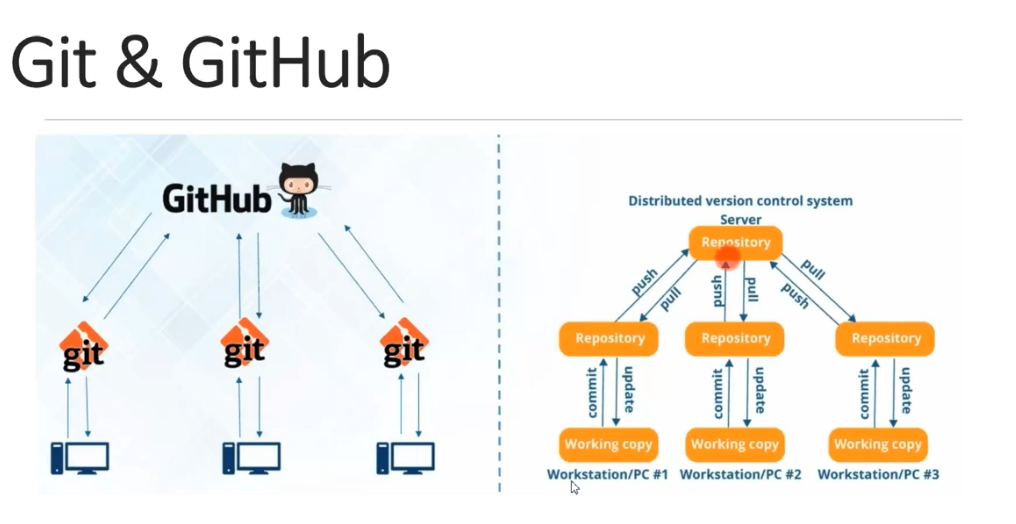
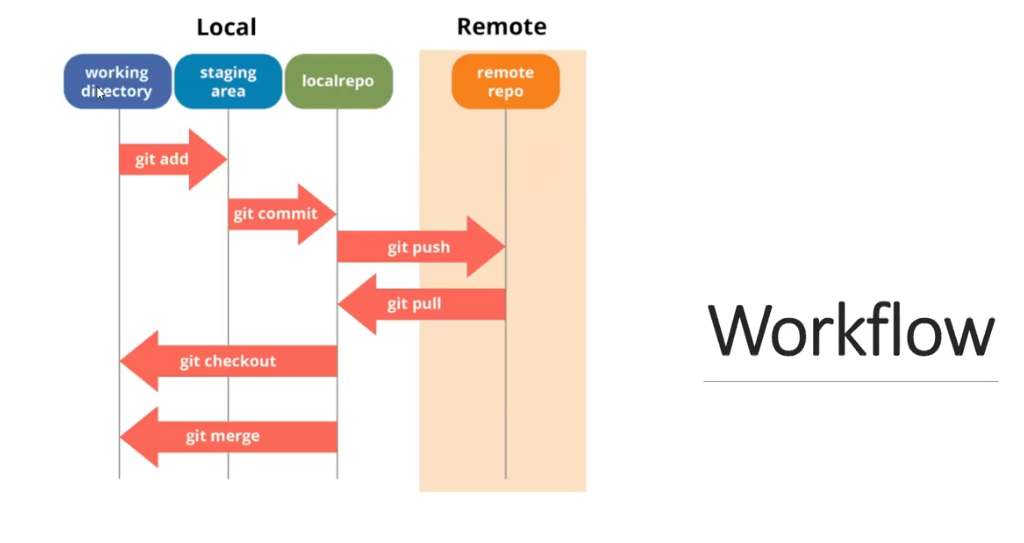
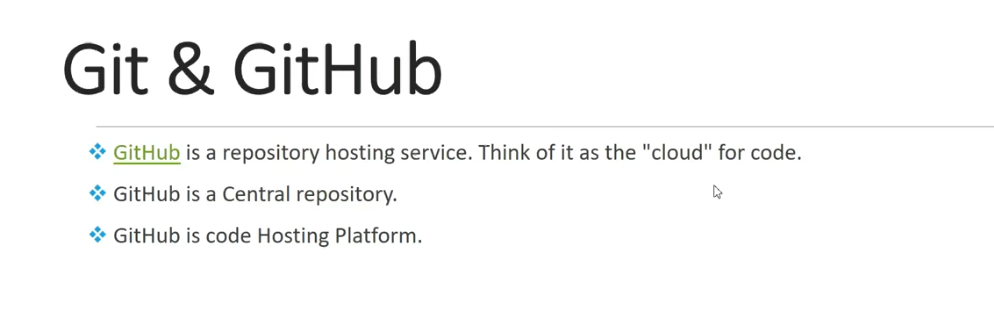
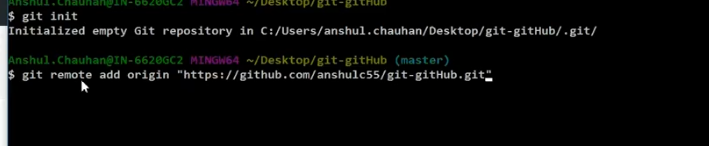
Explore GIT and GITHUB

