Week03

grepcommand

The grepfiltersearches afile for a particular pattern ofcharacters,anddisplays all lines that contain that pattern. The pattern that is searched in the file is referred to asthe regular expression (grep stands for global search for regular expression and print out).

Syntax:grep[options]pattern[files]

OptionsDescription

-c:Thisprintsonlyacountofthelinesthatmatchapattern

-h: Displaythematchedlines,butdonotdisplaythefilenames.

-i: Ignores,caseformatching

-l: Displayslistofafilenamesonly.

-n: Displaythematchedlinesandtheirlinenumbers.

-v: Thisprintsoutallthelinesthatdonotmatchesthepattern

-eexp: Specifiesexpressionwiththisoption.Canusemultipletimes.

-ffile: Takespatternsfromfile,oneperline.

-E: Treatspatternasanextendedregularexpression(ERE)

-w: Matchwholeword

-o : Print only the matched parts of a matching line, with each such part on a separate output line.

-An: Printssearchedlineandnlinesaftertheresult.

-Bn: Printssearchedlineandnlinebeforetheresult.

-Cn: Printssearchedlineandnlinesafterbeforetheresult.

awkcommand

AWK is suitable for pattern search and processing. The script runsto search one or more files to identify matching patterns and if the said patterns perform specific tasks.

WhatOperationscanAWKdo?

- Scanningfileslinebyline
- Splittingeachinputlineintofields
- Comparinginputlinesandfieldstopatterns
- Performingspecifiedactionsonmatchinglines

AWKCommandUsefulness

- Changingdatafiles
- Producingformattedreports

ProgrammingConceptsforawkcommand

- Formatoutputlines
- Conditionalandloops
- Arithmeticandstringoperations

AwkCommandExamples

```
root@nginx:~# file.txt
Item    Model   Country         Cost

1       BMW     Germany         $25000
2       Volvo   Sweden          $15000
3       Subaru  Japan           $2500
4       Ferrari Italy           $2000000
5       SAAB    USA             $3000
```

Printingspecific columns

Toprintthe2ndand3rdcolumns,executethecommandbelow.

```
$awk'{print$2"\t"$3}'file.txt
```

Output

```
root@nginx:~# awk '{print $2 "\t" $3}' file.txt
Model    Country

BMW      Germany
Volvo    Sweden
Subaru   Japan
Ferrari  Italy
SAAB     USA
```

Printingalllinesinafile

Ifyou wish tolistallthelinesand columns in afile,execute

```
$awk'{print$0}'file.txt
```

Output

```
root@nginx:~# awk ' {print $0}' file.txt
Item     Model    Country          Cost

1        BMW      Germany          $25000
2        Volvo    Sweden           $15000
3        Subaru   Japan            $2500
4        Ferrari  Italy            $2000000
5        SAAB     USA              $3000
```

Printingalllinesthatmatchaspecific pattern

Ifyouwanttoprintlinesthatmatchacertainpattern,thesyntaxisas shown

```
$awk'/variable_to_be_matched/{print$0}'file.txt
```

Forinstance,tomatchallentrieswiththeletter‘o’,thesyntaxwillbe

```
$awk'/o/{print$0}'file.txt
```

Output:

```
root@nginx:~#  awk '/o/ {print $0}' file.txt
Item     Model    Country          Cost
2        Volvo    Sweden           $15000
```

Tomatchall entrieswiththeletter‘e’

```
$awk'/e/{print$0}'file.txt
```

Output

```
root@nginx:~# awk '/e/ {print $0}' file.txt
Item    Model   Country         Cost
1       BMW     Germany         $25000
2       Volvo   Sweden          $15000
4       Ferrari Italy           $2000000
```

Printingcolumnsthatmatchaspecificpattern

When AWK locates a pattern match, the command will execute the whole record. You can change the default by issuing an instruction to display only certain fields. For example:

```
$awk'/a/{print$3"\t"$4}'file.txt
```

Theabovecommandprintsthe3rdand4thcolumnswheretheletter‘a’appearsin either of the columns

Output

```
root@nginx:~# awk '/a/ {print $3 "\t" $4}' file.txt
Germany $25000
Japan   $2500
Italy   $2000000
```

