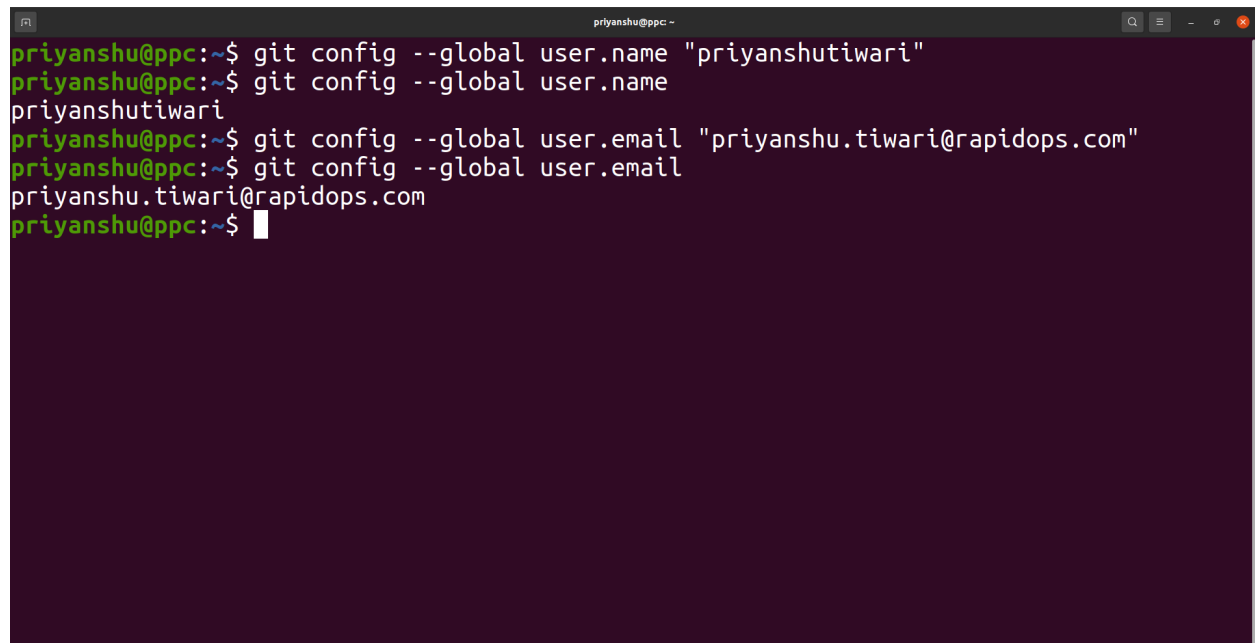


### 1. Configure your user name and email.

**git config --global user.name "enter your username"**

**git config --global user.email "enter your email"**

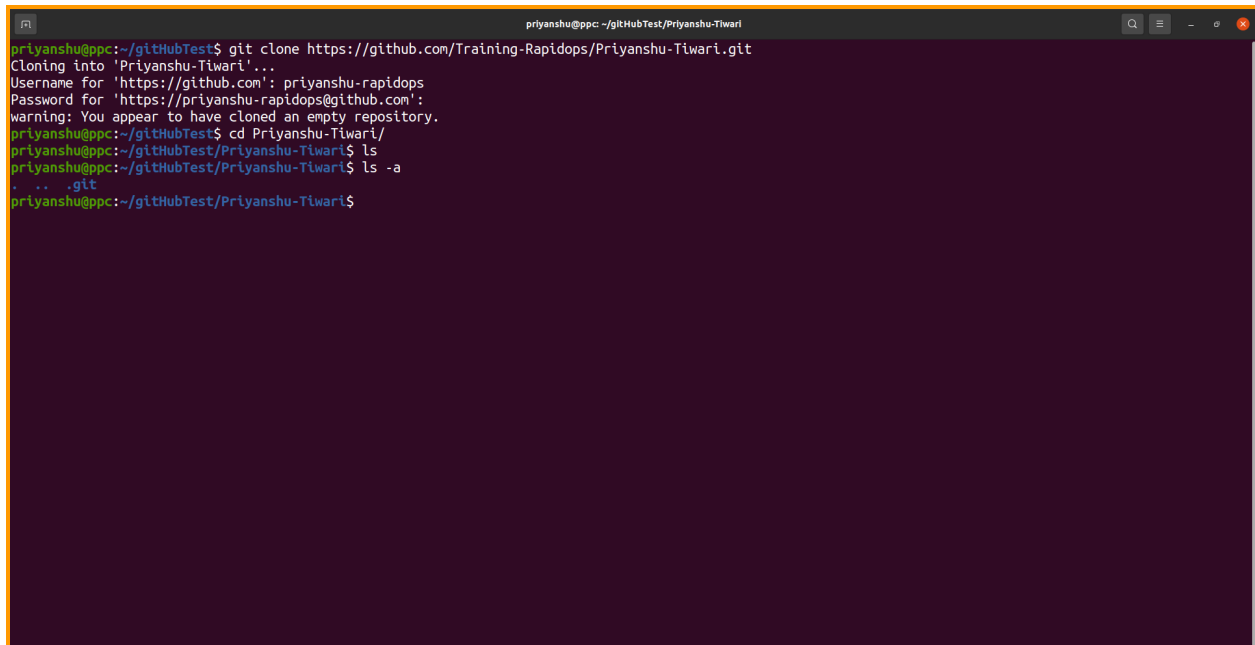
A terminal window with a dark purple background. The title bar reads 'priyanshu@ppc: ~'. The terminal shows the following commands and output:

```
priyanshu@ppc:~$ git config --global user.name "priyanshutiwari"
priyanshu@ppc:~$ git config --global user.name
priyanshutiwari
priyanshu@ppc:~$ git config --global user.email "priyanshu.tiwari@rapidops.com"
priyanshu@ppc:~$ git config --global user.email
priyanshu.tiwari@rapidops.com
priyanshu@ppc:~$
```

