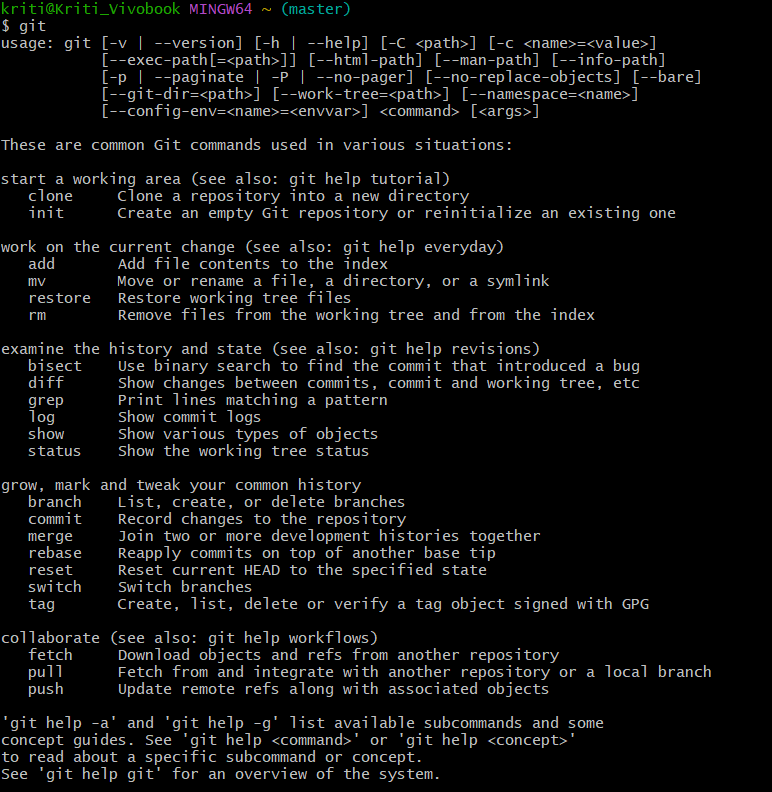
**SOURCE CODE MANAGEMENT**

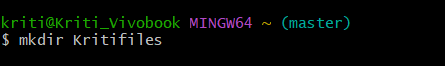
**git**

This command gives the basics of using git.



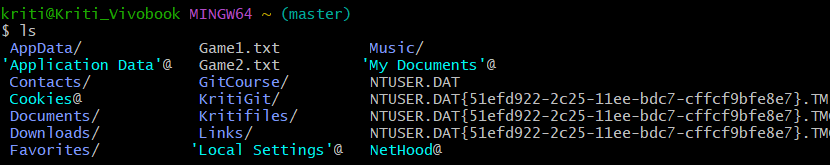
**mkdir (folder name)**

This is used to make a folder

****

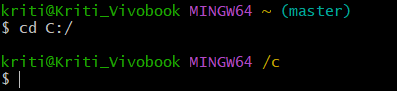
**ls**

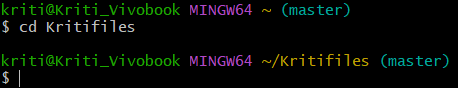
This is used to list the files present on the system.



**cd (folder name)**

This is used to enter into a folder or any drive.





**git --version**

This command gives the running version of git.



**git config –global user.name “ ”**

It is important to tell our computer who you are.



