date: 22-08-2023

git commands learnt today:

1. Git status
2. Git init // to initialize the files in the git repository
3. Git add –a / git add <name of file>

This is done to stage a particular file or all files

1. Git commit -m “any comment”

this is done to commit any file(s)

1. Git log: it is done to view the messages or comments you have last given
2. Git config –global user.name// shows the name
3. Git config –global user.email// shows the email
4. Rm -rf.git// this is a dangerous command used to delete and remove the git repo file from the folder
5. Git clone “some ul” // creating a repo by copying the contents of some other master’s files and folders
6. Pwd: this is linux command for present working directory .
7. Cd: change directory
8. Ls: list the files in the pwd
9. F5+tab to autocomplete?
10. **Always remember the file status lifecycle for git**
11. **Also, if you stage a file, but then modify it, it will still be in the staging area without the modification, ready to be committed until you add this unmodified part to the staging area**
12. Touch is a linux command used to create a new file within a given directory
13. Touch .gitignore: this is file we create wherein we store the names of files and folders we do not want to modify and commit
14. Then inside the .gitignore file you have to write <file name.ext> or dir/ for a folder
15. Say you want to ignore all files of type .log, then in the .gitignore file type \*.log
16. Git diff : compares staging area and working directory
17. It does not return anything if the dir is added to staging area and ready for commit
18. git diff-- staged : compares the previous commit to the current staging area
19. git rm : this removes the file
20. git mv: rename
21. :q! :This command means "quit without saving changes". It will exit Vim and discard any changes you made to the file.
22. Git rm – cached filename: lets say there is a file already being tracked, not you add it to .gitignore and then commit, now if you make any changes to it, it will still show modified and to be staged, thus for this use this command
23. Git log -p or git log -3 or any number, it is used to see those many number of changes along with the diff
24. Git log --stat again same, in short it tells the changes
25. Git log--pretty=one line : it shows in a neat one line per commit
26. Git log --pretty=full
27. git log --since=2.days : shows all the commits made in the past 2 days
28. git log --since=2.weeks
29. git log --pretty=format:"%h -- %an" : it will print it in this format along with the names , for email use %ae
30. :wq this is used to leave vim editor
31. Git restore - -staged filename : to remove the file from the staging area( unstage)
32. Git checkout - - filename: used to unmodify the particular file ( but here you will loose the current changed if you haven’t commit it , so it can be dangerous)
33. Git checkout -f : not particular file, but more than one files in the directory
34. git push -u origin master
35. git remote add origin <https://github.com/Dhruv-Sadhale/gitdc.git>
36. git remote -v
37. git config - -global alias.st status: to rename status as st
38. git config - -global alias.unstage ‘restore - -staged – -'
39. history: it displays the history
40. cat ./filepath : gives the contents of the txt file
41. exit : closes the terminal
42. catdoc, pandoc , antiword