### 2. Clone repo of your name from GitHub to the local system.

**git clone** cmd is used for the copy of the given repo in the github its help the copy of the local system

**git clone "enter the url given repo"**

A terminal window with a dark purple background. The title bar reads 'priyanshu@ppc: ~/githubTest/Priyanshu-Tiwari'. The terminal shows the following commands and output:

```
priyanshu@ppc:~/githubTest$ git clone https://github.com/Training-Rapidops/Priyanshu-Tiwari.git
Cloning into 'Priyanshu-Tiwari'...
Username for 'https://github.com': priyanshu-rapidops
Password for 'https://priyanshu-rapidops@github.com':
warning: You appear to have cloned an empty repository.
priyanshu@ppc:~/githubTest$ cd Priyanshu-Tiwari/
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari$ ls
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari$ ls -la
.  ..  .git
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari$
```

### 3. Create a file inside the repo, and make your first commit "My First Commit".

**git add .** is used for the select of all given file in your repo  
**git status** is used for check the file are tracked and untracked and show the file are add,modified , all change show on the git status cmd  
**git push** is used for help to host for project in github local system → github  
**git commit** is used for the save our change and save the file **git commit -m** is used for message what should we change in given file

```
priyanshu@ppc: ~/gitHubTest/Priyanshu-Tiwari
priyanshu@ppc:~/gitHubTest/Priyanshu-Tiwari$ git add .
priyanshu@ppc:~/gitHubTest/Priyanshu-Tiwari$ ls -l
total 0
-rw-rw-r-- 1 priyanshu priyanshu 0 Jan 17 12:45 index.html
priyanshu@ppc:~/gitHubTest/Priyanshu-Tiwari$ git status
On branch master

No commits yet

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

priyanshu@ppc:~/gitHubTest/Priyanshu-Tiwari$ git commit -m "add-index.html"
[master (root-commit) 982d488] add-index.html
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 index.html
priyanshu@ppc:~/gitHubTest/Priyanshu-Tiwari$ git status
On branch master
Your branch is based on 'origin/master', but the upstream is gone.
  (use "git branch --unset-upstream" to fixup)

nothing to commit, working tree clean
priyanshu@ppc:~/gitHubTest/Priyanshu-Tiwari$ git push origin
Username for 'https://github.com': priyanshu-rapidops
Password for 'https://priyanshu-rapidops@github.com':
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 223 bytes | 223.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Training-Rapidops/Priyanshu-Tiwari.git
 * [new branch]      master -> master
priyanshu@ppc:~/gitHubTest/Priyanshu-Tiwari$ git push master
fatal: 'master' does not appear to be a git repository
fatal: Could not read from remote repository.
```

```
priyanshu@ppc: ~/gitHubTest/Priyanshu-Tiwari
Please make sure you have the correct access rights
and the repository exists.
priyanshu@ppc:~/gitHubTest/Priyanshu-Tiwari$ git branch -M master
priyanshu@ppc:~/gitHubTest/Priyanshu-Tiwari$ git remote add origin https://github.com/Training-Rapidops/Priyanshu-Tiwari.git
error: remote origin already exists.
priyanshu@ppc:~/gitHubTest/Priyanshu-Tiwari$ git push -u origin master
Branch 'master' set up to track remote branch 'master' from 'origin'.
Everything up-to-date
priyanshu@ppc:~/gitHubTest/Priyanshu-Tiwari$ git status

Command 'gits' not found, did you mean:

  command 'gitls' from snap gitls (1.0.3)
  command 'gitk' from deb gitk (1:2.25.1-1ubuntu3.2)
  command 'gitg' from deb gitg (3.32.1-1)
  command 'git' from deb git (1:2.25.1-1ubuntu3.2)
  command 'gist' from deb yorick (2.2.04+dfsg1-10)
  command 'gits' from deb gnutit (4.9.5-3build3)
  command 'gitso' from deb gitso (0.6.2+svn158+dfsg-3)

See 'snap info <snapname>' for additional versions.

priyanshu@ppc:~/gitHubTest/Priyanshu-Tiwari$ git commit --amend -m "My First Commit"
[master 3a8ee96] My First Commit
 Date: Mon Jan 17 13:07:31 2022 +0530
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 index.html
priyanshu@ppc:~/gitHubTest/Priyanshu-Tiwari$ git push --force
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 228 bytes | 228.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Training-Rapidops/Priyanshu-Tiwari.git
 + 982d488...3a8ee96 master -> master (forced update)
priyanshu@ppc:~/gitHubTest/Priyanshu-Tiwari$
```

