

Git & Github

Git is a tool used to store the history of files and its changes. (version control system).

Github is a cloud based hosting service to manage git repositories.

Repository → Folder

git init → Initializes (creates) an empty git folder. (hidden)

git status → Shows us the tracked and untracked files.

git add. → add all the files to be committed (Recorded in history).

git add [file name] → add files individually. (add to stage).

git commit -m "your message" → commit or save the staged files into .git folder.

git log → shows history of commits.

git reset [hashcode] → resets to the hashcode commit & deletes the above or previous commits.

git stash → saves uncommitted changes for later use, and reverts back to previous commit.

git stash pop → brings back the uncommitted changes saved files from backstage.

git remote add origin [url] → links the repository in which we can save files in .git folder.

git remote -v → shows all the url's attached to the folder.

git push [url name] [branch name] → pushes the committed files into the repository.

master is the head or main branch.

git branch [name] → creates a new branch.

git checkout [branch name] → makes this branch the head branch.

all the commits will be added to head branch

git merge [branch name] → merges this branch to the head or main branch.

Fork → creates a copy of the original file into our own folder.

git clone [url] → gets a copy of the file into our system.

git remote add upstream [url] → links to the original repository from which we forked a copy.

pull request → a request to merge a branch to the original repository branch.

Only one pull request per branch.

git push origin [branch name] -f → force push (as online repository already has these commits)

git fetch --all --prune → fetches all the files or changes from the upstream repository even deleted ones.

git reset --hard upstream/main → merges my main with the fetched folder or file.

git pull upstream main [fetch and merge in github]

rebase -i [hashcode] → command to squash multiple commits to one.

s → squashes or merges this commit with above pick option.

pick → commit in which below commits will merge.