***Version Control Using Git and the Command Line***



First install git for windows and then initialize it.

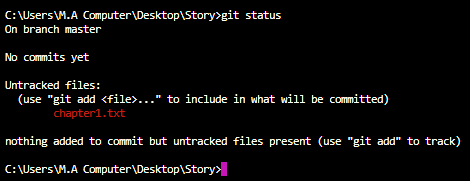
**Initialize Git:**

git init



**ls –a** command is linux find equivalent command for windows.

**git status**



Red files are undirected files. To track the changes use the command git add.

git add chapter1.txt

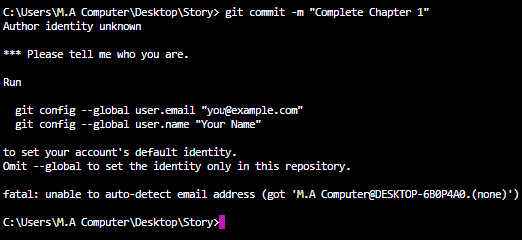
Now the file has been added and it is green now.



For all files **git add \*.txt**

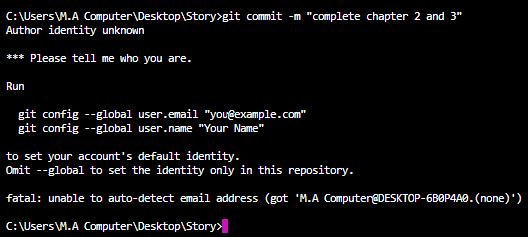
-m flag to add a commit message

**git commit –m “what changes are saved in initial and current save point”**



What commit you have made see by command **git log**

git add . //to add everything in the folder (all files).



Add author name

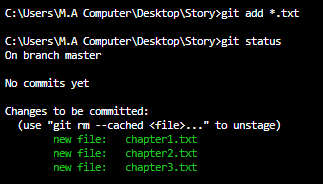
$ git config --global user.name "John Doe"

$ git config --global user.email "john@doe.org"

$ git config user.name "John Doe"

$ git config user.email "john@doe.org"

