

# Git-Task2

To delete a master branch in git- using “git branch -D master” command

```
root@ip-172-31-26-33:~/new
[root@ip-172-31-26-33 new]# git branch
* dev
  master
[root@ip-172-31-26-33 new]# git branch -D master
Deleted branch master (was 844b2eb).
[root@ip-172-31-26-33 new]# git branch
* dev
[root@ip-172-31-26-33 new]#
```

To delete other branches in the git “git branch -d dev”/”git branch -D dev”.

```
root@ip-172-31-26-33:~/new
[root@ip-172-31-26-33 new]# git branch
* master
[root@ip-172-31-26-33 new]# git branch dev
[root@ip-172-31-26-33 new]# git branch
dev
* master
[root@ip-172-31-26-33 new]# git branch -d dev
Deleted branch dev (was 7d85e6a).
[root@ip-172-31-26-33 new]# git branch
* master
[root@ip-172-31-26-33 new]#
```

To delete all other branches except the master branch “git branch | grep -v ‘master’ | xargs branch -D”

```
root@ip-172-31-26-33:~/new
[root@ip-172-31-26-33 new]# git branch
* master
[root@ip-172-31-26-33 new]# git branch dev
[root@ip-172-31-26-33 new]# git branch
dev
* master
[root@ip-172-31-26-33 new]# git branch -d dev
Deleted branch dev (was 7d85e6a).
[root@ip-172-31-26-33 new]# git branch
* master
[root@ip-172-31-26-33 new]# git branch dev
[root@ip-172-31-26-33 new]# git branch dev1
[root@ip-172-31-26-33 new]# git branch dev2
[root@ip-172-31-26-33 new]# git branch
dev
dev1
dev2
* master
[root@ip-172-31-26-33 new]# git branch | grep -v 'master' | xargs git branch -D
Deleted branch dev (was 7d85e6a).
Deleted branch dev1 (was 7d85e6a).
Deleted branch dev2 (was 7d85e6a).
[root@ip-172-31-26-33 new]# git branch
* master
[root@ip-172-31-26-33 new]#
```

**GIT STASH** -allows you to temporarily store changes in your working directory without committing them to a branch.

```
root@ip-172-31-28-6:~/new
[root@ip-172-31-28-6 new]# vi one
[root@ip-172-31-28-6 new]# vi two
[root@ip-172-31-28-6 new]# git add .
[root@ip-172-31-28-6 new]# git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   one
    deleted:    test
    new file:   two

[root@ip-172-31-28-6 new]# git stash
Saved working directory and index state WIP on master: ffdda92 2nd commit
[root@ip-172-31-28-6 new]# git stash list
stash@{0}: WIP on master: ffdda92 2nd commit
[root@ip-172-31-28-6 new]# git status
On branch master
nothing to commit, working tree clean
[root@ip-172-31-28-6 new]#
```

**“git stash” saved the files[one and two] locally in the working directory .**

**“git stash list” to list all the stashes in the working directory.**

```
root@ip-172-31-28-6:~/new
[root@ip-172-31-28-6 new]# vi one
[root@ip-172-31-28-6 new]# vi two
[root@ip-172-31-28-6 new]# git add .
[root@ip-172-31-28-6 new]# git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   one
        deleted:    test
        new file:   two

[root@ip-172-31-28-6 new]# git stash
Saved working directory and index state WIP on master: ffdda92 2nd commit
[root@ip-172-31-28-6 new]# git stash list
stash@{0}: WIP on master: ffdda92 2nd commit
[root@ip-172-31-28-6 new]# git status
On branch master
nothing to commit, working tree clean
[root@ip-172-31-28-6 new]# vi three
[root@ip-172-31-28-6 new]# git add .
[root@ip-172-31-28-6 new]# git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   three

[root@ip-172-31-28-6 new]# git stash
Saved working directory and index state WIP on master: ffdda92 2nd commit
[root@ip-172-31-28-6 new]# git stash list
stash@{0}: WIP on master: ffdda92 2nd commit
stash@{1}: WIP on master: ffdda92 2nd commit
[root@ip-172-31-28-6 new]#
```