**4.Create and switch to the branch 'test/development' (create from the master branch and it should be from origin).**

**git checkout** is used for the check the which branch your are at now

```
priyanshu@ppc: ~/githubTest/Priyanshu-Tiwari
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari$ git checkout
Your branch is up to date with 'origin/master'.
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari$ git checkout -b test/development
Switched to a new branch 'test/development'
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari$ git checkout
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari$ git checkout -a
error: unknown switch 'a'
usage: git checkout [<options>] <branch>
or: git checkout [<options>] [<branch>] -- <file>...

    -b <branch>          create and checkout a new branch
    -B <branch>          create/reset and checkout a branch
    -l                   create reflog for new branch
    --guess              second guess 'git checkout <no-such-branch>' (default)
    --overlay            use overlay mode (default)
    -q, --quiet          suppress progress reporting
    --recurse-submodules[=<checkout>]
                        control recursive updating of submodules
    --progress           force progress reporting
    -m, --merge           perform a 3-way merge with the new branch
    --conflict <style>   conflict style (merge or diff3)
    -d, --detach          detach HEAD at named commit
    -t, --track           set upstream info for new branch
    -f, --force           force checkout (throw away local modifications)
    --orphan <new-branch>
                        new unparented branch
    --overwrite-ignore   update ignored files (default)
    --ignore-other-worktrees
                        do not check if another worktree is holding the given ref
    -2, --ours            checkout our version for unmerged files
    -3, --theirs          checkout their version for unmerged files
    -p, --patch           select hunks interactively
    --ignore-skip-worktree-bits
                        do not limit pathspecs to sparse entries only
    --pathspec-from-file <file>
                        read pathspec from file
```

```
priyanshu@ppc: ~/githubTest/Priyanshu-Tiwari
usage: git checkout [<options>] <branch>
or: git checkout [<options>] [<branch>] -- <file>...

    -b <branch>          create and checkout a new branch
    -B <branch>          create/reset and checkout a branch
    -l                   create reflog for new branch
    --guess              second guess 'git checkout <no-such-branch>' (default)
    --overlay            use overlay mode (default)
    -q, --quiet          suppress progress reporting
    --recurse-submodules[=<checkout>]
                        control recursive updating of submodules
    --progress           force progress reporting
    -m, --merge           perform a 3-way merge with the new branch
    --conflict <style>   conflict style (merge or diff3)
    -d, --detach          detach HEAD at named commit
    -t, --track           set upstream info for new branch
    -f, --force           force checkout (throw away local modifications)
    --orphan <new-branch>
                        new unparented branch
    --overwrite-ignore   update ignored files (default)
    --ignore-other-worktrees
                        do not check if another worktree is holding the given ref
    -2, --ours            checkout our version for unmerged files
    -3, --theirs          checkout their version for unmerged files
    -p, --patch           select hunks interactively
    --ignore-skip-worktree-bits
                        do not limit pathspecs to sparse entries only
    --pathspec-from-file <file>
                        read pathspec from file
    --pathspec-file-nul  with --pathspec-from-file, pathspec elements are separated with NUL character

priyanshu@ppc:~/githubTest/Priyanshu-Tiwari$ git branch -a
* master
* test/development
remotes/origin/master
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari$
```