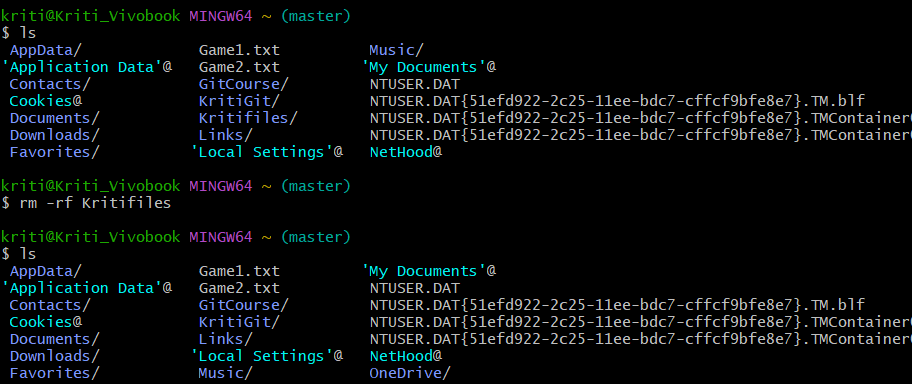
**git config –global user.email “ ”**

This command is used to configure your email id.



**rm -rf (folder name)**

This is used to remove the directory.

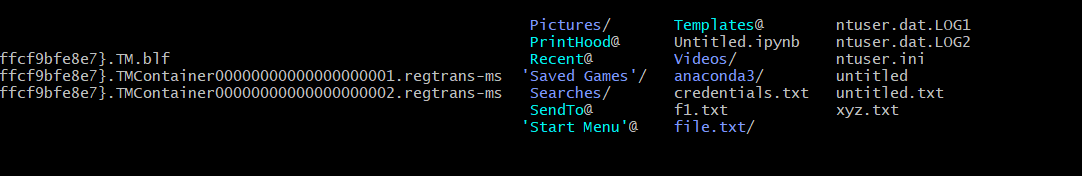
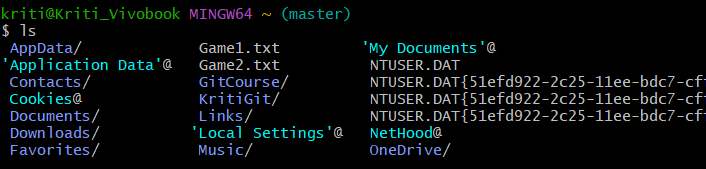


As you can see in the above picture, firstly ***ls*** command is used to check the files or folder available on the computer. Then we used ***rm -rf Kritifiles*** to remove this directory from the computer. As you can see that when again ***ls*** command is used that folder is deleted.

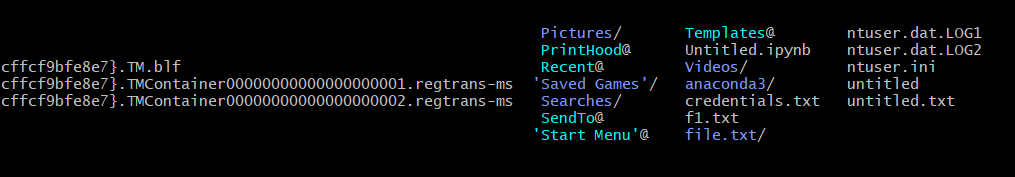
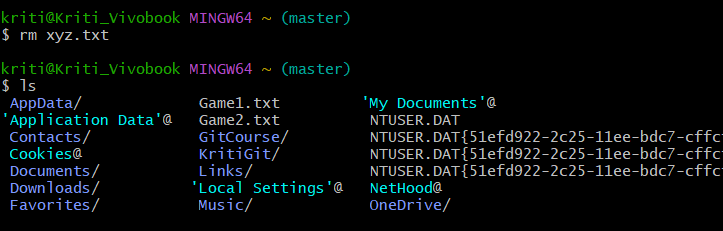
***-rf***( is used to signify that we are removing a directory instead of removing a file.

**rm(file name)**

This command is used to remove any file.



Here using ***ls*** command, we get a list of files available on the computer and from these files choose the file to remove from git.

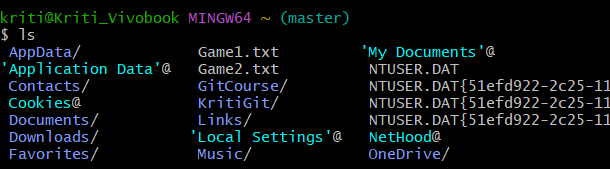


Here using ***rm xyz.txt*** , this file xyz.txt is removed and also its verified using ***ls*** command.

