

Untracked: Files which are not tracked by the GIT are known as untracked file. EXP: log files, automatic generated files, etc…. Newly created files are stored in this section. Untracked files are shown by the “git status” command.

Staged: the files which need to be tracked are placed in staged area. Staged area is used to commit the files. Commit is like the snapshot of the file. To add the file to staging area use the command “git add <file name>”. Then use the command “git commit” to start the tracking.

Unmodified: the files are complete means no further changes required those files are placed in unmodified. Files are moved from Staged area to unmodified by the commit command.

* Git add command to add the file to the staging area
* Git commit to move the file in un-modified area
* Git status to check the status of the files
* Git commit –A to commit all the files
* Git commit –m “message” short command to commit the file with message. Not to use the vim editor for messaging.
* Git checkout <file name> to recover the file from last commit
* Git checkout –f command is used to restore all the files from the last commit
* Git log command is used to see the history of the committed commands
* Git log –p -1 This command is used to filer the logs. We can check the last three commits by below command

Git log –p -3

🡪 git diff command is used to compare what the data has been changed. Means compare the data of working directory with staged data. If the file is commited then it will not show the result.

* Git diff –stage this command compared the file within the staged area
* Git rm –cached <file name> this command is used to remove the file from the staging area and place in working area.
* Git rm <file name> this command is used to delete the file from disk
* Gitignore is a file that can contain the name of the files which we don’t want to add the file is staging area. Just create the file with command “touch gitignore” and type the name of the file inside this file to ignore the file to place in staging area.
* Git branch <branch name> to create the new branch
* Git branch to view all the branches
* Git checkout <branchname> to enter into particular branch
* Git merge <branch name> to merge the content from one branch to another
* Git remote Add <name> <path> to add the remote repo.
* Git remote –v is used to check the path of the remote repo
* ssh -T [git@github.com](mailto:git@github.com) Command used to authenticate permanently with SSH
* git push origin master to upload the data to master branch
* git push origin review to upload the data to master branch