CIS45-Lab-2

1. Navigating the file system.

$cd /; ls -F;

$ cd /etc; ls -F

2. **Using the 'ls' command and the Pipe for interprocess communication**.
   The pipe '|' allows you to send the output from one command into another command as input to the other command. This example sends the output of the ls -F through to the 'more' command. The more command gives you one page at a time.

   $ ls -F | more

3. The -a option to the 'ls' command shows hidden files.
   $ cd /var/log; ls -a

4. The -l option to the ls shows a long listing. Note the permissions of files and directories. You can also pipe the output to more as in " ls -al |more ".
   $ ls -al

   $ ls -al |more

5. What kinds of files do you see under /usr/sbin?
   $ ls /usr/sbin

   $ ls -F /usr/sbin

6. In Linux, a hidden file is a filename that begins with a dot '.' To create a hidden file, use the following command.
   $ touch .mydatafile
   $ ls -l
   > do you see the file?
   $ ls -a
   > do you see the file?

   $ ls -al
   > who owns the file?
   > what is the size of the file?

   $ pwd; cal > .mydatafile

> We just redirected the output of the 'cal' command to the hidden .mydatafile
> How do you see what's in the .mydatafile at this point?
$ cat .mydatafile

Relative and Absolute Paths

7. What is the dot '.', and dotdot '. .' ? They allow you to move around the file system. The dot '.' always refers to the current directory. The dotdot '. .' refers to the parent directory. From your home directory you will move around with the dot and dotdot.
$ cd $HOME; pwd; ls -a

$ cd ..

$ pwd
/home

$ cd ..
$ pwd
/
8. The '-' option to the 'cd' command takes you back to where you were.
$ cd -
$ pwd

9. Relative = Relative to where you are . Absolutely Path always begins with the root '/' directory.

A. Use the absolute path to change to your home directory.
$ cd /home/<login_name>
$ pwd

Use the absolute path to change to the /var/log directory
$ cd /var/log
$ pwd

B. Use the relative path to change to the /var directory from where you are in /var/log
$ cd . .
$ pwd

C. Change directory to mr-tester home directory using the ~
$ cd ~mr-tester
$ pwd

D. Change directory back to where you were using the '-'

$ cd -

$ pwd

File Globbing & Wildcards

Using the asterisk and question mark in the filter is called file globbing. File globbing is the processing of pattern matching using wildcards. The wildcards are officially called metacharacter wildcards.

- [a-z] # Any single lower case letter from a to z.
- [0-9] # Any single digit from 0 to 9
- * # Any characters accept one that begins with a dot .
- ? # Any single character except one that begins with a dot .
- ! # Exclamation point says 'NOT' when used in brackets [!]

In your home directory, make a directory called globbers.

10. Making Directories

$ mkdir $HOME/globbers

$ ls -d $HOME/globbers

> Record the permissions of this directory.
>

In the globbers directory, run the following command to create some files.

11. **$ touch filex filey filez file55 file60 file70 file100 file200 foo fox**

12. List all files that starts with the letter 'f' follow by any two characters

$ ls f?? # Letter f follow by exactly any two characters

>

$ ls f*x # Letter f followed by any characters and ends with the letter x

> filex fox

$ ls f*[2]* # letter f followed by any characters followed by the number 2
followed by any other characters.

$ ls f*2* # f followed by any characters then the number 2 then followed by
any other characters.

$ ls f*[2-9] # f followed by any single digit from 2 to 9
> file55

$ ls f*[0-9][0-9] # f followed by any characters and ends with any two digits.

$ ls f*[0-9][0-9]* #

$ ls f[!i]* # Do a listing of all files that begins with the letter f followed by
any character that is not the letter 'i' then followed by any other
characters. The exclamation point is the first character in the
bracket.

Creating files

The touch command creates an empty file. We saw this command before. Create these two files in your globbers directory

13. **$ touch file_one file_two**

14. **$ ls -a**

Use the cat command to create a file. You will use the CTRL(contrl key) plus the letter 'd' to end the cat command.

15. **$ cat > tester_file**
    I am the first line in this file
    I am the second line in this file
    This is the last line in this file
    <Enter Control + d> to end this file.

16. **$ cat tester_file**

Use the nano editor to create another file. Call it tester_file_2. Use the CTRL+X to save the file when done.

