start --> Git --> GIT CMD

1.git config --global --list

if you get any details like below

fatal : uable to read......

2.git config --global user.name "viji lakshmi"

3.git config --global user.email "coolviji84@gmail.com"

4. Now give the same command

git config --global --list

5.define editor as notepad++

git config --global core.editor npp

6.set autocorrect true

git config --global help.autocorrect 1

7. open GitFundamentals directory

cd GitFundamentals

if it doesnot exits create it and open it

md GitFundamentals

cd GitFundamentals

8.git config --global color.ui auto

9.git config --global core.autocrlf true

if windows preferable is true

10. now check the config by using

git config --global --list

11.cat ~/.gitconfig

12. if you want to remove some configurations, then use

git config --unset core.autocrlf

git config --unset use.name

13. npp .git/config

Create Repository:

===================

1.echo "Hello, GIT" >README.txt

2.git status

3.Create file

git add README.txt

now check

git status

4.then

git commit

--u will get notepad with details, add "Added README.txt" at the first line.

5.git log

--it will show the changes

6.notepad README.txt

--once you get the notepad with "Hello, GIT"

--add one more line "GIT Started"

---close the notepad

7.git status

--it will give modification details

8.git add -u

--u indicates update

9.git status

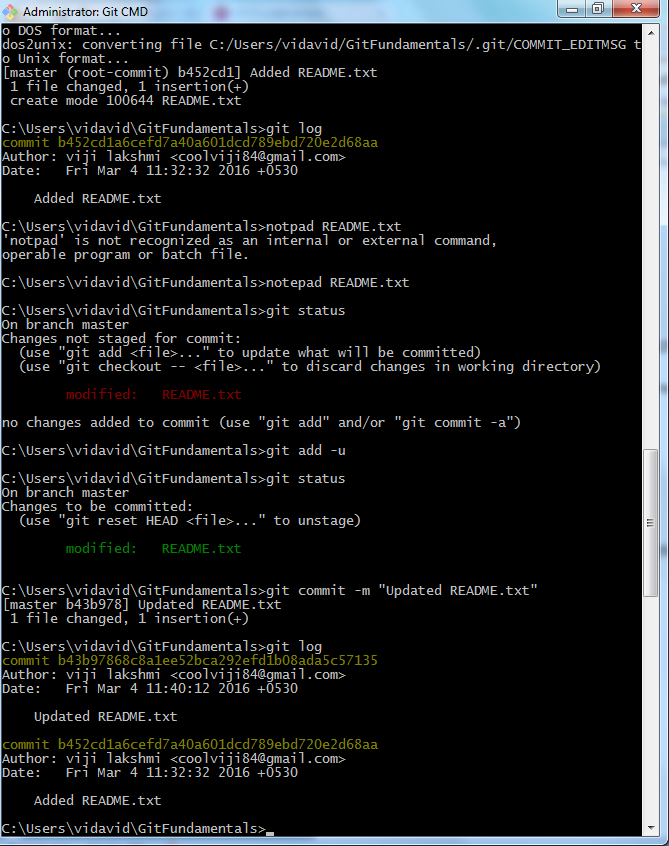
--u will get unstage details

10.git commit -m "Updated README.txt"

--commit the changes with the messages called "Updated README.txt"

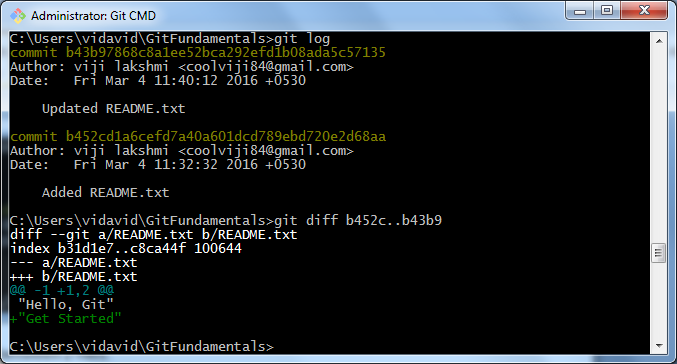
11.git log

--u could see 2 changes done in the README.txt file

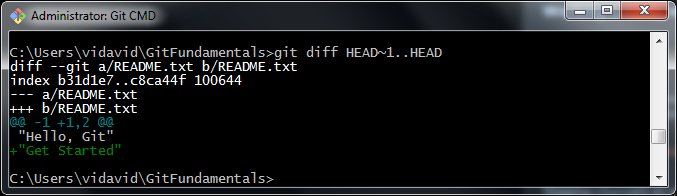


12. if you want to find the difference between 2 files.

