-----------------

|  |  |
| --- | --- |
| git Config  git config --global user.email [you@example.com](mailto:you@example.com)  git config --global user.name "Your Name" | To set up the configuration |
| Pwd | Present working directory |
| Cd  Cd”c/” | Change directory |
| git status | To see the status of the repository |
| git init | To make a local directory a git repository |
| git add  \*git add --a / git add.  \*\*git add <filename.ex>t | \*To stage all the files in a directory  \*\* to stage the single file |
| git commit -m ”comment” | To commit the change |
| git clone <the address> <folder\_name> | To clone a existing github repository in a local repository |
| git log | To trace the committed change in the files |
| rm -rf .git | To delete the git repository |
| git ignore  \*touch error.log  \*\*touch .gitignore  git add .gitignore | \* will create a error.log file in the repository  \*\* will create .gitignore file the the repository  If we merge the files which we just want to ignore we can merge them with .gitignore by ( specifying the names, address or mentioning ant patterns) |
| git diff | Compare the staging area with working directory  (if one file is modified after the staging- it will show what was the modification) |
| git dif --staged | Compare previous commit with the current staging |
| git commit -a -m ”comments”  Direct commit | It will commit all the tracked file in the repository but not the untracked one- which is new created or was never staged. |
| git rm <filename.ex> | To remove a file from the repository |
| git mv <prev\_file.ex> <new\_file.ex> | To rename a file presented in the repository |
| git rm --cached <filename.ext> | To stop the tracking of this file |
| \*git log -p  \*\*git log -p -n | \*it will show the log with git diff(difference)  \*\* it will show the diff for n commits |
| git log --stat | It will provide short overview of commits and difference |
| \*git log --pretty=0ne line  \*\*git log --pretty= short  \*\*\*git log --pretty=full | \*it will show the comments in one line  \*\* it will show shortly only author and commients the have made  \*\*\*it will show the author name (who has build the file) and commit name(who has made change) |
| git log --since=2.days/weeks/months | We can see the commit of different timestamps |
| git log --pretty=format:”%h-- %an” | It will show the log with <hash commit—author name > this format |
| git commit --amend | It will open a new terminal where we can edit the existing commits / merge the difference in a single commit |
| git restore –staged <file name> | To unstage a staged filr |
| \*git checkout -- <file name>  \*\* git checkout -f | \*it will reverse the file as previous commit  \*\* it will reverse all the files as previous commit  ! if we run this before commit we will loss the modification permanently |
| git remote add <url> | To adda remote repository |
| git remote -v | It will show the push(fetch) and pull repository |
| git push -u origin master  \*origin = name of thr repository  \* master/ main = brunch type | To push the code in the remote repository |
| Git config –global alias.st status  $ git config --global alias.unstage 'resote --staged--' | It will change the status alias as s  It will small down the unstaged commad as unstage |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |