

Edit an incorrect commit message

Git diff -🡪 shows diff. of changes made

Git commit –m “wrong message”

Git log -🡪 Check for log

Git commit –ammend –m “Corrected message”

**Verify:** Git log

How to undo or edit the last commit

Accidentally committing a change before

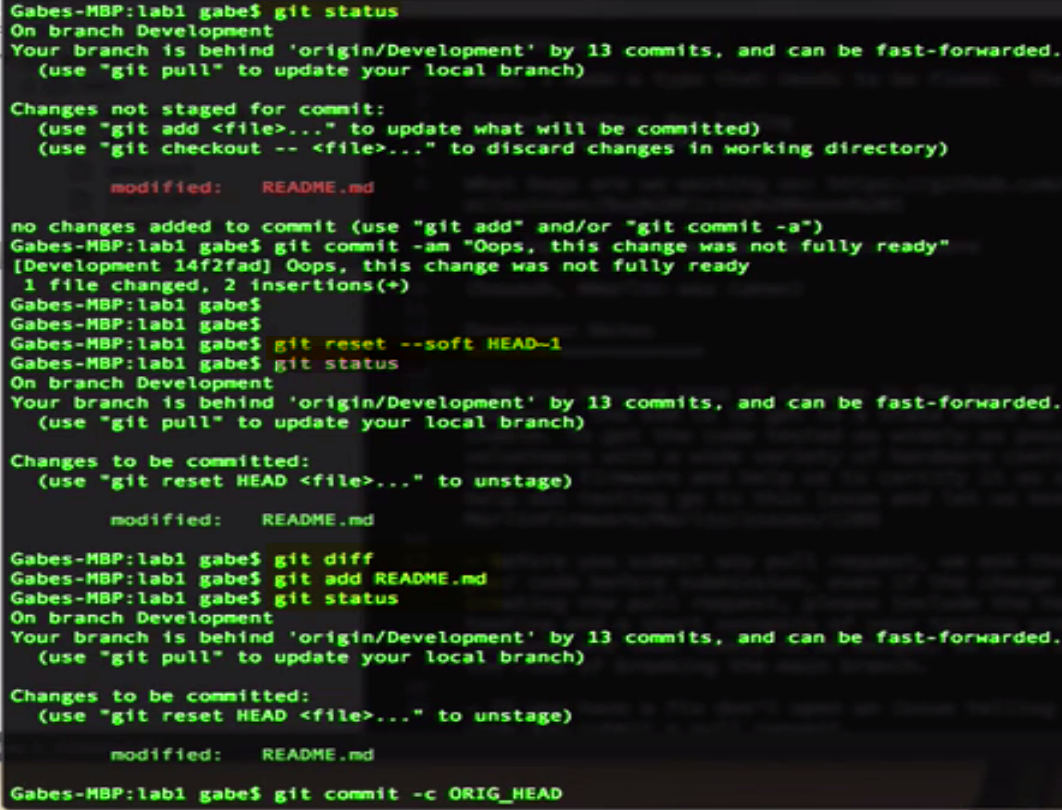
* I've committed all my files, so I may have **missed the file** or
* I may have **noticed a typo after have committed** to my local folder

and before i push it up to remote repository

So let's look at undoing a commit and then we'll redo the change

soft reset does not touch the working tree .

so that is all the files in your current folder and it does not touch the index file as well so **basically it leaves your all your files in the changes to be committed state** that get would see the mess.



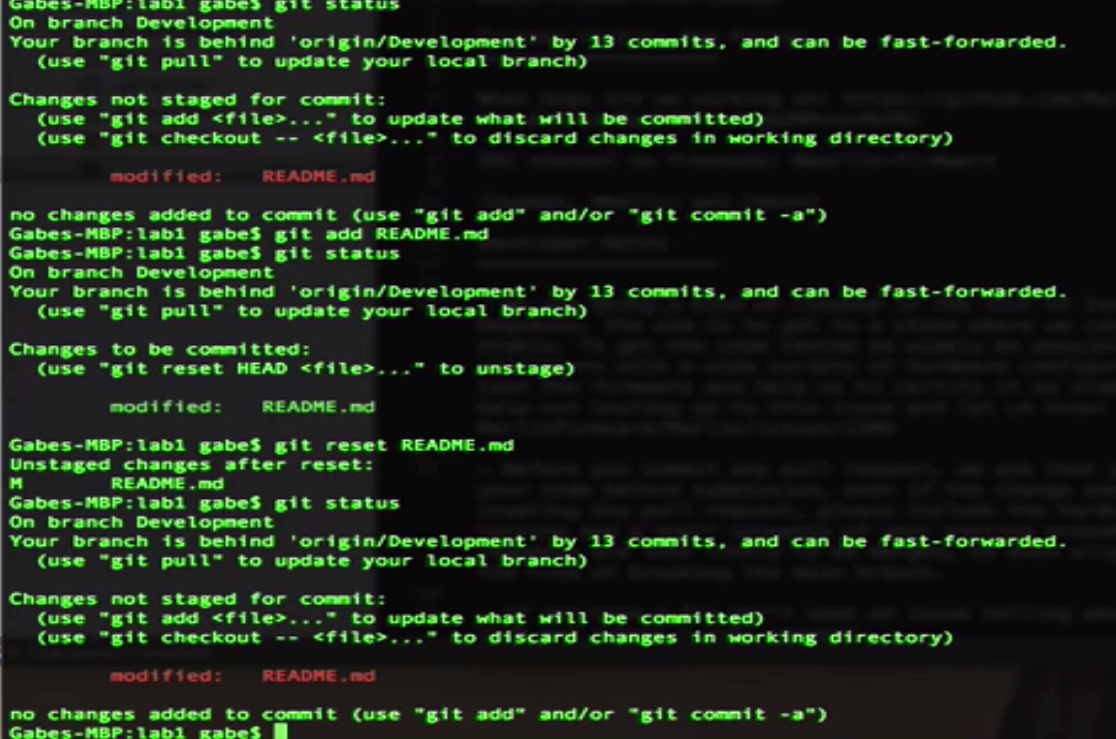
Undo an accidential 'git add' before a commit

