Git & GitHub

ANKIT
Date
Page

Git :- Git is a Version Control System is a tools that helps to track change in code

Github :- This is website that allow developers to store and manage their code using Git,

Configuring Git :-

git config --global user.name "Md. Akram"

git config --global user.email "____"

git config --list

This is the user name on global level

Date Page ANKIT

* Command :-

Clone & Status :-

* Clone :- Cloning a repository on our Local machine

git clon <- some link ->

cd :- change directory

Ls :- List files

Ls -a :- hidden files open

U :- Untracked files

* Status :- Display the State of the code

if modify/edit any files

then — add → commit
Stage → (1) (2)

ANKIT
Date
Page

- untracked :-
  new files that git doesn't yet track

- modified files :-
  changed ;

- unmodified :-
  unchanged ;

- staged ;
  file is ready to be Commit,

Add & Commit

- Add :- adds new or changed files in your working directory to Git staging area.

- git add <- File name ->

command

add → commit

staging area;

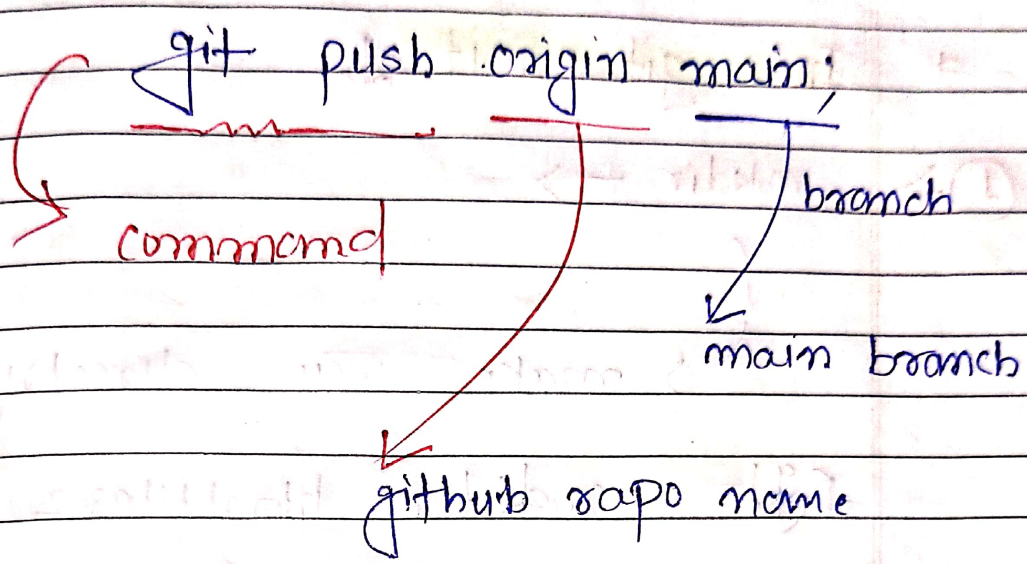
- Commit :- It is the record of change.

git commit -m "some message".

- Push Command :-

push - upload local repo content

to remote repo ;



Repogetry

Init Command :-

Init :- used to create a new git rapo

git init

git remote add origin <link>

git remote -V (to verify remote)

git branch (to check branch)

git branch -M main (to rename branch)

git push origin main

* In git init ↓

1.> mkdir →

→ make new directory.

eg:- mkdir htmlfiles2

2.> cd HTMLFILES

→ Repository name

> if you are write cd htmlfiles2 then you'r enter in htmlfiles2.

eg:- Users\mdakram\HTMLFILES\htmlfiles2

3.> git add .

4.> git status

ANKIT
Date
Page

see in git status;

your new file is added

3) git commit

git commit -m "Add new initial files"

Enter

command

git remote add origin <link>

link of new repo.

git remote -V

→ to verified

git branch

* master → show

git branch -M main

then check:-

git branch

see * main

then →

git push -u origin main

u → to setupstream

or

git push origin main

if through error.

then:- use command

→ git push --set-upstream origin main

basic roadmap

ANKIT
Date
Page

git add .

git commit -m "comment"

