Practical No: 1 Simple Filters

Q. Implement Linux command for Simple Filters and I/O redirection:

1) pr

2) head

3) tail

4) cut

5) paste

6) sort

7) tee

8) uniq

9) tr

-> Sample file

```
$ cat emp.lst
2233|a.k. shukla      |g.m.     |sales     |12/12/52|6000
9876|jai sharma       |director |production|12/03/50|7000
5678|sumit chakrobarty|d.g.m.   |marketing |19/04/43|6000
2365|barun sengupta   |director |personnel |11/05/47|7800
5423|n.k. gupta       |chairman |admin     |30/08/56|5400
1006|chanchal singhvi |director |sales     |03/09/38|6700
6213|karuna ganguly   |g.m.     |accounts  |05/06/62|6300
1265|s.n. dasgupta    |manager  |sales     |12/09/63|5600
4290|jayant Choudhury |executive|production|07/09/50|6000
2476|anil aggarwal    |manager  |sales     |01/05/59|5000
6521|lalit chowdury   |director |marketing |26/09/45|8200
3212|shyam saksena    |d.g.m.   |accounts  |12/12/55|6000
3564|sudhir Agarwal   |executive|personnel |06/07/47|7500
2345|j.b. saxena      |g.m.     |marketing |12/03/45|8000
0110|v.k. agrawal     |g.m.     |marketing |31/12/40|9000
```

1)pr command: it prepares the file for printing by adding suitable header, footers and formatted text.

Options

i)pr -t -k: prints in columns where k is an integer.

ii)-d: double space input, reduce clutter

iii)-n: Numbers lines.

iv)-o n: offset lines by n spaces increases left margin of the page

2) head command: displays the beginning of the file.

Options

i)-n 3(or any digit):specifies a line count

3) tail command:

4) cut command:

Options:

i)cutting the column (-c)

ii) cutting fields (-f)

5) paste command:

6) sort command:

7) tee command:

8) uniq command:

Options:

i)-d

ii) -u

9) tr command:

Options:

Answer the following questions.

1) Create a file named 'Numbers' containing numbers 1-20. Print data of this file in 5 columns.
   Command = cat > Numbers (enter 1-20)

   Command = $pr -t -5
2) Create a file 'employee' and enter the sample data given above.
   Command = $ cat>employee
3) Display employee file in printable format.
   Command = $pr employee
4) Display employee file with space and line numbers.
   Command = $pr -t -n -d -o 4 employee
5) Display first 10 records of employee file.
   Command = $ head employee
6) Display first 3 records of employee file.
   Command = $ head -n 3 employee
7) Display last 10 records of employee file.
   Command = $ tail employee
8) Display last 3 records of employee file.
   Command = $ tail -n 3 employee
9) Cut first 5 records of employee file and copy it in file named 'shortlist'
   Command = $head -n 5 employee | tee shortlist
10) Extract two columns of shortlist file.

Command = $cut -c 6-22,24-30 shortlist

11) Extract second and third column(fields) by specifying delimiter.
    Command = $ cut -d \| -f 2,3 shortlist | tee cutlist1
    Command = $ cut -d "|" -f 1,4 shortlist > cutlist2
12) Extract the fields numbered 1,4,5 and 6 and save the output in cutlist2.
    Command = cut -d "|" -f 1,4- shortlist > cutlist2
13) Paste above two files side by side.
    Command = $paste cutlist1 cutlist2
14) Put delimiter in above command.
    Command = $paste -d"|" cutlist1 cutlist2
15) Sort file 'shortlist'
    Command = $sort shortlist
16) Sort 'shortlist' file on second field.
    Command = $sort -t"|" -k 2 shortlist
17) Sort 'shortlist' file in reverse order.
    Command = $sort -t"|" -r -k 2 shortlist
18) Show sorting on secondary key.
    Command = $sort -t"|" -k 3,3 -k 2,2 shortlist
19) Create a 'dept' file with following records. Display only unique records.

```
$ cat dept.lst
01|accounts|6213
01|accounts|6213
02|admin|5423
03|marketing|6521
03|marketing|6521
03|marketing|6521
04|personnel|2365
05|production|9876
06|sales|1006
```