**5.Add a file in this branch and commit your changes with the message "Practice started in test branch".**

```
pryanshu@ppc: ~/githubTest/Priyanshu-Tiwari
pryanshu@ppc:~/githubTest/Priyanshu-Tiwari$ git checkout
Your branch is up to date with 'origin/test/development'.
pryanshu@ppc:~/githubTest/Priyanshu-Tiwari$ touch tindex.html
pryanshu@ppc:~/githubTest/Priyanshu-Tiwari$ git push
Everything up-to-date
pryanshu@ppc:~/githubTest/Priyanshu-Tiwari$ git status
On branch test/development
Your branch is up to date with 'origin/test/development'.

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

nothing added to commit but untracked files present (use "git add" to track)
pryanshu@ppc:~/githubTest/Priyanshu-Tiwari$ git add .
pryanshu@ppc:~/githubTest/Priyanshu-Tiwari$ git status
On branch test/development
Your branch is up to date with 'origin/test/development'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   tindex.html

pryanshu@ppc:~/githubTest/Priyanshu-Tiwari$ ls -l
total 0
-rw-rw-r-- 1 priyanshu priyanshu 0 Jan 17 12:45 index.html
-rw-rw-r-- 1 priyanshu priyanshu 0 Jan 17 13:36 tindex.html
pryanshu@ppc:~/githubTest/Priyanshu-Tiwari$ git push
Everything up-to-date
pryanshu@ppc:~/githubTest/Priyanshu-Tiwari$ git commit -m "Practice started in test branch"
[test/development 503d1e3] Practice started in test branch
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 tindex.html
pryanshu@ppc:~/githubTest/Priyanshu-Tiwari$ git remote https://github.com/Training-Rapidops/Priyanshu-Tiwari.git
error: Unknown subcommand: https://github.com/Training-Rapidops/Priyanshu-Tiwari.git
usage: git remote [-v | --verbose]
```

## 6. Now push your changes and this branch to the remote.

```
pryanshu@ppc: ~/githubTest/Priyanshu-Tiwari
or: git remote [-v | --verbose] show [-n] <name>
or: git remote prune [-n | --dry-run] <name>
or: git remote [-v | --verbose] update [-p | --prune] [(<group> | <remote>)...]
or: git remote set-branches [--add] <name> <branch>...
or: git remote get-url [--push] [--all] <name>
or: git remote set-url [--push] <name> <newurl> [<oldurl>]
or: git remote set-url --add <name> <newurl>
or: git remote set-url --delete <name> <url>

-v, --verbose            be verbose; must be placed before a subcommand

pryanshu@ppc:~/githubTest/Priyanshu-Tiwari$ git remote add https://github.com/Training-Rapidops/Priyanshu-Tiwari.git
usage: git remote add [<options>] <name> <url>

-f, --fetch              fetch the remote branches
--tags                  import all tags and associated objects when fetching
                        or do not fetch any tag at all (--no-tags)
-t, --track <branch>    branch(es) to track
-m, --master <branch>   master branch
--mirror[=(push|fetch)] set up remote as a mirror to push to or fetch from

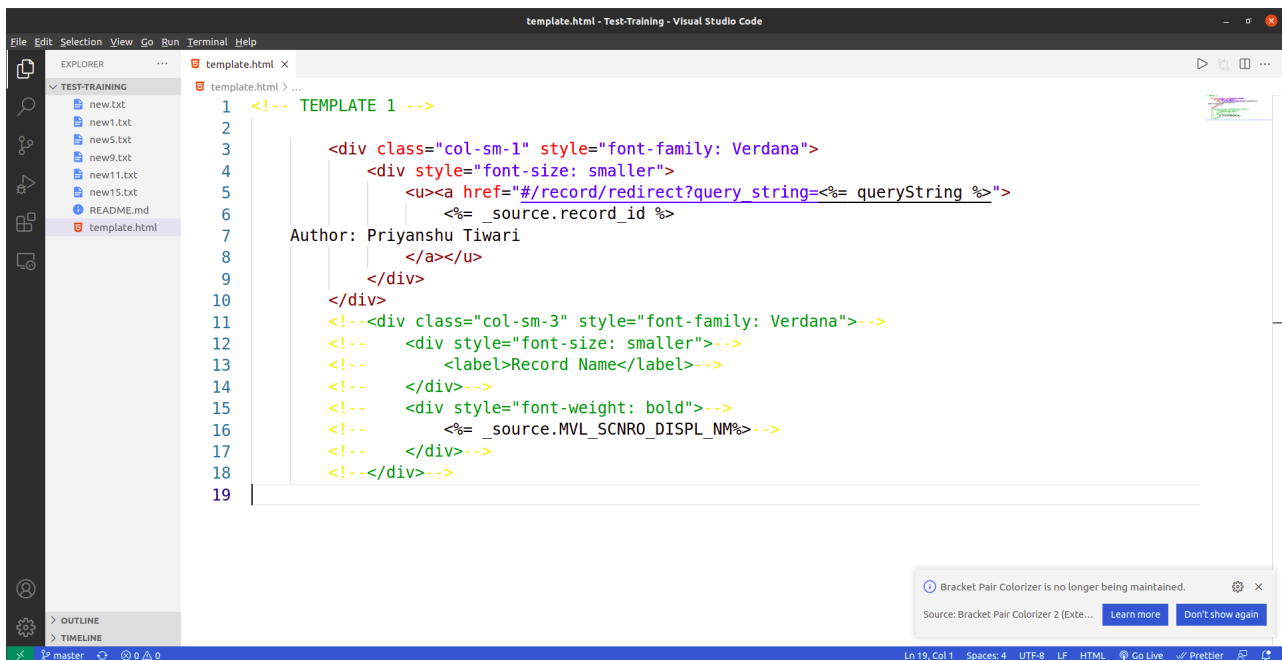
pryanshu@ppc:~/githubTest/Priyanshu-Tiwari$ git remote add origin https://github.com/Training-Rapidops/Priyanshu-Tiwari.git
error: remote origin already exists.
pryanshu@ppc:~/githubTest/Priyanshu-Tiwari$ git push -u origin test/development
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 262 bytes | 262.00 KiB/s, done.
Total 2 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Training-Rapidops/Priyanshu-Tiwari.git
3a8ee96..503d1e3 test/development -> test/development
Branch 'test/development' set up to track remote branch 'test/development' from 'origin'.
pryanshu@ppc:~/githubTest/Priyanshu-Tiwari$
```

