Git and gitHub

10 December 2022 19:05

git status = git status hume status batata hai current repository ka jispe abhi hum kaam kar rhe hai

git init = git init wo command hai jisse mai ek normal folder ko git repository me convert kar dunga

Agar kisi directory me hume enter karna hai with the help of the cmd then just type cd in cmd and drag and drop that folder there

.git folder = jo .git folder create hota hai humari repository me uske andar meta data hota hai of git repository ek folder and ek git repository me git floder ka hi fark rehta hai humara jo version control hota hai wo issi git folder pe watch karta rehta hai ki kaunsi file kitne baar change hue hai etc ye sab information git folder ke pass hota hai

Cls = this command is use for clearning the terminal

Ab agar hume kisi aur ka repository ko apne pc me lana hai hai kuch changes wagera karne ke liye to uske repository me jake code pe click kar ke uska link copy karlo then link copy karne ke baad cmd me
git clone link_of_repository folder_name/
ye command type karna hai folder_name wo folder_name hoga jisme hume uss repository ko dalna hai

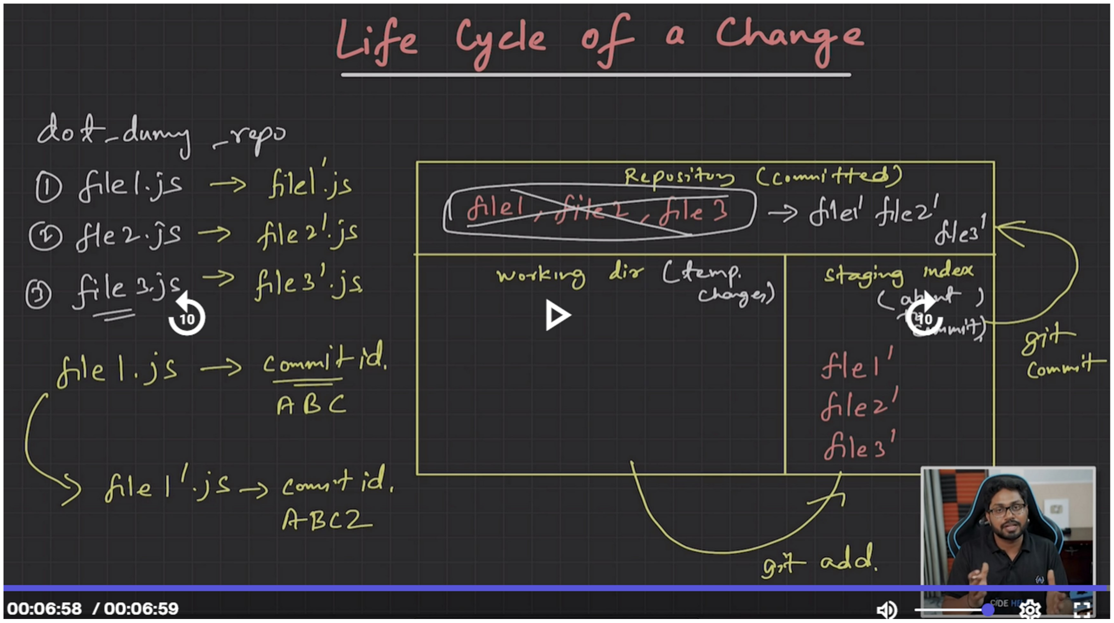
git diff = ye command agar type kiya to hume pata chalega ki kaunse file me kya changes hua hai and ye command wo jo humne clone folder banaya hoga uss directory me jake type karna hai

Life Cycle of Change

Jab hum kisi bhi file me change karte hai to usse permenantly save karne ke liye 3 stages hote hai

1. Working directory (temporary change)
2. Staging index (about of commit)
3. Repository (committed)

Jab hum file me kuch change karne hai to wo working dirrectory me rehta hai then hum jab git add. Command ka use karte hai to wo file staging index me chala jata hai and jab waha se hum commit command ka use karte hai to wo file Repository me permanent changes kar deta hai



Reviewing History in a Git Repository

git log : iss command se hume pata chalta hai ki humare repository me kab kab commit performe hua hai , git log se agar bahar jana hai to q press karna hai q press karte hi apne aap bahar aa jayega
git log -3 : isse hume sirf latest 3 commit dikhega agar 3 ki jagah 2 kardiya to sirf latest 2 commit dikhega jitne number waha pe dalenge utne latest commit dikhega
git log -p : isse hume ye to dikhega ki repository me kitne changes commits hue hai and sath me ye bhi dikhega ki kaunse commits me hume kya changes kiya tha matlap diff wala command bhi dikhega
git log --oneline : iss command se hume ye pata chalega ki commit id kya hai and msg kya hai jo jo commit hua tha commit ki puri info to nahi hogi but oneline info hogi
git log --stat : iss command se hume ye pata chalta hai ki kaunse kaunse commit hue hai and wo commit me kaunse kaunse files me change hua tha isme ye nahi pata chalega ki kya changes hua tha bas itna pata chalega ki kaunse kaunse files me changes hua tha
git show id_of_commit = iss command se hume ye pata chalega ki kisi ek particular commit me kya kya change hua hai hai (id_of_commit me jiss commit ki bare me info cahiye uska id dalna hai jisko hum sha id bhi bolte hai)

