Git and Git HUB

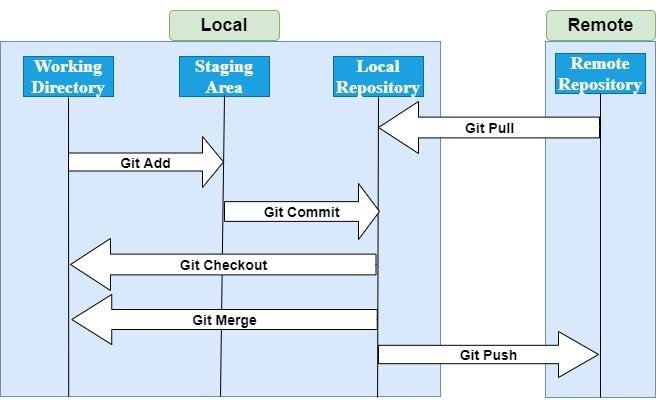
Git learning video <https://www.youtube.com/watch?v=LIhE7L__E6M>

Git bash install : https://git-scm.com/downloads

Git hub account creation : <https://github.com/login>

Putty and Puttygen install : <https://www.puttygen.com/download-putty#PuTTY_for_linux>

**Git workflow:**



Reference links for basic commands:<https://about.gitlab.com/images/press/git-cheat-sheet.pdf>

<https://www.javatpoint.com/git-cheat-sheet>

Basic steps to do in git bash  
echo "# deltenow" >> f1.txt

git init

git add f1.txt

git commit -m "first commit"

git branch -M main

git remote add origin https://github.com/venkatnarayana850/deltenow.git

git push -u origin main

Git clone:

git clone **<repository** URL**>**  (download complete repo)

git clone **<repository** URL**>**  "new folder(2)"  (download particular folder)

git clone -b **<Branch** name**><Repository** URL**>** (download particular branch)

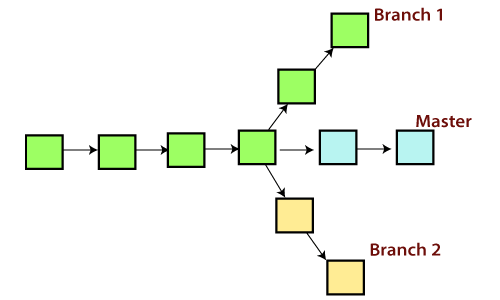
Git remote commands:

git remote show origin (or) git remote -v ( to check connection between github and gitbash)

git remote add origin <git url> (add connection to github from git bash)

git remote rm origin (add connection to github from git bash)

Git branches



git branch (or) git branch --list (list of branches)

git branch <branch name> (create branch)

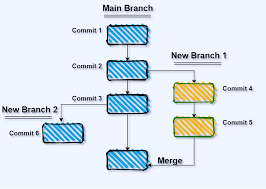
git checkout <branch name> (goto that branch)

git checkout -b <branch name> (create and goto that branch)

git branch -d <branch name> (delete branch)

git branch -m <old branch name> <new branch name> (re name branch)

Git merge and merge conflicts



git merge <commit\_id> (merge branch with commit id)

git merge <branch name> (merge branch with branch name)

Merge conflicts:  
when 2 emp’es working on same file. (or)

When 2 branches having same file name but different data.

Git rebase

Git merge and git rebase both are same in process but small difference is

Git merge get lot of commit id are generated.

Git rebase get linear shape graph and commit id’s genaerated very low(small amount commit id’s)

A screenshot of a computer

Description automatically generated

Git squash

# Git Stash

Sometimes you want to switch the branches, but you are working on an incomplete part of your current project. You don't want to make a commit of half-done work. Git stashing allows you to do so.

git stash //files goto bin

git stash list //check bin files list

git stash show // check bin file changes

git stash apply stash@{0}//(stash id) (or) git stash pop //get it from bin to working area

git stash drop //delete bin files.

Git Tags

2 types of tags

git tag <tag name>   //create a tag for your work (light weight tag)

git tag <tag name> -m “tag message” // create a tag for your work (Annotated weight tag)

git show <tag name> // check tag of the work

git tag // list of tags

git tag --delete <tag name> //delete the tag

git push origin <tag name> //push the work with tag name

git push origin --delete <tag name> //delete the work with tag name

Git diff:  
(before git add.)

git diff

git diff --color-words (for all file)

git diff --color-words <filename> (for specific file)

(before commit)

git diff --color-words --staged (for all file)

(after commit using logid)

git diff latest\_log\_id previous\_log\_id --color-words <filename>

(branches)

(add, commit after in 2 branches)

git diff master\_branch and ev\_branch

Git log

git log  (full details)

git log --oneline  (oneline with log id)

git log --graph --oneline  (online with graph)

Git status

Git status //check status of work

Git clone:

when you want to download an remote repository files(or) folders to your local computer.

Git pull:

When you want to take changes or updates done by other developer/team member on git repository, you have to use git pull.

Git fetch:

Only it will download to git bash commits, objects and refs and It fetches branches and tags from one or more repositories. Not updated in local repository. Git merge is used to download locally.

A diagram of a git bash

Description automatically generated

Git revert

Git chery-pick

Reference <https://www.youtube.com/watch?v=i657Bg_HAWI>

It is mainly used if you don’t want to merge the whole branch and you want some of the commits.

git cherry-pick <commit-hash>

A screen shot of a computer

Description automatically generated

Git Aliase:



Git config --global alias.<new alias name> “<command>”