7. Go to your GitHub repository and create a pull request to merge this branch in master. Also, add Ravindra & your mentor as reviewers.

8. Fork a public repository **Test-Training**

```
priyanshu@ppc: ~/githubTest/Priyanshu-Tiwari/Test-Training
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari$ git clone https://github.com/priyanshu-rapidops/Test-Training.git
Cloning into 'Test-Training'...
remote: Enumerating objects: 38, done.
remote: Counting objects: 100% (38/38), done.
remote: Compressing objects: 100% (28/28), done.
remote: Total 38 (delta 15), reused 19 (delta 7), pack-reused 0
Receiving objects: 100% (38/38), 6.80 KiB | 632.00 KiB/s, done.
Resolving deltas: 100% (15/15), done.
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari$ ls -l
total 4
-rw-rw-r-- 1 priyanshu priyanshu  0 Jan 17 12:45 index.html
drwxrwxr-x 3 priyanshu priyanshu 4096 Jan 17 14:03 Test-Training
-rw-rw-r-- 1 priyanshu priyanshu  0 Jan 17 13:36 tindex.html
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari$ cd Test-Training/
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari/Test-Training$ ls -l
total 8
-rw-rw-r-- 1 priyanshu priyanshu  0 Jan 17 14:03 new11.txt
-rw-rw-r-- 1 priyanshu priyanshu  0 Jan 17 14:03 new15.txt
-rw-rw-r-- 1 priyanshu priyanshu  0 Jan 17 14:03 new1.txt
-rw-rw-r-- 1 priyanshu priyanshu  0 Jan 17 14:03 new5.txt
-rw-rw-r-- 1 priyanshu priyanshu  0 Jan 17 14:03 new9.txt
-rw-rw-r-- 1 priyanshu priyanshu  0 Jan 17 14:03 new.txt
-rw-rw-r-- 1 priyanshu priyanshu 46 Jan 17 14:03 README.md
-rw-rw-r-- 1 priyanshu priyanshu 689 Jan 17 14:03 template.html
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari/Test-Training$
```

