

1. Git Setup

```
Activities Terminal Tue 12:06 shubham@shubham: ~
File Edit View Search Terminal Help
shubham@shubham:~$ sudo apt-get install git
[sudo] password for shubham:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  git-man liberror-perl
Suggested packages:
  git-daemon-run | git-daemon-sysvinit git-doc git-el git-email git-gui gitk gitweb git-cvs git-mediawiki git-svn
The following NEW packages will be installed:
  git git-man liberror-perl
0 upgraded, 3 newly installed, 0 to remove and 623 not upgraded.
Need to get 4,738 kB of archives.
After this operation, 33.9 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://in.archive.ubuntu.com/ubuntu bionic/main amd64 liberror-perl all 0.17025-1 [22.8 kB]
Get:2 http://in.archive.ubuntu.com/ubuntu bionic-updates/main amd64 git-man all 1:2.17.1-1ubuntu0.5 [803 kB]
Get:3 http://in.archive.ubuntu.com/ubuntu bionic-updates/main amd64 git amd64 1:2.17.1-1ubuntu0.5 [3,912 kB]
Fetched 4,738 kB in 1s (3,661 kB/s)
Selecting previously unselected package liberror-perl.
(Reading database ... 125727 files and directories currently installed.)
Preparing to unpack .../liberror-perl_0.17025-1_all.deb ...
Unpacking liberror-perl (0.17025-1) ...
Selecting previously unselected package git-man.
Preparing to unpack .../git-man_1%3a2.17.1-1ubuntu0.5_all.deb ...
Unpacking git-man (1:2.17.1-1ubuntu0.5) ...
Selecting previously unselected package git.
Preparing to unpack .../git_1%3a2.17.1-1ubuntu0.5_amd64.deb ...
Unpacking git (1:2.17.1-1ubuntu0.5) ...
Setting up git-man (1:2.17.1-1ubuntu0.5) ...
Setting up liberror-perl (0.17025-1) ...
Processing triggers for man-db (2.8.3-2) ...
Setting up git (1:2.17.1-1ubuntu0.5) ...
```

```
shubham@shubham:~$ git config --global user.name shubhamjain2211
shubham@shubham:~$ git config --global user.email shubham.jain1@tothenew.com
shubham@shubham:~$ ssh-keygen
Bad escape character 'ygen'.
shubham@shubham:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/shubham/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/shubham/.ssh/id_rsa.
Your public key has been saved in /home/shubham/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:9JwMhibUZx00dVUgJZ63E/NAX9tQSWW8NB0/qZvqADI shubham@shubham
The key's randomart image is:
+---[RSA 2048]----+
.. ..o.. o==%*|
..+o ...*+B
..oo=.. o.+
o o = .o *.o
E.S = o *
o . = .
. o .
..
.o
+---[SHA256]-----+
shubham@shubham:~$
```

Activities

Firefox Web Browser

Tue 12:20

SSH and GPG keys - Mozilla Firefox

Inbox (2) - shubham.jain x Learning | Dashboard x Introduction to Version x SSH and GPG keys x

GitHub, Inc. (US) | https://github.com/settings/keys

...

...

...

Search or jump to...

Pull requests

Issues

Marketplace

Explore

Personal settings

Profile

Account

Security

Security log

Emails

Notifications

Billing

SSH and GPG keys

Blocked users

Repositories

Organizations

Saved replies

Applications

SSH keys

New SSH key

This is a list of SSH keys associated with your account. Remove any keys that you do not recognize.

shubham@shubham

92:b8:18:bb:59:8e:60:1c:ea:05:08:b3:29:58:fd:83

Added on Feb 4, 2020

Never used — Read/write

SSH

Delete

Check out our guide to [generating SSH keys](#) or troubleshoot [common SSH Problems](#).

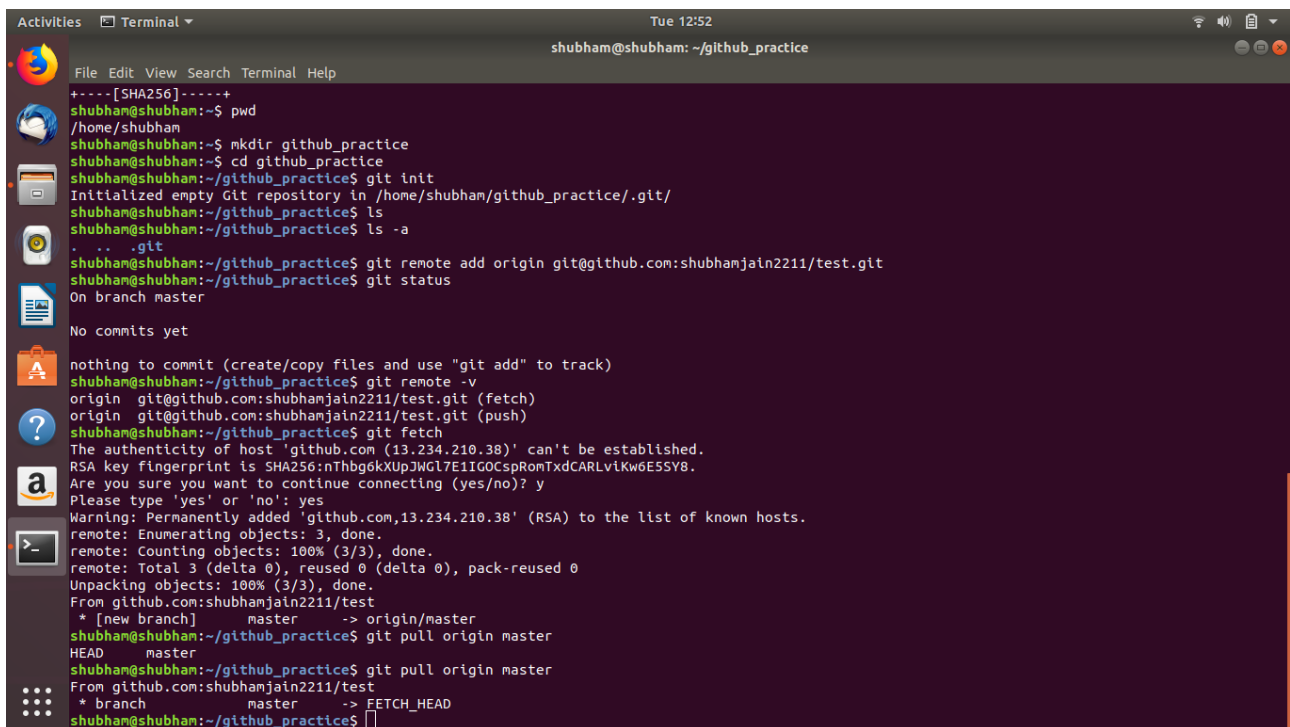
GPG keys

New GPG key

There are no GPG keys associated with your account.

Learn how to [generate a GPG key](#) and add it to your account.

