

HELLO WORLD

Parsa Partovi

Maktab / Linux Questions



سوال 1 : چگونه فایل هایمان را برعکس ترتیب الفبا نشان دهیم؟

```
parsa@parsa-sigware: ~  
parsa@parsa-sigware:~$ pwd  
/home/parsa  
parsa@parsa-sigware:~$ ls -r  
snap  Templates  Pictures  Downloads  Desktop  
Videos  Public      Music    Documents  
parsa@parsa-sigware:~$
```

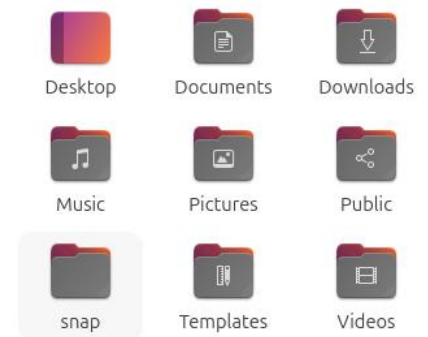
Explain :

at the files (section) we have some files included documents , ... with ls command and sorting option they get reversed .

pwd [command] → “print working dictionary” Displays the path (root /) of your current location in the filesystem .

ls [co] → “list dictionary” It shows the files or folders inside the current directory . by default it sorts them alphabetically.

-r [option] → “reverse” , -r just destroys the default order of ls and reverses that alphabetic order .



Before **ls** !

Other

options :

-t :

Sorts files by modification time (latest first).

Its for quickly seeing which files were recently changed .

-s :

Sort files by size (largest first).

finds the biggest files in a directory.

Hint :

For getting extra details even the date and mins about “lt -t “ , .. we write :

lt -lt

lt -ls

سوال 2 : چگونه لیست فایل هایمان را بر اساس جدیدترین زمان تغییر نمایش دهیم ؟

```
parsa@parsa-sigware: ~  
parsa@parsa-sigware:~$ ls -t  
snap  Pictures  Downloads  Public  Templates  
Desktop  Documents  Music      Videos  
parsa@parsa-sigware:~$ ls -lt  
total 36  
drwx----- 8 parsa parsa 4096 Nov 17 00:00 snap  
drwxr-xr-x 2 parsa parsa 4096 Nov 16 22:54 Desktop  
drwxr-xr-x 3 parsa parsa 4096 Nov 16 22:51 Pictures  
drwxr-xr-x 2 parsa parsa 4096 Nov 16 19:17 Documents  
drwxr-xr-x 2 parsa parsa 4096 Nov 16 19:14 Downloads  
drwxr-xr-x 2 parsa parsa 4096 Nov 16 19:07 Music  
drwxr-xr-x 2 parsa parsa 4096 Nov 16 19:07 Public  
drwxr-xr-x 2 parsa parsa 4096 Nov 16 19:07 Videos  
drwxr-xr-x 2 parsa parsa 4096 Nov 16 19:07 Templates  
parsa@parsa-sigware:~$
```

Explain :

Here we are showing the difference between the extra details and less details but both are displaying the same order due to the option we chose .

Some other related options :

-a	shows all files, including hidden ones (those starting with (.))	Ls -a
-A	Shows all files except current directories(.) (..)	Ls -A
-l	Long listing format (permissions, owner, size, date , "all included")	Ls -l
-R	Recursive listing (include subdirectories)	Ls -R
-u	Sort by access time (atime)	Ls -lu
-C	Sort by change time (ctime)	Ls -lc

سوال 3 : در مورد روند مخفی سازی فایل یا دایرکتوری توضیح دهید .

```
parsa@parsa-sigware: ~/Documents
parsa@parsa-sigware:~$ cd ~/Documents
parsa@parsa-sigware:~/Documents$ touch testfile
parsa@parsa-sigware:~/Documents$ mv testfile .testfile
parsa@parsa-sigware:~/Documents$ ls -a
.  ..  .testfile  script1.sh
parsa@parsa-sigware:~/Documents$ ls
script1.sh
parsa@parsa-sigware:~/Documents$
```

Explain :

any file or folder whose name starts with a dot (.) is considered hidden.

hidden files/folders do not show up in a normal "ls" listing.

