1. git init : untuk membuat atau inisialisasi sebuah repository Git yang masih kosong dan hidden /.git/
2. git status : untuk melihat status proyek seperti penambahan apa yang telah dibuat
3. Git add octocat.txt : menambah sesuatu pada file baru bernama octocat.txt.
4. Setelah melakukan perubahan seperti git add dan Git bisa mentrack file, bisa cek status file dengan git status
5. Git commit : menjalankan perintah commit agar file yang masih terdapat pada Staging Area bisa tersimpan secara permanen di dalam repository. Contoh: git commit -m “Add cute octocat story”, perintah ini mengirimkan pesan dimana kita bisa mengetahui apa yang telah kita buat.
6. Git add ‘\*.txt’ : menambahkan file-file dengan tipe yang sama dalam sebuah repository.
7. Setelah menambahkan file-file text baru dalam staging area, bisa commit lagi dengan perintah git commit -m ‘Add all the octocat txt files’
8. Git log : melihat perubahan-perubahan apa saja yang telah kita buat.
9. Git remote add origin <url> : menambah sebuah remote repository agar bisa push ke server GitHub
10. Git push : untuk memberi tahu Git dimana semua commit kita dan push ke origin repo di GitHub. Contoh: git push -u origin master, dimana origin adalah nama remote repository dan master adalah default local branch name. -u untuk membuat git masih mengingat paramater nya sehingga kita cuma menjalankan command git push saja.
11. Git pull : mengecek perubahan pada repository GitHub kita dan membuat perubahan baru lagi. Contoh : git pull origin master
12. Git diff : melihat perbedaan dari semua perubahan yang telah kita buat dalam repository kita. Contoh : git diff HEAD untuk melihat perubahan atau commit terakhir. Contoh lain : git diff --staged untuk melihat perubahan yang telah kita staged / dalam staging area.
13. Git reset <nama file>: meng-unstage files atau menghilangkan stage file. Contoh: git reset octofamily/octodog.txt.
14. Git checkout --<target> : mengembalikan keadaan files seperti semula yaitu sebelum commit terakhir.
15. Git branch <nama branch> : membuat copy file lain yang nantinya bisa dimerge kembali dengan master
16. Git checkout <nama branch> : pindah branch
17. Git rm ‘\*.txt’ : menghapus file baik dari disk maupun stage.
18. Setelah membuat banyak perubahan seperti git rm, harus commit branch changes lagi seperti git commit -m “remove all cats”
19. Git checkout master : merge atau copy semua perubahan dari branch lain ke master branch
20. Git merge <nama branch> : merge branch ke dalam master branch.
21. Git branch -d <nama branch> : menghapus branch yang sudah tak lagi diperlukan. Contoh: git branch -d clean up
22. Git push: meng push semua ke dalam remote repository kita (final push).

Nita Hidayat

26415115/Kelas A (Selasa)