CountingandPrintingMatchedPattern

You can use AWK to count and print the number of lines for every pattern match. For example, the command below counts the number of instances a matching pattern appears

```
$awk'/a/{++cnt}END{print"Count=",cnt}' file.txt
```

Output

```
root@nginx:~#  awk '/a/{++cnt} END {print "Count = ", cnt}' file.txt
Count =  3
```

PrintLineswithMoreorlessthana No.ofCharacters

AWK has a built-in length function that returns the length of the string. From the command $0 variable stores the entire line and in the absence of a body block, the default action is taken, i.e., the print action. Therefore, in our text file, if a line has more than 18 characters, then the comparison results true, and the line is printed as shown below.

```
$awk'length($0)>20'file.txt
```

Output

```
root@nginx:~# awk 'length($0) > 20' file.txt
Item    Model   Country         Cost
1       BMW     Germany         $25000
2       Volvo   Sweden          $15000
3       Subaru  Japan           $2500
4       Ferrari Italy           $2000000
```

SavingoutputofAWKtoadifferentfile

Ifyouwishtosavetheoutputofyourresults,usethe>redirectionoperator.For example

```
$awk'/a/{print$3"\t"$4}'file.txt>Output.txt
```

Youcanverifytheresultsusingthecatcommandasshown below

```
$cat output.txt
```

Output

```
root@nginx:~# cat output.txt
Germany $25000
Japan   $2500
Italy   $2000000
```

SED command in UNIX stands for stream editor and it can perform lots of functions on file like searching, find and replace, insertion or deletion. Though most common use of SED command in UNIX isforsubstitution or for findandreplace. By usingSED you can edit files even without opening them, which is much quicker way to find and replace something in file, than first opening that file in VI Editor and then changing it.

- SEDisapowerfultextstreameditor.Candoinsertion,deletion,searchand replace(substitution).
- SEDcommandinunixsupportsregularexpressionwhichallowsitperformcomplex pattern matching.

Syntax:
sedOPTIONS...[SCRIPT][INPUTFILE...]

Example:
Considerthebelowtextfileasaninput.

$cat>geekfile.txt
unix is great os. unix is opensource. unix is free os.

learn operating system.

unixlinuxwhichoneyouchoose.

unixiseasytolearn.unixisamultiuseros.Learnunix.unixisapowerful.

SampleCommands

1. **Replacing or substituting string :**Sed command is mostly used to replace the text in a file.Thebelow simple sed commandreplacestheword “unix” with “linux”in thefile.
2. **$sed 's/unix/linux/' geekfile.txt**
   Output :
   linux is great os. unix is opensource. unix is free os.

   learn operating system.

   linuxlinuxwhichoneyouchoose.

   linuxiseasytolearn.unixisamultiuseros.Learnunix.unixisapowerful.

   Here the “s” specifies the substitution operation. The “/” are delimiters. The “unix” is the search pattern and the “linux” is the replacement string.

   By default, the sed command replaces the first occurrence of the pattern in each line and it won’t replace the second, third…occurrence in the line.

3. **Replacing the nth occurrence of a pattern in a line :**Use the /1, /2 etc flags to replace the first, second occurrence of a pattern in aline. The below command replaces the second occurrence of the word “unix” with “linux” in a line.
4. **$sed 's/unix/linux/2' geekfile.txt**
   Output:
   unix is great os. linux is opensource. unix is free os.

   learn operating system.

   unixlinuxwhichoneyouchoose.

   unixiseasytolearn.linuxisamultiuseros.Learnunix.unixisapowerful.

5. **Replacing all the occurrence of the pattern in a line :**The substitute flag /g (global replacement) specifies the sed command to replace all the occurrences of the string in the line.
6. **$sed 's/unix/linux/g' geekfile.txt**
   Output :
   linuxisgreatos.linuxisopensource.linuxisfreeos.

learnoperatingsystem.

linuxlinuxwhichoneyouchoose.

linuxiseasytolearn.linuxisamultiuseros.Learnlinux.linuxisapowerful.