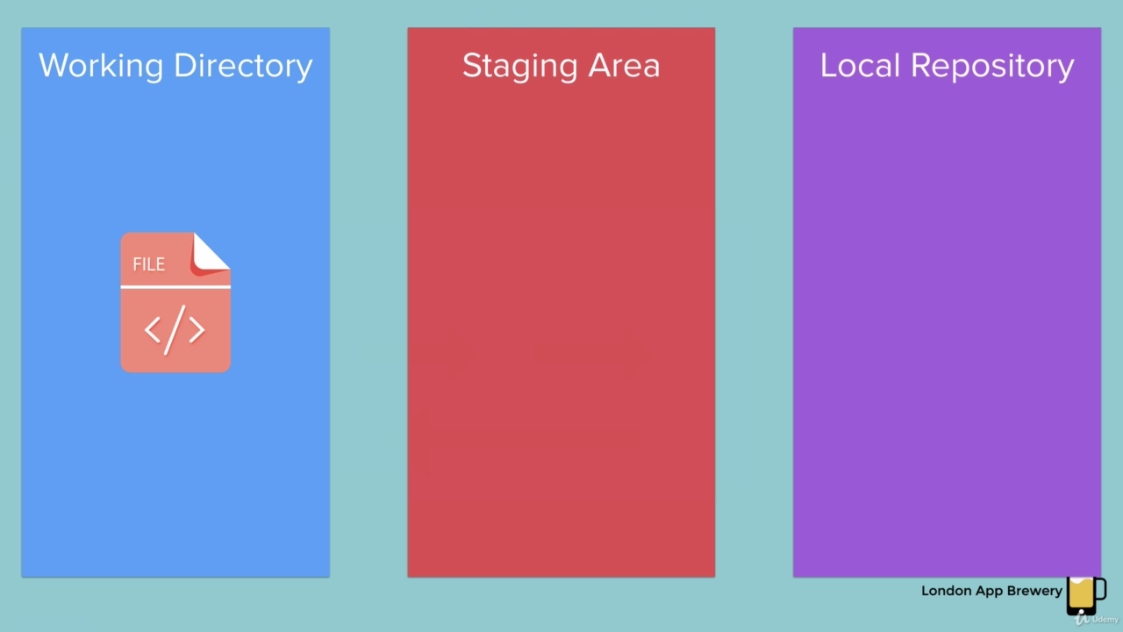
**git status**

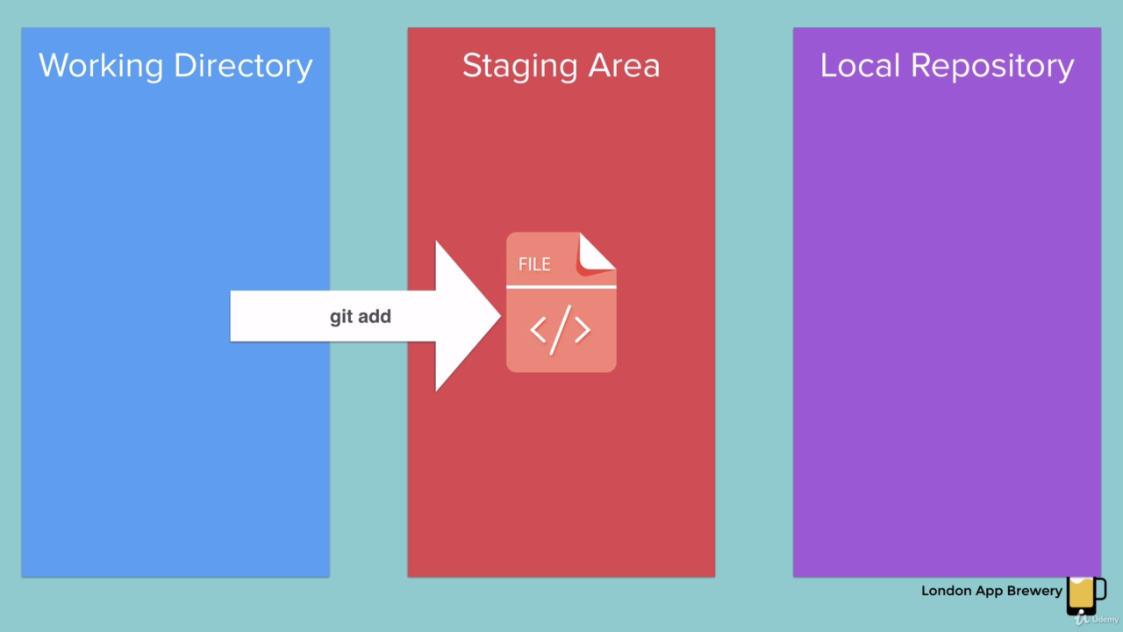


Let’s recap what we done

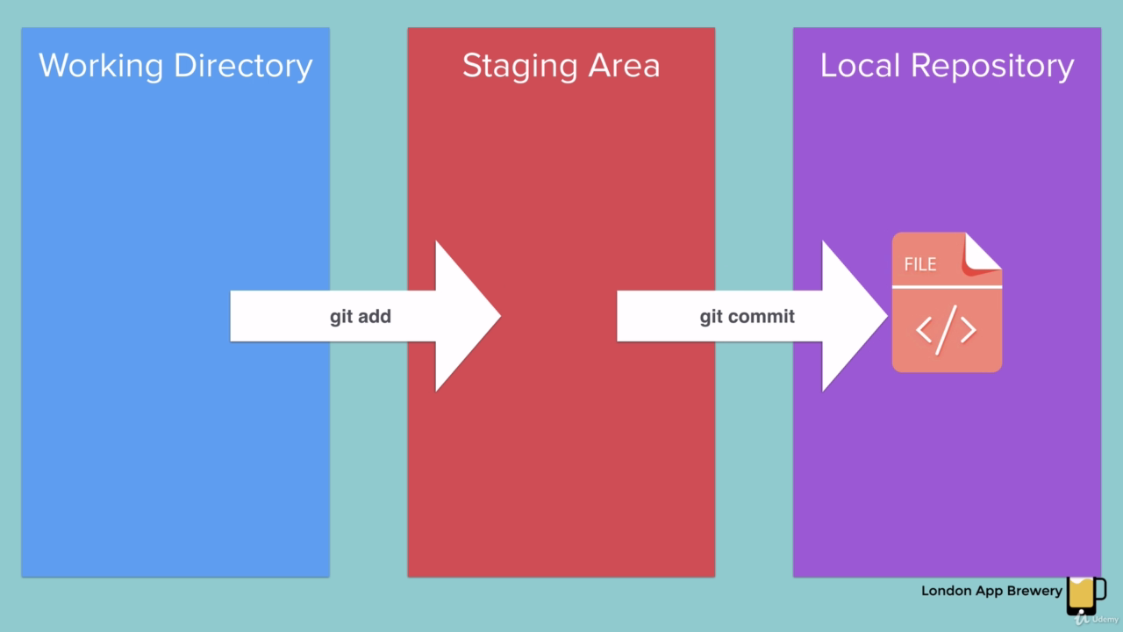
Working Directory:

