

# GIT CLASS 2

To untrack a file : `git rm --cached filename`

GIT LOG: This command is used to commits in git

`git log` : used to get all commits

`git log --oneline` : used to get only commit ids & messages

`git log --pretty=oneline` = used to get only full commit ids & messages

`git log -2` : used to get latest 2 commits

`git log --follow --all filename` : used to get all commits for a file

GIT IGNORE:

a ".gitignore" file is used in Git to tell Git which files and directories to ignore when tracking changes in a project.

Let's say you're developing a software project and using Git to manage your code changes. Sometimes, there are files or directories in your project that you don't want Git to keep track of, such as temporary files, compiled code, or sensitive information like passwords. Instead of manually excluding these files every time you commit changes, you can list them in a ".gitignore" file.

GIT CONFIG : Used to add our name and email for git history

`git config user.name "username"` : used to add our name

`git config user.email "email"` : used to add email

GIT AMEND: This command is used to change commit details

`git commit --amend --author "user <email>"` : To change author of the latest commit

`git commit --amend -m "new message"` : To change message of the latest commit

`git commit --amend --no-edit` : this command : To change author of the latest commit

GIT RESET: This command is used to delete latest commits

`git reset --hard HEAD~1` : used to delete latest commit along with changes

`git reset --soft HEAD~1` : used to delete only latest commit

`git update-ref -d HEAD`: used to delete all commits at a time (only commits)