**Uniq command**

In the uniq command helps you to print duplicate lines once in the output. It actually discards the lines which are repeated and prints the first adjacent repeated line, which enables us to view the output properly. File must be sorted before applying uniq command.

Syntax

uniq [OPTION]... [INPUT [OUTPUT]]

e.g

sort emp | uniq

**With -d, --repeated option**

The -d option prints only lines that are repeated. It discards non-duplicate lines.

Cut -d”|” -f 3 emp | Sort | uniq -d

**With -c, --count option**

Below, in the next example, we’re using the -c option to count the repeated lines.

**$** sort file2 | uniq -c

With -u, --unique option

Opposite of the above option, the -u option prints unique lines i.e., non-duplicate lines.

**$** cut -d”|” -f 4 emp |sort | uniq -u

**Grep command**

**Grep**is an acronym that stands for **G**lobal **R**egular **E**xpression **P**rint.

Grep is a Linux / Unix command-line tool used to search for a string of characters in a specified file. The text search pattern is called a regular expression. When it finds a match, it prints the line with the result. The grep command is handy when searching through large log files.

**To Ignore Case in Grep Searches(-i option )**

As grep commands are case sensitive, one of the most useful operators for grep searches is **-i.**Instead of printing lowercase results only, the terminal displays both uppercase and lowercase results. The output includes lines with mixed case entries.

An example of this command:

grep -i “clerk” emp

**Inverse grep Search (-v option)**

You can use grep to print all lines that **do not**match a specific pattern of characters. To invert the search, append **-v**to a grep command.

grep -v “manager” emp

**List Names of Matching Files (-l)**

Sometimes, you only need to see the names of the files that contain a word or string of characters and exclude the actual lines. To print only the filenames that match your search, use the **-l**operator:

grep -l “admin” \*.txt

**To Count the Number of Matches**

Grep can display the count of lines where it finds a match for your word.

Use the ***-c*** operator to count the number of matches:

grep -c “director” emp

**-e option to match multiple pattern:**

To match the multiple pattern.

$grep –e "Agarwal" –e "Aggarwal" –e "Agrawal" emp

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

$cat pattern.txt

Agarwal

Aggarwal

Agrawal

$grep –f pattern.txt emp

## Basic Regular expressions

Some of the commonly used commands with Regular expressions are tr, sed, vi and grep. Listed below are some of the basic Regex.

|  |  |
| --- | --- |
| **Symbol**  **Replace any single character(dot .)** |  |
| Grep “ap.le” a1 |  |
| **^pat**  Matches only at the beginning of record  Grep “^hello” a1 |  |
| **pat$**  matches at the end of the record  grep “hello$” |  |
| \*  Matches zero or more occurrence of previous character  Grep “ap\*le” a1 |  |

**[ ]**

**It matches the character which specified in range within brackets .**

**Grep ag[ar][ar]wal emp**

**[^ ]**

**It matches the character which is not specified in range within brackets .**

**Grep pat[^i]l emp**

**Extended Regular Expression**

when we want to use ERE with grep command then use egrep command instead of greap.

1. +

It specifies that one or more occurrence of previous character.

Egrep ap+le a1

1. ?

It specifies zero or one occurrence of previous charater.

Egrep ap?le a1

1. (|)

It matches the pattern with different combination .

**Tagged Regular Expression**

**Sed command**

SED command in UNIX is stands for stream editor and it can perform lot’s of function on file like, searching, find and replace, insertion or deletion. Though most common use of SED command in UNIX is for substitution or for find and replace

**Syntax**

**Sed option ‘adress action ‘ file**

**Actions**

**q :**

q means quit and print. It is same like head command.

Sed ‘5q’ emp

It display first 5 records.

**P:**

P means print. when we use p action it display selected record as well as entire file. To compress the entire file use -n option with p.

Sed -n ‘5p’ emp

It display the only 5th record .

Sed -n ‘5,9p’ emp

It display the only 5 to 9 record .

Sed -n ‘/clerk/p’ emp

It display the records who are clerk .

Sed -n ‘5p

/manager/p

/admin/ p’ emp

**d action :**

To delete the record.

Sed ‘3,5 d’ ep

Delete 3 to 5 record.

Sed ‘/account/ d’ emp

Remove all employees of account department.

Sed ‘3,5 d

/clerk/ d’ emp

Delete 3 to 5 record as well as all clerks.

**W action:**

W action copy the selected records to another file.

Sed -n ‘5,10 w p1’ emp

Copy record 5 to 10 to p1 file.

Sed -n ‘/director/w p1’ emp

Copy all directors to p1 file.

Sed -n ‘5,10 w p1

/admin/ w a1’ emp

Copy record 5 to 10 to p1 file and employees of admin department.

**I action**

I action is use to insert a record before specified line no.

Sed ‘5i\ hello’ emp

Insert hello word before 5 th line.

Sed ‘1i\ hello’ emp

Insert hello word as a first line.

Sed ‘i\ hello’ emp

Insert hello word before ecach line.

Sed ‘i\ ’ emp

Insert blank line before each line.

Sed ‘5i\ hello

\students ‘ emp

Insert hello and students word before 5th line.

**a action :**

a action is use to insert a record after specified line no.

Sed ‘5a\ hello’ emp

Insert hello word after 5 th line.

Sed ‘$a\ hello’ emp

Insert hello word as a last line.

Sed ‘a\ hello’ emp

Insert hello word after each line.

Sed ‘i\ ’ emp

Insert blank line after each line.

Sed ‘5i\ hello

\students ‘ emp

Insert hello and students word after 5th line.

S action :

S action is used to substitute the word in a file.

Sed ‘1,10s /hello/hi/’ emp

Here it substitutes all hello words with hi in first 10 line. But one problems occurred that it replaces the first occurrence of each line. Solve this problem with flag g

Sed ‘1,10s /hello/hi/g’ emp

Sed ‘/director/s/director/executive/g ‘ emp