# GitHub-Commands

This is a tutorial for all the GitHub commands

Git\_Commands

Quick reference for Git Commands

Prepare Repository

Initialize the current directory with an empty repository.

git init

Clone Repository

git clone <url> <dir>

Configure Repository

To view current configurations

git config --list

Set Username

'git config user.name <"username">'

Set Email

git config user.email <"email">

Change default editor

git config core.editor <editor>

Working with repository

Show working directory status

git status

Add file to index for next commit

git add <file>

git add . (to add all files at once)

Remove file from the index

git reset --<file>

Discard file modification

git checkout --<file>

Remove file from the current working directory

git rm <file>

To commit changes

git commit -m <"commit message">

Replace last commit of the current branch with current index

git commit --amend

To view commit logs

git log -n <number>

git shortlog (To view shorter commit log)

git shortlog -s (Short log summary)

git log --onelne (Shows commits in one line)

Remove untracked files from the repository

git clean -i

Manage Branch

Detach head from current branch

git checkout --detach

Create new branch

git checkout -b <branch-name>

Switch branch

git checkout <branch-name>

Checkout to n previous commits

git checkout <HEAD~n> ("n" could be any number)

List local branches

git branch --list

List remote tracking branches

git branch -r

List both remote and local branches

git branch -a

Delete branch

git branch -d <branch\_name>

git branch -D <branch\_name> (Force delete. Use with caution)

Merge branch

git merge <branch-name>

Rebase current branch with other branched interactively

git rebase -i <other-branch>

Merge using merge commit

git merge --no--ff <other-branch>

Tags

List tags

git tag

Add tag

git tag <tag\_name>

Delete tag

git tag -d <tag\_name>

Stash

Save working directory state to new stash

git stash save "stash\_message"

List stashes

git stash list

Restore last stash and apply to working directory

git stash pop

Remove last stash

git stash drop

Clear stashes

git stash clear

Collaborate

Show remote repositories

git remote -v

Add remote repositories

git remote add <remote-name> <url>

Push branch to remote

git push <remote-name> <branch-name>

Delete remote branch

git push --delete <remote-branch> <branch-name>

Push and create new remote branch at the same time

git push -u origin master

git push origin master

git push origin master --force

Push tags/tag

git push <remote-name> <tag\_name>

git push --tags <remote-name>

Delete remote tag

git push --delete <remote-name> <tag-name>

Fetch from remote (Update remote-tracking branches)

git fetch <remote-name>

Pull from remote (Retrieve objects from remote)

git pull <remote-name> <branch-name>

Good Practices

Optimize repository

git gc

Auto optimize

git gc --auto

Check repository

git fsck (Checks integrity of objects in the repository)