2. Initialize a Git Repository



```
shubham@shubham: ~/github_practice
+----[SHA256]-----+
shubham@shubham:~$ pwd
/home/shubham
shubham@shubham:~$ mkdir github_practice
shubham@shubham:~$ cd github_practice
shubham@shubham:~/github_practice$ git init
Initialized empty Git repository in /home/shubham/github_practice/.git/
shubham@shubham:~/github_practice$ ls
.  ..  .git
shubham@shubham:~/github_practice$ git remote add origin git@github.com:shubhamjain2211/test.git
shubham@shubham:~/github_practice$ git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)
shubham@shubham:~/github_practice$ git remote -v
origin  git@github.com:shubhamjain2211/test.git (fetch)
origin  git@github.com:shubhamjain2211/test.git (push)
shubham@shubham:~/github_practice$ git fetch
The authenticity of host 'github.com (13.234.210.38)' can't be established.
RSA key fingerprint is SHA256:nThbg6kXUpJWGL7E1IGOCspRomTxdCARLviKw6E5SY8.
Are you sure you want to continue connecting (yes/no)? y
Please type 'yes' or 'no': yes
Warning: Permanently added 'github.com,13.234.210.38' (RSA) to the list of known hosts.
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
From github.com:shubhamjain2211/test
* [new branch]      master      -> origin/master
shubham@shubham:~/github_practice$ git pull origin master
HEAD master
shubham@shubham:~/github_practice$ git pull origin master
From github.com:shubhamjain2211/test
* branch            master      -> FETCH_HEAD
shubham@shubham:~/github_practice$
```

3. Add files to the repository

```
Activities Terminal Tue 13:12 shubham@shubham: ~/github_practice
File Edit View Search Terminal Help
shubham@shubham:~/github_practice$ touch file
shubham@shubham:~/github_practice$ nano file
shubham@shubham:~/github_practice$ cat file
Hello World

shubham@shubham:~/github_practice$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)

        file
        file1

nothing added to commit but untracked files present (use "git add" to track)
shubham@shubham:~/github_practice$ git add file
shubham@shubham:~/github_practice$ git status
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

        new file:   file

Untracked files:
  (use "git add <file>..." to include in what will be committed)

        file1

shubham@shubham:~/github_practice$ git commit -m
error: switch 'm' requires a value
usage: git commit [<options>] [--] <pathspec>...

        -q, --quiet          suppress summary after successful commit
        -v, --verbose        show diff in commit message template

Commit message options
        -F, --file <file>    read message from file
        --author <author>    override author for commit
        --date <date>        override date for commit
```

```
Activities Terminal Tue 13:16 shubham@shubham: ~/github_practice
File Edit View Search Terminal Help
-i, --include                add specified files to index for commit
--interactive                interactively add files
-p, --patch                  interactively add changes
-o, --only                   commit only specified files
-n, --no-verify              bypass pre-commit and commit-msg hooks
--dry-run                    show what would be committed
--short                      show status concisely
--branch                     show branch information
--ahead-behind                compute full ahead/behind values
--porcelain                  machine-readable output
--long                       show status in long format (default)
-z, --null                   terminate entries with NUL
--amend                      amend previous commit
--no-post-rewrite            bypass post-rewrite hook
-u, --untracked-files[=<mode>] show untracked files, optional modes: all, normal, no. (Default: all)

shubham@shubham:~/github_practice$ git commit -m "Added new file for initial commit."
[master eda90cc] Added new file for initial commit.
1 file changed, 2 insertions(+)
create mode 100644 file
shubham@shubham:~/github_practice$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)

        file1

nothing added to commit but untracked files present (use "git add" to track)
shubham@shubham:~/github_practice$ git push origin master
Counting objects: 3, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 301 bytes | 301.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To github.com:shubhamjain2211/test.git
   15860be..eda90cc  master -> master
shubham@shubham:~/github_practice$
```

Activities

Firefox Web Browser

Tue 13:18

shubhamjain2211/test - Mozilla Firefox

Inbox (1) - shubham.jain ×

Learning | Dashboard ×

Introduction to Version ×

shubhamjain2211/test ×

+

← → ↻ 🏠

🔒 GitHub, Inc. (US) | https://github.com/shubhamjain2211/test

⋮ 📧 ☆

🔍 📄 ☰

shubhamjain2211 / test

👁 Unwatch 1 ⭐ Star 0 🍴 Fork 0

<> Code

🔔 Issues 0

🔗 Pull requests 0

⚙ Actions

📁 Projects 0

📖 Wiki

🛡 Security

📊 Insights

⚙ Settings

No description, website, or topics provided.

Edit

Manage topics

📌 2 commits

🌿 1 branch

📦 0 packages

📦 0 releases

👤 1 contributor

Branch: master

New pull request

Create new file

Upload files

Find file

Clone or download

shubhamjain2211 Added new file for initial commit. Latest commit eda0e6cc 5 minutes ago

