Lab8.
Text Processing Tools and Regular Expressions

Instructor :Murad Njoum

Objectives

After completing this lab, the student should be able to:
- Identify and use filters as valuable text processing tools.
- Use simple regular expressions to make text processing more efficient

Text Processing using Filters

In the pipes lab, we mentioned a group of commands called filters. These are basically commands that take some input and then filter it to produce the requested **output without changing the original source of input**. In this lab we will practice how to use some filters as useful tools for **text processing**

Filters:

head and tail: used to display lines from the beginning or end of a given input respectively.
cat: used to view or concatenate files.
grep: used to extract certain rows (lines) from a given input. We will concentrate on the options -i, -l (EL), -v.
cut: used to extract certain columns from a given input. We will use the options -d, -f, and –c.
tr: translates (changes) a given input to a specified output
wc: used to count lines, words, or characters in a given input.
sort: used for sorting a given input. We will present the options -i, -o, -u, -n, -k, and –t.
sed: used for stream editing (changing parts of an input to a specified output)

Create Students file(using vi)

ah6:506:Ahmad_Hamdan
sh5:345:Suha_HAMDAN
rd7:427:Ribhi_ahmad
hr4:234:hamdan_ribhi
ad6:386:Arwa_Ahmad
ad5:285:ahmadi_Ahmad

Execute the following commands:

head -2 students

```
mnjoum@ubuntu:~$ head -2 students

ah6:506:Ahmad_Hamdan
sh5:345:Suha_HAMDAN
View first 2 lines
```

Try

head -n 2 students

```
mnjoum@ubuntu:~$ head -n 2 students
ah6:506:Ahmad_Hamdan
sh5:345:Suha_HAMDAN
```

Or head –n +2 students

Try

head -n -2 students

```
mnjoum@ubuntu:~$ head -n -2 students
ah6:506:Ahmad_Hamdan
sh5:345:Suha_HAMDAN
rd7:427:Ribhi_ahmad
hr4:234:hamdan_ribhi
```

Execute the following commands:

- *tail -3 students*

```
mnjoum@ubuntu:~$ tail -3 students
hr4:234:hamdan_ribhi
ad6:386:Arwa_Ahmad
ad5:285:ahmadi_Ahmad

View last 3 lines from file
```

Try

tail -n 3 students

```
mnjoum@ubuntu:~$ tail -n 3 students
hr4:234:hamdan_ribhi
ad6:386:Arwa_Ahmad
ad5:285:ahmadi_Ahmad
```

Try

tail -n -3 students

Try

tail -n +3 students

```
mnjoum@ubuntu:~$ tail -n +3 students
rd7:427:Ribhi_ahmad
hr4:234:hamdan_ribhi
ad6:386:Arwa_Ahmad
ad5:285:ahmadi_Ahmad
```

Execute the following commands:

What command would you use to get the fourth line only from file students ?

Note: default for value on n is 10

Execute the following commands:

- *cat students*

  grep ahmad students

  Join both cat and grep with pipes to get the same result as the previous grep command:

grep -i Ahmad students

*grep -l Ribhi * (*:means all files in current directory)*

Describe Output? Give a solution for this case ? Try grep *-L Ribhi **

- *grep -v Ribhi students*

-v, --invert-match
Invert the sense of matching, to select non-matching lines.

grep -iv hamdan students

Execute the following commands:

- *cut -d: -f2 students*

- Q: *What command would you use to get the last names for all users in file students:*

Q: What command would you use to get the first names of all users with last name hamdan (all cases)

Execute the following commands:

cut -c2,3 students

-c, --characters=LIST
select only these characters

Continue

- *What command would you use to get the middle digit in the id numbers for all users with last name hamdan ?*
- *tr "a-z" "A-Z" < students (Describe output)*
- *What command would you use to get the first names (all in lower case) of all users that have the word ahmad (all cases) as part of their full name:*
- *wc -l students*

- *head -1 students | cut -d: -f3 | cut -d_ -f2 | wc -c*

  same as previous, try with wc –w ,what does mean –w,-l,-c,-m

- *What command would you use to count the number of files in your home directory?*

sort students (Describe output)

sort -o result students (What happened?)

