Git Commands

**To checkgit version -**

git version

**To see the origin of git and the url**

git remote -v

**Set git username globally -**

git config --global user.name CuelogicAmolWabale

**Set git Email globally -**

git config --global user.email amol.wabale@cuelogic.com

**To see all configuration value -**

git config --list

**Initialize from existing code or track code / folder / project -**

git init

**To know newly added files or change in files -**

git status

**To add file in staging area or to make file ready to be checked in you need to add the file -**

git add -A //To add all the file

git add fileName //To add specific file

**To add updated file in staging area**

git add -u

**To remove file from staging area -**

git reset

**To commit changes -**

git commit -m "your comment"

**To get latest commit done**

git show HEAD

**To see specific commits**

git show COMMITNUMBER

**To get latest code -**

git pull origin master

**Reset config settings -**

git config --global --unset-all core.editor

**To set code editor / Diff tool**

git config --global core.editor EDITORNAME

**To show correct suggestion when entered wrong git keywords -**

git config --global help.autocorrect 1

**to show colourized output in diff or status -**

git config --global color.ui auto

**To reset particular configuration -**

git config --global --unset-all CONFIGNAME

**To show x log-**

git log HEAD~x..Head

**To show after x log-**

git log HEAD~x

**To show one liner log-**

git log --oneline

**To get total number of commit - (does not work on windows)**

git log --oneline | wc -l

**Graphical view of commit -**

git log --oneline --graph

**To get Short logs (gives you contributors username and message) -**

git shortlog

**To get logs by Summary, Displaying by number of commits, email address -**

git shortlog -sne

**Revert changes to previous state if state prior to add (-u or -A)-**

git checkout FILENAME

**Revert changes to previous state if state is ready for commit-**

git reset --hard

**To go back to x commit state -**

git reset --hard HEAD~1

**To remove all unused files-**

git clean -f

**Check branches available remotely**

git branch -r

**To check the tags ( tags are used to pinpoint stable versions )**

git tag

**To tag a commit -**

git tag v1.1

**to add message / annotation with tag -**

git tag -a v1.1\_hello hi there

**Create New branch -**

git branch BRANCHNAME

**Rename branch -**

git branch -m OLDBRANCHNAME NEWNAME

**To recover deleted files**

- git reflog

- git branch BRANCHNAME COMMITNUMBER

**Delete branch**

git branch -d BRANCHNAME

**Delete remote branch**

git push origin :REMOTEBRANCHNAME

**Merge branch -**

**if A is to be merged with B**

**First checkout Branch B**

**and than run command**

git merge A

**stashing -**

**If you dont want to commit the changes but want to save it for future than use command -**

**It will roll back the changes and will save it as an shelvset -**

git stash

**to see all stash list -**

git stash list

**to apply changes saved in stash use command -**

git stash apply

**to apply changes saved in stash and to remove the top entry use following -**

git stash pop

**Rebasing - It is same as merging, only difference is it will not preserve it histroy, master branch will appear as clean**

**If B is to rebased on A-**

**got to branch B and use following command-**

git rebase A

**Than go to branch A and use following command**

git merge B

**Cherry picking- It is the technique of getting single commit from other branch to your main(Master) branch**

git cherry-pick LOGNUMBER