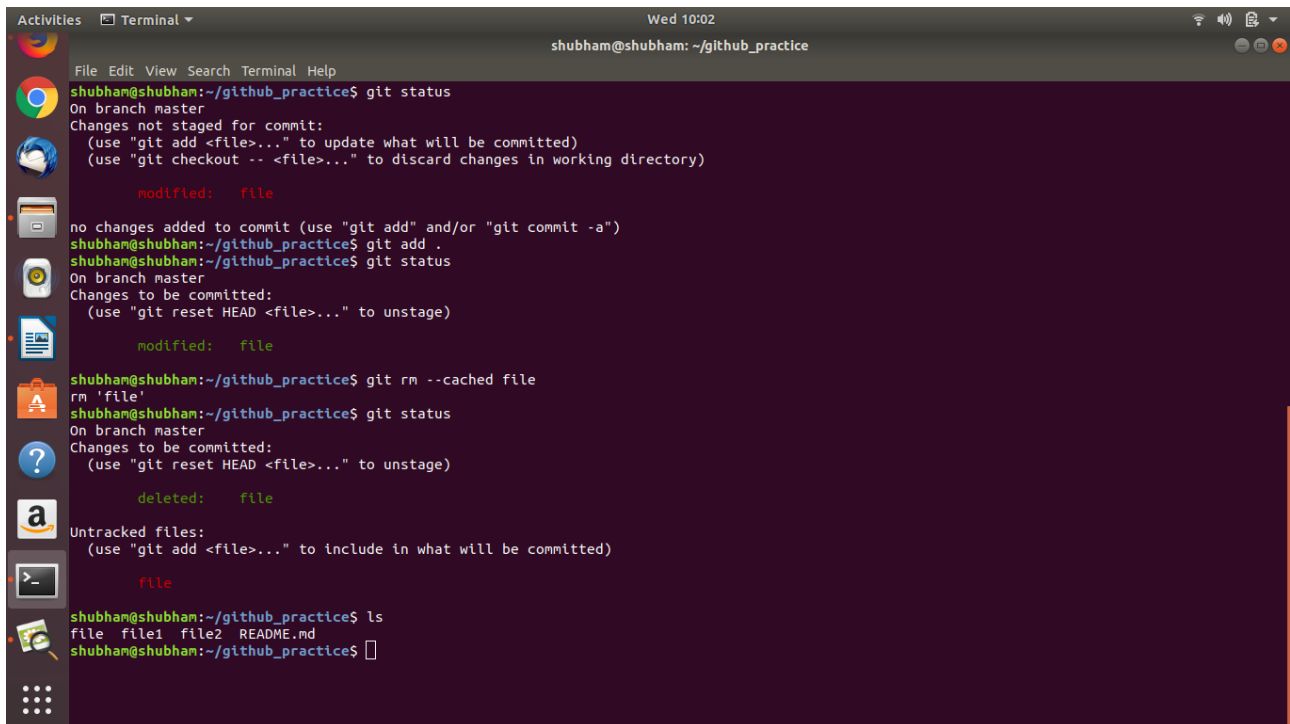
📄 README.md Initial commit 1 hour ago

📄 file Added new file for initial commit. 5 minutes ago

📄 README.md

test

4. Unstage 1 file



A terminal window titled 'shubham@shubham: ~/github_practice' showing a series of Git commands and their outputs. The terminal has a dark purple background with light green text. The left sidebar shows various application icons. The terminal output is as follows:

```
shubham@shubham:~/github_practice$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   file

no changes added to commit (use "git add" and/or "git commit -a")
shubham@shubham:~/github_practice$ git add .
shubham@shubham:~/github_practice$ git status
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

        modified:   file

shubham@shubham:~/github_practice$ git rm --cached file
rm 'file'
shubham@shubham:~/github_practice$ git status
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

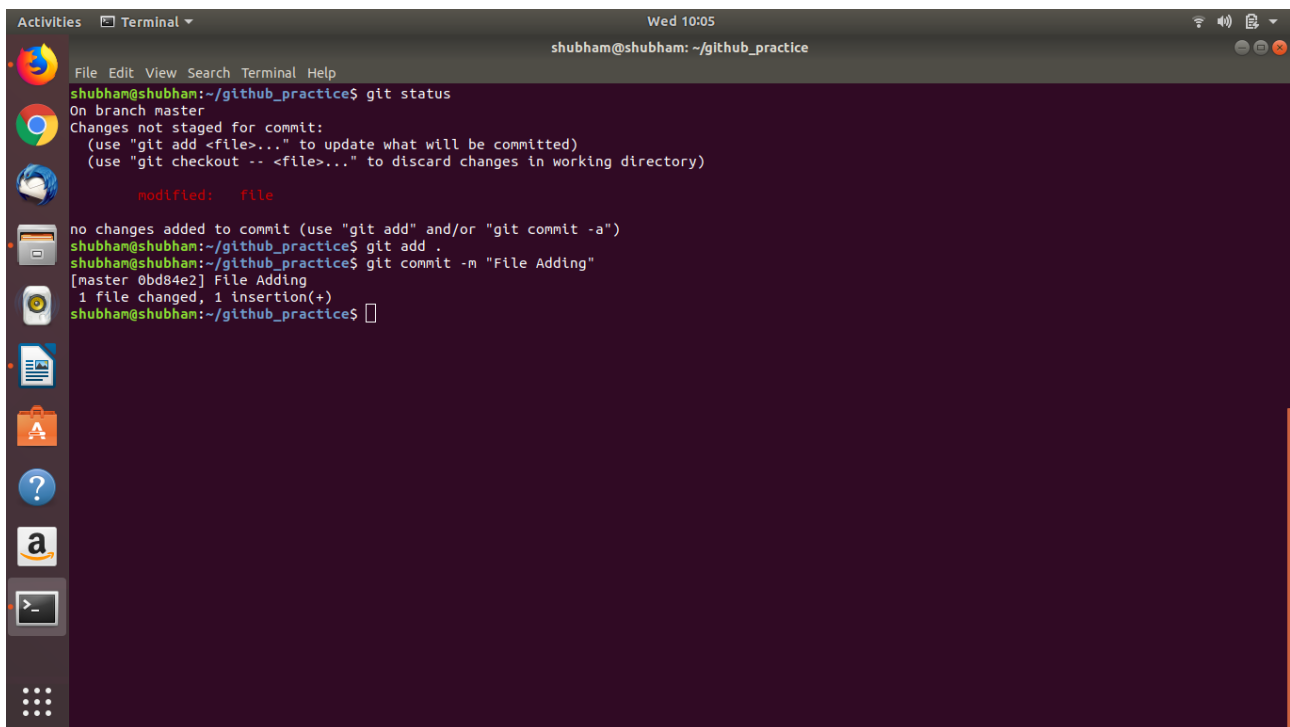
        deleted:    file

Untracked files:
  (use "git add <file>..." to include in what will be committed)

        file

shubham@shubham:~/github_practice$ ls
file  file1  file2  README.md
shubham@shubham:~/github_practice$
```

5. Commit the file



A terminal window titled "Terminal" with a menu bar (File, Edit, View, Search, Terminal, Help) and a status bar (Wed 10:05, shubham@shubham: ~/github_practice). The terminal shows the following commands and output:

```
shubham@shubham:~/github_practice$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   file

no changes added to commit (use "git add" and/or "git commit -a")
shubham@shubham:~/github_practice$ git add .
shubham@shubham:~/github_practice$ git commit -m "File Adding"
[master 0bd84e2] File Adding
1 file changed, 1 insertion(+)
shubham@shubham:~/github_practice$
```

6. Add a remote

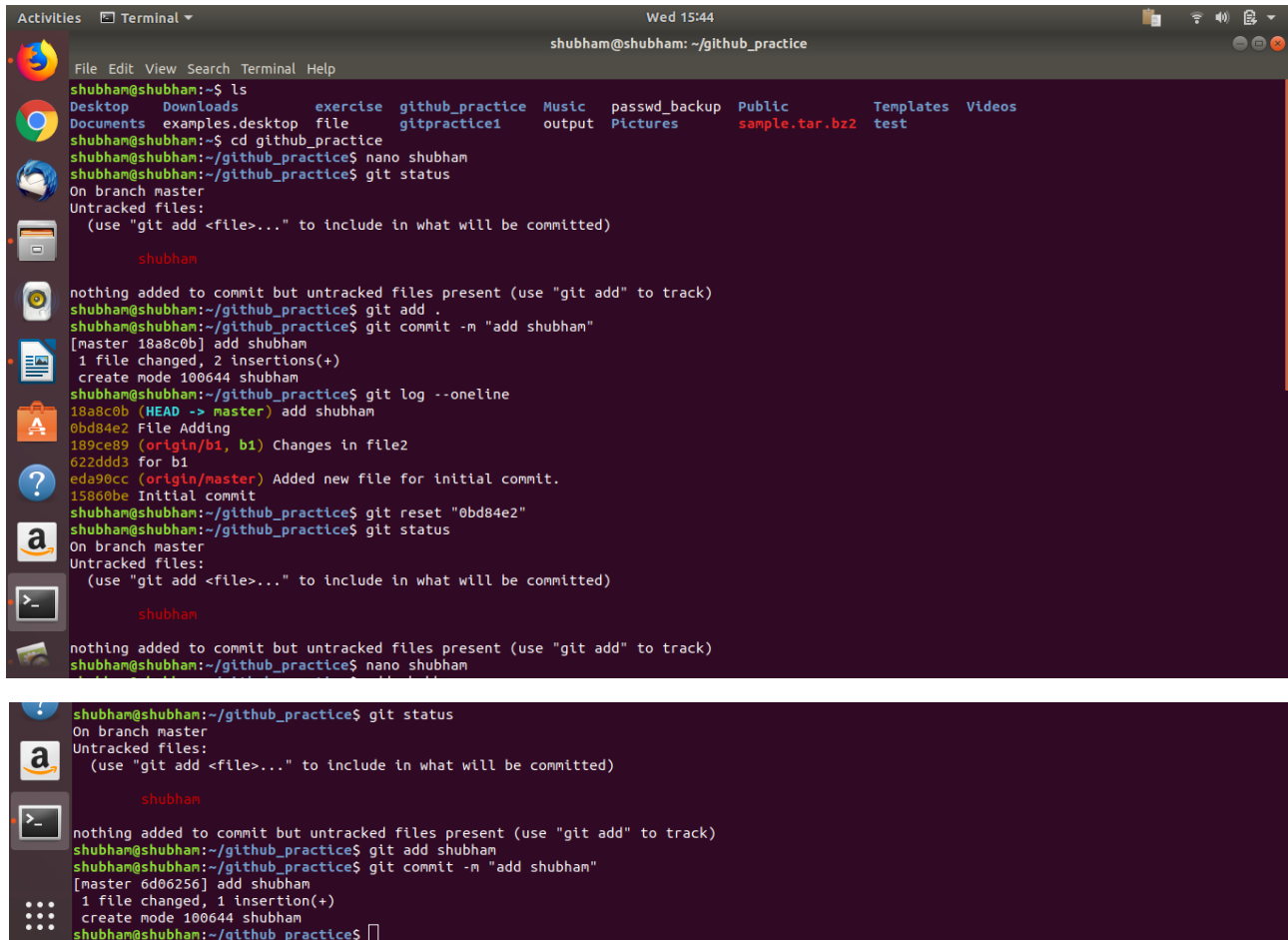


```
shubham@shubham:~/github_practice$ git remote add origin git@github.com:shubhamjain2211/test.git
shubham@shubham:~/github_practice$ git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)
shubham@shubham:~/github_practice$ git remote -v
origin  git@github.com:shubhamjain2211/test.git (fetch)
origin  git@github.com:shubhamjain2211/test.git (push)
```