**“git stash pop” to apply the most recent stash from the stash list and remove it from the stash list,**

**“git stash pop 1” to specifically apply the most recent stash from the stash list and remove it from the stash list.**

```
root@ip-172-31-28-6:~/new
stash@{0}: WIP on master: ffdda92 2nd commit
[root@ip-172-31-28-6 new]# git stash
Saved working directory and index state WIP on master: ffdda92 2nd commit
[root@ip-172-31-28-6 new]# git stash list
stash@{0}: WIP on master: ffdda92 2nd commit
stash@{1}: WIP on master: ffdda92 2nd commit
[root@ip-172-31-28-6 new]# git stash pop 0
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   one
    new file:   two

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

Dropped refs/stash@{0} (364888674b8941ff8547595409a884ef7de287f9)
[root@ip-172-31-28-6 new]# git stash
Saved working directory and index state WIP on master: ffdda92 2nd commit
[root@ip-172-31-28-6 new]# git stash list
stash@{0}: WIP on master: ffdda92 2nd commit
stash@{1}: WIP on master: ffdda92 2nd commit
[root@ip-172-31-28-6 new]# git stash pop 1
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   three

Dropped refs/stash@{1} (cb7ef5ddc73b8ac0205d8f3d0ae0958826d17972)
[root@ip-172-31-28-6 new]#
```

**“git stash apply”** it applies the most recent stash from the stash list to your working directory without removing it from the stash list.

```
root@ip-172-31-28-6:~/new
[root@ip-172-31-28-6 new]# git stash list
stash@{0}: WIP on master: ffdda92 2nd commit
stash@{1}: WIP on master: ffdda92 2nd commit
[root@ip-172-31-28-6 new]# git stash apply
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   three

[root@ip-172-31-28-6 new]#
```

“git stash apply stash@{ 3}” it applies the most recent stash from the stash list to your working directory without removing it from the stash list.

```
root@ip-172-31-28-6 new# git stash list
stash@{0}: WIP on master: ffdda92 2nd commit
stash@{1}: WIP on master: ffdda92 2nd commit
stash@{2}: WIP on master: ffdda92 2nd commit
stash@{3}: WIP on master: ffdda92 2nd commit
root@ip-172-31-28-6 new# git stash apply stash@{3}
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   one
    new file:   two

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

root@ip-172-31-28-6 new# git stash
Saved working directory and index state WIP on master: ffdda92 2nd commit
root@ip-172-31-28-6 new# git stash list
stash@{0}: WIP on master: ffdda92 2nd commit
stash@{1}: WIP on master: ffdda92 2nd commit
stash@{2}: WIP on master: ffdda92 2nd commit
stash@{3}: WIP on master: ffdda92 2nd commit
stash@{4}: WIP on master: ffdda92 2nd commit
root@ip-172-31-28-6 new#
```

