GIT

git checkout . <all changes in particular branch will be reverted>

git branch -d <branch-name> to delete a branch

Git Clone

git clone <url > (Master Branch)

git clone origin -u <url> (Origin)

Git Add

git add <filepath> for single change

git add <filename> for file

git add . 🡪for Adding all the files at a time

git remote add origin <url>

Git Commit

Both Add and Commit can be single line Command

git commit -am “Initial Commit”

git commit -v 🡪v refers verbose (Gives all details of your project)

git commit -m “Inline Commit”

🡪If we first commit in a project that is “root commit”

🡪By Mistake if we have committed

If we use notepad(npp) else vi

If we press “ i ” button to edit

Enter Esc to exit insert mode

Shift : wq (write quit) to

Git Push

git push -u origin/master filename

git push 🡪 move to that particular branch and type git push

Git Pull

git pull 🡪 move to that particular branch and type git pull for latest code

git pull -u origin/master <filename>

Git Branching

40 characters of commit ID

How can we create a branch

git branch <new branch-name>

git checkout -b <branch-name> 🡪creating and moving to that branch

git checkout <branch-name> 🡪 if we want to move for a particular branch

git merge <branch-name> Move to particular branch and and merge with choice branch both will be merged

git branch --merge 🡪shows What are the branches where merged

git branch --no-merge 🡪 Shows Which branches are not merged

git branch -a 🡪 will shows you all branches

git merge --abort 🡪 To abort the merge which we want to merge.

Git Stashing

Move to that Directory and type below command for Dustbin

git stash 🡪 to dustbin

git restore <filename> (For restoring)

git status 🡪 to check the status like restored or not

git stash apply (Changes on deleted file)

git stash list (Deleted list)

git stash pop (Change and drop from stash)

git stash drop (to remove from the latest stash)

git stash -u (untracked files to include to stash)

git stash apply stash@{0,1,2} for single value entry and applied for particular stash

Git LogHistory

We have to move for Directory and for project Q for quit from that project

git log -p 🡪What are the changes code changes in every Commit

git log -3 (last 3 commits)

git log -5 (last 5 commits)

git log -2 (last 2 commits)

After we give how many commits are there

git log --pretty =oneline

git log --pretty =short

git log --pretty =full

git log --pretty =fuller

|  |
| --- |
| git log --oneline --graph --all |
| All the flow of your project |

ls-al 🡪 for listing all the files in your cloned code (including hidden files)

ls 🡪 for listing all the files in a cloned code

git status to check added files or committed files

git init 🡪 this command is used for initializing the git with local repository

or we can call initialize the empty git repository

git

if you create new file in github

git fetch origin

changes will come but won’t apply to your local branch

git branch -a 🡪to check all the branches present in project

git merge origin /master 🡪 Now the changes will apply

git pull origin master 🡪if you need directly then you can go for pull option

Git Aliases

git config –global alias.(name we want)

***for example***

log -- oneline -- graph – all

git config –global alias.all commits “log -- oneline -- graph – all”

Git Rebase or Git Merge (are almost same)

git show c57a18 (shows all the commit history)

git rebase <branch name>

cat <branch name> 🡪 to check changes

git rebase --continue

Cleaning

git clean -f 🡪 forceful clean (All the untracked files will be deleted)

Folder or files if we need to delete

git clean -f -d 🡪 -d is directory

git clean -f -d -x 🡪 .git ignored files will be deleted or cleaned

git clean -f -n

or

git clean -f --dry-run 🡪 it will show what are the files are going to be delete or clean from u r directory

Git Tagging

git tag v1.0

git log --oneline --graph –all 🡪 to see all the tags present in the project

*Express commit*

Both Add and Commit can be single line Command

git commit -am “Initial Commit”

git tag -a v2.0 🡪 For description

git tag v2.0 -m “Updated v2.0” 🡪 if you don’t need a description

*compare between 2 tags*

git diff v1.0 v2.0 🡪 if you want to findout difference between v1.0 and v2.0

git tag -d v1.0 🡪If you want to delete light weight tag

if you want to check whether the tag is deleted or not

git log --oneline --graph –all

If you want to give tag to already released project or Running project

git tag -a v1.0 09cd897 here v1.0 🡪 tag name , 09cd897🡪 commit ID

tag update for by wrongly given to other tag

git tag -a v1.0 -f bocb7e8

git stash -u 🡪 untracked files to include to stash

*Git Diff and git status*

git status will shows modified content’s file, unmodified/added files

git diff will shows you content difference

git diff --cached

git diff –staged

git status --short or -s

?? 🡪 Untracked files

A 🡪 staging area

M 🡪 Modified files