**git init**

* Creates a new Git repository

**git clone**

* Clones a repository from a remote source to your local machine

**git status**

* Displays the current status of the repository

**git log**

* Shows the commit history of the repository

**git diff**

* Shows the differences between the working directory and the staging area or the repository

**git log --author**

* Shows the commit history for a specific author

**git log -s**

* git log --grep "String in commit msg"

Shows the commit history for commits with a specific commit message

**git add .**

* Adds changes to the staging area

**git add -A**

* Added only modified file

**git commit -m "MSG"**

* Commits changes to the local repository

**git branch <branch name>**

* Switches between branch

**git checkout -b <branch name>**

* Create new branch

**git push origin --delete <branch name>**

* Delete the branch from remote repo

**git branch -d <branch name>**

* Delete the branch from the local repo

**git reset <cummit id>**

* Unstaged chages from local repo

**git reset --hard<cummit id>**

* Delete changes from local repo to workind directory

**git config --local user.name "name"**

* Get name by config file in local

**git config --local user.email "email"**

***Connect local repo to remote***

1. **git remote add origin <https link>**
2. **git push origin master --force**

***Rename branch***

1. **git branch -m <current name> <new name>**
2. **git push origin : <oldname> <newname>**

***Fetch and full***

**Fetch** - **git fetch origin <branch\_name>**

* Get changes into local repo from the remote but does not merge them

**Merge** - **git merge origin/<branch name>**

* merge that changes into working directory

**Pull** -**git pull origin <branch name>**

* Remote to working directory

***Revort commit by remote repo***

1. **git revert <commit id>** //delete commit from local
2. Opend editor "Type msg" SAVE
3. **git status**
4. **git push**

***Detached Head***

1. git checkout <commit id> //Head will point to the specific commit
2. git switch -c <new brach name > // this will create new branch till that commit id

***Git cherry pick = Pick specific commit***

1. Switch to brach that you want to insert commit (eg.master)
2. git cherry-pick <commit id eg. dev branch>
3. git status
4. git push

==========

**Stage area to Unstaged**

git restore --staged <file path>

git reset --mixed

git reset

git reset --hard

=========

git rm -f <>

git rm --cached<>

**git rm --cached <filepath>**

Local repo to working directory unstage

**git reset <cummit id >**

all the above commits are unstaged