7. Undo changes to a particular file



```
shubham@shubham:~$ ls
Desktop  Downloads  exercise  github_practice  Music  passwd_backup  Public  Templates  Videos
Documents  examples.desktop  file  gitpractice1  output  Pictures  sample.tar.bz2  test

shubham@shubham:~$ cd github_practice
shubham@shubham:~/github_practice$ nano shubham
shubham@shubham:~/github_practice$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)

        shubham

nothing added to commit but untracked files present (use "git add" to track)
shubham@shubham:~/github_practice$ git add .
shubham@shubham:~/github_practice$ git commit -m "add shubham"
[master 18a8c0b] add shubham
1 file changed, 2 insertions(+)
create mode 100644 shubham
shubham@shubham:~/github_practice$ git log --oneline
18a8c0b (HEAD -> master) add shubham
0bd84e2 File Adding
189ce89 (origin/b1, b1) Changes in file2
622ddd3 for b1
eda90cc (origin/master) Added new file for initial commit.
15860be Initial commit
shubham@shubham:~/github_practice$ git reset "0bd84e2"
shubham@shubham:~/github_practice$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)

        shubham

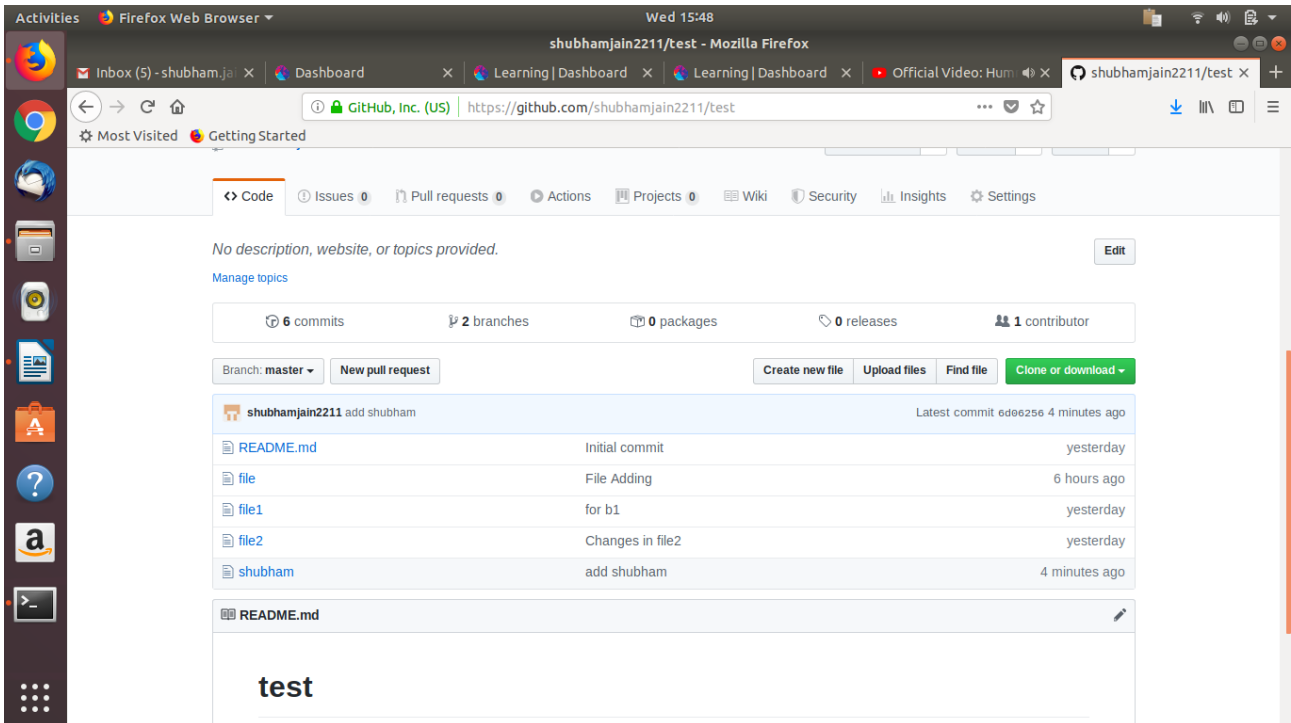
nothing added to commit but untracked files present (use "git add" to track)
shubham@shubham:~/github_practice$ nano shubham
shubham@shubham:~/github_practice$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)

        shubham

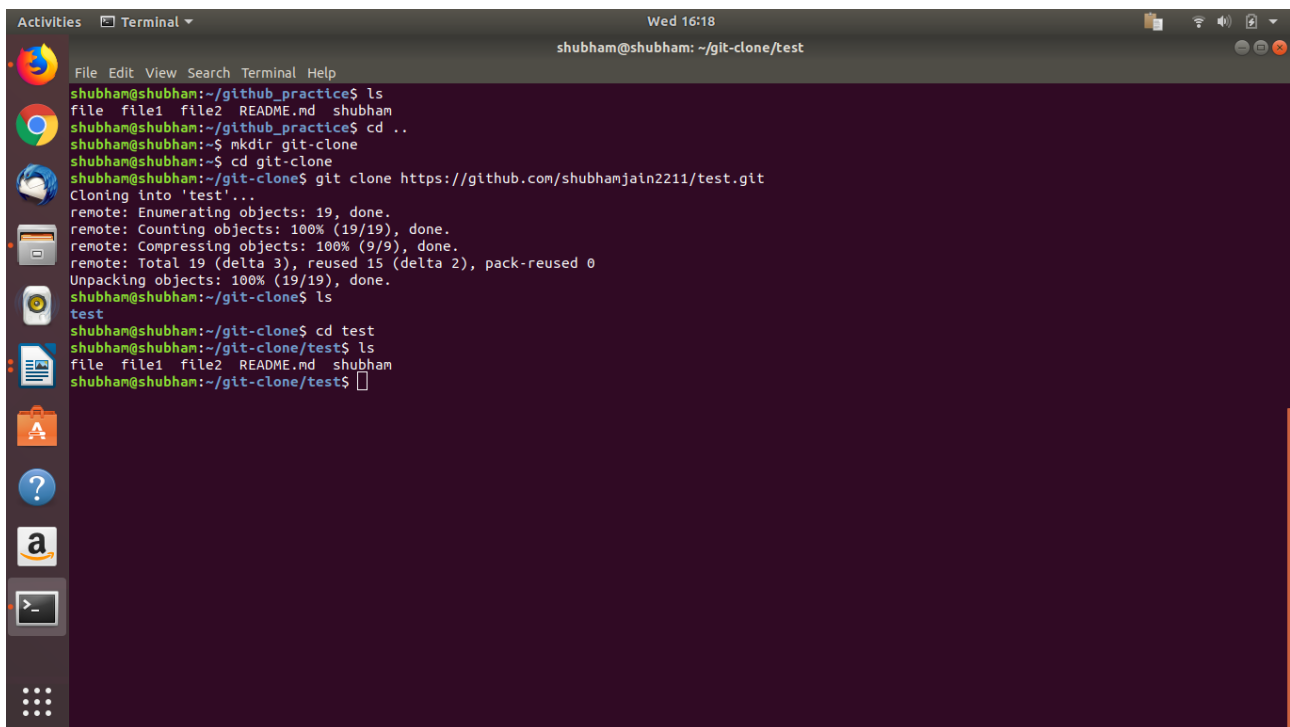
nothing added to commit but untracked files present (use "git add" to track)
shubham@shubham:~/github_practice$ git add shubham
shubham@shubham:~/github_practice$ git commit -m "add shubham"
[master 6d06256] add shubham
1 file changed, 1 insertion(+)
create mode 100644 shubham
shubham@shubham:~/github_practice$
```

