

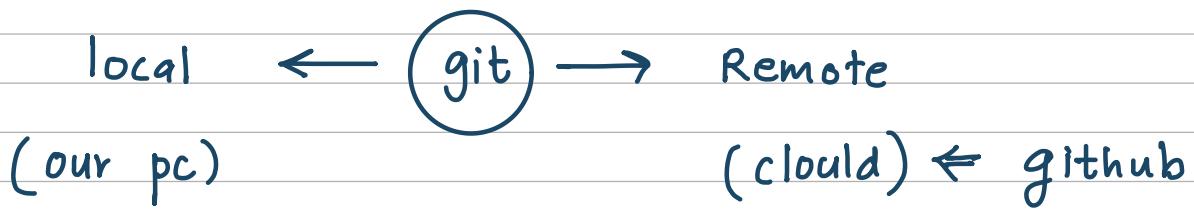
GitHub

Gitlab

Gitbucket

owned by MS

Cone Concept



1. working directory - the folders where you're working on your project.

2. Stage - changes are ready. they can move to the next step

3. local repository - temporary area where files sit between working directory and repository save

commit - permanently saving those changes to your local repository

Repository \Rightarrow where all the versions of files and their complete change history are stored.

git init \leftarrow start working on this folder
initialize

commit == save

stage every single changes
across the entire project

git add --all }
git add -A }

same => git reset

comes back to
working directory
from stage.

git add -

=> stage the changes within the
current directory you're in

current directory &
everything inside it.

git add (*)

new or modified ✓

not deleted ones.

stages all visible
changes except for
deleted files

git reset HEAD ~

=> to roll back to previous stage.
to modify or commit again.

git rm one.txt => remove add add to commit

git reset --hard => bring back the deleted
files to the wd directory

`git rm -f`

=> remove even the changes has
done locally.

`--forced` completely deletes the file

`--cached` only removes from the staging

`git rm -r < Folder>`
|
Recursive

`git log` => view commits.

Branching

`git merge main -m " "`

merging main branch into the other
branch

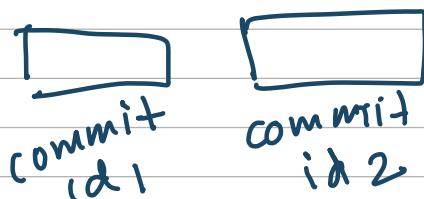
`git checkout`



back to newest version.

`git checkout main`

`git diff`

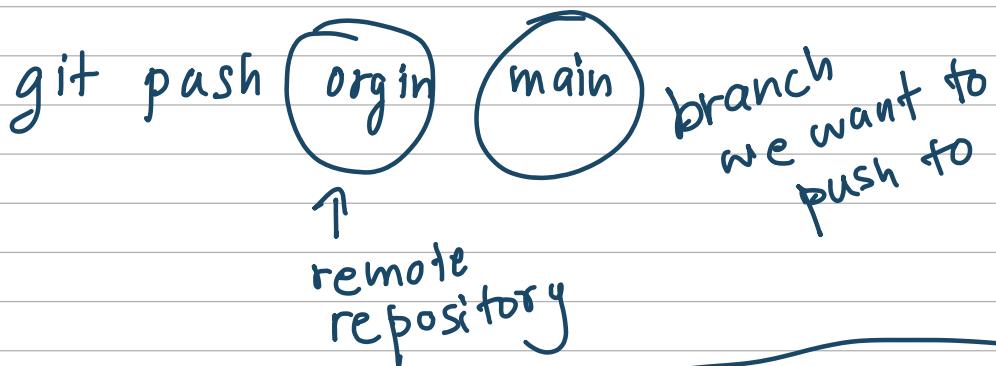


push fetch pull

push - sending local changes to the remote

fetch - bringing remote changes into your local repository, but not merging them yet.

pull - fetching plus merging - so your working directory immediately reflects the remote changes



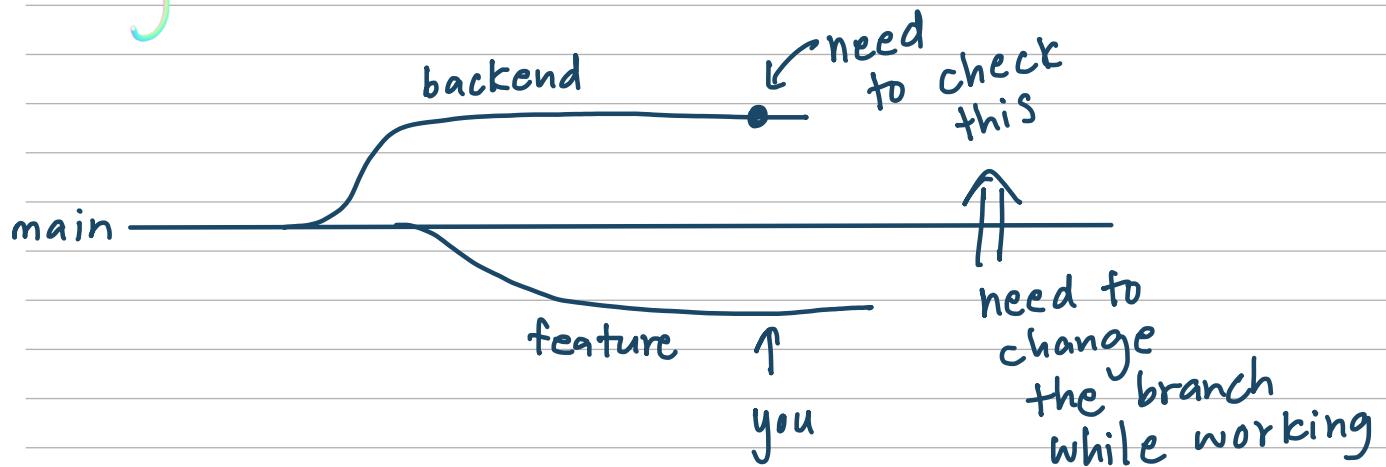
git pull = git fetch + git main

git Restore. (after add)

restore every file to its last committed stage.
(.)

git restore — staged (after add)

git stash



temporarily set aside your unfinished work,
switch to another branch to do something.

git stash

git checkout —
↓

git stash pop ⇒ remove from the stash list

git stash apply

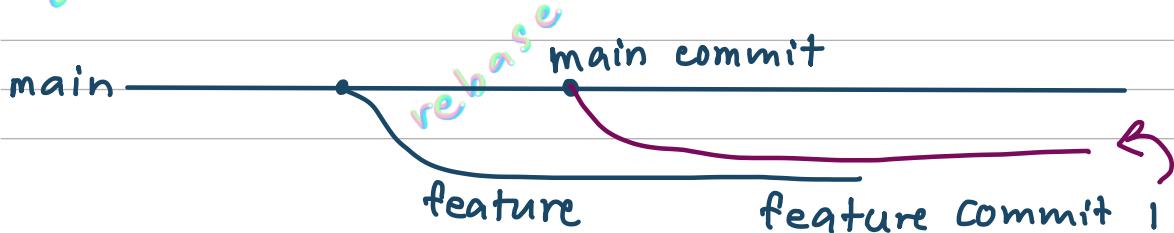
keep in the stash list for future work.

git stash list

git revert

git reset

git rebase



git rebase main ← specify the branch the
changes are in ("main branch")
need to be
in the branch
where changes are needed.

"feature branch"

not for public repositories.

pull request