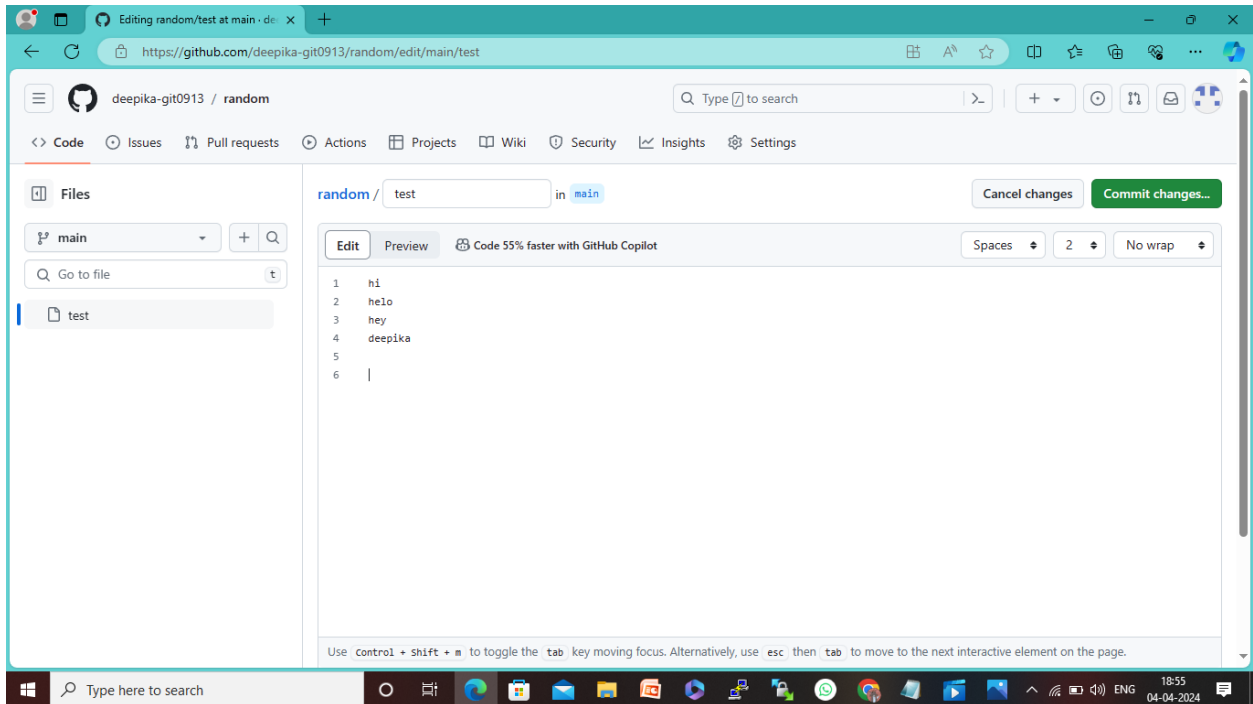
“git stash drop stash@{5}” drops(delete) the specific stash.

“git stash clear” clears(delete) all the stashes.