Git log

**reverting the file to be committed**

Git commit –a -🡪 a: all files

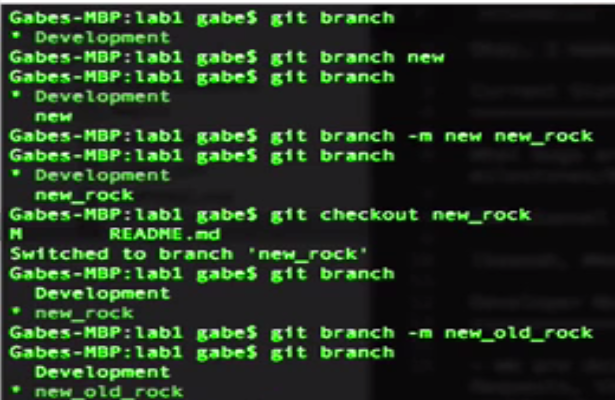
git reset read -🡪unstages the changes



Rename a local branch

I’m working with a branch that was part of a remote piece, and I realized so it's probably not the right name for it .

You can also rename the branch that you're currently on without switching.



Discover the difference between git pull and git fetch & git merge

git pull and

git fetch..

Situation: where git pull may stop working, meaning your changes aren't showing up.

You’re running into weird issues where someone may have something changes locally they made the change they push it up but for some reason it's not showing up locally.

What happens is?? git pull has an option where you can actually use two separate commands called get fetch and get merged .

git pulled you so let's take a quick peek so we'll do a quick example here I've got two tabs on my repository I have the same repository checked out into two different locations one called lab one and one called copy - and so this will let me simulate what would happen if i had a check out on my desktop and let's say a friend on your project has one checked on his project or in his desktop as well so we're simulating to people looking at the same repository so what I'm going to do is I'm going to open up my trusty readme file and say I made a change and type this into the file and save that to the disk so on my first client you can see I can see the readme file has been changed and so I'm going to commit that file and say I made a change so get commit am saying commit all file all changes and the message i made a change press ENTER and now i'm going to push those changes up to the remote deposit or e so publishing the changes for other people to get so now I run get push and so now I've pushed the change to development so now i'm going to jump to my other pc or my other tab here which is simulating another user looking at this i'm going to type get pull so I'm going to receive the change that that other user sent and you can see the readme file has changed and if i do a less on read me so let's look at the first top content of the file you can see I made a change is the very top there so as a developer my other friend made a change and I was able to receive that change using get pull now let's do that same thing again but let's do it with the get fetch and get merge and talk about how their difference so I'm going to first off make another change think i made another change and so i'm going to type again get commit - am i made another change commit that to my repository and then i'm going to type get push and push that to her in my remote deposit repository you can see it looks like a committed so now i'm going to go to my second tab here so this is simulating another user that's fetching this so get pull effectively does a get fetch and a get merge all in one instruction so the first example i'm going to type is get fetch and what gets fetched does is it connects to the repository and across all branches downloads all the changes that occurred and that's updating for your remote tracking branches so that's listed in your remotes configuration file so it pulls down everything that you're tracking and you can see that it's all change development that was pushed up to origin development origin meaning at describing the remote server that were speaking to in this case which is my github project so now i'm going to type get merge and i'm going to say get merged origins development now we type get merged we're always merging into our local branch so I'm taking the changes from origin development so from the remote server emerging into my local repository which is development and you can see it looks like read me . and he was changed and so if i type less read me you can see at the top here i made another changes there so you can see it's a git fetch get merge function just nearly identical to get pull the major difference is when to use git fetch and get merge is when you're running into trouble where for some reason files are missing or some things not working correctly or changes are not appearing that's a point where get fat and get merged helps a lot because I find that if get pulled over stops working or does weird things get that and get merged often times resolves the issue and that's the difference