**Git Tutorial:**

18/3/2019

<https://www.git-tower.com/learn/git/ebook/en/command-line/appendix/command-line-101>

Command line 101 for Git – very goodcd

<http://johnatten.com/2012/09/08/basic-git-command-line-reference-for-windows-users/>

6/8/2018

<https://medium.freecodecamp.org/git-cheat-sheet-and-best-practices-c6ce5321f52>

<https://www.atlassian.com/git/tutorials/setting-up-a-repository> -- git bare Vs Cloned repository

7/8/2018 – in the evening at home

How to delete file using Git

1. <https://stackoverflow.com/questions/2047465/how-can-i-delete-a-file-from-git-repo/2047477>

**Warning:** git rm is the right answer, but remember that the file will still be there in history. If you want to remove a file because it had sensitive information, you'll need to do something more drastic. (Changing history, especially for content you've already pushed, is a drastic action, and should be avoided if possible.) – [Keith Thompson](https://stackoverflow.com/users/827263/keith-thompson)[May 16 '13 at 21:06](https://stackoverflow.com/questions/2047465/how-can-i-delete-a-file-from-git-repo/2047477#comment23856805_2047465)

2.<https://git-scm.com/docs/git-rm>

3.<https://www.google.com.au/search?source=hp&ei=M5dpW_WiJtP4hwOG3rzQDA&q=git+how+to+ignore+files+already+in+repository&oq=git+how+to+ignore+&gs_l=mobile-gws-wiz-hp.1.1.0l5.1191.14383..16832...9.0..0.317.5752.0j16j11j1......0....1.......0..5j46j0i131j46i131j0i10.0-4Az-hZTOc>

.gitignore when Project and it’s files has been pushed... Good

4.<https://gist.github.com/tsrivishnu/a2f3adbbca9fcad5f3597af301ad1abb>

8/8/2018

Popular git command

<https://orga.cat/posts/most-useful-git-commands>

what does git cat –file stand for

<https://stackoverflow.com/questions/38177026/what-does-cat-file-stand-for-in-git>

**Rename/move and remove files**

git rm removeme.txt tmp/crap.txt

git mv file\_oldname.txt file\_newname.txt

git commit -m "deleting 2 files, renaming 1"

show content of a file – very good

<https://stackoverflow.com/questions/5153199/git-show-content-of-file-as-it-will-look-like-after-committing/5153379>