**9. Make a change in the template.html file by adding 'author: {your-name}' and add commit 'Author updated by {your-name}'.**



```
template.html - Test-Training - Visual Studio Code
EXPLORER
TEST-TRAINING
new.txt
new1.txt
new5.txt
new9.txt
new11.txt
new15.txt
README.md
template.html

template.html
1 <!-- TEMPLATE 1 -->
2
3 <div class="col-sm-1" style="font-family: Verdana">
4 <div style="font-size: smaller">
5 <u><a href="#/record/redirect?query_string=<%= queryString %>">
6 <%= _source.record_id %>
7 Author: Priyanshu Tiwari
8 </a></u>
9 </div>
10 </div>
11 <!--<div class="col-sm-3" style="font-family: Verdana">-->
12 <!-- <div style="font-size: smaller">-->
13 <!-- <label>Record Name</label>-->
14 <!-- </div>-->
15 <!-- <div style="font-weight: bold">-->
16 <!-- <%= _source.MVL_SCNR0_DISPL_NM%>-->
17 <!-- </div>-->
18 <!--</div>-->
19
```

```
priyanshu@ppc: ~/githubTest/Priyanshu-Tiwari/Test-Training
-rw-rw-r-- 1 priyanshu priyanshu 0 Jan 17 14:03 new11.txt
-rw-rw-r-- 1 priyanshu priyanshu 0 Jan 17 14:03 new15.txt
-rw-rw-r-- 1 priyanshu priyanshu 0 Jan 17 14:03 new1.txt
-rw-rw-r-- 1 priyanshu priyanshu 0 Jan 17 14:03 new5.txt
-rw-rw-r-- 1 priyanshu priyanshu 0 Jan 17 14:03 new9.txt
-rw-rw-r-- 1 priyanshu priyanshu 0 Jan 17 14:03 new.txt
-rw-rw-r-- 1 priyanshu priyanshu 46 Jan 17 14:03 README.md
-rw-rw-r-- 1 priyanshu priyanshu 689 Jan 17 14:03 template.html
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari/Test-Training$ code .
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari/Test-Training$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   template.html

no changes added to commit (use "git add" and/or "git commit -a")
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari/Test-Training$ git commit -am "Author updated by Priyanshu Tiwari"
[master 00a05cb] Author updated by Priyanshu Tiwari
 1 file changed, 1 insertion(+), 1 deletion(-)
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari/Test-Training$ git push -u origin master
Username for 'https://github.com': priyanshu-rapidops
Password for 'https://priyanshu-rapidops@github.com':
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 326 bytes | 326.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/priyanshu-rapidops/Test-Training.git
   65f8803..00a05cb master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari/Test-Training$
```

**10. Give PR to merge your forked repo in the Test-Training repo.**

**11. Come back to your repo, where you create a branch 'practice' and add 3 commits then rebase it with the master.**

**cmd= git brach**

```
priyanshu@ppc: ~/githubTest/Priyanshu-Tiwari/Test-Training
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari/Test-Training$ git branch practice
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari/Test-Training$ git checkout
Your branch is up to date with 'origin/master'.
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari/Test-Training$ git checkout practice
Switched to branch 'practice'
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari/Test-Training$ touch file1.txt
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari/Test-Training$ touch file2.txt file3.txt
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari/Test-Training$ ls -l
total 8
-rw-rw-r-- 1 priyanshu priyanshu 0 Jan 17 17:07 file1.txt
-rw-rw-r-- 1 priyanshu priyanshu 0 Jan 17 17:07 file2.txt
-rw-rw-r-- 1 priyanshu priyanshu 0 Jan 17 17:07 file3.txt
-rw-rw-r-- 1 priyanshu priyanshu 0 Jan 17 14:03 new11.txt
-rw-rw-r-- 1 priyanshu priyanshu 0 Jan 17 14:03 new15.txt
-rw-rw-r-- 1 priyanshu priyanshu 0 Jan 17 14:03 new1.txt
-rw-rw-r-- 1 priyanshu priyanshu 0 Jan 17 14:03 new5.txt
-rw-rw-r-- 1 priyanshu priyanshu 0 Jan 17 14:03 new9.txt
-rw-rw-r-- 1 priyanshu priyanshu 0 Jan 17 14:03 new.txt
-rw-rw-r-- 1 priyanshu priyanshu 46 Jan 17 14:03 README.md
-rw-rw-r-- 1 priyanshu priyanshu 689 Jan 17 14:27 template.html
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari/Test-Training$ git add file1.txt
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari/Test-Training$ git status
On branch practice
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   file1.txt

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        file2.txt
        file3.txt

priyanshu@ppc:~/githubTest/Priyanshu-Tiwari/Test-Training$ git commit -m "add file1"
[practice 061b165] add file1
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 file1.txt
```

```
priyanshu@ppc: ~/githubTest/Priyanshu-Tiwari/Test-Training
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari/Test-Training$ git commit -m "add file1"
[practice 061b165] add file1
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 file1.txt
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari/Test-Training$ git status
On branch practice
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        file2.txt
        file3.txt

nothing added to commit but untracked files present (use "git add" to track)
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari/Test-Training$ git add file2
fatal: pathspec 'file2' did not match any files
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari/Test-Training$ git commit -m "add file2"
On branch practice
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        file2.txt
        file3.txt

nothing added to commit but untracked files present (use "git add" to track)
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari/Test-Training$ git add file2.txt
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari/Test-Training$ git commit -m "add file2"
[practice e5a6d60] add file2
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 file2.txt
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari/Test-Training$ git add file3.txt
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari/Test-Training$ git commit -m "add file3"
[practice 71bf289] add file3
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 file3.txt
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari/Test-Training$ git status
On branch practice
nothing to commit, working tree clean
priyanshu@ppc:~/githubTest/Priyanshu-Tiwari/Test-Training$ git log
```

