11) to check the status of git repository

$ git status

12) to make your directory a git repository

$ git init

13) to make all of the files and folders in git repository ready for next commit (Or we can say putting them in staging area)

$ git add --a

14)To commit (Or we can say taking snapshot of your staged files)

$ git commit -m "<message>"

15)To see recent commits on git repository

$ git log

16)To stage particular files in your git for next commit

$ git add <file name>

17) If we want to delete all file from current folder(.git) then used below command.

$ rm -rf .git

==================================================================================================================

In this video we learn Following commands:

- ls (list): present all the content present in the folder.

- pwd (Present Working Directory): Used to show your present working directory.

- cd (Change Directory): Used to change Directory and folde.

- git clone : Used to clone Repository. Ex. git clone <URL of your Repo. in the form of HTTP>

PS: If you want to change the name of your cloned repository from shell? then type "git clone <URL of your Repo. in the form of HTTP> <your file name>"

- rm -rf .git (rm=REMOVE): Used to delete your current git repository. [Deadly Command, think before used]

=================================================================================================================

20)To ignore some files to track in git repository

i) -> Create text file named ".gitignore" using command (if not exist): touch .gitignore

ii)-> Add the name of the file to be ignore in .gitignore file

21)To ignore every file based on extension

ii) -> Write \*.<extension> in the .gitignore file

example : if we want to ignore all files having .log extension

open .gitignore file and add " \*.log " in that file.

22)To ignore folder inside git repository

ii) -> Write <foldername>/ in the .gitignore file

example : if we want to ignore all folder named "dir" inside git repository

open .gitignore file and add " dir/ " in that file.

23)To ignore outer folder inside git repository

ii) -> Write /<foldername>/ in the .gitignore file

example : if we want to ignore outer folder named "dir" inside git repository

open .gitignore file and add " /dir/ " in that file.

=====================================================================================================================

24)To compare same named files present in both staging area and working area

$ git diff

25)To add modified file in staging area again

$ git add .

26)to compare same named files present in staging area and last commit

$ git diff --staged

=====================================================================================================================

27)If you want to skip the "staging area" and want to "commit" file directly? then use this command (without staging)

$ git commit -a -m "commit message"

28)To Rename file

$ git mv <present-name> <new-name>

29)To Delete file

$ git rm <file-name>

30)To make the file untracked from tracked

$ git rm --cached <filename>

=================================================================================================================

Git log

rm -rf .git

Git clone url (new name)

Git log -p

Git log -p -(2/3/n)

Git log --stat

Git log --pretty=oneline

Git log --pretty=short

Git log --pretty=full

Git log --since(n).days

Git log --since(n).months

Git log --since(n).years

Git log -pretty=format:"%h - %an"

(Git commit --ammend

Press i to make changes, then escape then colon then w and q

In short (:wq).)

=================================================================================================================

39)To do modifications in previous commit message

$ git commit --amend

40)To make file unstaged from staged

$ git restore --staged <filename>

41)To make File restore/unmodify to previous commit

$ git checkout -- <filename>

42)To make all File restore/unmodify to previous commit

$ git checkout -f