**Git 🡪 Version Control System**

**GitHub 🡪 Cloud based web platform used to manage your projects**

1. git init 🡪 initializing empty git repo (hidden 🡪 .git) 🡪 to make a blank repository
2. git clone <URL> 🡪 to copy a repository to our desired folder 🡪 to work on an existing repository
3. git status
4. git diff <filename.extension>
5. git commit 🡪 cannot commit if changes are not staged
6. git add <filename.extension> ---🡪 to add our changes to stage area so that we can commit(lock) our change
7. git commit –m “file changed” --🡪m is for message printing
8. git log 🡪 shows all the old/previous commits
9. git log -5 🡪 shows recent 5 commits
10. git show <CommitID> 🡪 shows the changes done under that ID
11. git branch 🡪 shows current branch
12. git checkout –b TaniyaBranch 🡪 switching to a new branch

**(To delete that new branch, first go back to master branch, and then delete that):**

1. git checkout master 🡪 ‘checkout’ is used to switch between branches
2. git branch –d TaniyaBranch 🡪 TaniyaBranch deleted
3. git branch NewBranch 🡪 NewBranch added but we have not switched to it
4. git checkout NewBranch

**(The current branch will be highlighted by green colour.**

**After staging the changes, the file which is changed will be highlighted by green colour instead of red colour)**

**After making changes to new branch and staging and commiting, if we want to merge new branch with master branch:**

1. git checkout master
2. git merge NewBranch

**If we have Committed some changes but those changes were wrong, how to modigy it??**

**Let those were done in our new branch:**

1. git checkout NewBranch
2. git log -3 🡪 shows last 3 commits
3. git reset --hard HEAD^ 🡪 UNDO latest (most recent) commit

**If we want to remove only the commit and retain the changes made:**

1. git reset --soft <commitID>
2. git status
3. git restore --staged index.html 🡪 to remove the changes done in index.html from staged area 🡪 will be highlighted in red colour
4. git restore index.html 🡪 to remove the changes completely
5. git status 🡪 nothing to commit message will be printed
6. git checkout master

**To sync local repo with remote (original) repo:**

1. git pull

**To push local commit to remote branch:**

1. git push 🡪 will ask your username id and then it will be pushed to remote branch
2. git branch
3. git log -3

**Now if we want to modify (alter) the changes done in a file but we don’t want to generate a new commit :**

**First make the changes in your code, save it and then :**

1. git status 🡪 red colour highlighted
2. git diff index.html
3. git add index.html
4. git status 🡪 green coloured highlighted

**Here, if we use git commit –m “ “, then a new commit will be generated, but we don’t want to generate a new commit :**

1. git commit –amend

**To save entire thing :**

Esc + wq + Enter