cat myhdfile.txt

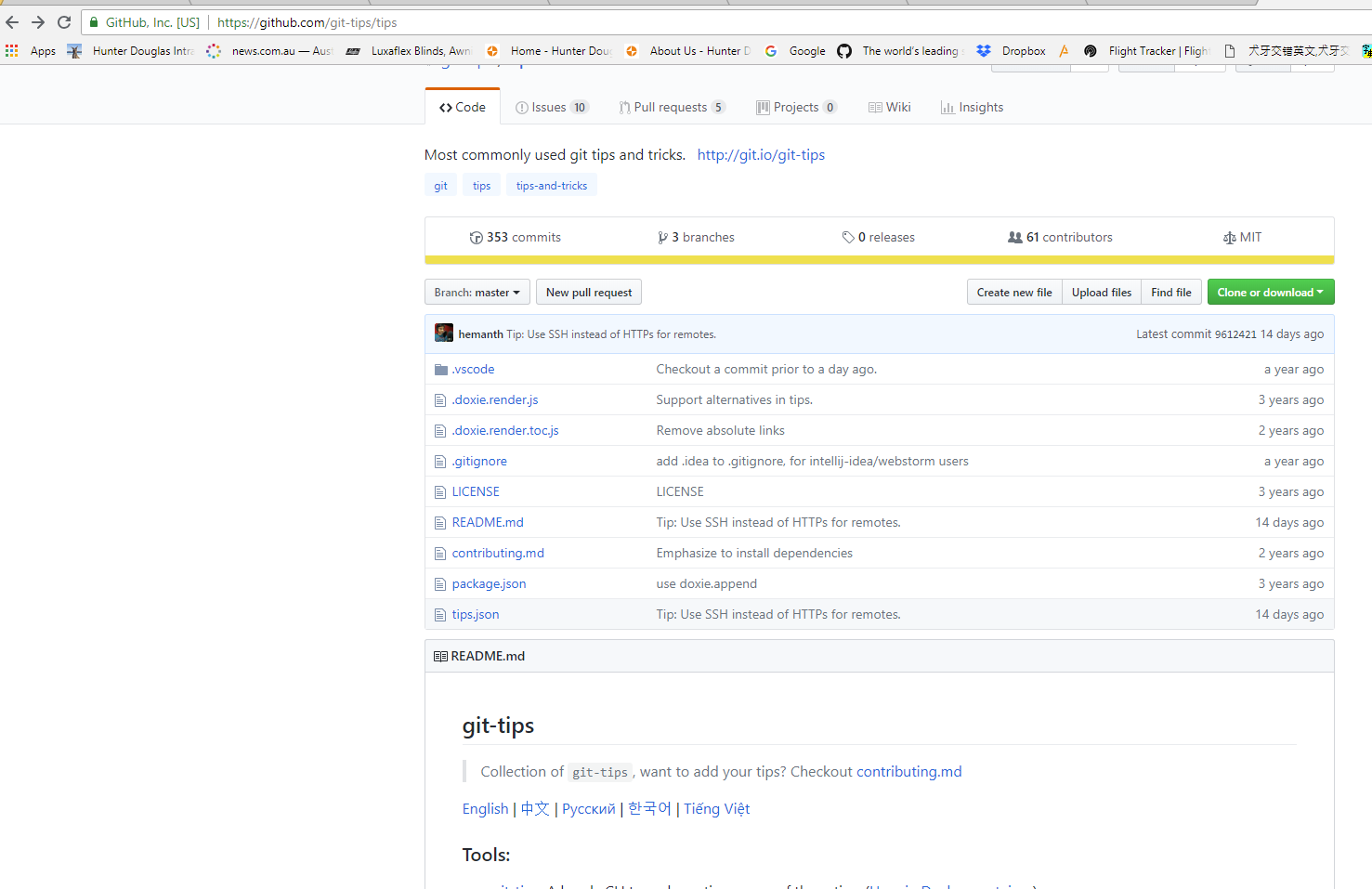
git show :myhdfile.txt

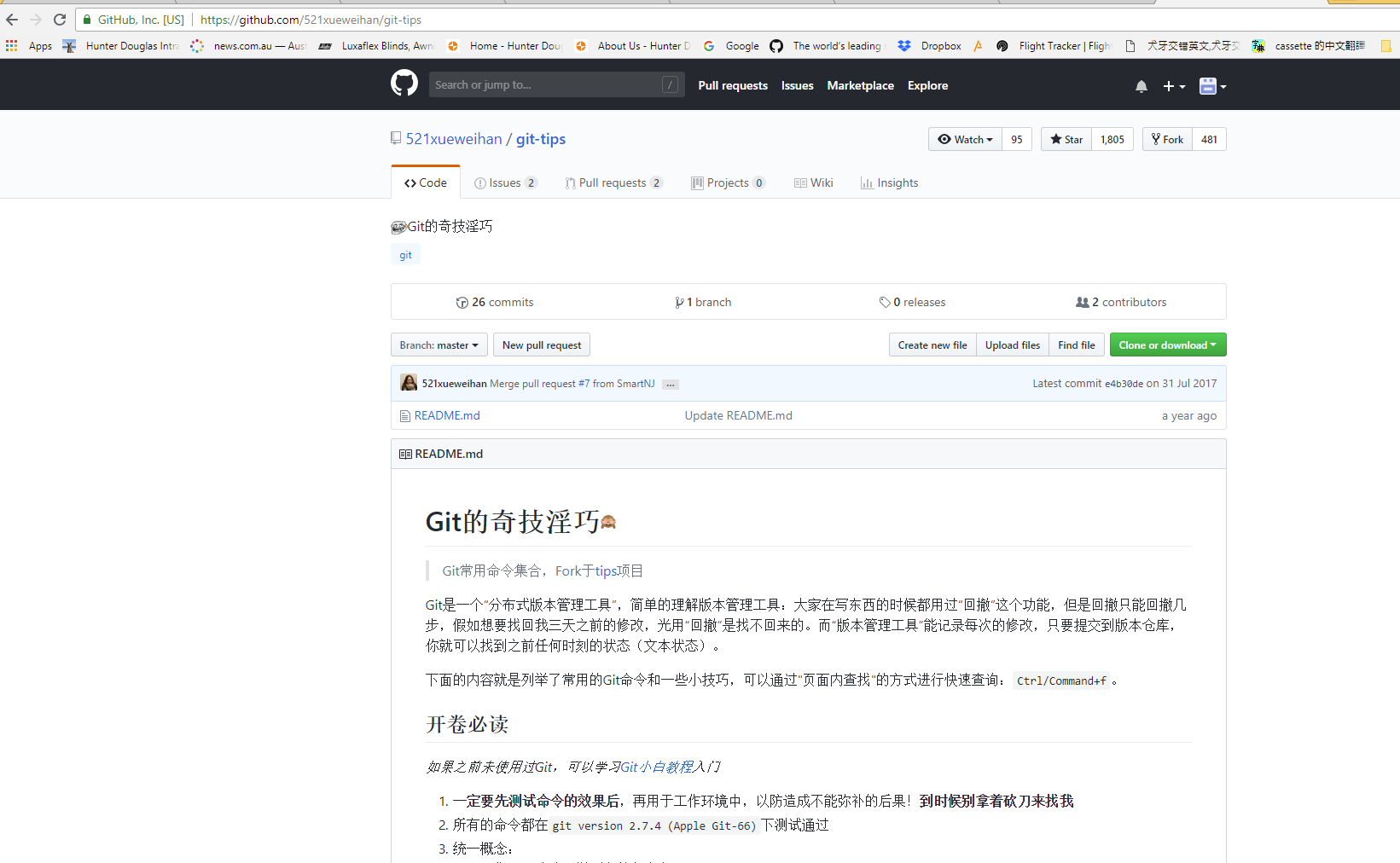
--

**Most commonly used git tips and tricks.**

<https://github.com/git-tips/tips>

can change to Chinese !