8. Push changes to Github

```
create mode 100644 shubham
shubham@shubham:~/github_practice$ git fetch
shubham@shubham:~/github_practice$ git push origin master
Counting objects: 6, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (6/6), 563 bytes | 563.00 KiB/s, done.
Total 6 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), done.
To github.com:shubhamjain2211/test.git
   eda90cc..6d06256 master -> master
shubham@shubham:~/github_practice$
```



9. Clone the repository

A terminal window titled 'shubham@shubham: ~/git-clone/test' showing a series of commands and their outputs. The user starts in the ~/github_practice directory, lists files (file, file1, file2, README.md, shubham), moves to the parent directory, creates a new directory named git-clone, and then clones a repository from GitHub. The clone operation shows progress for enumerating, counting, compressing, and unpacking objects. Finally, the user lists the contents of the newly created 'test' directory, which matches the original directory's contents.

```
shubham@shubham:~/github_practice$ ls
file file1 file2 README.md shubham
shubham@shubham:~/github_practice$ cd ..
shubham@shubham:~$ mkdir git-clone
shubham@shubham:~$ cd git-clone
shubham@shubham:~/git-clone$ git clone https://github.com/shubhamjain2211/test.git
Cloning into 'test'...
remote: Enumerating objects: 19, done.
remote: Counting objects: 100% (19/19), done.
remote: Compressing objects: 100% (9/9), done.
remote: Total 19 (delta 3), reused 15 (delta 2), pack-reused 0
Unpacking objects: 100% (19/19), done.
shubham@shubham:~/git-clone$ ls
test
shubham@shubham:~/git-clone$ cd test
shubham@shubham:~/git-clone/test$ ls
file file1 file2 README.md shubham
shubham@shubham:~/git-clone/test$
```

10. Add changes to one of the copies and pull the changes in the other.

```
Activities Terminal Wed 16:43
shubham@shubham: /home

File Edit View Search Terminal Help
shubham@shubham:~/github_practice$ nano ques0
shubham@shubham:~/github_practice$ git add .
shubham@shubham:~/github_practice$ git commit -m "adding ques10"
[master 8f90ed9] adding ques10
2 files changed, 1 insertion(+), 1 deletion(-)
create mode 100644 ques0
delete mode 100644 shubham
shubham@shubham:~/github_practice$ git push origin master
Counting objects: 2, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 233 bytes | 233.00 KiB/s, done.
Total 2 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:shubhamjain2211/test.git
8e83204..8f90ed9 master -> master
shubham@shubham:~/github_practice$ cd ..
shubham@shubham:~$ cd ..
shubham@shubham: /home$ ls
lost+found sample.tar.bz2 shubham test
shubham@shubham: /home$
```

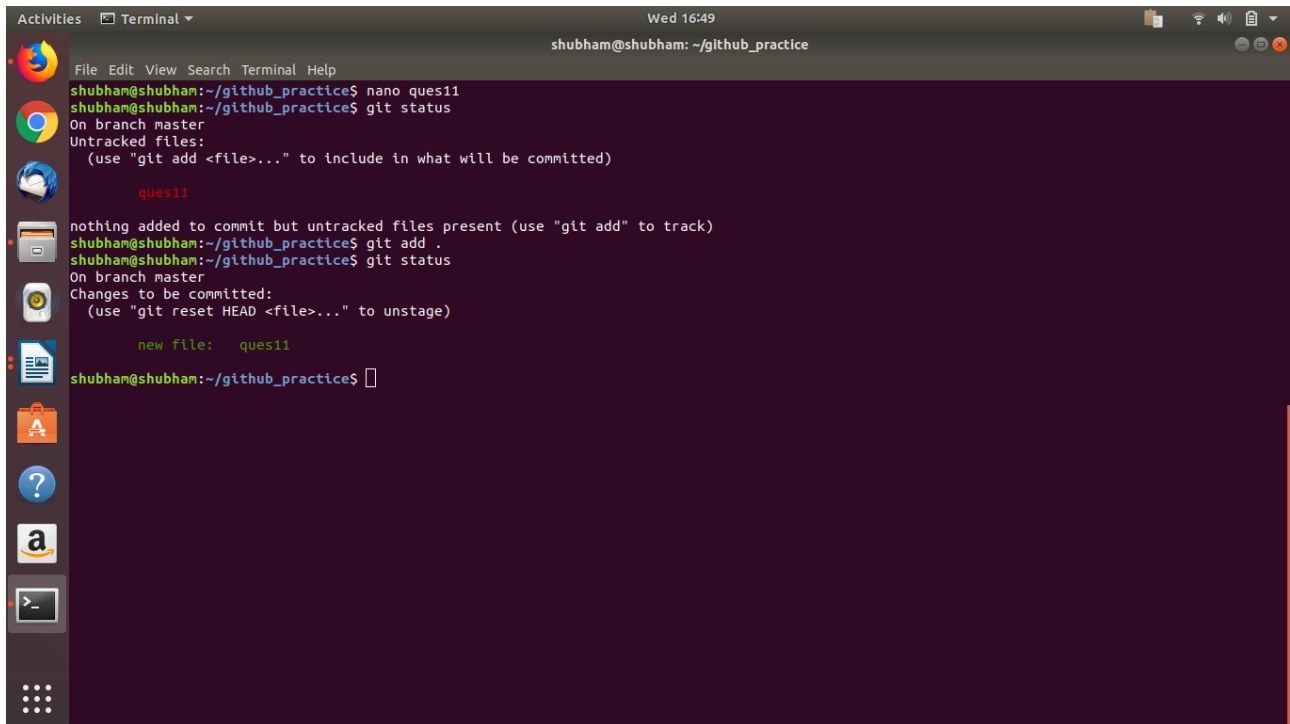
```
Activities Terminal Wed 16:44
shubham@shubham: ~/git-clone/test

File Edit View Search Terminal Help
shubham@shubham:~/git-clone/test$ git fetch
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (1/1), done.
remote: Total 2 (delta 1), reused 2 (delta 1), pack-reused 0
Unpacking objects: 100% (2/2), done.
From https://github.com/shubhamjain2211/test
8e83204..8f90ed9 master -> origin/master
shubham@shubham:~/git-clone/test$ git pull origin master
From https://github.com/shubhamjain2211/test
* branch master -> FETCH_HEAD
Updating 6d06256..8f90ed9
Fast-forward
 new | 1 +
 ques0 | 1 +
 shubham | 1 -
3 files changed, 2 insertions(+), 1 deletion(-)
create mode 100644 new
create mode 100644 ques0
delete mode 100644 shubham
shubham@shubham:~/git-clone/test$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
shubham@shubham:~/git-clone/test$ ls
file file1 file2 new ques0 README.md
shubham@shubham:~/git-clone/test$
```

