git status :- to get the current staus of the working directory

git add "--a" or " . " or "<filename>":- to add the files to the staging area and start tracing the file if not traced

git commit -m "<description>" :- to commit the file

git diff :- to see the changes that have been made line by line between staging area and working directory

rm -rf .git :- delete the git repository and untrack all the files from git repository the working directory is unaffected

.gitignore :- this is a file that is needto be made in the working directory to add the name of file and folders to get ignored by the git

pwd :- to get the present working directory that is opened

ls :- this command is used to list all the files in the directory

git diff --staged :- this command is used to compare the staging area with the commited files

git commit -a -m "<description>" :- this command is used to skip staging area and directly committing works only for tracked file

git rm <fileName> :- to remove file from the working directoy and staging the change directely

git mv <present-name> <new-name> :- this command is used to change orrename the file from the working directory and automatically staging it

git rm --cached <filename> :- to untrack the particular file from the working directory

view commits methods :-

git log -p :- this command is used to show the commits that have been made so far along with the diff all the changes that have been made

git log -p -n :- this command does same as the above command just the n variable represents an integer value that defines the number of commits to be shown it start from latest

git log --stat :- shown with description and the number of line changed insertion or deletion

git log --pretty=oneline :- shows each commit in oneline with commit\_id and description

git log --pretty=short :- shows each commit in some lines with commit\_id author details and description

git log --pretty=full :- shows each commit in some lines with commit\_id ,author details ,commiter details and description

git log --since=1.hours :- this command can be used to filter the commit by hours/days/weeks/months/years "1" specify the value and separated by dot the parameter

git log --prety=format:<specify format> :- search git documetaion for more details

git commit --amend :- yhis command is used to ammend and existing commit we can commit a new file in the existing commit

the vim editor will be opened => type "i" first to start editing you can change the description after that click on "esc" button and then type ":wq" to save the changes and exit the editor

git restore --staged <file> :- to unstage a file

git restore <file> to restore the file before staging