--- show history of Git Command

<https://stackoverflow.com/questions/7435452/history-or-log-of-commands-executed-in-git>

check difference in different version of files with diff - good

<https://veerasundar.com/blog/2011/06/git-tutorial-comparing-files-with-diff/>

what is origin/master , master , origin

<https://stackoverflow.com/questions/18137175/in-git-what-is-the-difference-between-origin-master-vs-origin-master?noredirect=1&lq=1>

<https://stackoverflow.com/questions/10588291/git-branching-master-vs-origin-master-vs-remotes-origin-master>

**9/8/2018**

# [**How to revert a Git repository to a previous commit**](https://stackoverflow.com/questions/4114095/how-to-revert-a-git-repository-to-a-previous-commit)

<https://stackoverflow.com/questions/4114095/how-to-revert-a-git-repository-to-a-previous-commit>

**Error message :**

# [**How to resolve git error: “Updates were rejected because the tip of your current branch is behind”**](https://stackoverflow.com/questions/22532943/how-to-resolve-git-error-updates-were-rejected-because-the-tip-of-your-current)

<https://stackoverflow.com/questions/22532943/how-to-resolve-git-error-updates-were-rejected-because-the-tip-of-your-current> -**this link has topics about stash**

**about Git Stash**

<https://www.git-tower.com/learn/git/faq/save-changes-with-git-stash>

**Error message :**

# [**Why does git say “Pull is not possible because you have unmerged files”?**](https://stackoverflow.com/questions/26376832/why-does-git-say-pull-is-not-possible-because-you-have-unmerged-files)

<https://stackoverflow.com/questions/26376832/why-does-git-say-pull-is-not-possible-because-you-have-unmerged-files>

Remember that if you need to ignore a file or a folder , you need to add it after ignore file created, any file created before this file ( gitignore )was created , it will not be ignored but still showing up.

---

**Git command:**

Pwd

Git - - version

Git status

Git add . git –A

So, For demonstration purpose, you can do following:

**Touch** 🡪 to create a text file under Git bash environment

**Echo** 🡪 to edit text file under Git bash environment ( of course you can use text edit to edit text file and comes back to Git environment to ‘commit’ the change made in text file )

**Cat** 🡪 to display the content of the text file

Rm –r <folder name> 🡪 to remove a folder in a Git environment ( folder is physically gone )

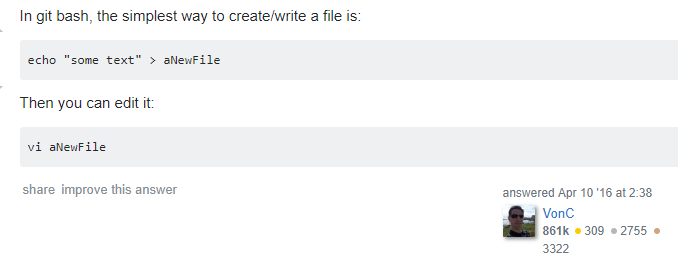
Rm <file name > 🡪 to remove a file in a Git environment ( folder is physically gone )

Mkdir <folder name> 🡪 to physically add a folder in a Git environment

Touch .gitignore -- create .gitignore file

Touch <file name> -- to create a file

echo "some text" > aNewFile -- to create file

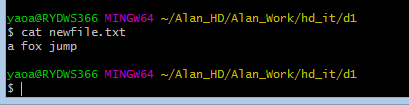


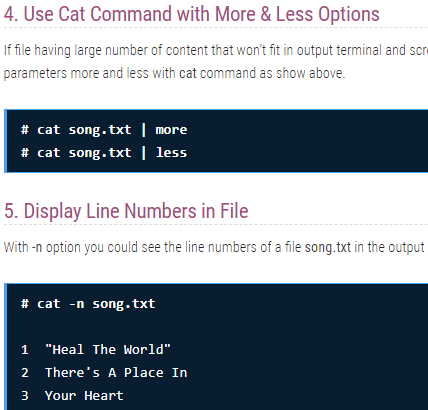
if you use vi to edit file use below command to exit text editor:





**Use cat to display text file content:**





**What is ‘Cat’ stand for ?**



Git commit –m “”

Git push

Git pull

Git init

Git clone

Git log

Git log -- stat

Git diff

Git merge

Git diff - - staged

Git rm

Git rm –rf .git - remove .git so no longer track your project

Git init - put it back, start to track your project

Git mv -- change name /move

git mv old\_filename new\_filename

git status

git commit -m "Rename file"

git push origin your-branch

Git checkout

Git branch

Git reset - while file is in staging area, not commit yet

History ( git ) / doskey /history ( cmd ) -- show command history

clear

git show : f2.txt - show content of the file

cat f2.txt

--

Ls –la

Git config –global

git reflog

Git diff – using before commit tells you or show you the difference between your Working directory And Local repository

When use .gitignore , after you add file names to .gitignore, you need to add this .gitignore file to staging area striaight away to make it take effect, otherwise it will not work ( however you might add txt file to staging area already )

Use .gitignore Before you add file to staging area or make commit.. then effectively it will include those files in ignore area to Not track its change.

Also you cannot .ignore file once it is commited to repository or push to remote

Note that if you already commit the change to the file, you can git rm the file, but if you just add . or create the file, you have to use git rm –f to force to delete the file.

