Some Useful Linux Commands

Simple Vim workflow example

If you haven't had a chance to play around with Vim much yet, you might be wondering what a simple workflow looks like when using it. It's relatively simple:

1. Open a new or existing file with `vim filename`.
2. Type `i` to switch into insert mode so that you can start editing the file.
3. Enter or modify the text with your file.
4. Once you're done, press the escape key `Esc` to get out of insert mode and back to command mode.
5. Type `:wq` to save and exit your file.

‘echo’ command in Linux

echo is one of the most commonly and widely used built-in command for Linux bash and C shells, that typically used in scripting language and batch files to display a line of text/string on standard output or a file.

The syntax for echo is:

```
echo [option(s)] [string(s)]
```

1. Input a line of text and display on standard output

```
$ echo Tecmint is a community of Linux Nerds
```

Outputs the following text:

```
Tecmint is a community of Linux Nerds
```

2. Declare a variable and echo its value. For example, Declare a variable of **x** and assign its value=**10**.

```
$ x=10
```

echo its value:

```
$ echo The value of variable x = $x



The value of variable x = 10
```

Note: The ‘**-e**‘ option in Linux acts as interpretation of escaped characters that are backslashed.
3. Using option ‘**\b**‘ – backspace with backslash interpretor ‘**-e**‘ which removes all the spaces in between.

```
$ echo -e "Tecmint \bis \ba \bcommunity \bof \bLinux
\bNerds"



TecmintisacommunityofLinuxNerds
```

4. Using option ‘**\n**‘ – New line with backspace interpretor ‘**-e**‘ treats new line from where it is used.

```
$ echo -e "Tecmint \nis \na \ncommunity \nof \nLinux
\nNerds"



Tecmint

is
```

```
a

community

of

Linux

Nerds
```

5. Using option ‘**\t**‘ – horizontal tab with backspace interpretor ‘**-e**‘ to have horizontal tab spaces.

```
$ echo -e "Tecmint \tis \ta \tcommunity \tof \tLinux 
\tNerds"



Tecmint         is      a       community       of      Linux           Nerds
```

6. How about using option new Line ‘**\n**‘ and horizontal tab ‘**\t**‘ simultaneously.

```
$ echo -e "\n\tTecmint \n\tis \n\ta \n\tcommunity \n\tof 
\n\tLinux \n\tNerds"



        Tecmint

        is

        a

        community
```

```
      of

      Linux

      Nerds
```

7. Using option ‘**\v**‘ – vertical tab with backspace interpretor ‘**-e**‘ to have vertical tab spaces.

```
$ echo -e "\vTecmint \vis \va \vcommunity \vof \vLinux
\vNerds"



Tecmint

        is

           a

             community

                       of

                          Linux

                                Nerds

```

8. How about using option new Line ‘**\n**‘ and vertical tab ‘**\v**‘ simultaneously.

```
$ echo -e "\n\vTecmint \n\vis \n\va \n\vcommunity \n\vof
\n\vLinux \n\vNerds"
```

```
Tecmint




is




a




community




of




Linux




Nerds
```

Note: We can double the vertical tab, horizontal tab and new line spacing using the option two times or as many times as required.

9. Using option ‘**\r**‘ – carriage return with backspace interpretor ‘**-e**‘ to have specified carriage return in output.

```
$ echo -e "Tecmint \ris a community of Linux Nerds"
```

```


is a community of Linux Nerds
```

10. Using option ‘**\c**‘ – suppress trailing new line with backspace interpretor ‘**-e**‘ to continue without emitting new line.

```
$ echo -e "Tecmint is a community \cof Linux Nerds"



Tecmint is a community avi@tecmint:~$
```

11. Omit echoing trailing new line using option ‘**-n**‘.

```
$ echo -n "Tecmint is a community of Linux Nerds"

Tecmint is a community of Linux
Nerdsavi@tecmint:~/Documents$
```

12. Using option ‘**\a**‘ – alert return with backspace interpretor ‘**-e**‘ to have sound alert.

```
$ echo -e "Tecmint is a community of \aLinux Nerds"

Tecmint is a community of Linux Nerds
```

