Chapter-15

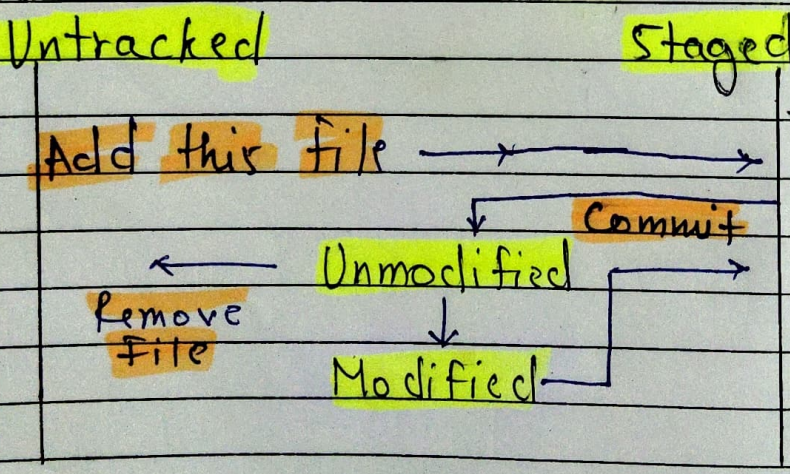
Git And Github

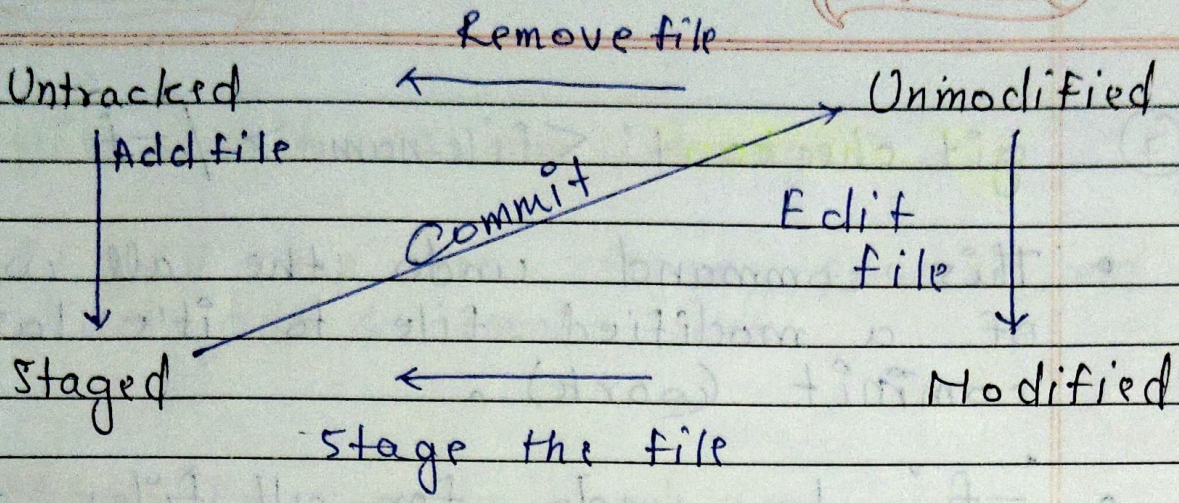
It is used to manage different versions of a project.

Commands - (Initial)

1. git init - To initiate a repo.
2. git config --global user.email <set>
3. git config --global user.name <name>

Flow chart of File Tracking.





Commands

④ git add -A / <filename>

To add file into stagging area.

⑤ git status

To check status of repo Modified, unmodified, stagged or all clear means commit.

⑥ git commit -m <message>

- commits all staged files with a commit message.
- If you don't write "-m <message>" then open a vim editor where you can write 'I' to input git message and ":wq" to commit.

(7) git checkout <file name> / -f

- This command undo the all work of a modified file to it's last commit (work).
- '-f' to undo for all files of repo to it's last commit.

(8) git log.

- To show details about commits that you have done.
- You can filtre out the logs, by various commands.

(9) git diff

It will compare working tree with stagging area.

(10) git diff --staged

compare stagging area with last commit.

(11) git commit -a -m "message"

It will skip the stagging process and direct add & commit files.

* Note: It can be used as linux command line.

(12) git rm / git rm --cached <name>

① command to remove a file completely from hdd.

② command use to remove a file from stagging area of repo. It means that file will be untracked.

(13) git status -s

tell short summary of repo status.

Git Ignore

To ignore some unnecessary & unwanted files from git to stage or commit we use [.gitignore] file.

All the file and folders name that are written in ".gitignore" folder is ignored by git.

* step 1 → create a .gitignore file using touch command.
* step 2 → write file namer that you want to ignore in .gitignore

BRANCHING

Branches are the clone or copy of repo where you can do change it does not affect differ branches.

You can create meltiple branches which help you to do experiments in your project without changing your core project.

Commands

(1) git branch <branch name>

create a new branch with name provided.

② git branch

shows all present branches in your repo.

③ git branch checkout <branch name>

it will change your working branch to name that you provided at last of command.

④ git merge <branch name>

it will merge the branch that is given to master branch.

⑤ git checkout -b <branch name>

This will create as well as switch to created branch.

GITHUB - PULL / PUSH

Step-1 : You have to creat a ssh key for your computer by using command line and taking geuidance from github docks.

Step-2 : Create a new repo and then push / pull your branches

① git remote add origin <url>

This command used to set a link for online repo to local repo.

② git push -u origin master

This command push your local repo of master branch to the online repo with link set in origin.

③ git push

Or You can only use this command push your branches to origin but if you want to change online repo link then you need use ② command

④ git clone <online repo link>

This command is used to clone a repo to your machine.