In which we initialize file.

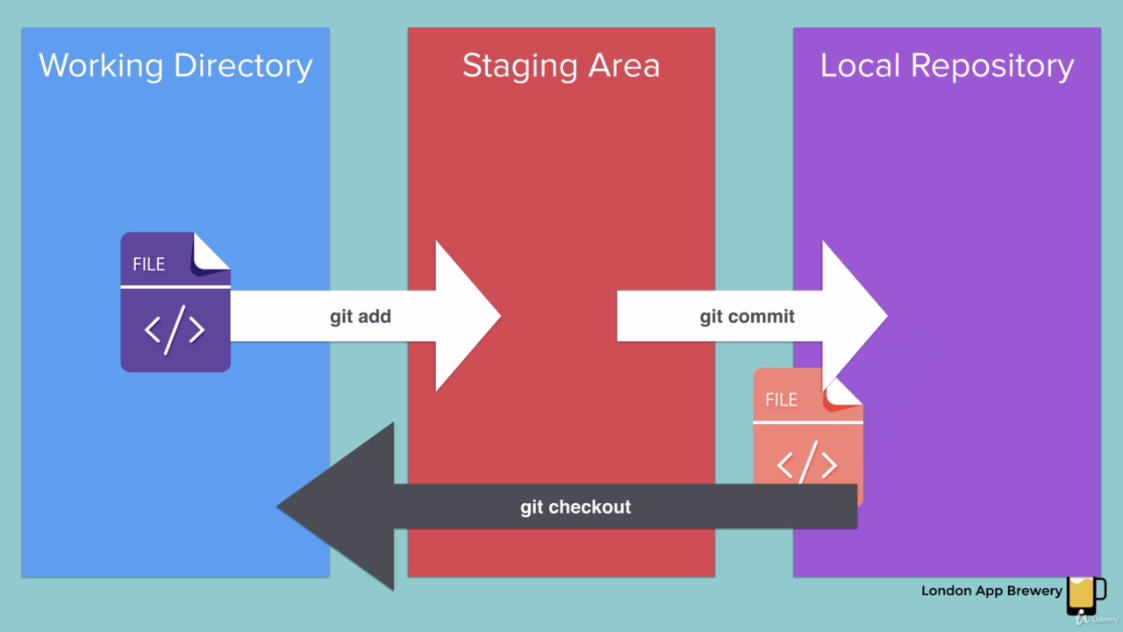




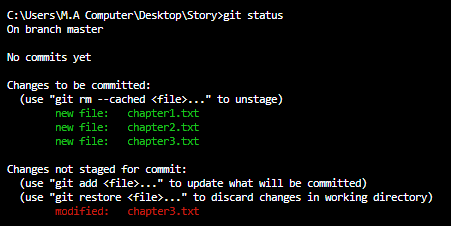
**Staging Area:** Sometimes you want to add all of your files to be tracked. Staging area is best place to trying figure out things you want git to ignore.