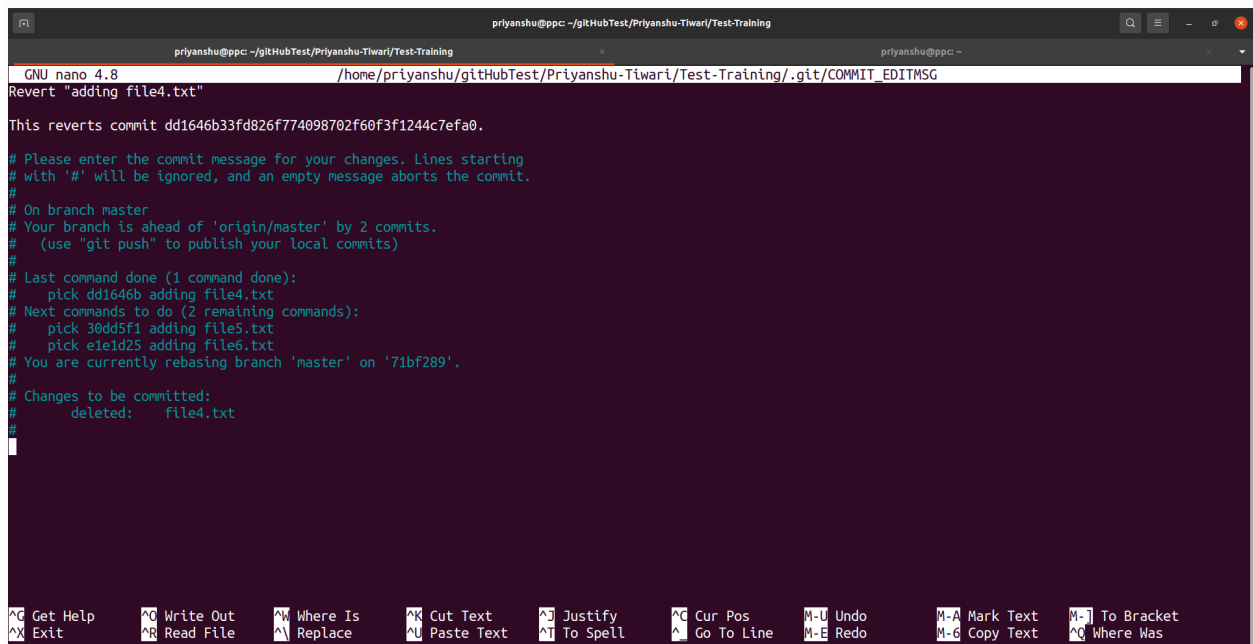
**12.Push your changes to remote, then Add 3 commits again and squash them into the first commit by keeping the message "Rebase squash done".**

**Cmd = git push origin master**

**Cmd = git rebase -i HEAD~3**

**13.Now revert these changes but note that the changes must be retained in the commit history. (use default revert commit message)**

**Cmd = git revert "commit id"**



```
GNU nano 4.8 /home/priyanshu/gitHubTest/Priyanshu-Tiwari/Test-Training/.git/COMMIT_EDITMSG
Revert "adding file4.txt"

This reverts commit dd1646b33fd826f774098702f60f3f1244c7efa0.

# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# On branch master
# Your branch is ahead of 'origin/master' by 2 commits.
#   (use "git push" to publish your local commits)
#
# Last command done (1 command done):
#   pick dd1646b adding file4.txt
# Next commands to do (2 remaining commands):
#   pick 38dd5f1 adding file5.txt
#   pick e1e1d25 adding file6.txt
# You are currently rebasing branch 'master' on '71bf289'.
#
# Changes to be committed:
#   deleted:   file4.txt
#
```

**14.Reset your last commit without losing the changes and then commit with the message "Finished revert with staging changes".**

