

# GIT COMMANDS

- to initialize the give in a folder: **git init**

init command will create a .git hidden file in the initialized folder the purpose of this file is to convert the normal folder to version controlled folder. Aside from this folder git commands will not work

- to set a user name globally: **git config --global user.name "sankar"**
- to set an email globally: **git config --global user.email "sankar@gmail.com"**
- to set a user name particular repo: **git config user.name "sankar"**
- to set an email particular repo: **git config user.email "sankar@gmail.com"**

if we use --global in the above it will set the user name and user email globally to all repos, if individual repos don't have there own user name.

## **configured individually**

repo1 : user.name and user.email - sankar1, sankar1@gmail.com

repo2:user.name and user.email -sankar2 sankar2@gmail.com

## **configured individually**

repo3: user.name and user.email - sankar3, sankar3@gmail.com

## **Not configured**

repo4: user.name and user.email - no user name email

if we check for user name and email for this we will global user name and email

user.name and user.email - sankar3, sankar3@gmail.com

- to see the user name of git: **git config user. name**
- to see the email of git user: **git config user.email**
- to see the status of files: **git status**

- to add particular file working dir to the staging area: **git add <file name>**

```
[root@ip-172-31-42-207 myrepo]# git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        sankar1
        sankar2
        sankar3
        sankar4

nothing added to commit but untracked files present (use "git add" to track)
[root@ip-172-31-42-207 myrepo]# git add sankar1
[root@ip-172-31-42-207 myrepo]#
```

- to add all files to the staging area: **git add .** (or) **git add --all** (or) **git add -A**

```
[root@ip-172-31-42-207 myrepo]# git add --all
[root@ip-172-31-42-207 myrepo]# git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   sankar1
        new file:   sankar2
        new file:   sankar3
        new file:   sankar4

[root@ip-172-31-42-207 myrepo]#
```

**note:** git add . the command is used to add the file in a particular directory inside the working directory  
for example my\_repo /sankar1/sankar2/sanka3

i have done changes in sankar1 and sankar2 and sankar3 . Now I want to move only sankar3 changes to the staging area we have to use **git add .** it will only move the changes within the sankar3 dir. **and one more thing is this command will not add deleted files to the staging area.**

if we use the other 2 commands anywhere in the sankar1, sankar2, or sankar3 it will move all changes to the staging area.

- to commit the staged changes: **git commit -m "your message"**

note: when we commit the changes only staged changes will be committed.

```
[root@ip-172-31-42-207 myrepo]# git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   sankar1
        new file:   sankar2
        new file:   sankar3
        new file:   sankar4

[root@ip-172-31-42-207 myrepo]# git commit -m "my first commit"
[master (root-commit) ca8ba81] my first commit
 4 files changed, 24 insertions(+)
 create mode 100644 sankar1
 create mode 100644 sankar2
 create mode 100644 sankar3
 create mode 100644 sankar4
[root@ip-172-31-42-207 myrepo]#
```

- to see the history of commits: **git log**

```
[root@ip-172-31-42-207 myrepo]# git status
On branch master
nothing to commit, working tree clean
[root@ip-172-31-42-207 myrepo]# git log
commit ca8ba8106933eb2fd0fe6eb46acefa9e3036cfb5 (HEAD -> master)
Author: sankar <sankar@gmail.com>
Date:   Sat Sep 23 18:07:19 2023 +0000

    my first commit
[root@ip-172-31-42-207 myrepo]#
```

- to see the last commit: `git log -1`
- to see the last 2 commits: `git log -2`

```
my first commit
[root@ip-172-31-42-207 myrepo]# git log -1
commit 3424243b8e2d27eee65e7fdbbe03c7b5feld6e6e (HEAD -> master)
Author: sankar <sankar@gmail.com>
Date: Sat Sep 23 18:26:38 2023 +0000
```

```
my 2nd commit
[root@ip-172-31-42-207 myrepo]# git log -2
commit 3424243b8e2d27eee65e7fdbbe03c7b5feld6e6e (HEAD -> master)
Author: sankar <sankar@gmail.com>
Date: Sat Sep 23 18:26:38 2023 +0000
```

```
my 2nd commit

commit ca8ba8106933eb2fd0fe6eb46acefa9e3036cfb5
Author: sankar <sankar@gmail.com>
Date: Sat Sep 23 18:07:19 2023 +0000
```

```
my first commit
[root@ip-172-31-42-207 myrepo]#
```

- to view the full commit in one line: **git log --oneline**

```
nothing to commit, working tree clean
[root@ip-172-31-87-26 myrepo]# git log --oneline
404f521 (HEAD -> master) 3rd commit
7e3e826 2st commit
bbe02c7 1st commit
[root@ip-172-31-87-26 myrepo]#
[root@ip-172-31-87-26 myrepo]#
```