Now file is in local repository **.git**

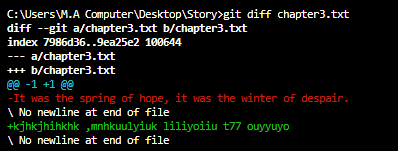


Change text in chapter3.txt and roll back using git.



Above file is currently in local repository.

Check difference between previous and current files.



Red part is the part that we deleted and green part is currently.

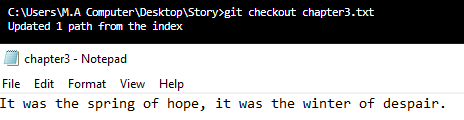
**git checkout** command



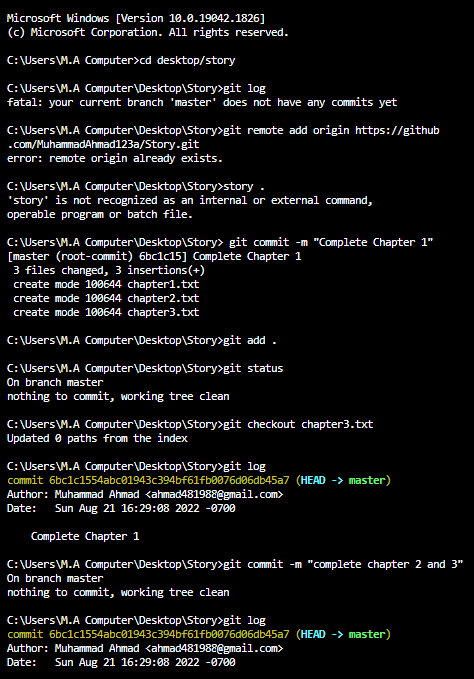
This command says to roll back the repository (chapter3.txt) to the last version committed in a local repository

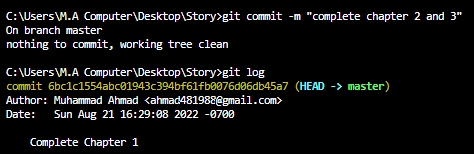
Check file is roll backs.

