

Basic Git commands

git clone

clones a remote Repository

git add file

adds file from WDir to staging area

git status

shows if there're untracked changes

git commit -m "Informative Text"

commits the changes to local Repository

git push

syncs the local with the remote repository(uploads changes)

git mv oldFilename newFilename

renames the file, on the lvl of the operating system there is no renaming, files locations get deleted and the new name inserted at the same place. if no new name is inserted (actual deletion) the data stays and gets overwritten when new data is created in the memory space it is. Git cant discern between deleting and renaming if its done outside of the git command because it just gets a new Address (name) for the Data, which would lead to the creation of new file in the repository with the same data but a new name. This can cost a lot of space and more important does not connect the old file to new named one. with git mv instead of just mv this is avoided,

git pull

updates the local repository to the state of the remote one

git diff HASH -- filename

shows the difference from the file at point HASH to the main file

git log

shows all previous Log entries for the repository

git checkout HASH -- filename

gets the file as it was at point HASH (works also without filename but then gets the whole Commit at point HASH) with main (master in GitLab) instead of HASH the most currently pushed file gets used again. if you want to go back to a former version of file make sure the the Head(current WDir file) is on the Main(most recently pushed

file) so that you stay in the main branch (so you don't actually go back to the file but reupload it as the newest Version).