Lets Make a Commit

git add : add file to track , git add kya karti hai agar koi new file aaye to wo file add to track karti hai and agar koi old file hai to wo usko staging index pe dal deti hai
    git add file_name isse hum file ko add karte hai eg: git add main.cpp
git commit -m "message" : git commit command se hum apne file ko commit kar sakte hai and -m ka matlab hai message and "message" yaha pe msg dalna hai ki tumne file me kya changes kiya hai
git add . : ye kya karta hai jitni bhi file modified hai sabko staging index me dal do but ye karna recommended nahi hai because supoose aise bhi hosakta hai ki kuch file aisi hai jo modified hai but mujhe usko staging index me nahi dalna tha.
Head : isme head kar ke ek cheez hota hai and ye head jo hota hai wo recent commit ko pe point karta rehta hai. Head humesha top commit dikhata hai wo batata hai ye latest commit hai
git restore file_name : suppose humne ek file me changes kiya but abhi tak uss file ko humne commit nahi kiya hai and wo changes hume nahi cahiye to hum ye git restore ka use kar ke apne file restore kar sakte hai and wo file restore hogi jisko recently commit kiya tha
.gitignore : ab suppose humare folder kuch aisi file bhi hai jo hum cahte hai ki git uss file ko read wagera na kare to hum .gitignore naam se ek file bana kar jo file hume git se ignore karwana hai wo sari files ka naam yaha pe dal denge to git unn sari files ko ignore karega

Let's Create a Branch, Merge it and Tag a commit

git branch : ye hume batata hai ki humare abhi tak kitne braches hai
git branch branch_name : isse hum new branch bana sakte hai and branch name kuch de dena hai
git checkout branch_name : iski madat se hum branch ko switch kar sakte hai jo bhi branch pe switch karna hai uska naam branch_name me likh dena hai
git checkout -b branch_name : isse kya hoga ki new branch bante hi usse branch pe switch hojayenge
git merge branch_name : agar koi branch ko master branch pe add karna hai to pehele master branch pe switch karna hoga then uspe switch karne ke baad ye merge wala command use karna hai jo branch ko merge karna hai uss branch ka name dal dena hai
git branch -d branch_name : agar kisi branch ko mujhe delete karna hai to ye wala command ka use karunga and usse branch ka naam likh dunga
git tag -a name_of_tag sha_id -m "message" : iss command ka use karke hum kisi bhi commit ko tag de sakte hai name_of_tag me hume dena hai ki usse hum kya tag dena cahte hai and sha_id uss commit ki id deni hai and messege me kuch msg likhna hai
git commit -am "message" : phele agar hume kuch commit karna hota tha to pehele hum add command ka use karte the then commit command ka use karte the but agar dono chezze ek sath karni hai to iss command ka use kar

sakte hai -am means add and message

git tag -d tag_name : iss command se hum tag ko delete kar sakte hai jo bhi tag delete karna hai uss tag ka naam likh dena hai

git stash : suppose humne github se ek repository uthaya and uspe kuch changes kar rhe hai then koi aakar original wale repository pe kuch changes kar deta hai to jab hum humara kiya hua changes uss uss original wale repository se git pull karne ko conflict hojayega to iske liye hum kya kar sakte hai ki hum git stash me apna file store kar lenge jo humne changes kiya hai and original wale repository ko wapas se apne computer me lelenge and then usko unko wapas se merge karenge and merge me conflict aaya to wo hum handle kar sakte hai

git push -u origin master : agar humne apne repository pe kuch changes kiye hai but wo humare original github wale repository pe show nahi ho rha hai to hum ye command ka use kar ke original wale ko change kar sakte hai

Undo commits

Hum jab commit karte hai to wo chiz final hoti hai but humne jo commit kiya hai usko bhi hum undo kar sakte hai although ye ek achi practice nahi hai but hum kar sakte hai ye bahot rarely used hota hai

3 operations hum use kar sakte hai undo commits ke liye

1. git commit --amend (most recent commit ko undo kar deta hai)
2. git revert (revert given commit)
3. git reset (delete the commit (it is the most dangerous command))

git revert sha_id : ye revert kar dega jo bhi sha id hum dalenge isko enter karne ke baad :wq press kar ke enter karna hai tab jake wo confirm hoga

git reset --soft sha_id : ye reset command usse commit ko humesha ke liye delete kar dega and jo sha_id hai to wo wali sha_id dalna hai jiske upar ka saara commit hume delete karna hai

Ab ye --soft , --mixed , --hard aise 3 commands hote hai

--hard : agar humne ye hard use kiya to ho sakta hai humne apne working directory me kuch changes kiya hai but abhi tak usko add ya commit nahi kiya hai to wo sab changes discard hojayega isliye ye command ka hum jayada use nahi karte

--mixed : humne jo changes kiye hai wo as it is rahenge modification show hota rahega git status type kiya to

--soft : humara jo changes hai that will show as staged mostly yahi used hota hai

git diff HEAD~ : ye command hume wo show karta hai ki humne abhi recent commit kaha pe kiya hai and kya add kiya hai

git commit --amend : ab agar humne koi recent commit kiya hai and usme jo msg dala hai wo galat dal diya hai to agar usse change karna hai to ye command ka use kar sakte hai and name change karne ke liye pehel i click karna hai tab name edit hoga

How to push a commit on github destop

1. Github account
2. Empty repository
3. Local system
    a. git config --global user.name "RitikSingh30"
    b. git config --global user.email "ritikraj.rr87@gmail.com"
4. Git ko local system se link kar ke le lena hai
    a. git remote add origin https://github.com/RitikSingh30/myrepo.git (aisa kuch link milega jab empty repo create karne tab)
5. git push -u origin master
    a. Username? -> RitikSingh30
    b. Password? -> github me jake token generate kar ke uska link yaha pe dalna hai and token me saare options tick kar dena hai

    Ye upar ka 5th point tab use karna hai jab github pe push karne me error aata hai
6. git pull : agar humare original github ke file me kuch changes hua to hum usse apne working directory me dikhane ke liye ye use karte hai