Rm –r <folder name> to remove a folder in a Git environment ( folder is physically gone )

Rm <file name > to remove a file in a Git environment ( folder is physically gone )

Mkdir <folder name> to physically add a folder in a Git environment

**Basic Git Command Line Reference for Windows Users**

<https://www.codeproject.com/Articles/457305/Basic-Git-Command-Line-Reference-for-Windows-Users>

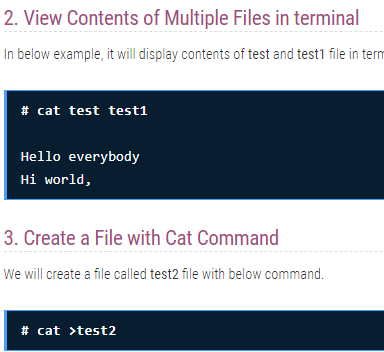
**Good**

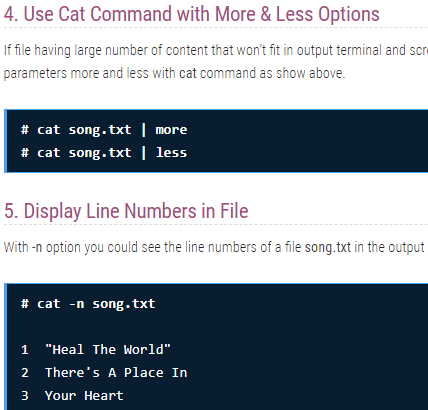


**13 Basic Cat Command Examples in Linux**

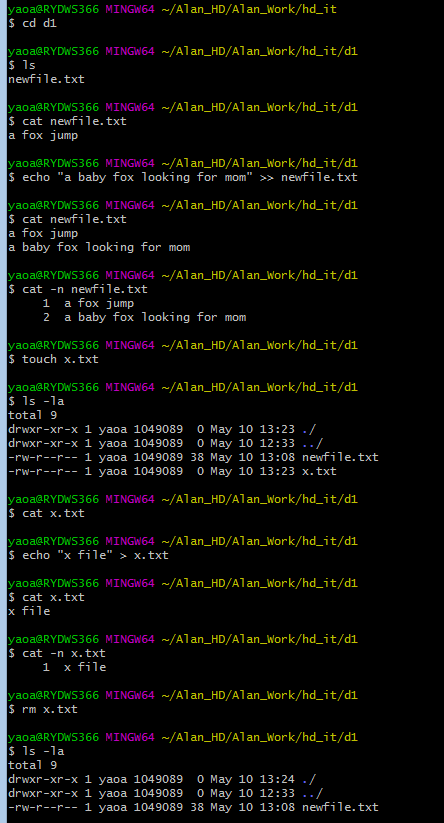
<https://www.tecmint.com/13-basic-cat-command-examples-in-linux/>

Good





**Example of add folder /delete folder/add txt file/edit txt file/delete txt file**



**--**

**9/8/2018**

.gitignore

Note: when you want to revive a file from ignore file and put it back into active condition, git will treat it as new file; so even if you make some change to this newly revived file ( F2.txt ) , you will not be able to use git diff to tell the difference between this file in working directory and file in remote.

You have to add and commit and push make it an active file ( like complete its life cycle) then if you make further changes after initialization, you are good to go – any further changes you make to F2.txt ( as re-activated ) you can use ‘git diff ‘ command.

.gitignore

## How can I ignore a file that is already committed to the repo?

Below are the steps on how to ignore this file (lets say the filename is config.py):

* Add it to .gitignore:

$ echo "config.py" >> .gitignore

* Now tell git to not track this file by removing it from the index:

$ git rm --cached config.py

* Doing git status shows you that you have deleted the file. However, if you do ls you will still see the file is still present on your local filesytem.
* Now, add .gitignore to index and commit:
* $ git add .gitignore

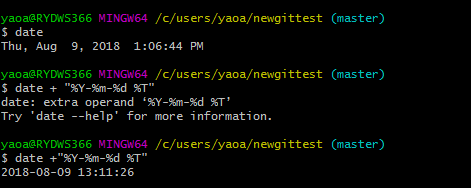
$ git commit -m "Ignore config.py"

**Note:** When other developers pull this commit, the file config.py will be deleted from their local filesystem.

--

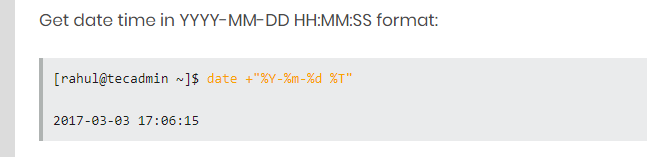
# **How to Get Current Date and Time in Bash Script – very good**

<https://tecadmin.net/get-current-date-and-time-in-bash/>

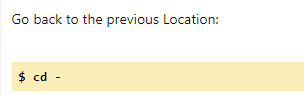


$ date +"--------------------------------%Y-%m-%d %T ------------------------------------"

<https://tecadmin.net/get-current-date-and-time-in-bash/> - in git bash



**Cmd command**



Pwd

Cd ..

Cd

Dir

Dir –all

Ls –la

Type

Echo “ ‘’ >> myfile.txt append more line

Echo “ ‘’ > myfile.txt overwrite all line

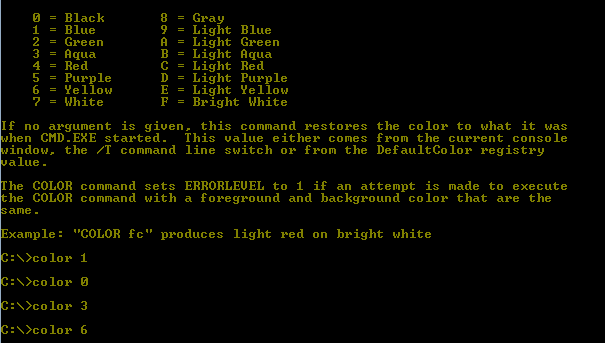
Type f2.txt -- Open & Display the file content

