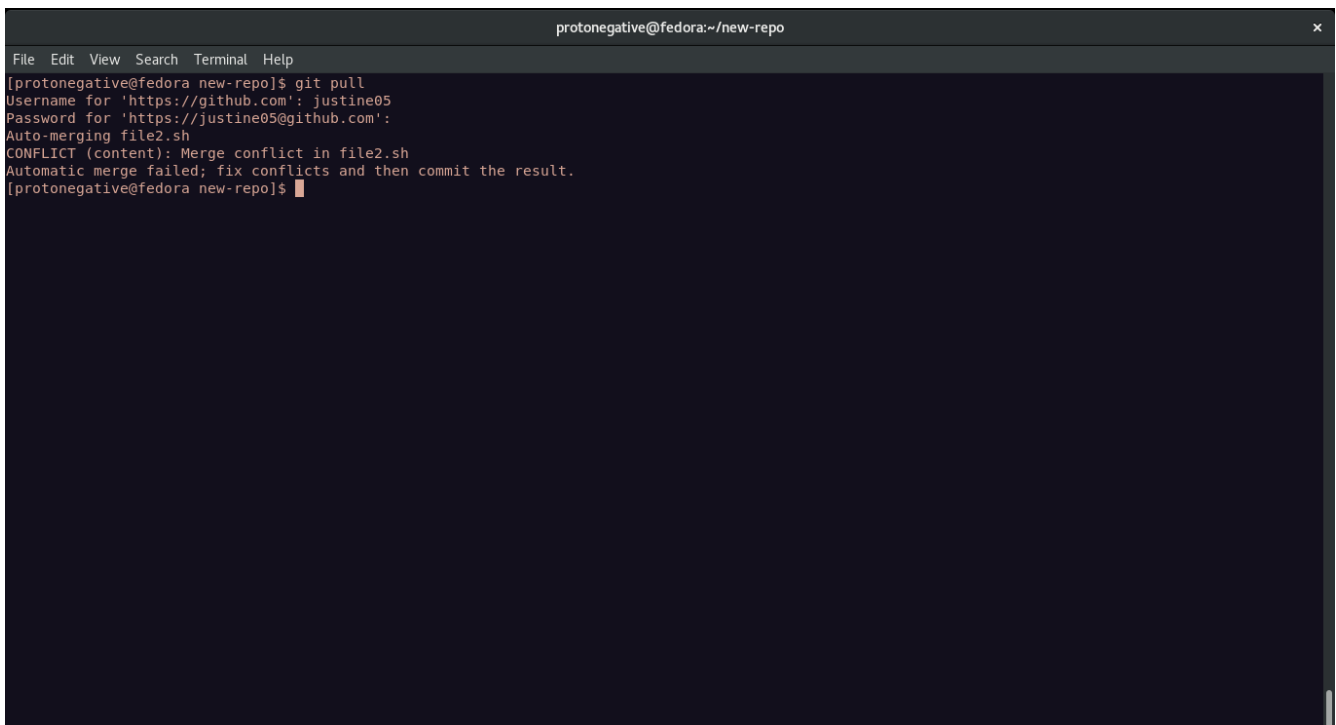
8. Conflicts and conflict resolution.



A terminal window titled 'protonegative@fedora:~/new-repo' with a menu bar (File, Edit, View, Search, Terminal, Help). The terminal shows the command 'cat file2.sh' being executed, which outputs the content of the file: '#!/bin/bash' followed by a blank line and a prompt character. The prompt is a red square.

```
protonegative@fedora:~/new-repo
File Edit View Search Terminal Help
[protonegative@fedora new-repo]$ cat file2.sh
#!/bin/bash
[protonegative@fedora new-repo]$
```

From the above two screenshots the content of file2.sh is conflicting hence pulling file2.sh to local repo will cause a merge conflict.

A terminal window titled 'protonegative@fedora:~/new-repo' with a menu bar (File, Edit, View, Search, Terminal, Help). The terminal shows the command 'git pull' being executed. It prompts for a username and password for 'https://github.com': justine05. It then shows 'Auto-merging file2.sh' followed by a conflict message: 'CONFLICT (content): Merge conflict in file2.sh' and 'Automatic merge failed; fix conflicts and then commit the result.' The prompt is a red square.

```
protonegative@fedora:~/new-repo
File Edit View Search Terminal Help
[protonegative@fedora new-repo]$ git pull
Username for 'https://github.com': justine05
Password for 'https://justine05@github.com':
Auto-merging file2.sh
CONFLICT (content): Merge conflict in file2.sh
Automatic merge failed; fix conflicts and then commit the result.
[protonegative@fedora new-repo]$
```

And hence the file shows the following.

```
protonegative@fedora:~/new-repo
File Edit View Search Terminal Help
1 <<<<<< HEAD
2 1 #!/bin/bash
3 2 =====
4 3 #!/bin/sh^M
5 4 >>>>>> c7f9f03d98ac78c8fa07190b1d60302cc093d51b

"file2.sh" 5L, 93C 1,1 All
```

Now select which line do you want to keep and push again.

```
protonegative@fedora:~/new-repo
File Edit View Search Terminal Help
[protonegative@fedora new-repo]$ git pull
Username for 'https://github.com': justine05
Password for 'https://justine05@github.com':
Auto-merging file2.sh
CONFLICT (content): Merge conflict in file2.sh
Automatic merge failed; fix conflicts and then commit the result.
[protonegative@fedora new-repo]$ vim file2.sh
[protonegative@fedora new-repo]$ git add .
[protonegative@fedora new-repo]$ git commit -m "Changing file2.sh"
[master 8a3450d] Changing file2.sh
[protonegative@fedora new-repo]$ git push
Username for 'https://github.com': justine05
Password for 'https://justine05@github.com':
Counting objects: 5, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (5/5), 601 bytes | 601.00 KiB/s, done.
Total 5 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/justine05/new-repo.git
   c7f9f03..8a3450d  master -> master
[protonegative@fedora new-repo]$
```

Hence this merge conflict is solved.

Result: Understood how to set up and manage a GIT repository.

eof