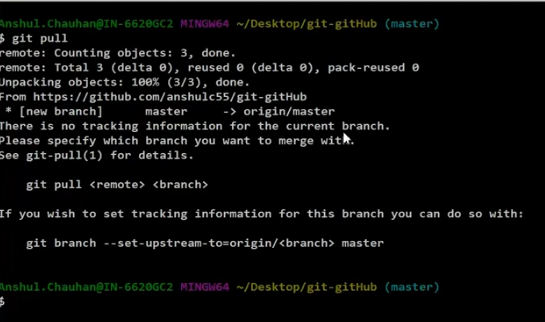






Create centralized repository :





$ git pull origin master

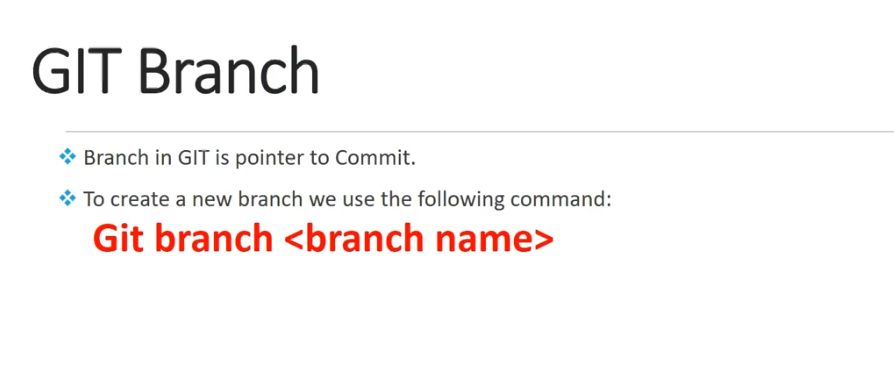
Before push always pull from origin master the files which are commited in the central repository.

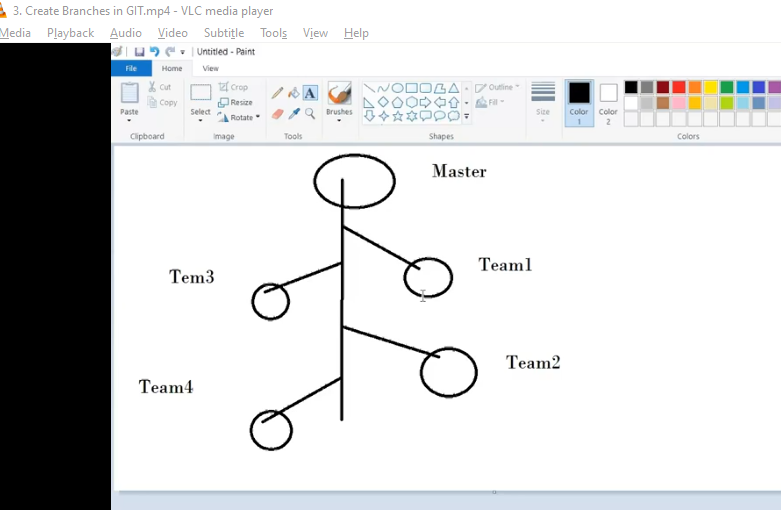
Create new file in central repo

Git pull

And then git push.

