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

Table

Description automatically generated

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: it is used for cutting out the sections from each lines of file and writing the result to standard output.

#cut -b 1,2,3 txt1 or cut -b -3 txt1

Cut -b 1-3,5-7 txt1

Cut -b 1- txt1

Options:

i)cutting the column (-c)

syntax: #cut -c [(k)-(n),(n)/(n)] filename

k denotes the starting position of the character and n denotes the ending position of the character in each line, if k and n are separated by “-“ otherwise they are only the position of character in each line from the file taken as input.

#cut -c 2,4 emp

#cut -d | -f 1 emp

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 Numbers

1. Create a file ‘employee’ and enter the sample data given above.

Command = $ cat>employee

1. Display employee file in printable format.

Command = $pr employee

1. Display employee file with space and line numbers.

Command = $pr -t -n -d -o 4 employee

1. Display first 10 records of employee file.

Command = $ head employee

1. Display first 3 records of employee file.

Command = $ head -n 3 employee

1. Display last 10 records of employee file.

Command = $ tail employee

1. Display last 3 records of employee file.

Command = $ tail -n 3 employee

1. Cut first 5 records of employee file and copy it in file named ‘shortlist’

Command = $head -n 5 employee | tee shortlist

1. Extract two columns of shortlist file.

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

1. 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

1. Extract the fields numbered 1,4,5 and 6 and save the output in cutlist2.

Command = cut -d “|” -f 1,4- shortlist > cutlist2

1. Paste above two files side by side.

Command = $paste cutlist1 cutlist2

1. Put delimiter in above command.

Command = $paste -d”|” cutlist1 cutlist2

1. Sort file ‘shortlist’

Command = $sort shortlist

1. Sort ‘shortlist’ file on second field.

Command = $sort -t”|” -k 2 shortlist

1. Sort ‘shortlist’ file in reverse order.

Command = $sort -t”|” -r -k 2 shortlist

1. Show sorting on secondary key.

Command = $sort -t”|” -k 3,3 -k 2,2 shortlist

1. Create a ‘dept’ file with following records. Display only unique records.

Text

Description automatically generated

Command = cat > dept

Command = $cat dept

Command = $ uniq dept

1. Display only non-repeated lines from ‘dept’.

Command = $ uniq -u dept

1. Display only one copy of repeated records.

Command = $ uniq -d dept

1. Count the frequency of occurrence of repeated records.

Command = $ uniq -c dept

1. Replace “|” symbol with ‘~’ sign and ‘/’ with ‘-‘symbol in file employee.

Command = $ tr “|/” “~-“ < employee | head -n 3

1. 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

Description automatically generated

Table

Description automatically generated with medium confidence

Answer the following questions.

1. Display lines containing sales in ‘employee’.

Command = $ grep “sales” employee

1. Display lines containing ‘accounts’ from ‘employee’ and ‘dept’.

Command = $ grep “accounts” employee dept

1. Display lines containing ‘agrawal’ in employee.

Command = $ grep -i ‘agrawal’ employee

1. Display all lines except ‘accounts’ from ‘dept’.

Command = $ grep -v ‘accounts’ dept

1. Display line numbers containing ‘marketing’ from employee.

Command = $ grep -n ‘marketing’ employee

1. How many directors are there in the file ‘employee’?

Command = $ grep -c ‘director’ employee

1. Display only file names containing ‘manager’.

Command = $ grep -l ‘manager’ \*

1. Display all the different ‘aggrawal’ from employee.

Command = $ grep -e “Agarwal” -e “Aggarwal” -e “Agrawal” employee

Command = $ egrep ‘Ag(arwal|garwal|rawal)’ employee

1. 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 OR $ fgrep -f file2 file1

1. Display ‘agrawal’ with the help of regular expression.

Command = $ grep “[aA]g[ar][ar]wal” employee

Command = $ grep “[aA]gg\*[ar][ar]wal” employee

1. Display name ‘J \_\_\_\_ Saxena’ from employee.

Command = $ grep “j.\*Saxena” employee

1. Display employee records whose number starts from 2.

Command = $ grep “^2” employee

1. Display employee records whose salary is between 7000 – 7999.

Command = $ grep “7…$” employee

1. Display employee records whose number doesn’t start from 2.

Command = $ grep “^[^2]” employee

1. Display the records of Sengupta and Dasgupta.

Command = $ grep -E ‘sengupta|dasgupta’ employee