|  |  |
| --- | --- |
| command | uses |
|  |  |
| Displaying |  |
| git log --graph --decorate --oneline | Gives all commit with structure |
| Git log |  |
| Git log -n |  |
| Git log --oneline |  |
| Git log -p | gives all history what we added removed  changes done for every commit  + for added line  - for removed file |
| Git log -p -n |  |
| Git log -p –grep="any specific word which there in commit" |  |
| git log --author="" |  |
| git log --since="date"  git log --until="date" | all commit after the date ==> can use before/after  all commit on and before date |
| git touch | Create a new file |
| git init | Initialize git repo |
| git add . OR git add \* OR git add --all | Add files in staging area |
| git commit -m “msg” | Add files in local repo |
|  |  |
| git branch | List all branches |
| git remote |  |
| git branch <name> | Create a new branch |
| git checkout <branch\_name>  git switch <branchname> | Switch to brnach |
| git checkout -b <branch name> | to create a new branch and checkout to that |
| git branch -m <old branch name> <new branch name> | Rename branch |
| git branch -D <branch name> | delete a branch |
| Git checkout -b <branch\_name> <sha value> | to recover deleted branch |
| Git merge <branch\_name> |  |
| Git rebase master  Git merge |  |
| git rm --cached <file> | rm removes file from both machine and git repo  cached 🡪keep file in machine |
| git restore --staged <file> | Undo add (from STAGE to WD) |
| UNDOING COMMIT   1. Safe way-can undo this 2. Unsafe way-can’t go back once deleted   Delete specified no.of commit | git revert <SHA VALUE>  git reset --hard <sha value>  git reset –hard HEAD~<number> |
|  |  |
| Tagging -Label on commit |  |
| Git tag <name> <sha> |  |
| Git tag <name> -m “msg” |  |
| Git tag | List all tags without msg |
| Git tag -n |  |
| Git tag -l “pattern” | Display tags with specific pattern |
|  |  |
| STASHING 🡪 store local changes that are not committed  Latest stash with index 0  Git stash |  |
| Git stash list | Display all stash list |
| Git stash apply  Git stash apply id  Git stash pop |  |
| Git stash -m “msg” | To give msg to stash |
| Git stash clear | Delete entire stash list |
|  |  |
| Check remote  Git remote  Git remote -v  Git branch -a | List all branches along with remote branches |
|  |  |
|  |  |
|  |  |

|  |  |
| --- | --- |
| **Taking remote repo**  1)Git init  Git remote add <name(origin)> <url>  Git fetch name/branch | Git fetch origin/main |
|  |  |
| 2)  Git clone <url>  Git pull origin  Git push origin  Git push origin <branch name> | Pull changes from remote repo(RR)  Push changes from local repo to RR |
|  |  |
| Git remote rename <old name> <new name> | To rename remote branch |
| Git remote rm <remote\_branch\_name> | To delete remote branch |
|  |  |
| >git bundle create repo.bundle HEAD main  git clone repo.bundle <name to repo> | share changes among peers without remote repo  create packet of your local repo  to clone a shared bundle in peers machine. open git babsh in that folder |

