

----- GIT Commands -----

Git working process

1. Create a folder
2. Open git bash terminal (Rightclick ---> More options ----> Click on 'Open Gitbash here' ----> Git terminal will open)
3. Initiate the empty git repository (git init)
3. Create a file (touch <FileName>)
4. Check the status of the file (git status)
5. Add the file to the staging area (git add) (git add .) (git add <FileName>)
6. Commit the file to local repository (git commit -m "<Enter Commit Message>")
7. Send the file to the remote repository

Note: Configure the git with username and user-email in the git

git config --global user.name "<UserName>"

git config --global user.email "<UserEmail>"

How to check the configuration details

Method 1:

git config user.name

git config user.email

Method 2:

git config --list

How to add multiple files at the same time to the staging area?

git add .

How to add specific file to the staging area?

`git add <FileName>`

How to add specific set of files (python files) to the staging area?

`git add <FileStartingName>*`

Ex: `git add python*`

How to commit multiple files at the same time to the repository area?

`git commit -m "<EnterCommitMessage>"`

How to commit specific file to the repository area?

`git commit -m "<EnterCommitMessage>" <FileName>`

How to commit specific set of files (python files) to the repository area?

`git commit -m "<EnterCommitMessage>" <FileName>*`

Ex: `git commit -m "added python files" python*`

How to rename a file in git?

`git mv <OldFileName> <NewFileName>`

How to delete a file in git?

`git rm <FileName>`

How to see the commit history?

`git log`

Note: Press 'down arrow' or 'up arrow' to see the older/latest commits.

Note: To come out of the commit history, press 'q'

Commit ID is a 40 character alphanumeric id

How to see the commit history in a simplified way?

```
git log --oneline
```

You will see only the commit id (first 6 characters) and the commit message

How to see the commit history in reverse order

```
git log --oneline reverse
```

```
git log reverse
```

How to see the commit history of a specific file

```
git log --oneline -- <filename>
```

```
git log --oneline -- <commit id>
```

How to see the top commits

```
git show HEAD
```

How to remove .git folder?

```
rm -rf .git
```

Note: it is not recommended to delete the .git folder.

Working with git restore

=> Restore is a polymorphic command

If the file is added to the staging area, if you want to unstage the file;

```
git restore --stage <FileName>
```

If the file is unstaged, and you want to remove the changes in the file;

```
git restore <FileName>
```

How to skip the staging area?

Note: This method is not recommended

How to see the status of a file in shortcut?

`git status -s`

If you see A in green colour, this means the file is added to the staging area

If you see ?? in red colour, this means the file is not being tracked by the git (unstaged area)

By
Kastro Kiran V

LinkedIn: <https://www.linkedin.com/in/kastro-kiran/>

YouTube: https://www.youtube.com/playlist?list=PLs-PsDpuAuTdOcZa-DDgG8KRbtMI_XRrC