11. Check differences between a file and its staged version

- When some changes are done to a file, git marks the file as “modified”
- After we stage the modified file using `git add <file-name>`, the modified files are now staged and are ready to be committed before pushing it to remote repository



A terminal window titled "shubham@shubham: ~/github_practice" showing the following commands and output:

```
shubham@shubham:~/github_practice$ nano ques11
shubham@shubham:~/github_practice$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)

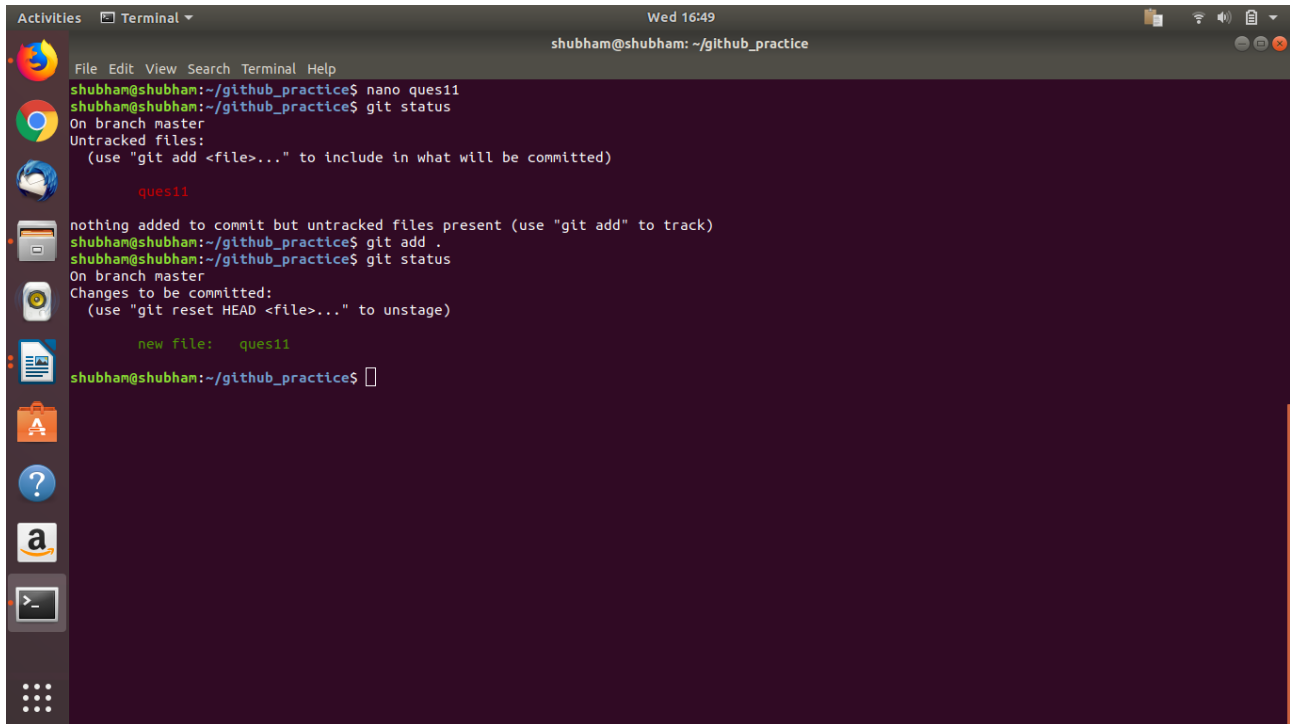
        ques11

nothing added to commit but untracked files present (use "git add" to track)
shubham@shubham:~/github_practice$ git add .
shubham@shubham:~/github_practice$ git status
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

        new file:   ques11

shubham@shubham:~/github_practice$
```

12. Ignore a few files to be checked in



A terminal window titled "shubham@shubham: ~/github_practice" showing the following commands and output:

```
shubham@shubham:~/github_practice$ nano ques11
shubham@shubham:~/github_practice$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)

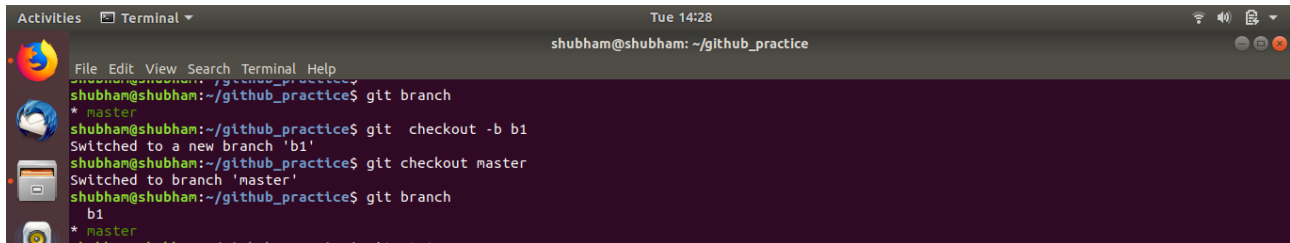
        ques11

nothing added to commit but untracked files present (use "git add" to track)
shubham@shubham:~/github_practice$ git add .
shubham@shubham:~/github_practice$ git status
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

        new file:   ques11

shubham@shubham:~/github_practice$
```

13. Create a new branch.

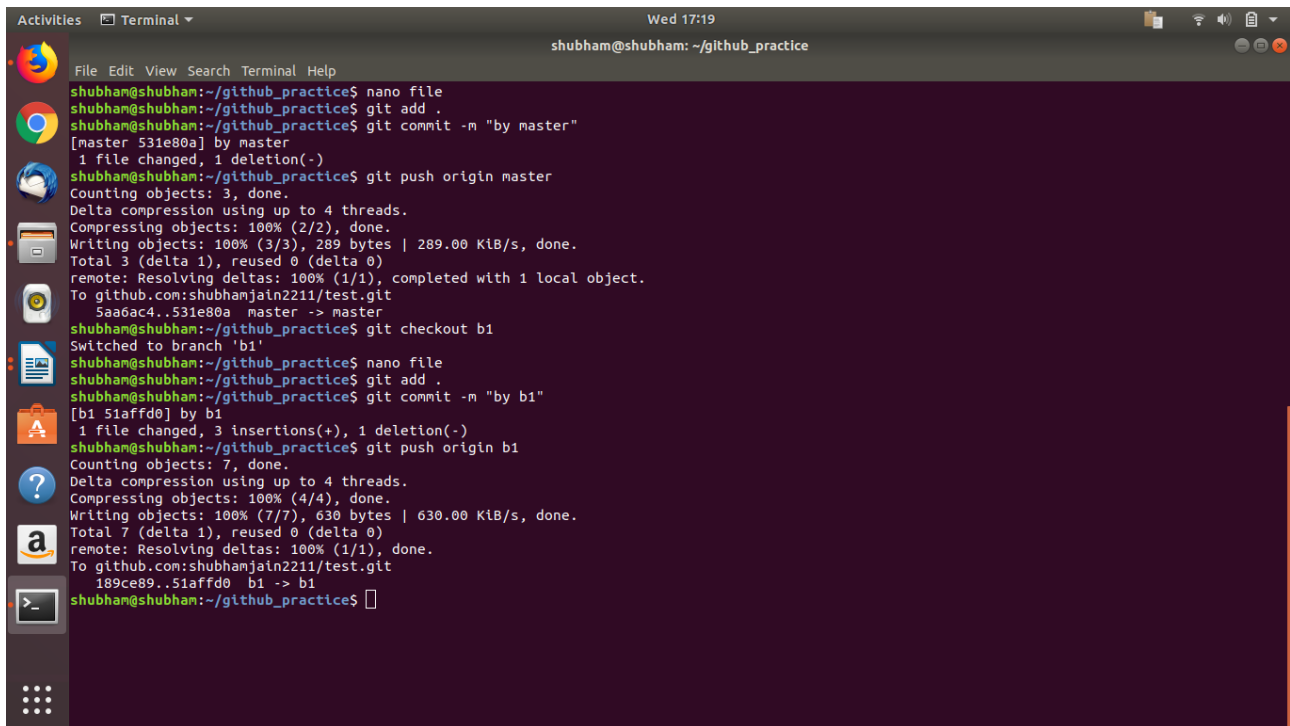
A terminal window titled 'Terminal' with a menu bar (File, Edit, View, Search, Terminal, Help) and a status bar (Tue 14:28, shubham@shubham: ~/github_practice). The terminal shows the following commands and output:

```
shubham@shubham:~/github_practice$ git branch
* master
shubham@shubham:~/github_practice$ git checkout -b b1
Switched to a new branch 'b1'
shubham@shubham:~/github_practice$ git checkout master
Switched to branch 'master'
shubham@shubham:~/github_practice$ git branch
  b1
* master
```