Or simple

C:\>F2.txt 🡪 then ‘Enter’ to Open & Display the text file

Del txt.file

Color <integer>



Cls ( to clear command history )

doskey /history ( cmd ) - show command history

<https://helpdeskgeek.com/how-to/redirect-output-from-command-line-to-text-file/> - save command

doskey /listsize=0

save command history to a file

doskey /history > C:\Users\name\Desktop\commands.txt

out put file:

ipconfig /all > c:\my\_directory\network\_info.txt

output file :

**NOTE:** If you saved the file to a different directory, be sure to put the full path to the file after “**notepad.exe**” on the command line as follows:

notepad.exe network\_info.txt

notepad.exe C:\my\_directory\network\_info.txt

**NOTE:** If you use the same filename a second time, that file will be overwritten and any output it previously contained will be lost. To preserve your previous output, use a different filename or you can append the output to the end of an existing file. To append data to an existing file, use two redirection symbols as shown below:

dir c:\my\_music >> network\_info.txt

**NOTE:** Redirection is also useful for saving directory listings. The previous command saves a listing of the **c:\my\_music** directory into the end of the **network\_info.txt** file. Enjoy!

Cat f2.txt -- cmd command line to display file content. Cat stand for concatenate

<https://stackoverflow.com/questions/38177026/what-does-cat-file-stand-for-in-git>

While [cat](https://en.wikipedia.org/wiki/Cat_(Unix)) does stand for "concatenate", what it actually does is simply display one or multiple files, in order of their appearance in the command line arguments to cat. The common pattern to *view*the contents of a file on Linux or \*nix systems is:

cat <file>

The main difference between cat and Git's [cat-file](https://git-scm.com/docs/git-cat-file) is that it only displays a single file (hence the -file part). Git's cat-file doesn't really stand for "concatenate"; it simply is a reference to the behavior of the cat command.

# [**Basic DOS Commands**](https://perishablepress.com/basic-dos-commands/) **-- good**

<https://perishablepress.com/basic-dos-commands/>

**Enter data and time in Cmd**

C:\Users\yaoa\newGITTest>

C:\Users\yaoa\newGITTest>time/t

12:45 PM

C:\Users\yaoa\newGITTest>date /t

Thu 09/08/2018

C:\Users\yaoa\newGITTest>date /t && time/t

Thu 09/08/2018

12:50 PM

C:\Users\yaoa\newGITTest>echo %date% %time%

Thu 09/08/2018 12:50:18.41

Jump out Ctrl + U then enter twice to jump out : enter new time



9/8/2018

--- get current data time in CMD

echo %DATE%

echo %TIME%

set datetimef=%date:~-4%\_%date:~3,2%\_%date:~0,2%\_\_%time:~0,2%\_%time:~3,2%\_%time:~6,2%

echo %datetimef%

which gives:

28/07/2009

8:35:31.01

2009\_07\_28\_\_ 8\_36\_01

<https://stackoverflow.com/questions/1192476/format-date-and-time-in-a-windows-batch-script>

**CMd – how to display time, date as your requirement**

<https://stackoverflow.com/questions/36147552/what-does-date-4-4date-10-2date-7-2-time0-2time3-2-mean> -- very good

# **2.**

# Format Windows command line date

<https://www.dev-notes.com/blog/2008/10/08/format-windows-command-line-date/>

Also, above link shows you - batch file to display Date / Time using CMD

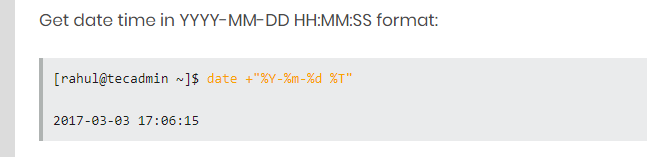
set backupcmd=c:windowssystem32xcopy.exe /c /d /e /h /r /y

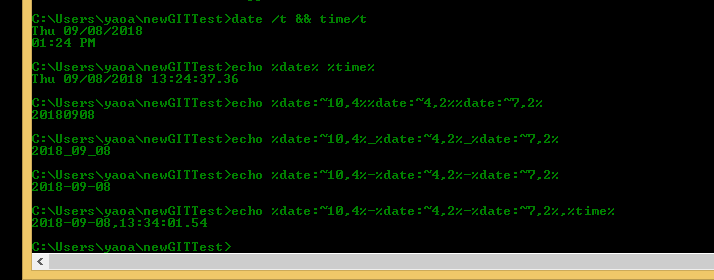
set dt=%date:~10,4%%date:~4,2%%date:~7,2%

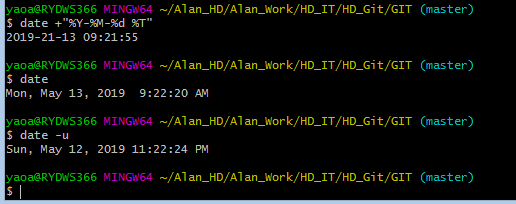
%backupcmd% C:stuff\*%dt%.txt D:backupstuff

