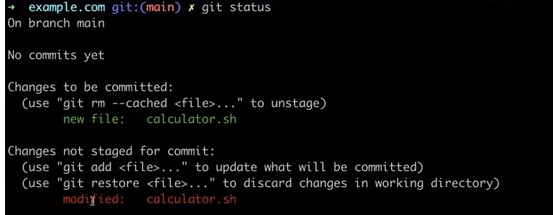
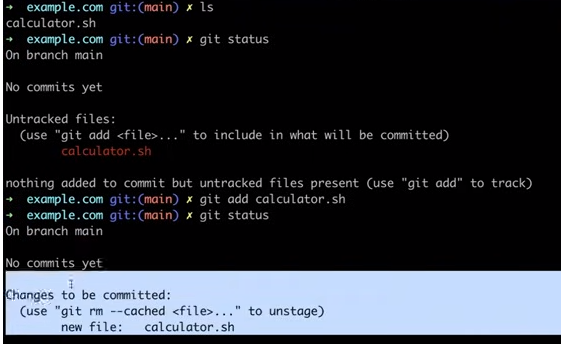
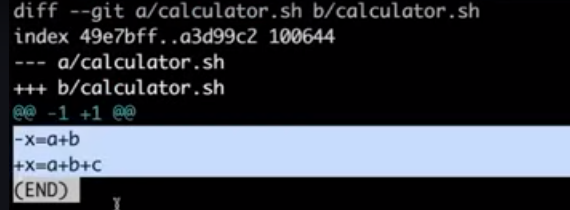
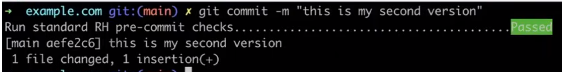
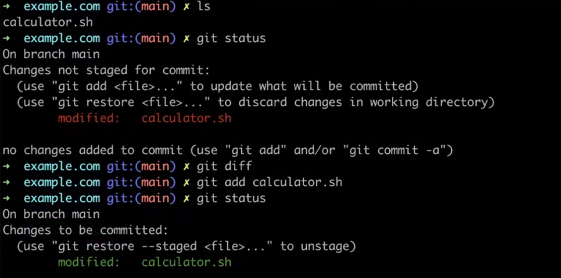
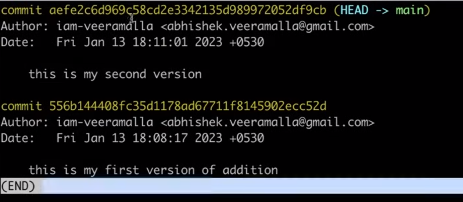
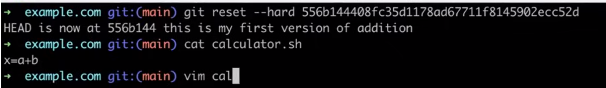
**Git & Github :**  
\* Git is a tool used for source code management. It is a free and open-source version control system used to handle small to very large projects efficiently. Git is used to tracking changes in the source code, enabling multiple developers to work together on non-linear development,  
as it keeps track of every version of your codebase  
\* GitHub is a web-based hosting service for git repositories. i.e solution built on Git  
\* We can use Git without Github but we can’t use Github without Git  
\* GitHub is an online software development platform. It's used for storing, tracking, and collaborating on software projects. It makes it easy for developers to share code files and collaborate with fellow developers on open-source projects  
\* They have built Github to overcome asability, raising issues, commenting, talking to peers, Reviweing the code and project management (like project tracking etc)  
\* What is Fork?  
@A fork is a copy of a repository that allows you to make your own changes without impacting the original project  
\* In Github comman commands are **git push**,**git add** & **git commit** , Everthing or every file we create in Git is tracked as Objects  
\***Git hooks** are scripts that run automatically every time a particular event occurs in a Git repository. They let you customize Git's internal behavior and trigger customizable actions at key points in the development life cycle or they can help you automate your development lifecycle  
\*The **Git config** command is a convenience function that is used to set Git configuration values on a global or local project level. These configuration levels correspond to .gitconfig text files. Executing git config will modify a configuration text file.  
\***Git status** is a command which shows any files not getting tracked in repos and says it add that file to track it by using git status filename and from now on it looks after its changes and shows its changes of a file if we use command **git add** filename   
  
modified file will be shown in red color   
  
\***Git diff** is a command used to see changes in a file or check the changes u have modified to a file here first it was x=a+b but changed to x=a+b+c  
\* when we do some changes to a file we need to add changes by git add filename and see status by using git status then if it shows as changes to be committed then use git commit –m “this is first/second version” here we r commiting this changes as versioning , in 2nd pic we can see that   
\* If we want to know the changes or exact commits we made we can use **git log** to know the no of versions it has and this way got can track & commit 100s of versions   
\*If we want to go back to old version we need use commit id(highlighted one) and command for it is **git reset –hard commitId** & this one way of versioning   


\* If we want to share or send data from local repo to github use **git push origin master/branchname** and before that use **git remote add origin githubrepolink** from github to connect ur local repository to online github repo  
\* In git there are 4 process – **untracked>modified>staged>commit**  
@untracked means in this the new files which we added will be present but not saved   
@ staged means in this files will be saved but not committed   
@commit means in this after committing this files new versioning will start with first commit on each file(**git commit –m “messagefor version like first/second version**”  
@modified means in this we can get committed file to re-edit and use **git add** to get staged  
& then commit that file after staged   
\* **git add .** is used to convert files from modified untracked files to staged then commit them to modified to commit it  
\* after editing a file use **git add** . as command to convert modified to staged and use **git commit** command to add oneline text in editor and save it and use command **git log –oneline** u will get text message in oneline u edited saved last version   
\*command **git branch** is used to know the branches present in repository and to create branches use **git branch newbranchname** and to switch from one branch to another use **git checkout branchname**   
\*to move from untracked to staged use command **git status** and if we need to remove files from staged then use **git rm --cached filename**(it will be removed ur file from staged & it will move to untracked)  
\*git log is used to check no of committed files and before commiting at first time we should use configure it use git config –global user.email “ssrssr@gmail.com” & git config –global user.name “ssrssr”  
\*to check oue connected mail username use **git config –global user.email/user.name**   
\* to connect from ec2 to github we need to do first see the location in which ur staying like ur in pem file location or not when u connect to ec2 and then use **git clone linkofgitrepo**>**export username=”username of github**”>**export token=”token no”**  
\*to connect from from local repo of git to github use **git clone linkof online/githubrepo** > **git config –global user.email=” “> git config –global user.name=” “> git init(only if u want ur local repos to show in github then use this orelse just move from last step to add file & next add>commit>log>remote(only first time)>>push>check in github.  
\*** 