17. **$ nano tester_file_2**
    I am first line in tester_file_2

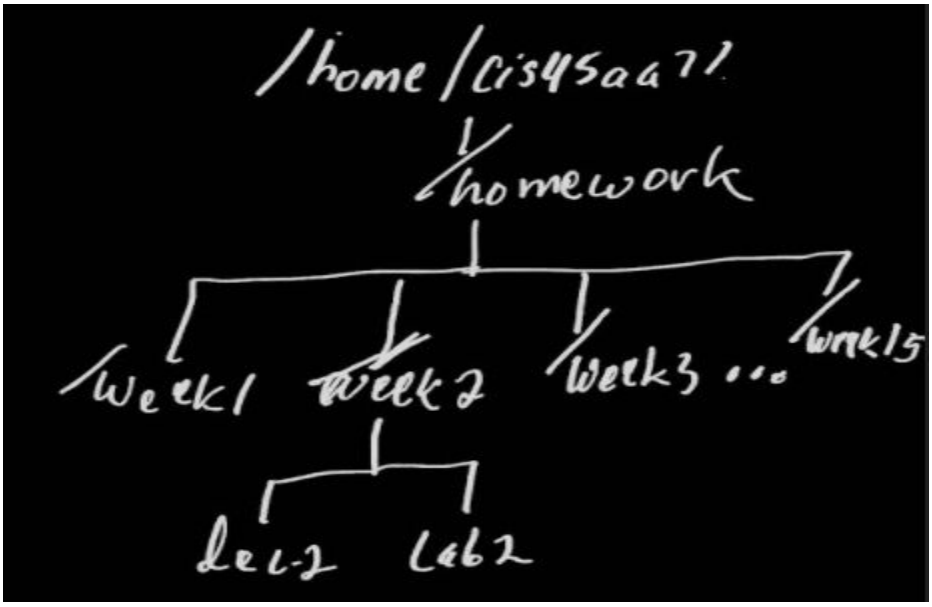
I am second line in tester_file_2

Use the vi editor or vim editor to create a file called tester-v. Use the letter **'i'** to insert characters. Use the **'esc'** key to escape so that you can delete any mistakes . You will type the letter **'x'** to delete a single character in escape mode. Use the letters **'dd'** to delete a whole line. Use the letters **'yy'** to yank/copy a line and use the letter **'p'** to put the 'yanked or copied line back. Use the capital letter **'O'** to open a line on top of the line you are currently on. Use the lowercase **'o'** to open a line below the line you are currently on. In escape mode, use **'esc' 'shift' ':'** to get to the last line in the editor, then type 'wq' to write and quit. To quit without saving the changes, type 'q!'
(i,esc,x,dd,yy,p,O,o, esc, shift, : , wq, q!)

18. $ **vi tester-v (i to insert)**
    I am tester line one
    I am tester line 2
    This is line 3
    esc shift :
    wq!

Create the following directory structure under your home directory. Note : You need to create week1 to week15

Each week will have a Lecture and a lab directory. Week1 will have a Lec-1 and Lab-1 directory. week2 will have a lec-2 and lab-2 directory. Every week will have a lecture and lab directory. Keep track of how long it takes you to do this!



Making directories

19. Using the absolute path
    $ mkdir /home/<your_home>/dirx

    $ cd $HOME/dirx

20. Using the relative path. Make sure you are in dirx.
    $pwd
    $ mkdir ../diry

21. How can you make a directory in /tmp while you are in $HOME/diry directory? Give the directory the name of your Linux login account. In my case, my login is campbell.
    $ mkdir ../../../hcampbell-tester

22. Make another directory in /tmp using the absolute path. Give it the number 2 at the end.
    $ mkdir /tmp/hcampbell-tester2

23. Create a file called data-file in the /tmp/<login>/<login>-tester directory
    $ touch /tmp/hcampbell-tester/data-file

    If the top level or parent directory does not exist, you must use the '-p' option to the mkdir command to make the directories you want to make.

24. What happens when you execute this command?
    $ mkdir /tmp/dirz/dirz

25. What happens when you execute this command? Why?
    $ mkdir -p /tmp/dirR/dirS

Removing a directory

26. Remove your /tmp/<login_name>tester directory
    $ rm /tmp/hcampbell-tester

27. Try to remove the /tmp/<login_name>-tester2 directory
    $ rm /tmp/hcampbell-tester2
        > What happened?

28. Try the rm -r option

$ rm -r /tmp/hcampbell-tester2
$ ls /tmp/hcampbell-tester2

29. To remove a file, you use the ‘rm’ command.
    $ touch Afile Bfile Cfile
    $ ls
    $ rm Afile
    $ ls
    $ rm Bfile
    $ ls
    $ rm Cfile