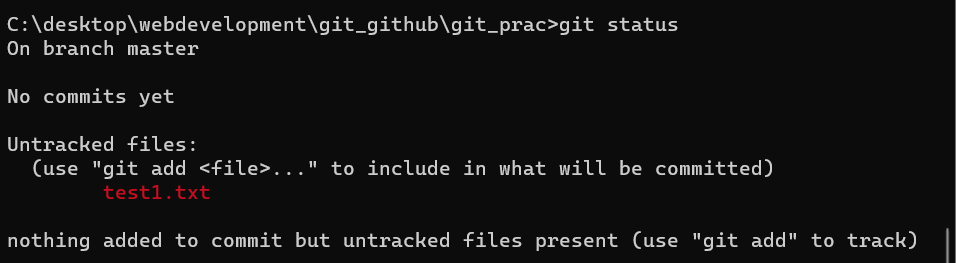
Git local repository creation

First create the folder and add the git into it by using the command **git inti**



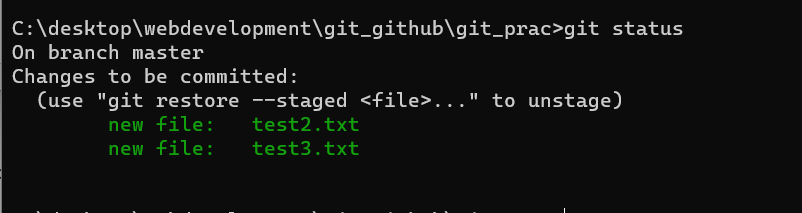
**git status** command is used to see the status of the repository or a particular file  
  


Next we need to add all the files in the working directory to the staging directory to keep track of their changes by git

We can do that by using **git add <filename>**

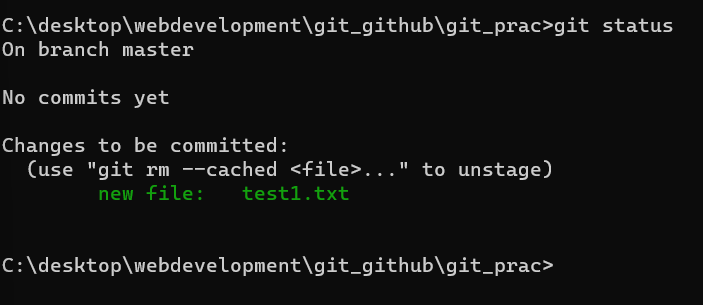
****

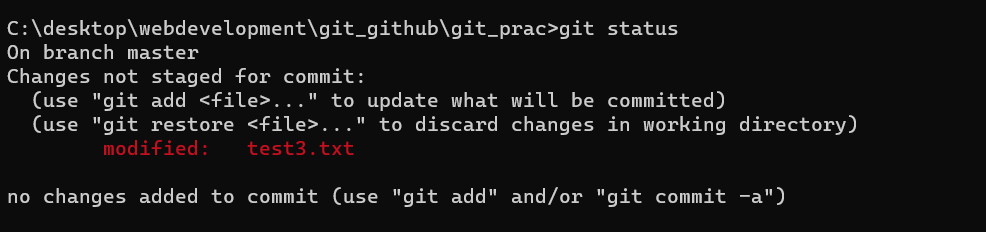
We can use the command <**git add .** >to add all the file in the working directiory into staging directory.