→ comment

↓

related to work /change /add

-git push

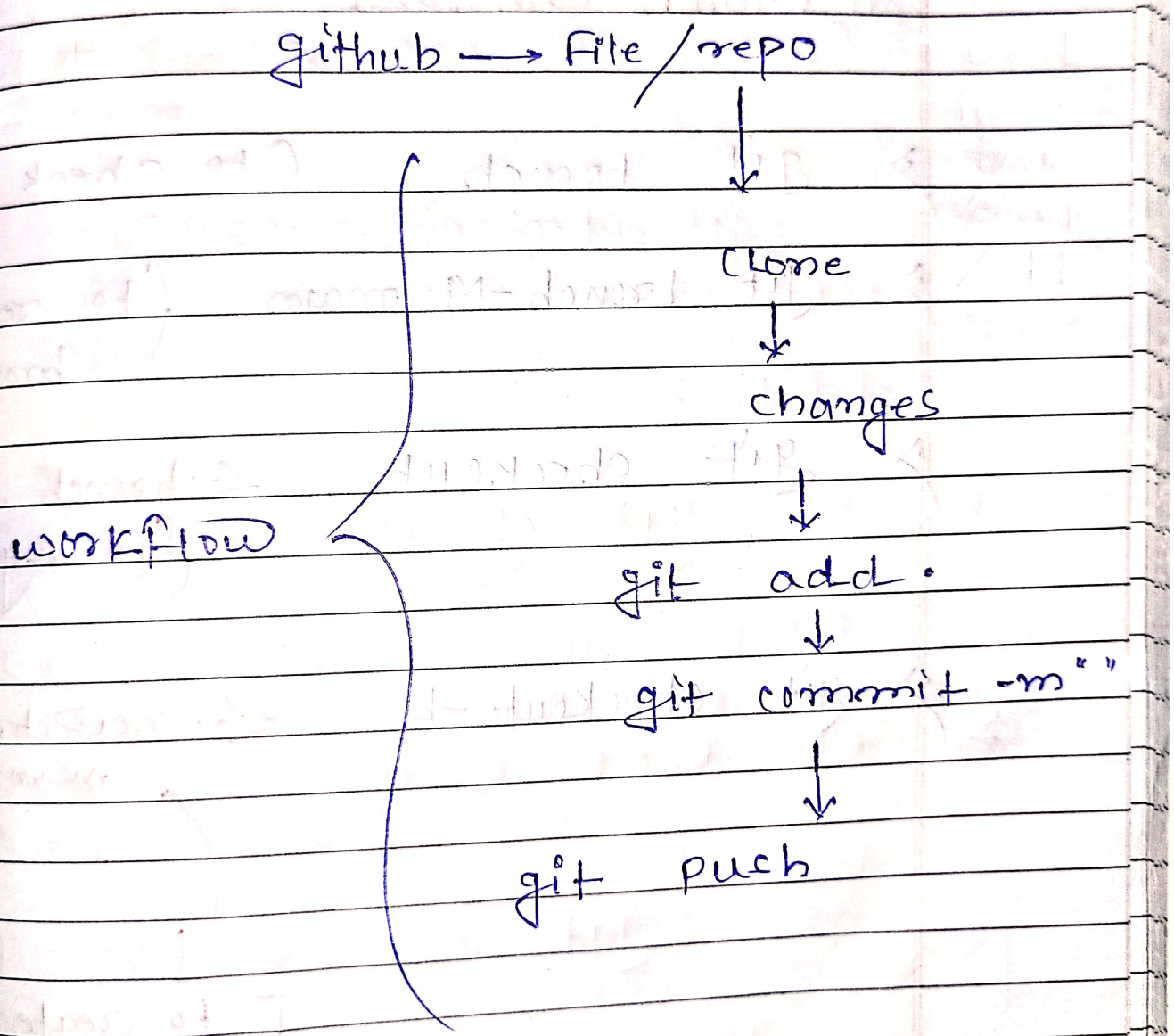
or

git push origin main

ANKIT
Date
Page

Workflow :—

Local git :—



ANKIT
Date
Page

Git Branches :-

Branch Command :-

- git branch (to check branch)
- git branch -M main (to rename branch)
- git checkout <- branch name -> (to navigative)
- git checkout -b <- new branch name -> [to create new branches.]
- git branch -d <- branch name -> (to delete branch)

ANKIT
Date
Page

To create branch work flow:-

git checkout -b repo2.1

name of branch which you'r create

git branch

then show all branches

like
main
* repo2.1
repo2.0

meaning of (*) that mean you are present in this branch currently

if you are then go to branch repo2.0 use this command

git Checkout repo2.0

you are enter repo2.0

ANKIT

Date
Page

> if you are delete some branch like repo2.1

> Then goto next branch
(जिसमें आप delete नहीं करेंगे)

git checkout repo2.0

delete :-

git checkout -d repo2.1

delete successfully repo2.1

> check branch :—

git branch

Merging code :

way - (01)

- git diff <- branch name ->
  - to compare commits, branches, files & more
  - command Line

- git merge <- branch name ->
  - enter whose branch I want to do merge

way - (02)

- create pull request

- Pull Request :-

it lets you tell other about changes you've pushed to a branch in a repository on Github

ANKIT
Date
Page

Pull Command

git pull origin main

↳ command name hai

:> used to fetch and download content from a remote repo and immediately update the local repo to match that content,

* Resolving Merge Conflicts

An event that takes place when git is unables to automatically resolve differences in code between two commits

1. Pull request
2. git merge.

ANKIT
Date
Page

> git merge <files name>

Command name

Undoing Changes :-

> Case-(01) : staged changes (add)

git reset <- File name ->

git reset ——— Whole file you want to Reset

git reset → all reset

> Case-(02) :- Commited change (for one commit)

git reset HEAD~1 → file

→ command

git log → exit → q

Date Page ANKIT

→ if you write git log command then exit in files your present

click — q. in keyboard.

→ case-(03): Commited changes
(For many commits)

→ git reset <commit hash>

command

git log pres and copy yellow clor font text

git reset --hard <commit has

→ all rest from github & V.S Code

ANKIT
Date
Page

Fork :-

A fork is a new repository that shares code and visibility setting with the original "upstream".

> Fork is a rough copy.