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 |