Try : sort –r *result students* or sort –r students > result

- *sort -k2 -t: -n students (Describe output)*

sort -k2 -t: -n students
hr4:234:hamdan_ribhi
ad5:285:ahmadi_Ahmad
sh5:345:Suha_HAMDAN
ad6:386:Arwa_Ahmad
rd7:427:Ribhi_ahmad
ah6:506:Ahmad_Hamdan

- *What command would you use to list all the last names of users in file students sorted based on lower case letters and without repetition*

cut -d : -f3 students|cut -d _ -f2 |sort -f -u

-f : fold lower case to upper case characters
-u :unique

- *What is different when we run the same command with the i (ignore case) option, as follows:*
  sed 's/ahmad/damha/i' students

```
sed s/ahmad/damha/i students
ah6:506:damha_Hamdan
sh5:345:Suha_HAMDAN
rd7:427:Ribhi_damha
hr4:234:hamdan_ribhi
ad6:386:Arwa_damha
ad5:285:damhai_Ahmad
```

What is different when we run the same command with the g (global) option, as follows:
sed 's/ahmad/damha/ig' students ,

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. By using SED you can edit files even without opening it, which is much quicker way to find and replace something in file, than first opening that file in VI Editor and then changing it.

- SED is a powerful text stream editor. Can do insertion, deletion, search and replace(substitution).

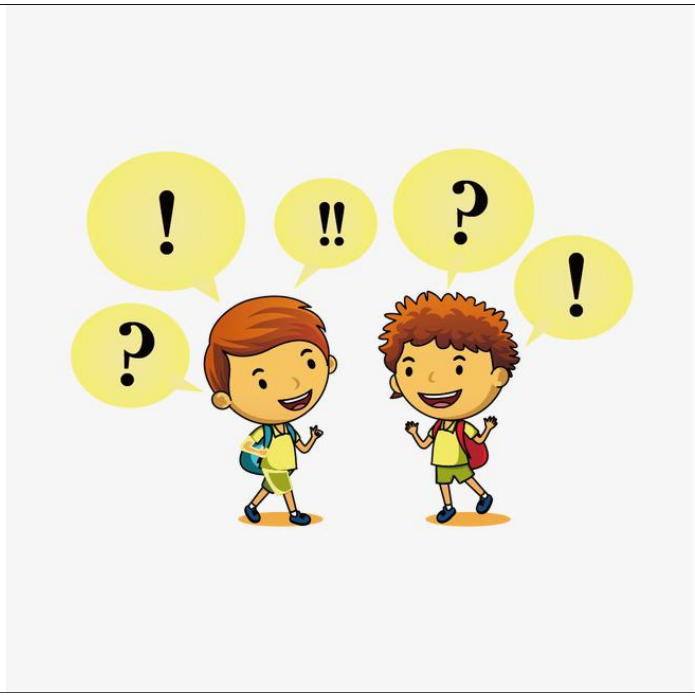
- SED command in Unix supports regular expression which allows it perform complex pattern matching.

Regular Expressions

- Some of the filters mentioned above such as ***grep*** and ***sed*** may use what we call regular expressions to be more powerful and precise. To get more information about the power and extent of regular expressions, you can read the man pages using the command:
  man regex
- **pattern$** : applied to a pattern if it is at the **end of a given line**.
  ^pattern: applied to a pattern if it is at the beginning of a given line.
  [abc]: means a or b or c
  [^abc]: means all characters except a, b, or c.

Command Cont…

- grep -i 'hamdan$' students
- cut -d: -f3 students | grep -i '^ahmad'
- cut -d: -f3 students | cut -d_ -f1 | grep -i '^ahmad$'
- cut -d: -f1 students | grep a[dh][^6]
- cut -d: -f3 students | sed 's/^ahmad/sameer/ig'
- sed 's/ahmad$/Sameer/i' students



exciting

EDUCATION

INFORM ENRICH INSPIRE