**Cmd -git reset –soft HEAD~1**

**15.Create one commit then perform a hard reset such that you're back to the commit with the message "Rebase squash done".**



```
priyanshu@ppc: ~/gitHubTest/Priyanshu-Tiwari/Test-Training
priyanshu@ppc:~/gitHubTest/Priyanshu-Tiwari/Test-Training$ git reset --hard HEAD~1
HEAD is now at dd1646b adding file4.txt
priyanshu@ppc:~/gitHubTest/Priyanshu-Tiwari/Test-Training$
```

**16. Create a file index.html, and add it to the staging index then stash it.**

```
priyanshu@ppc: ~/gitHubTest/Priyanshu-Tiwari/Test-Training
priyanshu@ppc:~/gitHubTest/Priyanshu-Tiwari/Test-Training$ touch index2.html
priyanshu@ppc:~/gitHubTest/Priyanshu-Tiwari/Test-Training$ ls -l
total 8
-rw-rw-r-- 1 priyanshu priyanshu 0 Jan 17 17:15 file1.txt
-rw-rw-r-- 1 priyanshu priyanshu 0 Jan 17 17:27 file2.txt
-rw-rw-r-- 1 priyanshu priyanshu 0 Jan 17 17:15 file3.txt
-rw-rw-r-- 1 priyanshu priyanshu 0 Jan 17 17:28 file4.txt
-rw-rw-r-- 1 priyanshu priyanshu 0 Jan 17 22:54 index2.html
-rw-rw-r-- 1 priyanshu priyanshu 0 Jan 17 14:03 new11.txt
-rw-rw-r-- 1 priyanshu priyanshu 0 Jan 17 14:03 new15.txt
-rw-rw-r-- 1 priyanshu priyanshu 0 Jan 17 14:03 new1.txt
-rw-rw-r-- 1 priyanshu priyanshu 0 Jan 17 14:03 new5.txt
-rw-rw-r-- 1 priyanshu priyanshu 0 Jan 17 14:03 new9.txt
-rw-rw-r-- 1 priyanshu priyanshu 0 Jan 17 14:03 new.txt
-rw-rw-r-- 1 priyanshu priyanshu 46 Jan 17 14:03 README.md
-rw-rw-r-- 1 priyanshu priyanshu 689 Jan 17 14:27 template.html
priyanshu@ppc:~/gitHubTest/Priyanshu-Tiwari/Test-Training$ git stash
No local changes to save
priyanshu@ppc:~/gitHubTest/Priyanshu-Tiwari/Test-Training$ git add index2.html
priyanshu@ppc:~/gitHubTest/Priyanshu-Tiwari/Test-Training$ git stash
Saved working directory and index state WIP on master: dd1646b adding file4.txt
priyanshu@ppc:~/gitHubTest/Priyanshu-Tiwari/Test-Training$ git stash list
stash@0: WIP on master: dd1646b adding file4.txt
stash@1: WIP on master: 8184396 rebase squash done
priyanshu@ppc:~/gitHubTest/Priyanshu-Tiwari/Test-Training$
```

**17. Check the list of stash, what changes are there in the stash, then bring your changes from stash.**

```
priyanshu@ppc: ~/gitHubTest/Priyanshu-Tiwari/Test-Training
priyanshu@ppc:~/gitHubTest/Priyanshu-Tiwari/Test-Training$ git stash list
stash@{0}: WIP on master: dd1c46b adding file4.txt
stash@{1}: WIP on master: 8184396 rebase squash done
priyanshu@ppc:~/gitHubTest/Priyanshu-Tiwari/Test-Training$ git stash pop 0
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
(use "git push" to publish your local commits)

Last command done (1 command done):
  pick dd1c46b adding file4.txt
Next commands to do (2 remaining commands):
  pick 30dd5f1 adding file5.txt
  pick e1e1d25 adding file6.txt
(use "git rebase --edit-todo" to view and edit)
You are currently editing a commit while rebasing branch 'master' on '71bf289'.
(use "git commit --amend" to amend the current commit)
(use "git rebase --continue" once you are satisfied with your changes)

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   index2.html

Dropped refs/stash@{0} (ea490ac69dd6320d9ba0ed1e87ffc76a7960fa2b)
priyanshu@ppc:~/gitHubTest/Priyanshu-Tiwari/Test-Training$
```

**18. Commit with a message "Revert, Reset & Stash done".**

```
priyanshu@ppc: ~/gitHubTest/Priyanshu-Tiwari/Test-Training
priyanshu@ppc:~/gitHubTest/Priyanshu-Tiwari/Test-Training$ git commit -m "Revert, Reset & Stash done"
[master 3126ad0] Revert, Reset & Stash done
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 index2.html
priyanshu@ppc:~/gitHubTest/Priyanshu-Tiwari/Test-Training$
```