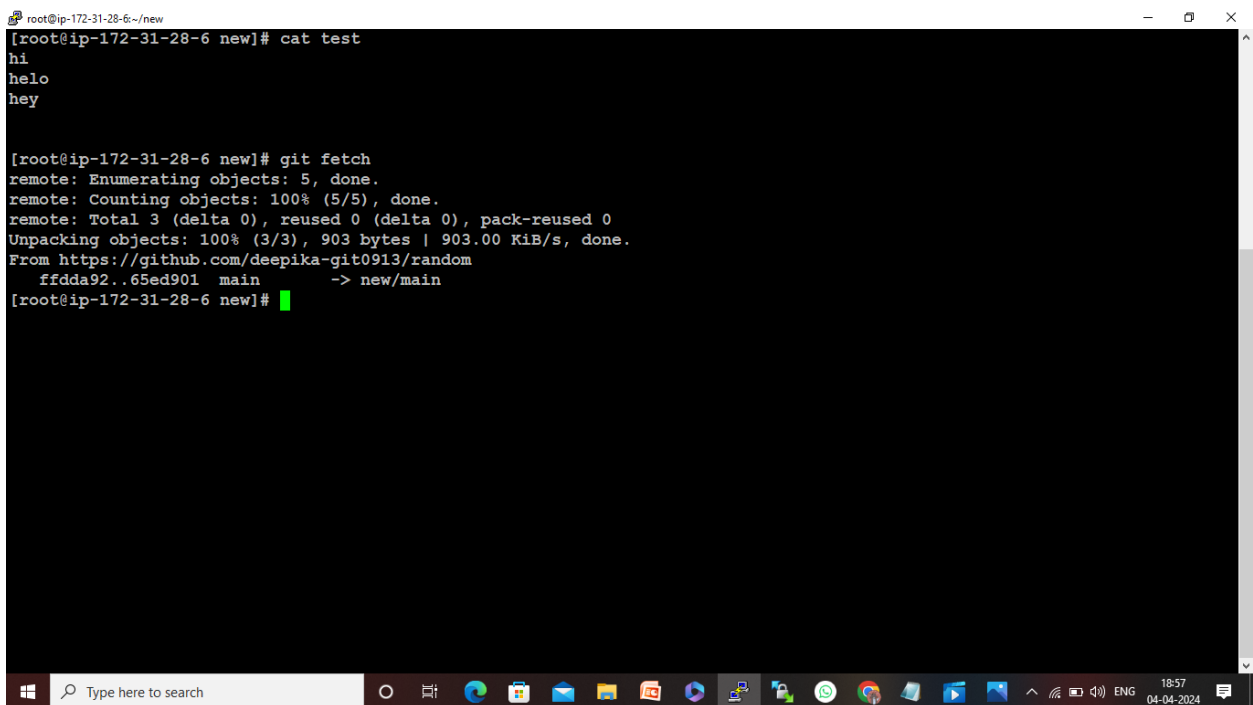
```
root@ip-172-31-28-6 new# git stash list
stash@{0}: WIP on master: ffdda92 2nd commit
stash@{1}: WIP on master: ffdda92 2nd commit
stash@{2}: WIP on master: ffdda92 2nd commit
stash@{3}: WIP on master: ffdda92 2nd commit
stash@{4}: WIP on master: ffdda92 2nd commit
stash@{5}: WIP on master: ffdda92 2nd commit
root@ip-172-31-28-6 new# git stash drop stash@{5}
Dropped stash@{5} (680ceeed6be044920f606e4aeda8d2abb6719750)
root@ip-172-31-28-6 new# git stash list
stash@{0}: WIP on master: ffdda92 2nd commit
stash@{1}: WIP on master: ffdda92 2nd commit
stash@{2}: WIP on master: ffdda92 2nd commit
stash@{3}: WIP on master: ffdda92 2nd commit
stash@{4}: WIP on master: ffdda92 2nd commit
root@ip-172-31-28-6 new# git stash clear
root@ip-172-31-28-6 new# git stash list
root@ip-172-31-28-6 new#
```

**GIT-FETCH-** used to retrieve the latest changes from a remote repository without merging them into your current branch. (like notification)