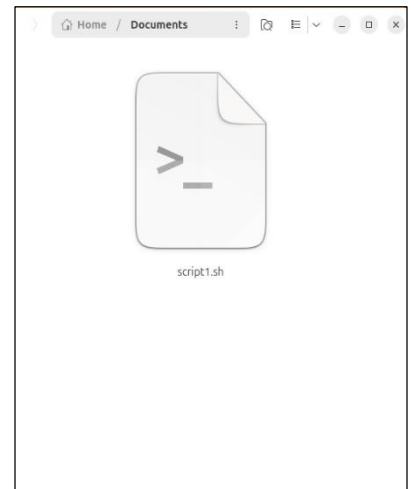
cd [co] → "change directory", it moves from your current directory to another one .

cd ~ → go to home directory

cd / → "root", system level directories under root.

touch ---- [co] → creating a new empty file , it literally means "touch the file", if the file doesn't exist, touch will create a new empty file, if it does exists touch will update it (updates changes time) .

mv ---- ---- [co] → "move", It's used to move or rename files and directories. It has a source (----) and a destination (----) .

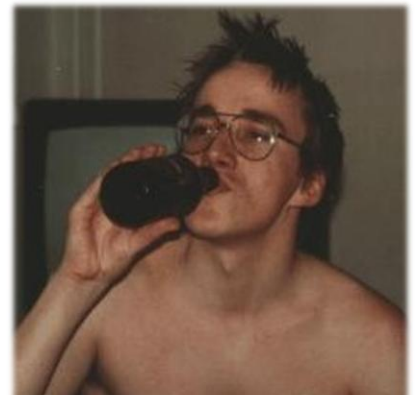


The hidden are invisible!

Hint :

We used "**ls -a**" to show all files even hidden ones

And used "**ls**" to show the files we can see , hidden are not included by that .



"talk is cheap show me the code"

سوال 4 : چگونه تعداد فایل های یک دایرکتوری را نمایش دهیم ؟

```
parsa@parsa-sigware: ~/Documents
parsa@parsa-sigware:~$ cd ~/Documents
parsa@parsa-sigware:~/Documents$ ls | wc -l
1
parsa@parsa-sigware:~/Documents$ ls -A |wc -l
2
parsa@parsa-sigware:~/Documents$
```

Explain :

by using this commands we are having different accesses to amount of files , for the first one , unhidden ones , and for the second one higher level of access(even hidden ones) .

`ls |` [*co +pipe*] → you put another command that can process the list of filenames ;

Pipe : the pipe takes the **output of one command** (here `ls`) and sends it as the **input to another command**.

`wc [co]` → “word count” , it’s used to count text statistics from input (files). It also can show:

`-l` → lines

`-w` → words

`-m` → characters

`-c` → bytes

`ls -A` → This is cleaner if you don’t want those two special entries (current directories (.) , parent directories(..) “comes after `cd`”) cluttering the output.

Hint :

Only using “`wc`” by itself will show the following options above .

سوال 5 : در مورد پایپ و اجرای ترکیبی آن با گرپ توضیح دهید .



Pipe ?

A **pipe (|)** connects the **output of one command** to the **input of another** . Instead of printing results to the screen, the first command "pipes" its output into the next command.

`command1 | command2`

grep stands for **Global Regular Expression Print**. It searches text for **patterns** (words, phrases, regex) and prints matching lines. Its also is kinda like a filter for example :

`grep "error" file1.txt` → Shows only lines containing error .

?grep



Both with
each other ?



Now here you can filter the output of any command using grep!!

Example :

`ls | grep ".txt"` → lists only files ending with .txt

by using it you will increase your efficiency ; You don't need to open files manually — just filter directly.

More flexibility! ; Works with any command that produces text output.

More precision; Lets you extract exactly what you need from large outputs.

سوال 6 : در مورد دستور سودو توضیح دهید .

SUDO

“**superuser do**” , It allows a regular user to run commands with elevated (root/administrator) **privileges**.

It lets you run commands as root safely, without permanently switching to the root account.

+ Some other options

sudo -s → start a program that takes your commands and passes them to the operating system(shell) as root.

sudo -u username command → run a command as another user.

sudo !! → re run the last command using sudo.

-Instead of logging in as the root user, you can temporarily execute a command as root using sudo. many system-level tasks (modifying system files, ...)require **root access**. sudo gives you that access **only** for the command you **run** , which is safer than staying logged in as root. ☐ You type a command with sudo. the system checks if your user is in the sudoers file (list of users allowed to use sudo). If yes, it asks for your password. the command runs with **root privileges**.

Basic use for installation(package) :

“sudo [co] “

→ **sudo apt install vim**

Edit a system file:

→**sudo nano file**

Restart a service:

→ **sudo systemctl restart apache2.**

Its finished!

Thank you