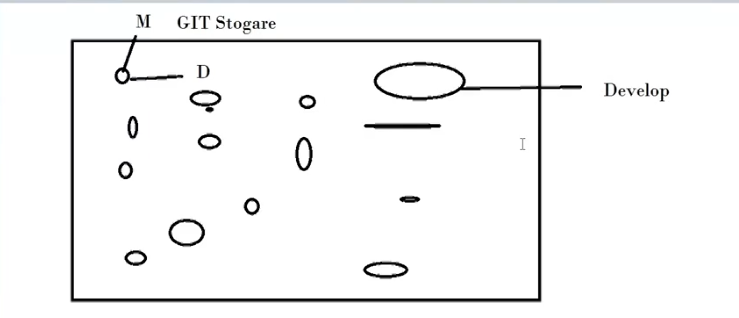
GIT BRANCH :





$ git branch develop

Branching is just the pointers, each and every file in master will create pointers to developers branch.



$ git checkout develop

$ git branch

\*develop

Master

$ create branch newBran

newBran

\*develop

Master

$ git checkout master

Readme.md sampletest1.txt Testfile

GIT MERGE BRANCHES IN GIT

$ git merge <branch-name>

To merge the develop branch with master branch

$ git checkout master

$ git merge develop

$ git push origin master

You can also pull from the main branch to other branches later using :

$ git pull origin master

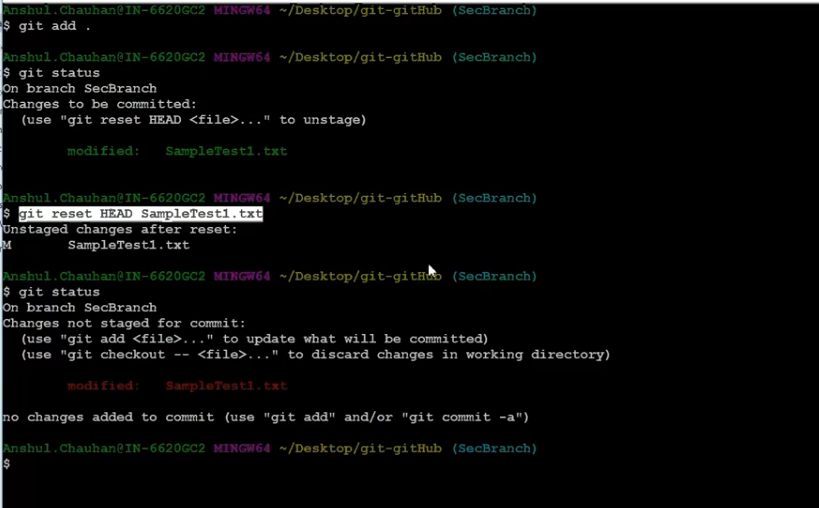
Or push to other branches using

$ git push origin devel

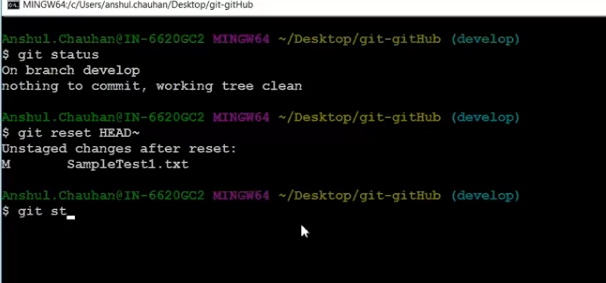
REVERT a COMMIT in GIT

$ git reset HEAD test1/text7.txt – this command to run before commit any change

$ git reset HEAD~ - this command to reset after commit is done.



To remove the last commit. Do the below :



Git status

Git log

To see the logs.

$ git status

On branch uat

Changes not staged for commit:

(use "git add <file>..." to update what will be committed)

(use "git restore <file>..." to discard changes in working directory)

modified: ../Devops - Explore GIT and GITHUB - day4.docx

modified: text7.txt

Untracked files:

(use "git add <file>..." to include in what will be committed)

../~$vops - Explore GIT and GITHUB - day4.docx

../~WRL2557.tmp

no changes added to commit (use "git add" and/or "git commit -a")