Note: Make sure to check Volume key, before firing.
13. Print all the files/folder using echo command (ls command alternative).

```
$ echo *
```

```
103.odt 103.pdf 104.odt 104.pdf 105.odt 105.pdf 106.odt
106.pdf 107.odt 107.pdf 108a.odt 108.odt 108.pdf 109.odt
109.pdf 110b.odt 110.odt 110.pdf 111.odt 111.pdf 112.odt
112.pdf 113.odt linux-headers-3.16.0-
customkernel_1_amd64.deb linux-image-3.16.0-
customkernel_1_amd64.deb network.jpeg
```

14. Print files of a specific kind. For example, let’s assume you want to print all ‘**.jpeg**‘ files, use the following command.

```
$ echo *.jpeg



network.jpeg
```

15. The echo can be used with redirect operator to output to a file and not standard output.

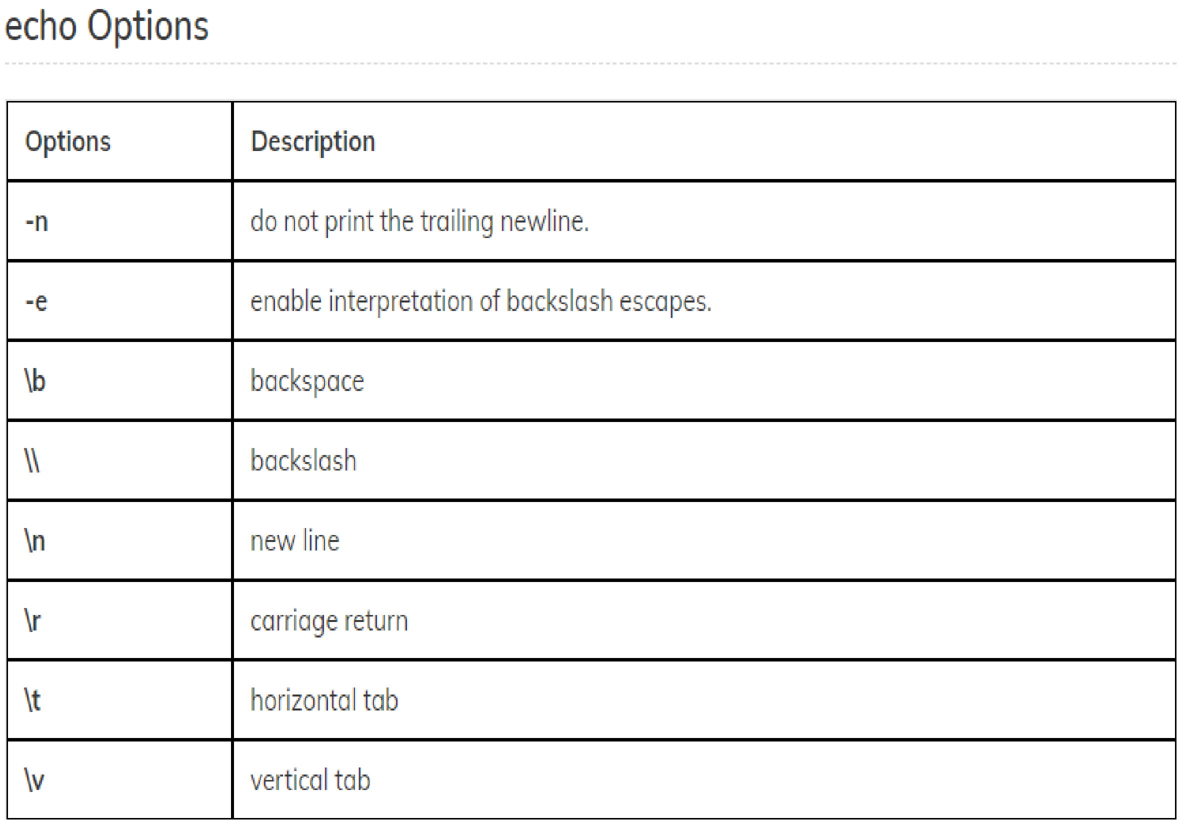
```
$ echo "Test Page" > testpage



## Check Content

avi@tecmint:~$ cat testpage
```

```
Test Page
```



echo Options

Options	Description
-n	do not print the trailing newline.
-e	enable interpretation of backslash escapes.
\b	backspace
\\	backslash
\n	new line
\r	carriage return
\t	horizontal tab
\v	vertical tab

mkdir Command in Linux

How to Make a New Directory In Linux

To create a directory using the terminal, pass the desired name to the ***mkdir*** command.