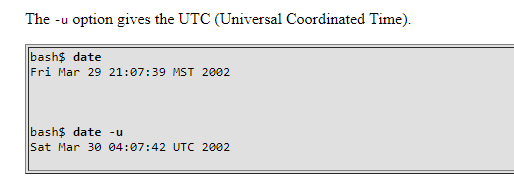
**3.**

<https://tecadmin.net/get-current-date-and-time-in-bash/> - in git bash









<https://www.tldp.org/LDP/abs/html/timedate.html>

**good**

**13/8/2018**

Remove folder in CMD

Rm -r <folder name>

Add a folder – Mkdir d1

Del a folder – del d1

Remove folder in Git

Rm –r<folder name>

**18/3/2019**

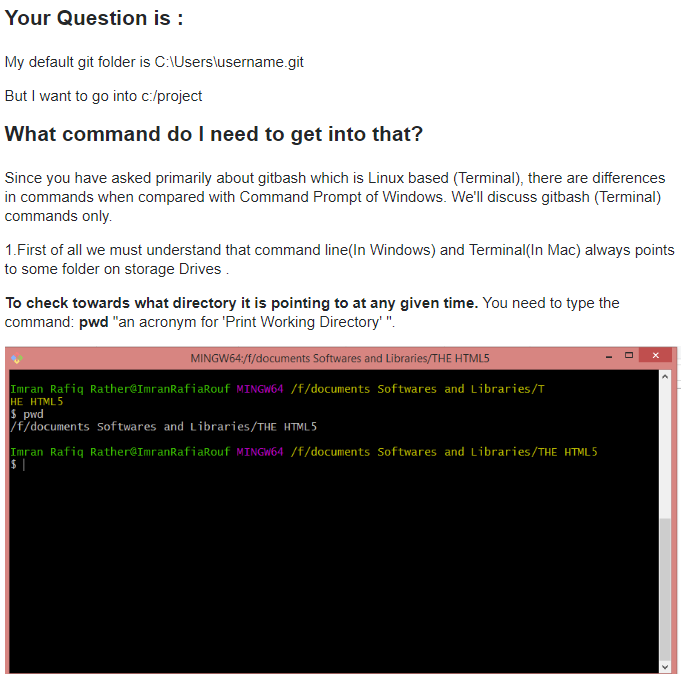
# [**How do I exit the results of 'git diff' in Git Bash on windows? [duplicate]**](https://stackoverflow.com/questions/9929457/how-do-i-exit-the-results-of-git-diff-in-git-bash-on-windows)

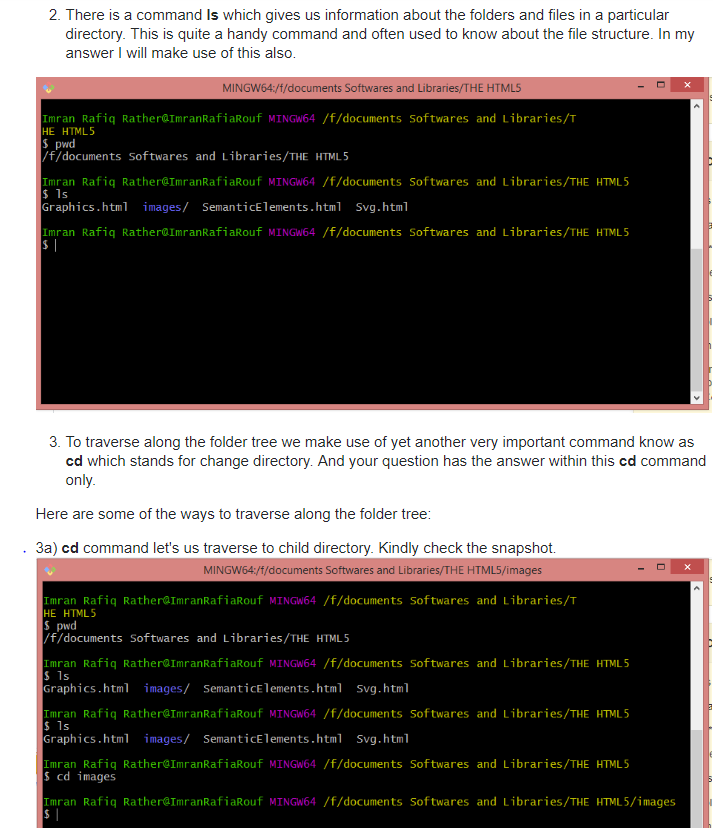
957

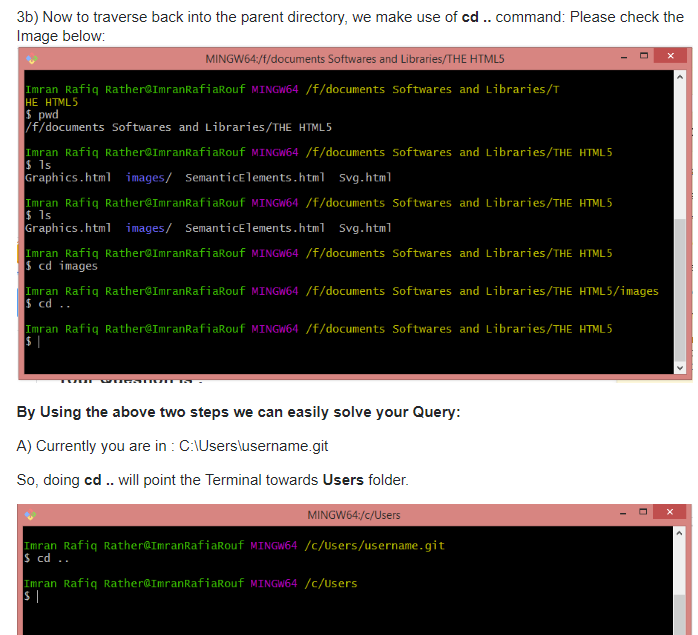
I think pressing Q should work.

