## ****Syntax of grep Command in Unix/Linux****

The basic syntax of the `**grep`** command is as follows:

**grep [options] pattern [files]**

* [**options**]: These are command-line flags that modify the behavior of grep.
* [**pattern**]: This is the regular expression you want to search for.
* [**file**]: This is the name of the file(s) you want to search within. You can specify multiple files for simultaneous searching.

## Options Available in grep Command

| **Options** | **Description** |
| --- | --- |
| **-c** | This prints only a count of the lines that match a pattern |
| **-h** | Display the matched lines, but do not display the filenames. |
| **-i** | Ignores, case for matching |
| **-l** | Displays list of a filenames only. |
| **-n** | Display the matched lines and their line numbers. |
| **-v** | This prints out all the lines that do not matches the pattern |
| **-e exp** | Specifies expression with this option. Can use multiple times. |
| **-f file** | Takes patterns from file, one per line. |
| **-E** | Treats pattern as an extended regular expression (ERE) |
| **-w** | Match whole word |
| **-o** | Print only the matched parts of a matching line, with each such part on a separate output line. |
| **-A n** | Prints searched line and nlines after the result. |
| **-B n** | Prints searched line and n line before the result. |
| **-C n** | Prints searched line and n lines after before the result. |

### ****Sample Commands****

Consider the below file as an input.

cat > hellofile.txt

*unix is great os. unix was developed in Bell labs.*

*learn operating system.*

*Unix linux which one you choose.*

*uNix is easy to learn.unix is a multiuser os.Learn unix .unix is a powerful.*

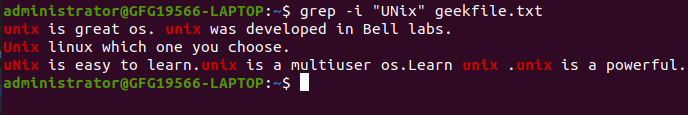
## Pratical Example of grep Command in Linux

## ****1. Case insensitive search****

The -i option enables to search for a string case insensitively in the given file. It matches the words like "UNIX", "Unix", "unix".

grep -i "UNix" hellofile.txt

**Output:**

Case insensitive search

## ****2. Displaying the Count of Number of Matches Using grep****

We can find the number of lines that matches the given string/pattern

grep -c "unix" hellofile.txt

**Output:**

IMG_257Displaying the count number of the matches

## ****3. Display the File Names that Matches the Pattern Using grep****

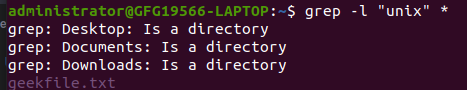
We can just display the files that contains the given string/pattern.

grep -l "unix" \*

**or**

grep -l "unix" f1.txt f2.txt f3.xt f4.txt

**Output:**

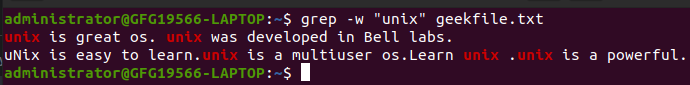
The file name that matches the pattern

## ****4. Checking for the Whole Words in a File Using grep****

By default, grep matches the given string/pattern even if it is found as a substring in a file. The -w option to grep makes it match only the whole words.

grep -w "unix" hellofile.txt

**Output:**

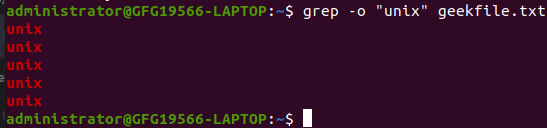
checking whole words in a file

## ****5. Displaying only the matched pattern Using grep****

By default, grep displays the entire line which has the matched string. We can make the grep to display only the matched string by using the -o option.

grep -o "unix" hellofile.txt

**Output:**

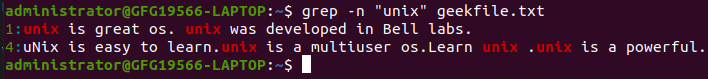
Displaying only the matched pattern

## ****6. Show Line Number While Displaying the Output Using grep -n****

To show the line number of file with the line matched.

grep -n "unix" hellofile.txt

**Output:**

Show line number while displaying the output

## ****7. Inverting the Pattern Match Using grep****

You can display the lines that are not matched with the specified search string pattern using the -v option.

grep -v "unix" hellofile.txt

**Output:**

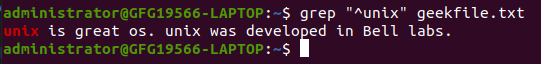
IMG_262Inverting the pattern match

## ****8. Matching the Lines that Start with a String Using grep****

The ^ regular expression pattern specifies the start of a line. This can be used in grep to match the lines which start with the given string or pattern.

grep "^unix" hellofile.txt

**Output:**

Matching the lines that start with a string

## ****9. Matching the Lines that End with a String Using grep****

The $ regular expression pattern specifies the end of a line. This can be used in grep to match the lines which end with the given string or pattern.

grep "os$" hellofile.txt

## ****10.Specifies expression with -e option****

Can use multiple times :

grep –e "Agarwal" –e "Aggarwal" –e "Agrawal" hellofile.txt

## ****11. -f file option Takes patterns from file, one per line****

cat pattern.txt

*Agarwal  
Aggarwal  
Agrawal*

grep –f pattern.txt hellofile.txt

## ****12. Print n Specific Lines from a File Using grep****

-A prints the searched line and n lines after the result, -B prints the searched line and n lines before the result, and -C prints the searched line and n lines after and before the result.

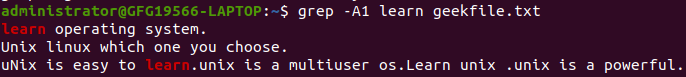
**Syntax:**

grep -A[NumberOfLines(n)] [search] [file]   
grep -B[NumberOfLines(n)] [search] [file]   
grep -C[NumberOfLines(n)] [search] [file]

**Example:**

grep -A1 learn hellofile.txt

**Output:**

Print n specific lines from a file

## ****13. Search Recursively for a Pattern in the**** D****irectory****

**-R**prints the searched pattern in the given directory recursively in all the files.

**Syntax:**

grep -R [Search] [directory]

**Example :**

grep -iR hellos /home/hellos

**Output:**

./hellos2.txt:Well Hello hellos  
./hellos1.txt:I am a big time hello  
----------------------------------  
-i to search for a string case insensitively  
-R to recursively check all the files in the directory.