    Command = cat > dept
    Command = $cat dept
    Command = $ uniq dept
20) Display only non-repeated lines from 'dept'.
    Command = $ uniq -u dept
21) Display only one copy of repeated records.
    Command = $ uniq -d dept
22) Count the frequency of occurrence of repeated records.
    Command = $ uniq -c dept
23) Replace "|" symbol with '~' sign and '/' with '-'symbol in file employee.
    Command = $ tr "|/" "~-" < employee | head -n 3
24) Convert small letter alphabet to uppercase in 'dept' file.
    Command = $ head -n 3 dept | tr '[a-z]' '[A-Z]'

Practical No: 2 grep Family

Grep:

Egrep:

Fgrep:

Table 13.2 The Basic Regular Expression (BRE) Character Subset

Symbols or Expression	*Matches*
*	Zero or more occurrences of the previous character
g*	Nothing or g, gg, ggg, etc.
.	A single character
.*	Nothing or any number of characters
[*pqr*]	A single character *p*, *q* or *r*
[*c1-c2*]	A single character within the ASCII range represented by *c1* and *c2*
[1-3]	A digit between 1 and 3
[^*pqr*]	A single character which is not a *p*, *q* or *r*
[^a-zA-Z]	A nonalphabetic character
^*pat*	Pattern *pat* at beginning of line
pat$	Pattern *pat* at end of line
bash$	bash at end of line
^bash$	bash as the only word in line
^$	Lines containing nothing

Table 13.3 The Extended Regular Expression (ERE) Set Used by **grep**, **egrep** and **awk**

Expression	*Significance*
ch+	Matches one or more occurrences of character *ch*
ch?	Matches zero or one occurrence of character *ch*
exp1|*exp1*	Matches *exp1* or *exp2*
GIF|JPEG	Matches GIF or JPEG
(*x1*|*x2*)*x3*	Matches *x1x3* or *x2x3*
(lock|ver)wood	Matches lockwood or verwood

Answer the following questions.

1) Display lines containing sales in ‘employee’.
   Command = $ grep “sales” employee
2) Display lines containing ‘accounts’ from ‘employee’ and ‘dept’.
   Command = $ grep “accounts” employee dept
3) Display lines containing ‘agrawal’ in employee.
   Command = $ grep -i ‘agrawal’ employee
4) Display all lines except ‘accounts’ from ‘dept’.
   Command = $ grep -v ‘accounts’ dept
5) Display line numbers containing ‘marketing’ from employee.
   Command = $ grep -n ‘marketing’ employee
6) How many directors are there in the file ‘employee’?
   Command = $ grep -c ‘director’ employee
7) Display only file names containing ‘manager’.

Command = $ grep -l ‘manager’ *

8) Display all the different ‘aggrawal’ from employee.
   Command = $ grep -e “Agarwal” -e “Aggarwal” -e “Agrawal” employee
   Command = $ egrep ‘Ag(arwal|garwal|rawal)’ employee
9) Accept pattern from a file for matching and display output.
   Command = $ grep -f pattern employee
   Example = $ cat > file1
   Hello World. This file has random Text.
   $ cat > file2
   World
   Text
   Purpose
   Command = $ fgrep file2 file1
10) Display ‘agrawal’ with the help of regular expression.
    Command = $ grep “[aA]g[ar][ar]wal” employee
    Command = $ grep “[aA]gg*[ar][ar]wal” employee
11) Display name ‘J ____ Saxena’ from employee.
    Command = $ grep “j.*Saxena” employee
12) Display employee records whose number starts from 2.
    Command = $ grep “^2” employee
13) Display employee records whose salary is between 7000 – 7999.
    Command = $ grep “7…$” employee
14) Display employee records whose number doesn’t start from 2.
    Command = $ grep “^[^2]” employee
15) Display the records of Sengupta and Dasgupta.
    Command = $ grep -E ‘sengupta|dasgupta’ employee