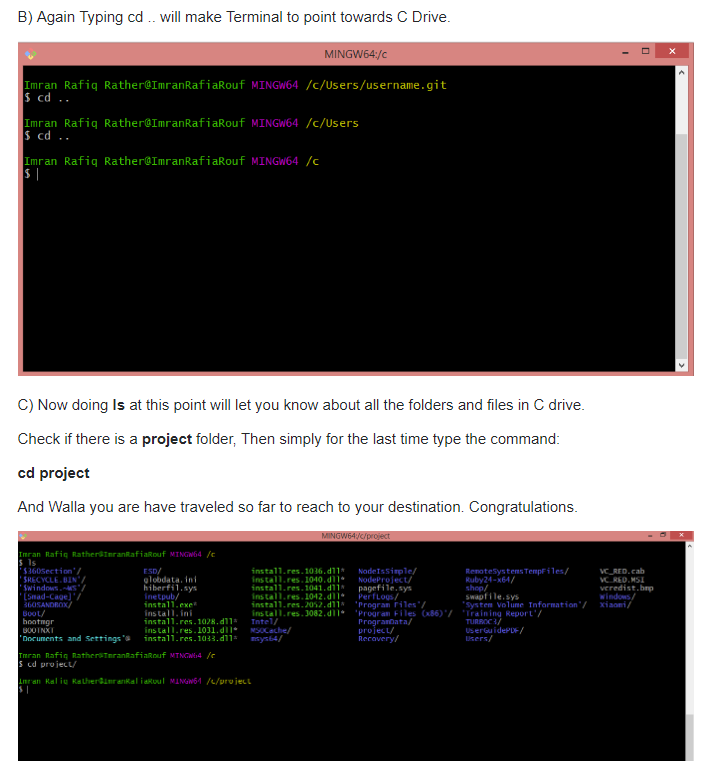
---

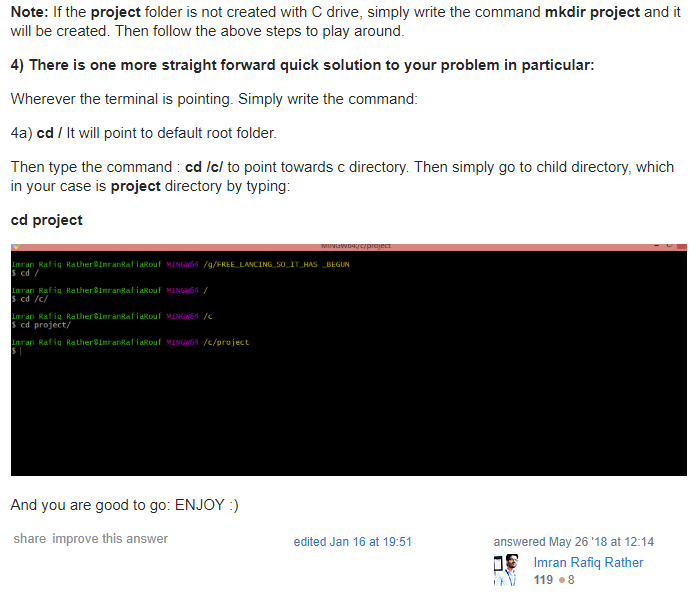
**How to traverse with different folder and file using Git command**



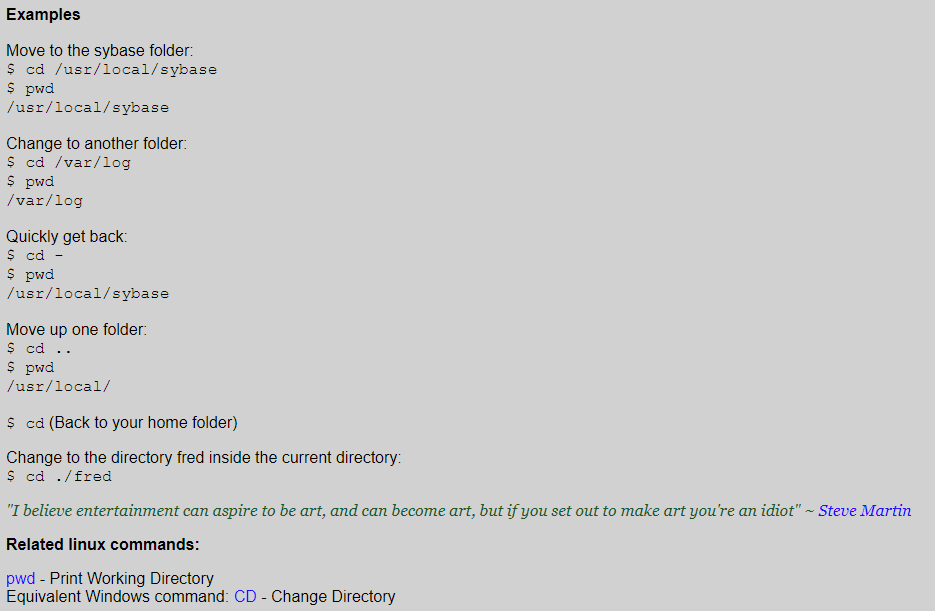








1. <https://ss64.com/bash/cd.html>



1. <https://stackoverflow.com/questions/40396613/forward-slash-vs-backward-slash-for-file-path-in-git-bash>