14. Diverge them with commits

```
shubham@shubham:~/github_practice$ git checkout b1
A       file1
A       file2
Already on 'b1'
shubham@shubham:~/github_practice$ git commit -m "for b1"
[b1 622ddd3] for b1
 2 files changed, 3 insertions(+)
 create mode 100644 file1
 create mode 100644 file2
shubham@shubham:~/github_practice$ git status
On branch b1
nothing to commit, working tree clean
shubham@shubham:~/github_practice$ ls
file  file1  file2  README.md
shubham@shubham:~/github_practice$ nano file2
shubham@shubham:~/github_practice$ git checkout master
error: Your local changes to the following files would be overwritten by checkout:
      file2
Please commit your changes or stash them before you switch branches.
Aborting
shubham@shubham:~/github_practice$ git add file2
shubham@shubham:~/github_practice$ git commit -m "Changes in file2"
[b1 189ce89] Changes in file2
 1 file changed, 2 insertions(+)
shubham@shubham:~/github_practice$ git push origin b1
Warning: Permanently added the RSA host key for IP address '13.234.176.102' to the list of known hosts.
Counting objects: 7, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (7/7), 621 bytes | 621.00 KiB/s, done.
Total 7 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), done.
remote:
remote: Create a pull request for 'b1' on GitHub by visiting:
remote:   https://github.com/shubhamjain2211/test/pull/new/b1
remote:
```


15. Edit the same file at the same line on both branches and commit



```
shubham@shubham: ~/github_practice
File Edit View Search Terminal Help
shubham@shubham:~/github_practice$ nano file
shubham@shubham:~/github_practice$ git add .
shubham@shubham:~/github_practice$ git commit -m "by master"
[master 531e80a] by master
1 file changed, 1 deletion(-)
shubham@shubham:~/github_practice$ git push origin master
Counting objects: 3, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 289 bytes | 289.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:shubhamjain2211/test.git
5aa6ac4..531e80a master -> master
shubham@shubham:~/github_practice$ git checkout b1
Switched to branch 'b1'
shubham@shubham:~/github_practice$ nano file
shubham@shubham:~/github_practice$ git add .
shubham@shubham:~/github_practice$ git commit -m "by b1"
[b1 51affd0] by b1
1 file changed, 3 insertions(+), 1 deletion(-)
shubham@shubham:~/github_practice$ git push origin b1
Counting objects: 7, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (7/7), 630 bytes | 630.00 KiB/s, done.
Total 7 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), done.
To github.com:shubhamjain2211/test.git
189ce89..51affd0 b1 -> b1
shubham@shubham:~/github_practice$
```

16. Try merging and resolve merge conflicts

- When we merged both the branches (using "git merge"); git detected different content in the file at the same line, but in different branches.

```
Activities Terminal Wed 17:40 shubham@shubham: ~/github_practice
shubham@shubham:~/github_practice$ git branch
* b1
shubham@shubham:~/github_practice$ git merge b1
Auto-merging file
CONFLICT (content): Merge conflict in file
Automatic merge failed; fix conflicts and then commit the result.
shubham@shubham:~/github_practice$ cat file
Hello World

<<<<<< HEAD
changes by master
=====
changes by branch b1
>>>>>> b1
shubham@shubham:~/github_practice$ nano file
shubham@shubham:~/github_practice$ cat file
```

```
shubham@shubham:~/github_practice$ cat file
Hello World

changes by master
changes by branch b1
shubham@shubham:~/github_practice$ 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)

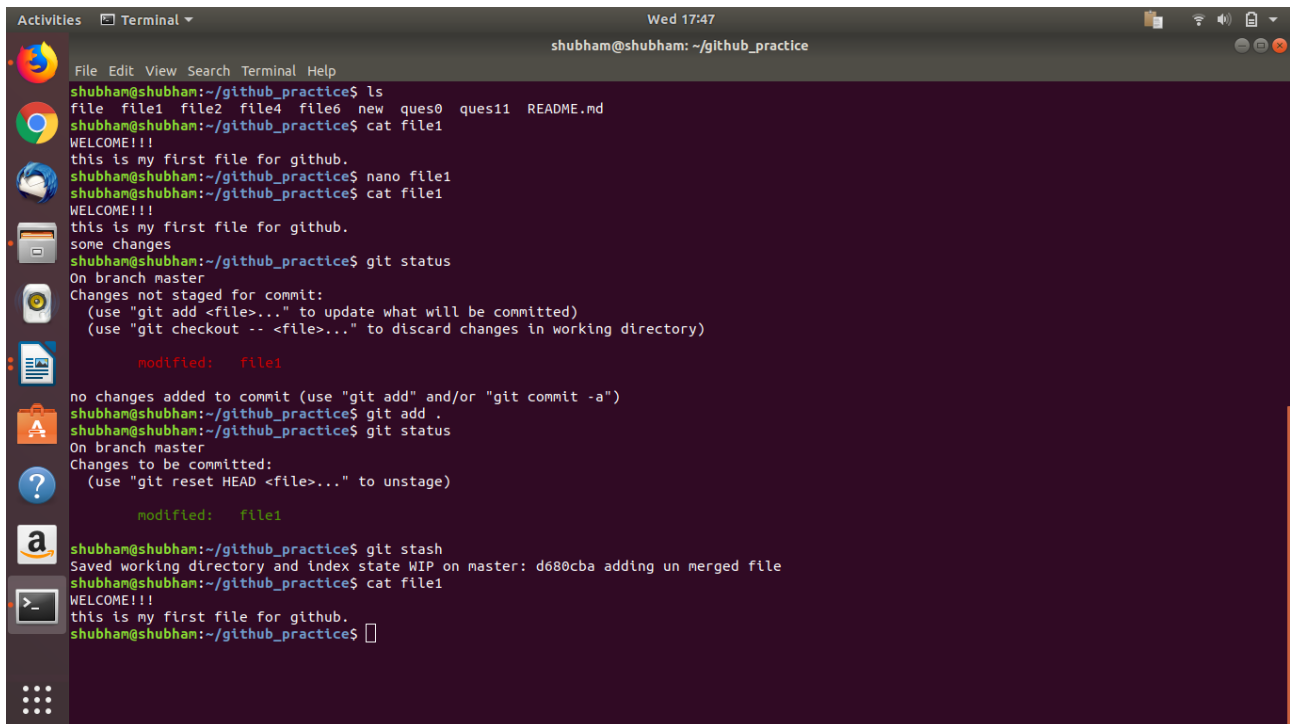
        both modified:   file

no changes added to commit (use "git add" and/or "git commit -a")
shubham@shubham:~/github_practice$
```

```
shubham@shubham:~/github_practice$ git add file
shubham@shubham:~/github_practice$ git commit -m"adding un merged file"
[master d680cba] adding un merged file
shubham@shubham:~/github_practice$ git push origin master
Counting objects: 12, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (9/9), done.
Writing objects: 100% (12/12), 1.06 KiB | 544.00 KiB/s, done.
Total 12 (delta 5), reused 0 (delta 0)
remote: Resolving deltas: 100% (5/5), completed with 1 local object.
To github.com:shubhamjain2211/test.git
  531e80a..d680cba master -> master
shubham@shubham:~/github_practice$
```