**git ls-files**

Steps To Commit A File In GITBASH:

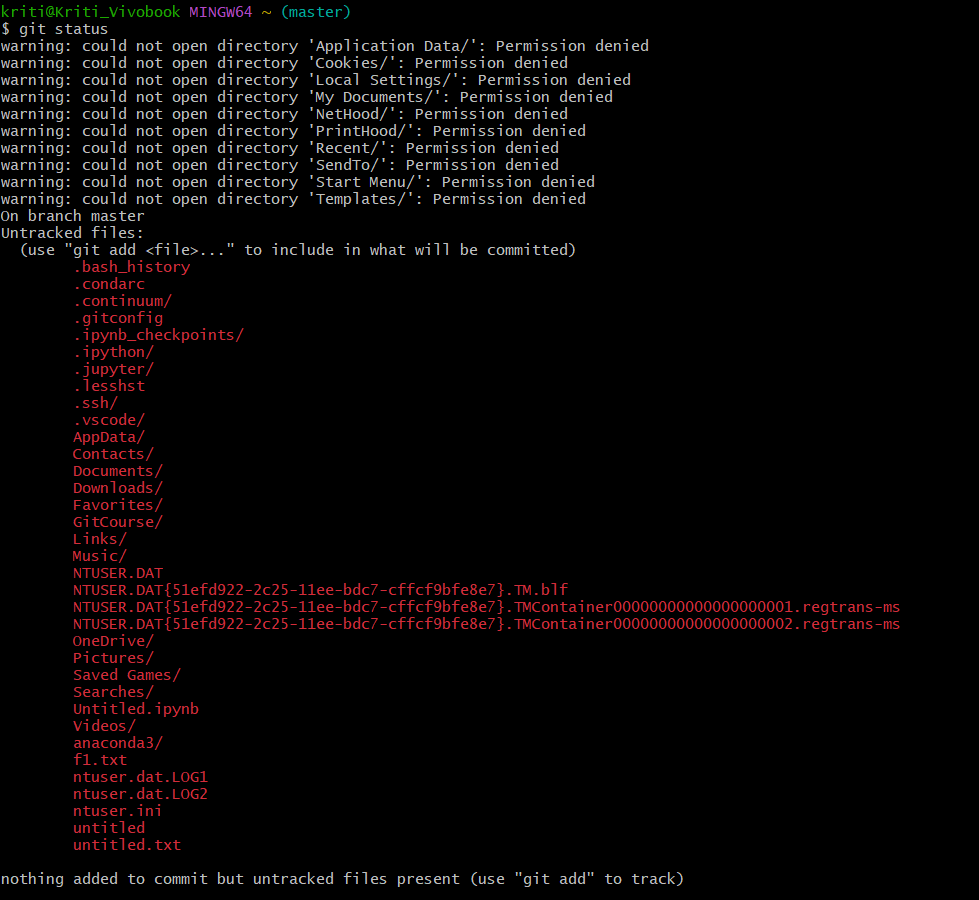
Firstly, run ***ls*** command to check the files on the computer.



Now to commit any file among these available files follow the following commands:

**git status**

This command will display the files available to commit and are in process to commit so we will run this command to check the files available to commit.



Now, select any file from the list of files being displayed to commit.

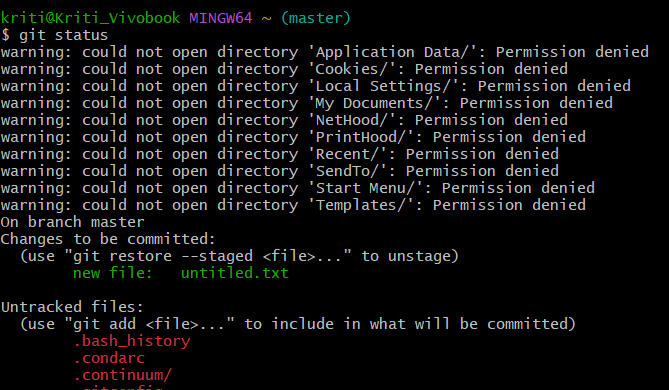
**git add(file name)**

This command is used to add the file we want to commit.