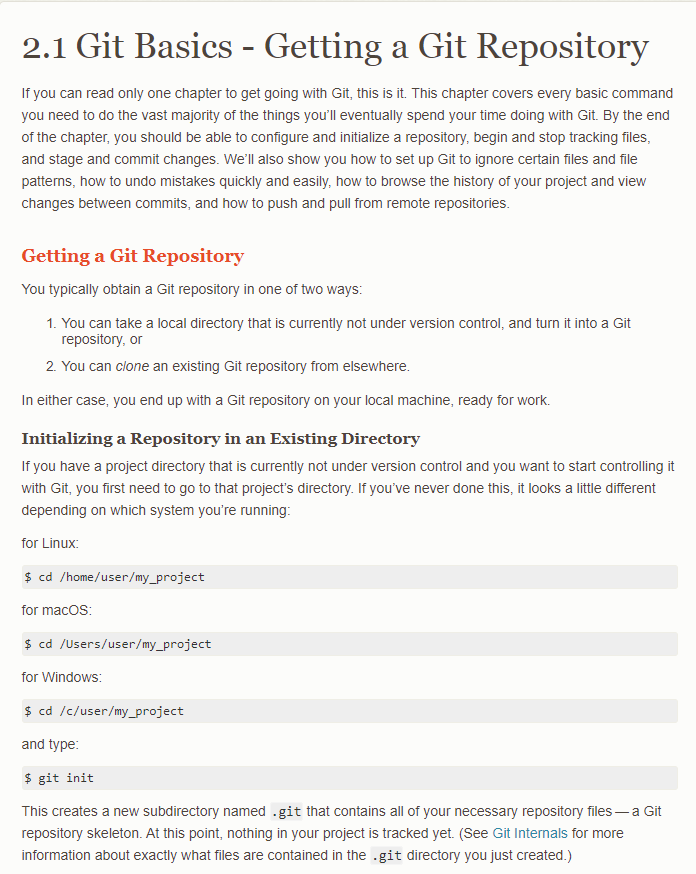
--

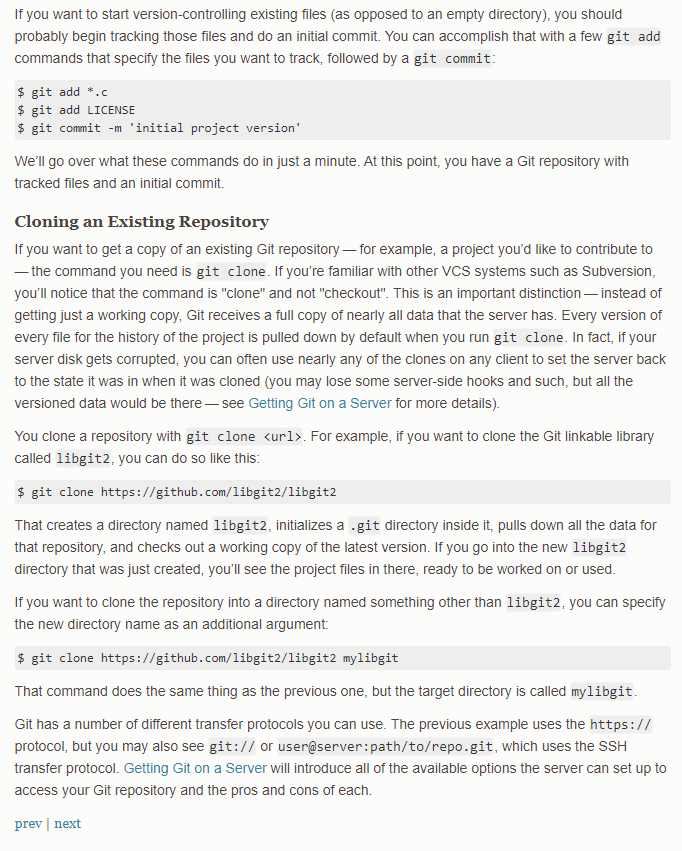
11/4/2019

**Create a repository and connect to server**

**Good**

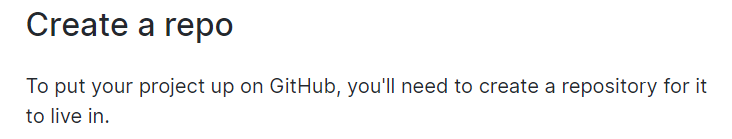
<https://git-scm.com/book/en/v2/Git-Basics-Getting-a-Git-Repository>





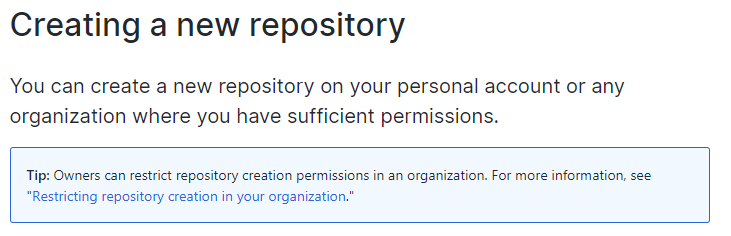
Good

<https://help.github.com/en/articles/create-a-repo>



Good

<https://help.github.com/en/articles/creating-a-new-repository>



Create a repository from Local project

<https://kbroman.org/github_tutorial/pages/init.html>

Create a repository from Local project – similar as above

<https://help.github.com/en/articles/adding-an-existing-project-to-github-using-the-command-line>

---

Create a local repository