17. Stash the changes and pop them

- Introducing some changes in the file and then using git stash to clean the working tree and stash the changes



```
shubham@shubham:~/github_practice$ ls
file file1 file2 file4 file6 new ques0 ques11 README.md
shubham@shubham:~/github_practice$ cat file1
WELCOME!!!
this is my first file for github.
shubham@shubham:~/github_practice$ nano file1
shubham@shubham:~/github_practice$ cat file1
WELCOME!!!
this is my first file for github.
some changes
shubham@shubham:~/github_practice$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

    modified:   file1

no changes added to commit (use "git add" and/or "git commit -a")
shubham@shubham:~/github_practice$ git add .
shubham@shubham:~/github_practice$ git status
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    modified:   file1

shubham@shubham:~/github_practice$ git stash
Saved working directory and index state WIP on master: d680cba adding un merged file
shubham@shubham:~/github_practice$ cat file1
WELCOME!!!
this is my first file for github.
shubham@shubham:~/github_practice$
```

- git stash pop : to pop the stashed content

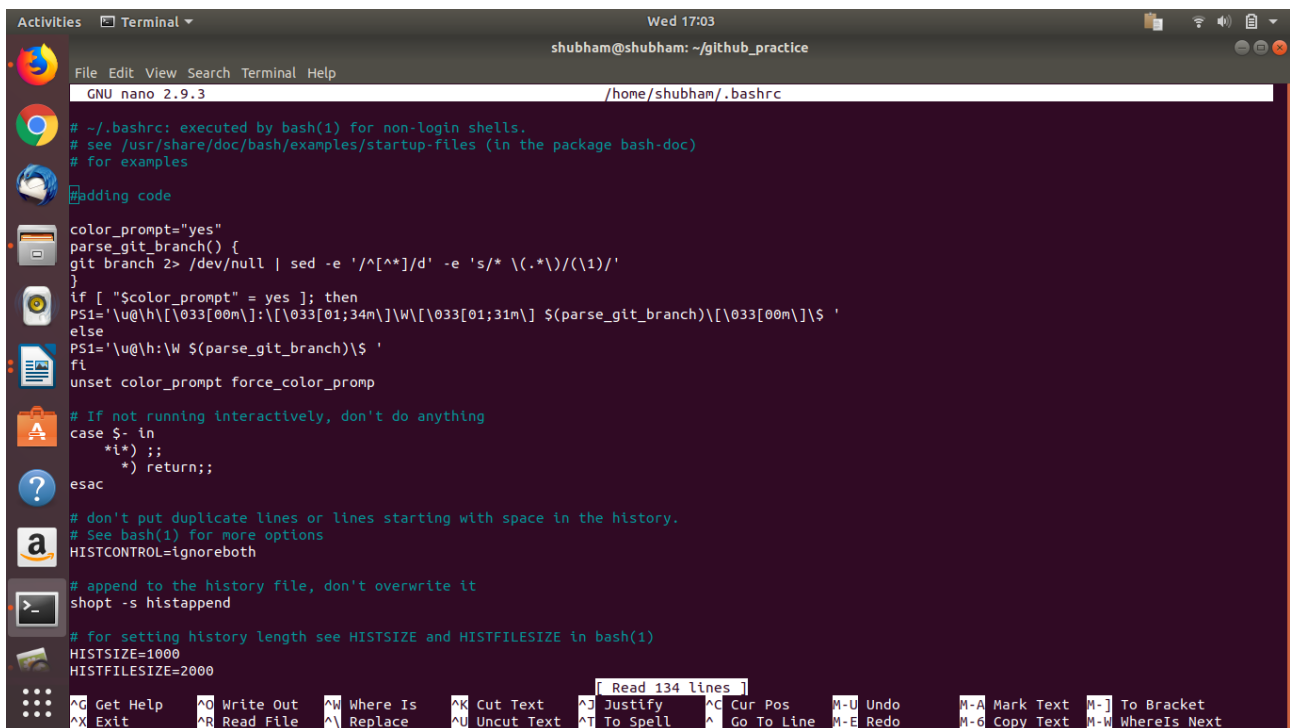


```
shubham@shubham:~/github_practice$ git stash
Saved working directory and index state WIP on master: d680cba adding un merged file
shubham@shubham:~/github_practice$ cat file1
WELCOME!!!
this is my first file for github.
shubham@shubham:~/github_practice$ git stash pop
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

    modified:   file1

no changes added to commit (use "git add" and/or "git commit -a")
Dropped refs/stash@{0} (78363f52d3b9398a0e7fe65b750ec9d42be421b8)
shubham@shubham:~/github_practice$ cat file1
WELCOME!!!
this is my first file for github.
some changes
shubham@shubham:~/github_practice$
```

18. Add the following code to your .bashrc file : color_prompt="yes"
- ```
parse_git_branch() {
git branch 2> /dev/null | sed -e '/^[^*]/d' -e 's/* \(.*)/(\\1)/'
}
if ["$color_prompt" = yes]; then
PS1='\u@\h\[\033[00m\]:\[\033[01;34m\]\W\[\033[01;31m\] $(parse_git_branch)\[\033[00m\]\$ '
else
PS1='\u@\h:\W $(parse_git_branch)\$ '
fi
unset color_prompt force_color_prompt
```



```
GNU nano 2.9.3 /home/shubham/.bashrc

~/.bashrc: executed by bash(1) for non-login shells.
see /usr/share/doc/bash/examples/startup-files (in the package bash-doc)
for examples

#adding code

color_prompt="yes"
parse_git_branch() {
git branch 2> /dev/null | sed -e '/^[^*]/d' -e 's/* \(.*)/(\\1)/'
}
if ["$color_prompt" = yes]; then
PS1='\u@\h\[\033[00m\]:\[\033[01;34m\]\W\[\033[01;31m\] $(parse_git_branch)\[\033[00m\]\$ '
else
PS1='\u@\h:\W $(parse_git_branch)\$ '
fi
unset color_prompt force_color_prompt

If not running interactively, don't do anything
case $- in
 (i) ;;
 *) return;;
esac

don't put duplicate lines or lines starting with space in the history.
See bash(1) for more options
HISTCONTROL=ignoreboth

append to the history file, don't overwrite it
shopt -s histappend

for setting history length see HISTSIZE and HISTFILESIZE in bash(1)
HISTSIZE=1000
HISTFILESIZE=2000

Read 134 lines
Get Help Write Out Where Is Cut Text Justify Cur Pos Undo Mark Text To Bracket
Exit Read File Replace Uncut Text To Spell Go To Line Redo Copy Text WhereIs Next
```