7. **Replacingfromnthoccurrencetoalloccurrencesinaline:**Usethecombinationof/1, /2 etc and /g to replace all the patterns from the nth occurrence of a pattern in a line. The following sed command replaces the third, fourth, fifth… “unix” word with “linux” word in a line.
8. **$sed 's/unix/linux/3g' geekfile.txt**
   Output:
   unix is great os. unix is opensource. linux is free os.

   learn operating system.

   unixlinuxwhichoneyouchoose.

   unixiseasytolearn.unixisamultiuseros.Learnlinux.linuxisapowerful.

9. **Parenthesize first character of each word :**This sed example prints the first character of every word in parenthesis.
10. **$echo"WelcomeToTheGeekStuff"|sed's/\(\b[A-Z]\)/\(\1\)/g'**
    Output:

    (W)elcome(T)o(T)he(G)eek(S)tuff

11. **Replacing string on a specific line number :** You can restrict the sed command to replace the string on a specific line number. An example is
12. **$sed '3 s/unix/linux/' geekfile.txt**
    Output:
    unix is great os. unix is opensource. unix is free os.

    learn operating system.

    linuxlinuxwhichoneyouchoose.

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

    The above sed command replaces the string only on the third line.

13. **Duplicating the replaced line with /p flag :** The /p print flag prints the replaced line twice on the terminal. If a line does not have the search pattern and is not replaced, then the /p prints that line only once.
14. **$sed 's/unix/linux/p' geekfile.txt**
    Output:
    linux is great os. unix is opensource. unix is free os.

    linuxisgreatos.unixisopensource.unixisfreeos.

learnoperatingsystem.

linux linux which one you choose.

linuxlinuxwhichoneyouchoose.

linux is easy to learn.unix is a multiuser os.Learn unix .unix is a powerful.

linuxiseasytolearn.unixisamultiuseros.Learnunix.unixisapowerful.

15. **Printing only the replaced lines :**Use the -n option along with the /p print flag to display onlythe replaced lines.Here the-n option suppresses the duplicate rows generated by the /p flag and prints the replaced lines only one time.
16. **$sed -n 's/unix/linux/p' geekfile.txt**
    Output:
    linux is great os. unix is opensource. unix is free os.

    linux linux which one you choose.

    linux is easy to learn.unix is a multiuser os.Learn unix .unix is a powerful.If

    you use-n alone without /p, then the sed does not print anything.

17. **Replacing string on a range of lines :**You can specify a range of line numbers to the sed command for replacing a string.
18. **$sed '1,3 s/unix/linux/' geekfile.txt**
    Output:
    linux is great os. unix is opensource. unix is free os.

    learn operating system.

    linuxlinuxwhichoneyouchoose.

    unixiseasytolearn.unixisamultiuseros.Learnunix.unixisapowerful.

    Herethesedcommandreplacesthelineswithrangefrom1to3.Anotherexampleis

    $sed '2,$ s/unix/linux/' geekfile.txt
    Output:
    unix is great os. unix is opensource. unix is free os.

    learn operating system.

    linuxlinuxwhichoneyouchoose.

    linuxiseasytolearn.unixisamultiuseros.Learnunix.unixisapowerful

    Here $ indicates the last line in the file. So the sed command replaces the text from second line to last line in the file.

19. **Deleting lines from a particular file :** SED command can also be used for deleting lines from a particular file. SED command is used for performing deletion operation without even opening the file
    Examples:
    1.ToDeleteaparticularlinesayninthisexample
20. Syntax:
21. $sed'nd'filename.txt
22. Example:
23. $sed'5d'filename.txt

    2. To Delete a last line

    Syntax:

    sed'd'filename.txt

    3. To Delete line from range x to y

    Syntax:

    $ sed 'x,yd' filename.txt

    Example:

    $sed'3,6d'filename.txt

    4. To Delete from nth to last line

    Syntax:

    $ sed 'nth,$d' filename.txt

    Example:

    $sed'12,$d'filename.txt

    5. To Delete pattern matching line

    Syntax:

    $ sed '/pattern/d' filename.txt

    Example:

    $sed'/abc/d'filename.txt