In Linux and Unix Systems Grep, short for “global regular expression print”, is **a command used in searching and matching text files contained in the regular expressions**.

**Grep Command in Linux**

Grep command can be used to find or search a regular expression or a string in a text file. To demonstrate this, let’s create a text file ***welcome.txt*** and add some content as shown.

Welcome to Linux !

Linux is a free and opensource Operating system that is mostly used by

developers and in production servers for hosting crucial components such as web

and database servers. Linux has also made a name for itself in PCs.

Beginners looking to experiment with Linux can get started with friendlier linux

distributions such as Ubuntu, Mint, Fedora and Elementary OS.

Great! Now we are ready to perform a few grep commands and manipulate the output to get the desired results. To search for a string in a file, run the command below **Syntax**

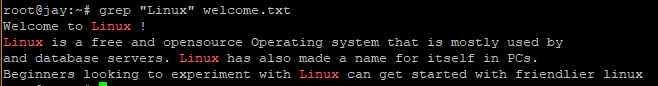
$ grep "string" file name

OR

$ filename grep "string"

**Example**:

$ grep "Linux" welcome.txt

**Output** As you can see, grep has not only searched and matched the string “Linux” but has also printed the lines in which the string appears. If the file is located in a different file path, be sure to specify the file path as shown below

$ grep "string" /path/to/file

## Colorizing Grep results using the --color option

If you are working on a system that doesn’t display the search string or pattern in a different color from the rest of the text, use the --color to make your results stand out. Example

$ grep --color "free and opensource" welcome.txt

## Searching for a string recursively in all directories

If you wish to search for a string in your current directory and all other subdirectories, search using the - r flag as shown

$ grep -r "string-name" \*

For example

$ grep -r "linux" \*

## Ignoring case sensitivity

In the above example, our search results gave us what we wanted because the string “Linux” was specified in Uppercase and also exists in the file in Uppercase. Now let’s try and search for the string in lowercase.

$ grep "linux" file name

Nothing from the output, right? This is because grepping could not find and match the string “linux” since the first letter is Lowercase. To ignore case sensitivity, use the -i flag and execute the command below

$ grep -i "linux" welcome.txt

## Count the lines where strings are matched with -c option

To count the total number of lines where the string pattern appears or resides, execute the command below

$ grep -c "Linux" welcome.txt

## Number the lines that contain the search pattern with -n option

To number the lines where the string pattern is matched , use the -n option as shown

$ grep -n "Linux" welcome.txt

## Search for exact matching word using the -w option

Passing then -w flag will search for the line containing the exact matching word as shown

$ grep -w "opensource" welcome.txt

**Output** grep command usage examplesHowever, if you try

$ grep -w "open" welcome.txt

## Using pipes with grep

The grep command can be used together with pipes for getting distinct output. For example, If you want to know if a certain package is installed in Ubuntu system execute

$ dpkg -L | grep "package-name"

For example, to find out if OpenSSH has been installed in your system pipe the dpkg -l command to grep as shown

$ dpkg -L | grep -i "openssh"

## Displaying number of lines before or after a search pattern Using pipes

You can use the **-A** or **-B** to dislay number of lines that either precede or come after the search string. The **-A** flag denotes the lines that come after the search string and **-B** prints the output that appears before the search string. For example

$ ifconfig | grep -A 4 ens3

## Using grep with regual expressions (REGEX)

The term REGEX is an acronym for **REG**ular **EX**pression. A REGEX is a sequence of characters that is used to match a pattern. Below are a few examples:

^ Matches characters at the beginning of a line

$ Matches characters at the end of a line

"." Matches any character

[a-z] Matches any characters between A and Z

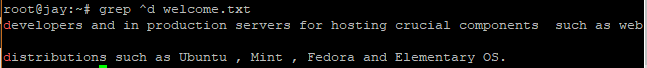
[^ ..] Matches anything apart from what is contained in the brackets

**Example** To print lines beginning with a certain character, the syntax is;

grep ^character file\_name

For instance, to display the lines that begin with the letter “d” in our welcome.txt file, we would execute

$ grep ^d welcome.txt

**Output** To display lines that end with the letter ‘x’ run

$ grep x$ welcome.txt

**Output** grep regex end of line

## Getting help with more Grep options

If you need to learn more on Grep command usage, run the command below to get a sneak preview of other flags or options that you may use together with the command.

$ grep --help