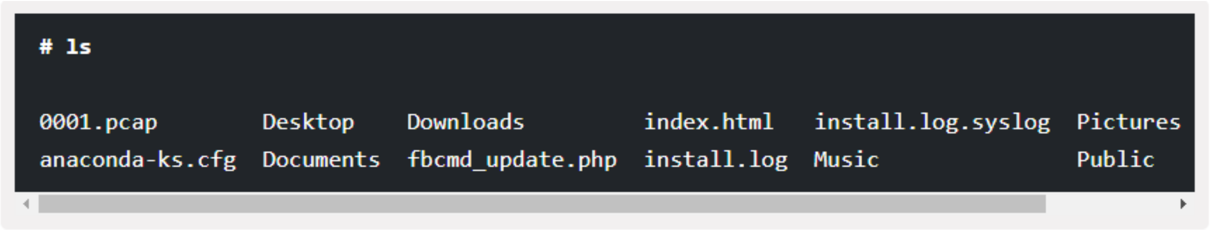
In this example, we created a directory *Linux* on the desktop. Remember commands in Linux and options are **case sensitive**.

```
mkdir Linux
```

If the operation is successful, the terminal returns an empty line. To verify, use `ls`.

List Files using ls

ls with no option list files and directories in bare format where we won’t be able to view details like file types, size, modified date and time, permission and links etc.



```
# ls

0001.pcap        Desktop    Downloads         index.html   install.log.syslog  Pictures
anaconda-ks.cfg  Documents  fbcmd_update.php  install.log  Music               Public
```

Cat Command Examples

The cat (short for “**concatenate**“) command is one of the most frequently used command in Linux/Unix like operating systems. **cat** command allows us to create single or multiple files, view contain of file, concatenate files and redirect output in terminal or files. In this article, we are going to find out handy use of **cat** commands with their examples in Linux.

General Syntax

```
cat [OPTION] [FILE]...
```

1. Display Contents of File

In the below example, it will show contents of **/etc/passwd** file.

```
# cat /etc/passwd

root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
narad:x:500:500::/home/narad:/bin/bash
```

2. View Contents of Multiple Files in terminal

In below example, it will display contents of **test** and **test1** file in terminal.

```
# cat test test1

Hello everybody
Hi world,
```

3. Create a File with Cat Command

We will create a file called **test2** file with below command.

```
# cat >test2
```

Awaits input from user, type desired text and press **CTRL+D** (hold down **Ctrl Key** and type ‘**d**‘) to exit. The text will be written in **test2** file. You can see content of file with following **cat** command.

```
# cat test2

hello everyone, how do you do?
```

mv command in Linux

To move a file into a directory using the `mv` command pass the name of the file and then the directory. In the following example the file `foo.txt` is moved into the directory `bar`.

```
tree -F .
.
├── bar/
└── foo.txt
mv foo.txt bar
tree -F
.
└── bar/
    └── foo.txt
```

How to move multiple files into a directory

To move multiple files using the `mv` command pass the names of the files or a pattern followed by the destination.

```
mv file1.txt folder
```

If the operation is successful, the terminal returns an empty line.
To verify, use `ls`.

Rm Command in Linux

rm is a command-line utility for removing files and directories. It is one of the essential commands that every Linux user should be familiar with.

In this guide, we will explain how to use the rm command through examples and explanations of the most common rm options.

How to Use the rm Command

To delete a single file, use the `rm` command followed by the file name as an argument:

```
rm filename
```

Removing Directories with rmdir

rmdir is a command-line utility for deleting empty directories. It is useful when you want to delete a directory only if it is empty, without needing to check whether the directory is empty or not.

To delete a directory with rmdir, type the command followed by the name of the directory you want to remove. For example, to delete a directory named dir1 you would type:

```
rmdir dir1
```

If the directory is not empty, you will get the following error:

```
rmdir: failed to remove 'dir1': No such file or directo
ry
```

In this case, you will need to use the rm command or manually remove the directory contents before you can delete it.