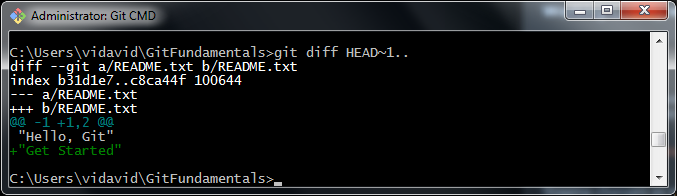
**git diff b452c..b43b9**



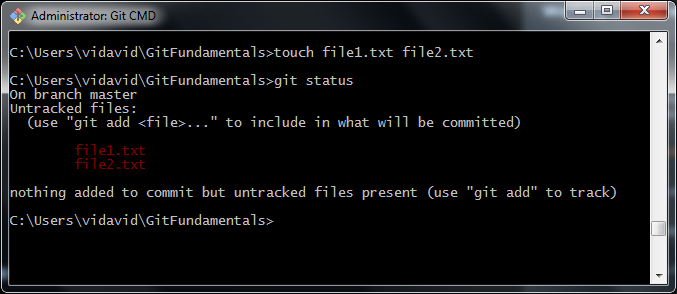
13. If you want to get the latest commit information , then give the following command



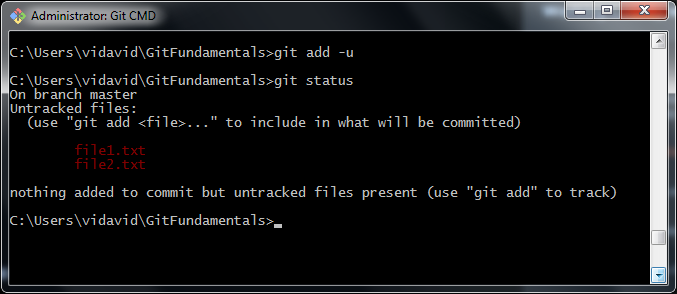
14. You can get the same result , by using the following



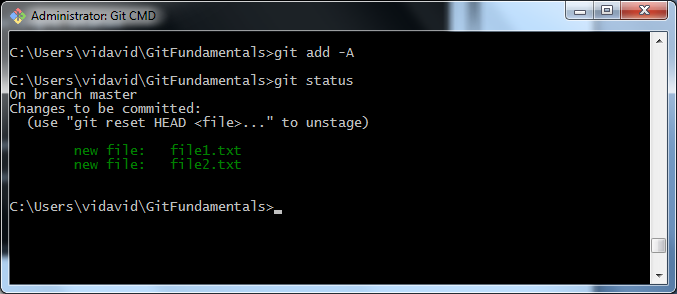
15.If you want to check other files(file1.txt and file2.txt)



Here aging if you give the command **git status**, it wont reflect anything, because it is applicable only for updated files

16.Try the below****

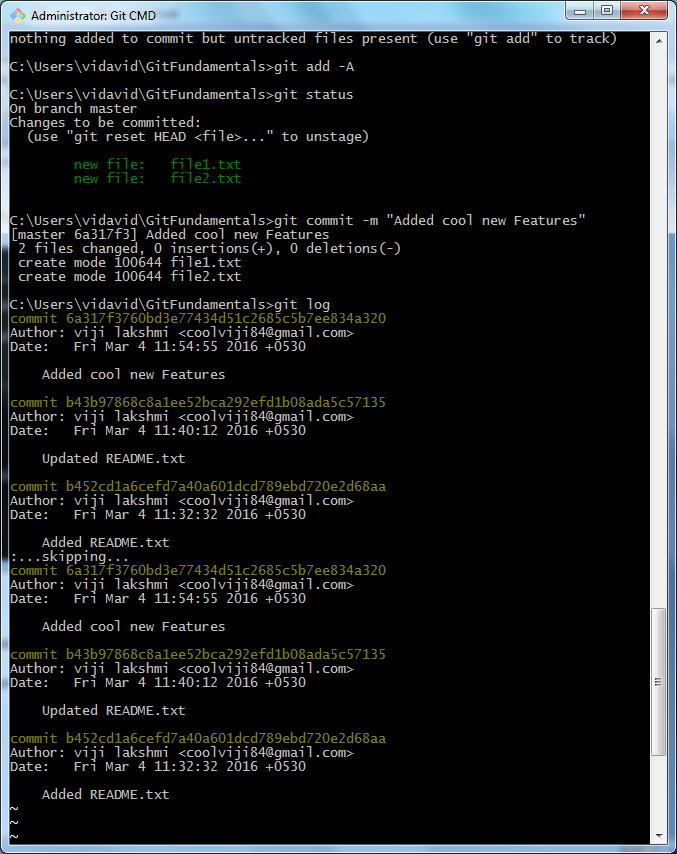
17. To add file1.txt and file2.txt , use the below command



18.Now commit it and see the log by using the below commands.

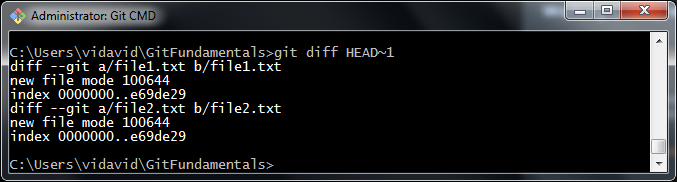
git commit -m "Added cool new Features"

git log



19. Now use the diff command

git diff HEAD~1



20. Now use **cat file1.txt** command, it does not show anything because those files are not added into repository.

