**Perintah yang sering digunakan di git!**

Git adalah sebuah sistem version control yang memungkinkan anda untuk melacak perubahan pada file atau proyek anda, sehingga anda dapat bekerja secara kolaboratif dengan orang lain dan mengelola versi dari proyek anda. berikut adalah beberapa perintah dasar pada git dan fungsi yang mereka lakukan:

1. **git init:** digunakan untuk membuat repository git baru pada direktori/folder kerja anda. perintah ini akan membuat direktori .git di dalam direktori kerja anda, yang akan berisi semua data git yang diperlukan untuk melacak versi proyek anda.
2. **git add:** perintah ini digunakan untuk menambahkan file yang sudah diubah atau file baru ke dalam index git, sehingga git dapat melacak perubahan pada file tersebut.
3. **git commit:** untuk menyimpan perubahan yang ada pada file ke dalam repository git. setiap commit akan memiliki pesan yang berisi deskripsi singkat tentang perubahan yang telah dilakukan pada file.
4. **git status:** untuk menampilkan status dari file yang ada dalam direktori kerja anda, termasuk file-file yang sudah di-add ke dalam index git, file-file yang sudah di-commit, file-file yang berubah, dan file-file yang belum di-add ke dalam index git.
5. **git log:** perintah ini digunakan untuk menampilkan riwayat commit dari repository git. perintah ini akan menampilkan daftar semua commit yang telah dilakukan, beserta pesan commit dan detail lainnya.
6. **git branch (namabranch):** untuk membuat, menampilkan, atau menghapus branch pada repository git. branch adalah fitur pada git yang memungkinkan anda untuk bekerja pada bagian dari proyek tanpa mempengaruhi bagian yang lain.
7. **git merge (namabranch):** untuk menggabungkan dua branch yang berbeda ke dalam satu branch. dengan perintah ini, perubahan pada branch yang satu dapat digabungkan ke dalam branch yang lain, sehingga proyek menjadi lebih terintegrasi.
8. **git clone:** untuk membuat salinan dari repository git yang sudah ada pada mesin lain/komputer. dengan perintah ini, anda dapat mengunduh proyek dari mesin lain dan mulai bekerja pada proyek tersebut.
9. **git push:** untuk mengunggah perubahan yang telah dilakukan ke repository git yang terhubung ke mesin lain/akun github, sehingga perubahan tersebut dapat dilihat dan diakses oleh anggota tim yang lain.
10. **git pull:** untuk mengambil perubahan terbaru yang telah dilakukan pada repository git yang terhubung ke mesin lain, sehingga proyek anda selalu diperbarui dengan perubahan terbaru.
11. **git branch -d (nama branch) :** untuk mendelete branch yang sudah tidak digunakan/dipakai lagi
12. **git branch -D (nama branch):** untuk mendelete branch yg sudah/ belum di merge. dalam kata lain tidak peduli sudah di merge atau belum
13. **git branch --merged:** untuk mengetahui bahwa branch tersebut sudah dimerged
14. **git checkout (nama branch):** untuk berpindah branch
15. **alias graph=”git log --all --decorate --oneline --graph”:** untuk membuat alias atau menambahkan graph secara visual
16. **git remote -v :** digunakan untuk menampilkan daftar repo yang terhubung dengan repo local saat ini beserta URL-nya.
17. **git remote add (namaremote) (URL) :** perintah ini untuk menambahkan rermote repository baru kedalam repository local kita
18. **git remote -u :** untuk mengirim perubahan pada repositori lokal ke repositori jarak jauh (remote repository) pada branch yang ditentukan.
19. **git pull --rebase (namaremote) (namabranch) :** adalah perintah git pull yang digunakan untuk mengambil perubahan pada repositori jarak jauh (remote repository) dan melakukan penggabungan (merge) pada branch lokal Anda dengan melakukan rebase pada perubahan Anda terhadap perubahan terakhir di repositori jarak jauh.
20. **git mv nama\_file nama\_folder/ :** untuk memindahkan beberapa file kedalam folder yang sudah dibuat.
21. **touch (nama\_file) :** untuk membuat file baru pada git
22. **git rm (nama\_file/folder) :** untuk menghapus file/folder
23. **git rm -r folder/ :** untuk menghapus folder beserta direktorinya
24. **git rm --cached (file.txt) :** menghapus file.txt dari Git tetapi tidak dari direktori kerja
25. **git rm -r --cached (folder):** menghapus folder beserta isinya dari Git tetapi tidak dari direktori kerja
26. **git clean -f (file):** untuk menghapus file yg masih untracked secara paksa
27. **git clean -f -d (folder):** untuk menghapus folder dan isi didalamnya yg masih untracked secara paksa
28. **git clean -n:** untuk menampilkan daftar file/folder yang akan di hapus tanpa benar-benar menghapusnya
29. **git clean -i:** untuk menghapus folder secara interaktif
30. **git clean -df:** untuk menghapus file/folder yang masih untracked beserta isinya
31. **git commit --amend -m (pesan) :** untuk merubah/membatalkan commit yang sudah di commit dan kita bisa memperbaiki commitnya (isi pesannya).
32. **git log --oneline :** untuk melihat commit dalam format yang ringkas.
33. **gitk :** untuk melihat commit log secara visual.
34. **git reset HEAD <nama\_file>:** untuk menarik / membatalkan / sudah di add ke dalam stagging index
35. **git restore --staged <nama\_file>** untuk menarik / membatalkan / sudah di add ke dalam stagging index
36. **git add /d/Project/file.txt :** untuk menambahkan file diluar direktori saat ini dan yang masih untracked. Contoh seperti diatar /d adalah merupakan direktori d. /Project adalah sebuah folder. Dan /file.txt merupakah sebuah filenya yang masih untracked.
37. **git branch -m <nama-branch-lama> <nama-branch-baru> :** untuk mengganti nama branch.
38. **git push origin -u <nama-branch-baru>:** untuk mengunggah perubahan nama branch baru agar up to date local maupun remote.
39. **git remote remove <nama\_remote> / git remote rm <nama\_remote>:** perintah ini untuk menghapus remote pada git
40. **git config --list:** untuk menampilkan daftar konfigurasi Git yang terkait dengan repositori Git yang sedang aktif atau konfigurasi global pada komputer Anda.
41. **git config --local user.name “helloworld”:** untuk merubah konfigurasi username/akun hanya untuk satu folder git
42. **git config --local user.email “helloworld@gmail.com”:** untuk merubah konfigurasi email/akun hanya untuk satu folder git
43. **git config --global user.name “helloworld”:** untuk merubah konfigurasi username/akun untuk semua folder git
44. **git config --global user.email “helloworld@gmail.com”:** untuk merubah konfigurasi email/akun untuk semua folder git
45. **git log -3/n:** untuk menampilkan commit terakhir (contohnya 3)
46. **ssh-keygen:** untuk membuat ssh key

itulah beberapa perintah dasar pada git dan fungsi yang mereka lakukan. terdapat perintah lainnya yang dapat digunakan pada git, namun perintah-perintah di atas sudah cukup untuk memulai penggunaan git.