Now the all the files are added into the staging directory

**git status**  after adding it into the staging directory

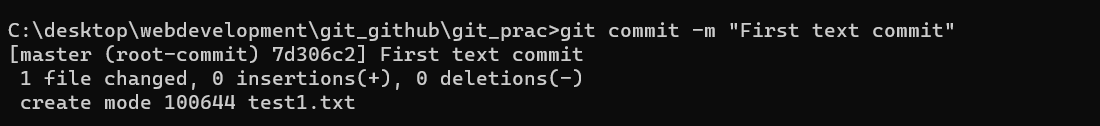




**Git status** showing the uncommited file

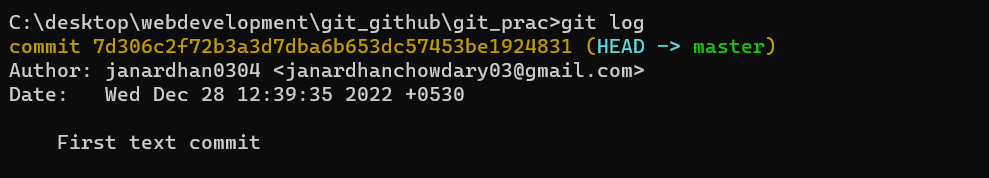
Now we have to commit the changes to make a save point at that situation/stage .For this we need to use cmd **git commit -m “Message ” (\* as a best practice keep message in present tense \*)**

**Eg git commit -m “First text commit”**

****

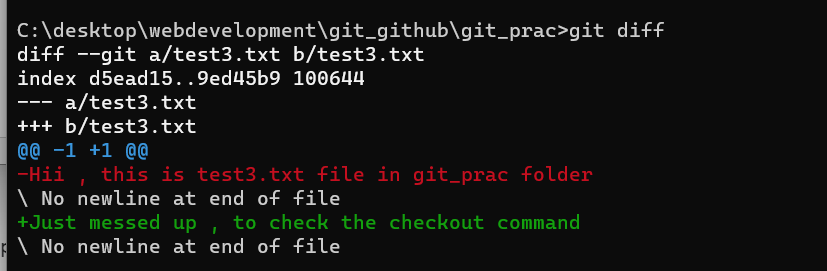
Now a check point will be formed at this point of time , in layman it will be version 1.

We can use  **git log**  command to see the commits we made



We will be shown the hash and the HEAD in the working version details and also the commit message

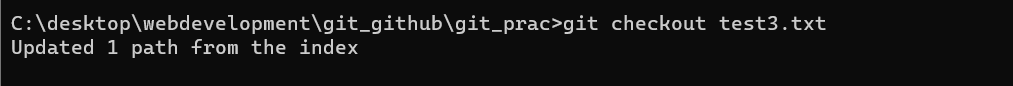
**git diff**  to check the changes in the repo



**git checkout <filename>**  is the command used to go back to the last version of our code.

Briefly here I have first commited the file test3.txt and then by mistake I have erased the text in it and entered some wrng data which I don’t need, so now I need to go back to the previous version.

I can do this by using checkpoint command



Now let us make the repository in our github that is to move local repository to github

So far we have created local repository now our task is to add this local repo to the github /remote and the command we use for that is

**git remote add origin <and the url of the repo>**

here origin can be anything ,it is just a convention to keep it as origin

origin is the name of the remote

this will create a remote for us in our github

now we can push our local repo to this remote repo which is origin

the command for that is

**git push -u origin master**

**origin -> the name of the repo**

**master is the name of the brance**

there is PAT in github ,which is used instead of password inorder to access our github through git CLI

if any doubt refer to this site [**source**](https://stackoverflow.com/questions/68775869/message-support-for-password-authentication-was-removed-please-use-a-personal)

**GIT IGNORE**

First create git ignore file using command is

**Type nul > .gitignore**

**attrib -h .gitignore**

in windows.

**git clone url**  of the repo

we can even view the git logs of that repo if we clone it

Git branching

**git branch <new branch name>**

to create new branch

**git checkout <branch name >**

takes us to the branch which we want to go also bring back the files as if they were in the that branch.

**git merge <child branch name>**

in order to merge the branch fist go to master branch and then use the above command

**IN ORDER TO SEE THESE CHANGES IN GRAPHICAL FORM :**

**GITHUB> REPO> INSIGHTS> NETWORK**

Advanced topic

**Git fork** this is used just to bring someones remote (github repo) to your github and then we can clone it and make changes to it ,here these changes will only be reflected in our github repo but not on the owners repo. Inorder to make our changes to be added into the repo by the owner we first need to make pull request and if the owner is happy with the changes he can approve the changes

The owner can see how many have forked his repo

**git pull :**

this is a way to communicate with the owner to tell them I have made changes (improvements to your code ) if you find it ok then merge it.If the owner feels changes are good then she accepts the request and merges it with the master /main branch