Now, we will check whether file is in staging area.

**git status**

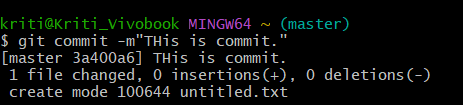


Here, we can see the text highlighting in green informs that our file is being pushed to staging area.

Now, we will run another command to finally commit this file.

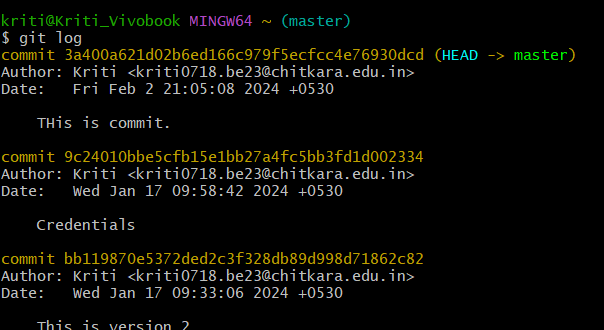
**git commit -m “ ”**

This is the command used to commit the file. Its necessary to add the message to commit any file.

****

**git log**

This showcase the total no. of commits done .



**git diff**

This command gives the difference between the content of the committed file and the modifies file.

**git dig –staged**

This command is used to get difference between the committed and staged file.

**touch(file name)**

This command is used to make a new file.

**echo” ”>>(file name)**

This command is used to add text to a file and a new file can also be created with this command.

**git branch -m main**

This changes the branch to main.

**git push -u origin main**

This command is used to push our committed file to git repository on GitHub.

**cat (file name)**

This command read the content of the file.

**git show(tag name)**

This command is used to create a tag.

**git push origin(tag name)**

This is used to push our tag to git repository on GitHub.

**git tag**

This shows all the tags created in that repository.

**BRANCHING**

There are basically two commands used for branching:

Older command: checkout

New command: switch

**git switch -c (branch name)**

With this command we can simultaneously we can create a new branch and switch to it.

**git switch (branch name)**

This command is used to move to the branch name which is mentioned.

**git branch -d (branch name)**

This command is used to delete the branch whose name is mentioned given that we are presently not on that branch.

**git push origin (branch name)**

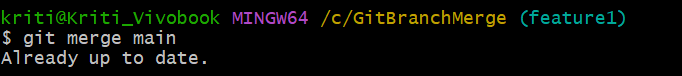
This command is used to push branch to GitHub repository.

**Merging Branches**

Two branches are merged so that the main branch contains the content of the second branch as well.

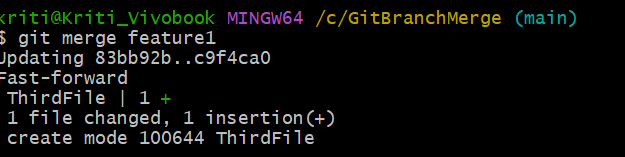
**\***It is necessary to be on main branch while merging any branch. Otherwise it will fail.

**\***It is important to note that this command will only work when you are on the main branch and it will merge the named branch with the main branch.



**git merge(branch name)**

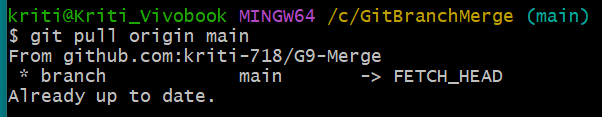
This command is used to merge two branches, when you are already on main branch.

****

**How to push changes on remote repository**

It is a good practice to use the pull command before using the push command.

**git pull origin main**

****