**Making some changes in the remote repository**



**“git fetch” shows the changes to the local repository.**



**“git clone”- is used to initially create a local copy of a remote repository.**

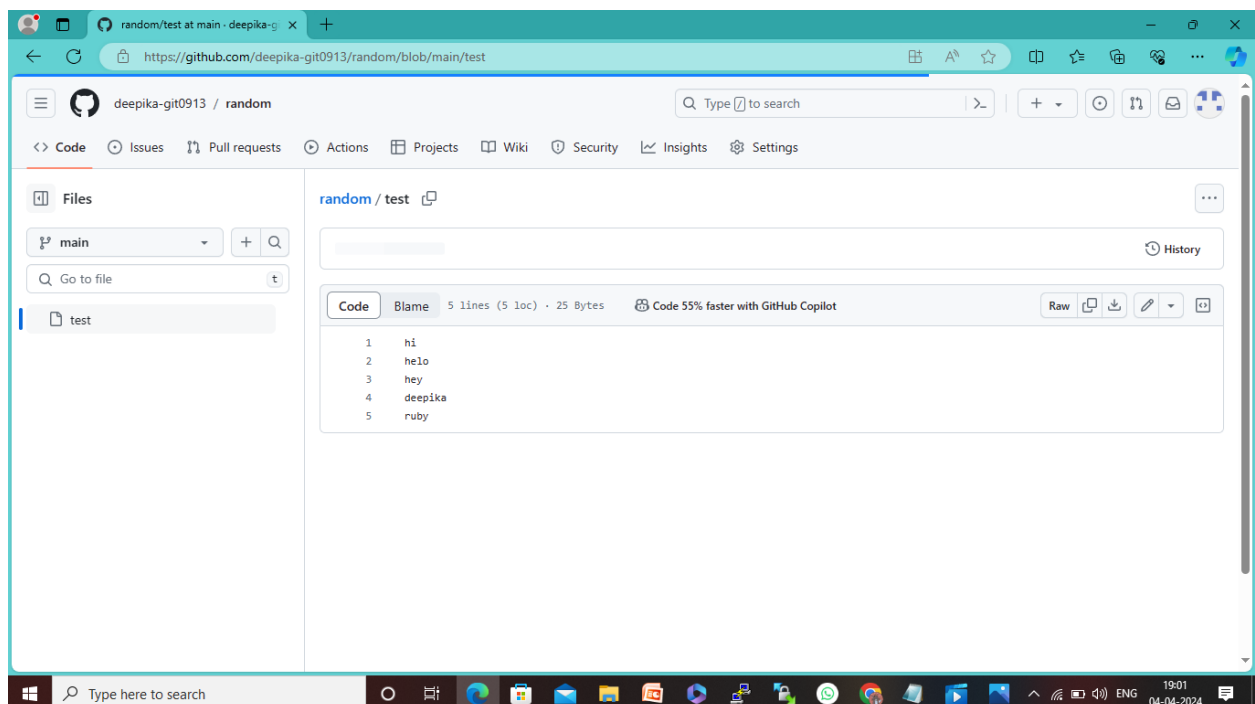
```
root@ip-172-31-28-6:~/new/random
hi
helo
hey

[root@ip-172-31-28-6 new]# git fetch
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 903 bytes | 903.00 KiB/s, done.
From https://github.com/deepika-git0913/random
ffdda92..65ed901  main      -> new/main
[root@ip-172-31-28-6 new]# git clone https://github.com/deepika-git0913/random.git
Cloning into 'random'...
remote: Enumerating objects: 9, done.
remote: Counting objects: 100% (9/9), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 9 (delta 1), reused 6 (delta 1), pack-reused 0
Receiving objects: 100% (9/9), done.
Resolving deltas: 100% (1/1), done.
[root@ip-172-31-28-6 new]# ls
random test
[root@ip-172-31-28-6 new]# cd random
[root@ip-172-31-28-6 random]# ls
test
[root@ip-172-31-28-6 random]# cat test
hi
helo
hey
deepika

[root@ip-172-31-28-6 random]#
```

**GIT –PULL -is used to fetch the latest changes from a remote repository and merge them into your current branch.**

**Making another changes in the remote repository.**



**“git fetch” does not update any changes in the local repository.**

```
root@ip-172-31-28-6:~/new/random
deepika

[root@ip-172-31-28-6 random]# cd ..
[root@ip-172-31-28-6 new]# cat test
hi
helo
hey

[root@ip-172-31-28-6 new]# ls
random test
[root@ip-172-31-28-6 new]# cd random
[root@ip-172-31-28-6 random]# cat test
hi
helo
hey
deepika

[root@ip-172-31-28-6 random]# git fetch
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 909 bytes | 909.00 KiB/s, done.
From https://github.com/deepika-git0913/random
 65ed901..fd38632  main      -> origin/main
[root@ip-172-31-28-6 random]# cat test
hi
helo
hey
deepika

[root@ip-172-31-28-6 random]#
```

**“git pull”-is used to fetch the latest changes from a remote repository and merge them into your current branch.**

```
root@ip-172-31-28-6:~/new/random
hi
helo
hey
deepika

[root@ip-172-31-28-6 random]# git fetch
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 909 bytes | 909.00 KiB/s, done.
From https://github.com/deepika-git0913/random
 65ed901..fd38632  main      -> origin/main
[root@ip-172-31-28-6 random]# cat test
hi
helo
hey
deepika

[root@ip-172-31-28-6 random]# git pull
Updating 65ed901..fd38632
Fast-forward
 test | 2 + -
 1 file changed, 1 insertion(+), 1 deletion(-)
[root@ip-172-31-28-6 random]# ls
test
[root@ip-172-31-28-6 random]# cat test
hi
helo
hey
deepika
ruby
[root@ip-172-31-28-6 random]#
```