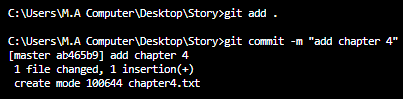




**Corrected of Above**

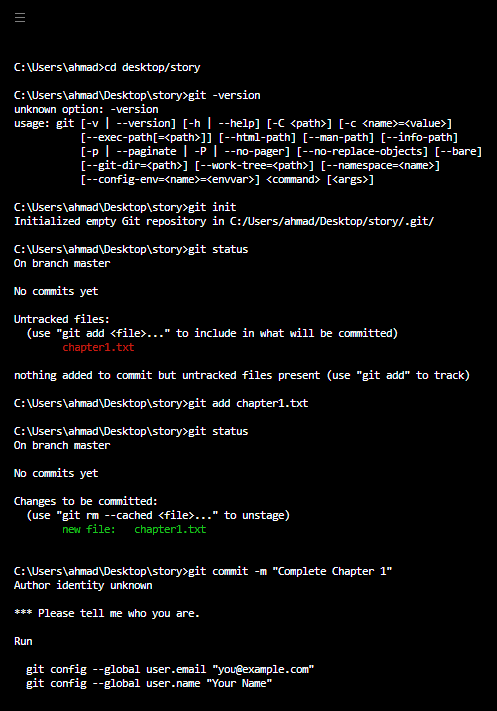


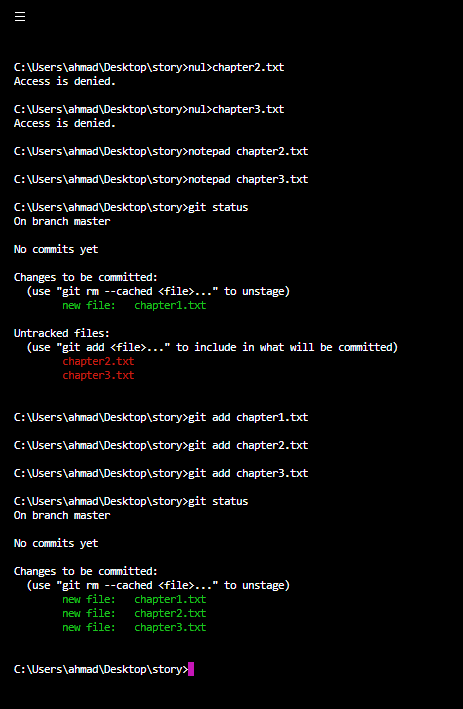


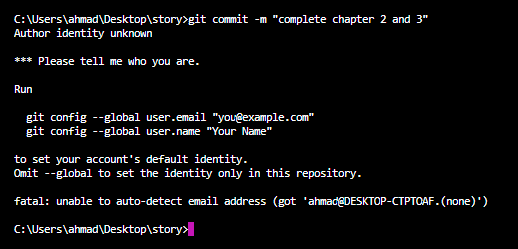


**New notes 2023:**

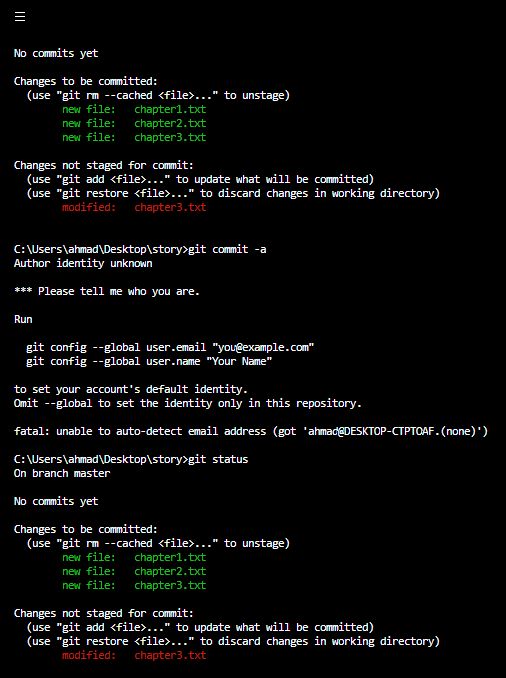


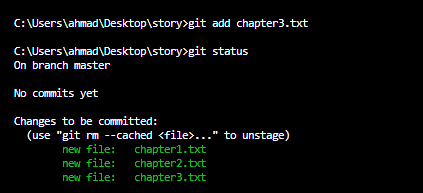


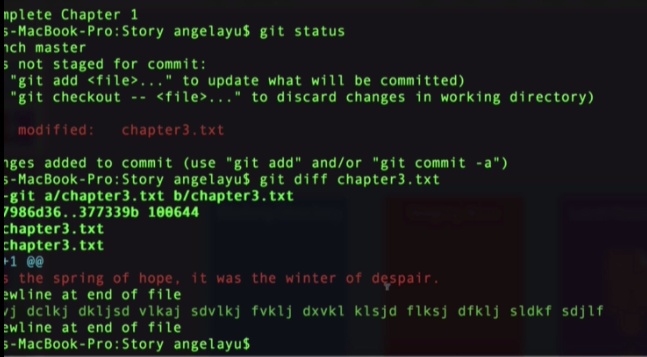




Now change the text of chapter3.txt file and save it. We can see that changes are not modified until we commit them.

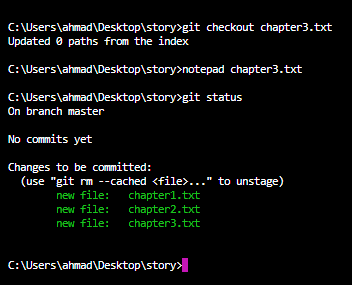






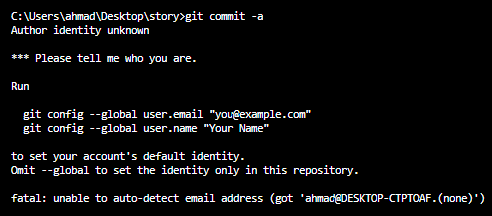
Now delete changes by using command (Restore the file)

git checkout chapter3.txt



To check commits configure email and name by running commands config

Configure name and email





Now commit running

