Samnang San

Linux Admin SED Assignment #3

October 3, 2021

Sed is a command in Linux administration that stands for stream editor. The Sed command can perform multiple functions on files like search, find, substitution, insert or even deleting a file. The command does not change the original file by default unless it’d redirected to another file. We will be testing the SED command with the following file name datebook.

Text

Description automatically generated

Text

Description automatically generated

1. We are going to Change the name Meg to Meghan in the file.

Command: sed s/Meg/Meghan/ datebook

Sed is the is the command, “s” is a function in sed which is used for substituting. Meg is the name or pattern we are looking for. Meghan is the in between the slashes for which we are substituting meg with. Followed by datebook which is the name of the file.

Text

Description automatically generated

1. Deleting the last 3 lines of the file.

Command: sed ‘$d;$d;$d’ datebook

We will be calling SED command on almost all exercises. We will be using the delete function on that exercise. We start with the sed command followed by a two single quotes. In between the quotes, we will add a range of which lines we want to delete. “$” sign stands for last line and “d” stands for delete. So here, I did $d three times between the quotes to delete the last three lines. Followed by the name of the file.

Text

Description automatically generated

Text

Description automatically generated

1. Prints lines 2 through 4.

Command: sed -n ‘2,4p’ datebook

Call sed followed by an -n, which stops automatic printing and only prints what we asked it to. Followed by two single quotes. In between the quotes, we provide the range. In this example, we use 2 and 4 then a p. The “p” function is for printing. Followed by the name of the file again which is datebook.

Graphical user interface, text, application

Description automatically generated

1. Delete lines containing Place.

Command: sed ‘/Place/d’ datebook

Sed is called upon again. We will be using the delete function. Sed followed by single quotes and two slashes inside the singles quotes. In this example, we are using Place. Putting place in between the slashes followed by a “d” after the last slash, tell it to delete the lines containing the word place.

Text

Description automatically generated

1. Print all lines where the birthdays are in the first week of the month. The format should be MM/DD/YY.

Command: sed -n -r ‘/[0-9]{5}\:[0-9]+\/[1-7]\//p’ datebook

n- stop automatic printing, -r is using the extended regex in sed. First, we match the first 5 numbers with [0-9]{5} followed by a colon. Next, [0-9] for month that follows the 5 numbers and a colon. [1-7] stands for the first week of the month. P command prints the line out. Followed by the name of the file. See sources.

Text

Description automatically generated

1. Append three asterisks to the end of lines starting with Sir.

Command: sed ‘/^Sir/a\\*\*\*’ datebook

Sed command is called. In this example, we are looking for a pattern, so we are going to use the slashes again. ‘/^Sir/, the “^” is the beginning of the line anchor, so it means the beginning of line and name Sir. “a\” means to append. Like this example, we are appending 3 \*s so we typed in “\*\*\*”. followed by name of file= datebook.

Text

Description automatically generated

Text

Description automatically generated

7.Replace line containing “Westley Pirate “with the phrase “As you wish”.

Command: sed ‘/Westley Pirate/c\ As you Wish’ datebook

Sed command is called. Pattern we are looking for is Westley Pirate, so we do ‘/Westley Pirate/. Then we use function “c\” which replaces the whole line with what we enter after. In this case, as you wish replaces Westley pirate. Always add the name of file at the end. Note that it replaces the whole line with as you wish.

Text

Description automatically generated

8. Change Minerva McGonagall’s birthday to 12/25/1960. Assume you don’t know original birthday. Use regular expressions to search.

Command: sed ‘/Minerva/s,[0-9]\+/[0-9]\+/[0-9]\+,12/25/60,’ Datebook

Sed command is called. Pattern we are looking for is Minerva so we put that in between slashes. The “s’ stands for substitute. Since we don’t know the original birthdate, we use [0-9] so sed can search for that range. Example [0-9]/[0-9]/[0-9]. Then we add that date 12/25/60. Followed by the name of the file.

Text

Description automatically generated

9. Delete all blank lines.

Command: sed ‘/^$/d’ datebook

Command sed is called. We are looking for a pattern again, so we need to put something in between slashes. “^” is the beginning and “$” is the end. We didn’t put anything after the “^” or before the “$”, So sed is looking for start and end with blank spaces. “d” function deletes the lines that match which is the blank spaces. Followed by name of file.

Text

Description automatically generated

10. Sed Script, I will begin with the commands.

a. Insert above the first line the title -Great Literary Characters-

command: sed ‘1i/-Great Literary Characters-/’ datebook

The 1 in the beginning refers to the first line. “I” is to insert. We are inserting -Great Literary Characters- in between the //. In the first line, we are inserting -Great Literary Characters-.

b. Remove duplicate lines

Command: sed ‘/^$/d’ datebook

This command removes all blank lines like number 9 our tutorial.

Command: sed ‘17d’ datebook

This command removes line 17 using the delete command which is a “d”. That’s the only duplicate in datebook.

c. Print content of file with first name then phone number.

I have been stuck on this problem for weeks and I can’t seem to figure it out.

After completing the awk section. Only way I can think of this to work is piping to awk.

I created a script named test.sed

Command: Sed -f (name of script) datebook | awk -F: ‘{print $1,$2}’

We will run this last…

D. Append “Happily Ever after. The End” at the end.

Sed ‘$ a Happily Ever after. The End’ datebook

$ means the end. Where will append Happily Ever after. The End.

Begin by typing vim (name of file).sed, here I created a file called test.sed

To enter insert, you have type “I” in the command prompt.

First type in #!/usr/bin/sed and enter commands like the example below. Notice how you don’t have to type Sed before the script? When typing in script, you don’t have to. Now to save, hit the esc key followed by a colon and wq. For example, :wq and enter.

Text

Description automatically generated

Now, to run the script. Type in: sed -f (name of file).sed (name of file your screening)

In this example, sed -f test.sed datebook

Text

Description automatically generated

And now for Part C with the script we made.

Text

Description automatically generated

**Sources used**

https://flylib.com/books/en/4.356.1.40/1/

<https://www.cs.ait.ac.th/~on/O/oreilly/unix/sedawk/ch02_05.htm>

Question #5, I had to seek a tutor to help me figure out using regex extended. Question 10 took me a very long time. Tutors were not able to assists me with that one. After Completing the awk assignment, it gave me the idea to pipe with awk.