Robert Call

Linux Admin

10/9/21

**Lab 3 Sed Lab**

1. Change the name Meg to Megan

Code: sed s/Meg/Megan/g datebook

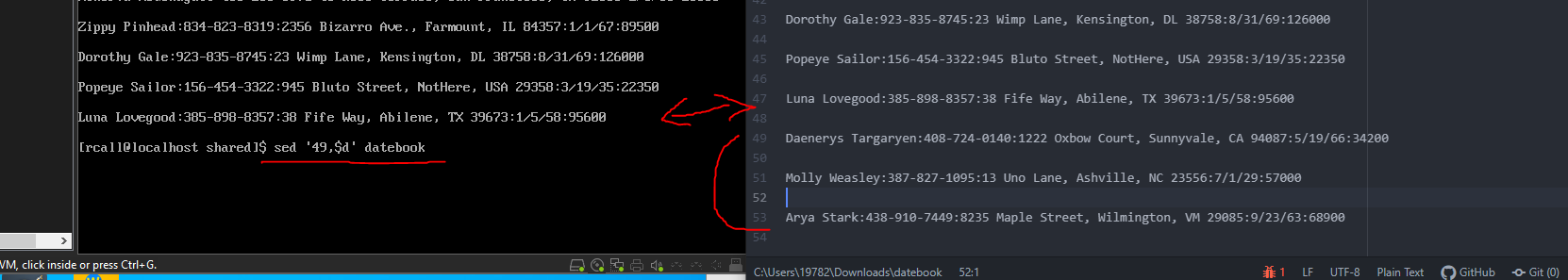
Explanation: using sed with the s for substitute we put Meg which is what we want to substitute for Megan and the g is for global which means it will find Meg if there was more than one in the file datebook and change it to Megan for the output printed.



2. Delete the last 3 lines.

Code: sed ’49,$d’ datebook

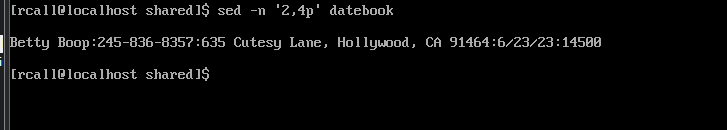
Explanation: using sed we specific we want from line 49 to the last line $ deleted d from datebook. This gives use the effect of from line 49 down those lines are deleted.

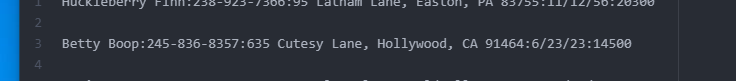


3. Print lines 2 through 4 .

Code: Sed ‘2,4p’ datebook

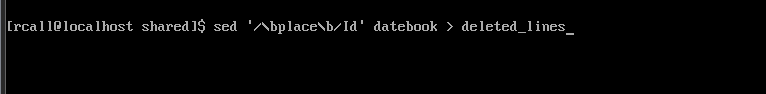
Explanation: using sed we request that only lines 2 through four to be printed this only prints out betty boop because the blank lines still count as lines see below.





4. Delete lines containing Place.

Code: sed ‘/\bplace\b/Id’ datebook > deleted\_lines

Explanation: using the command we use sed to find lines with place in them and print them. The \b \b is for borders meaning I only want place the word by itself. I then use datebook > deleted\_lines which uses the command on datebook but sends output into a new created file called deleted\_lines. We can then CAT the file deleted\_lines which displays the output of the command we wrote to view the output easier.

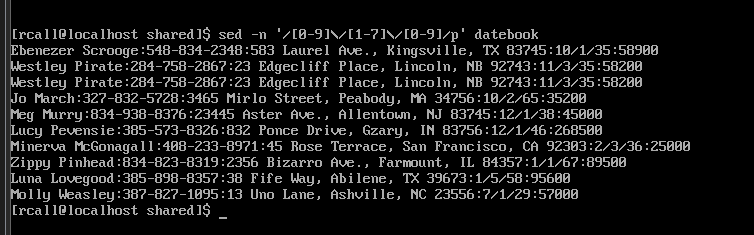
Text

Description automatically generated

5. Print all lines where the birthdays are in the first week of the month. Be careful of the dates for birthdays, the format is MM/DD/YY

Code: sed -n ‘/[0-9]\/[1-7]\/[0-9]/p’ datebook

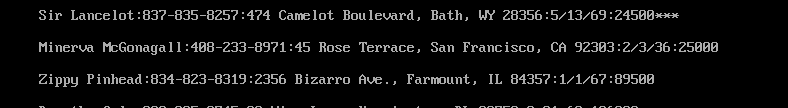
Explanation: we use the sed command to print out lines with a format of a number followed by a / and a number between [1-7] for the first week only of the month followed by another / for the year. We use \ which is a delimiter so that the command reads the / as a slash not a special character. Then we use p to print the lines.



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

Code: sed ‘ /Sir/ s/$/\*\*\*/’ datebook

Explanation: we use Sir to locate the line we want add the three asterisks too. Then using s for substitute and $ for end of line and specify to add three \*\*\*.



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

Code: sed ‘s/Westley Pirate/As you wish/’ datebook > wishpirate

Explanation: using sed we use s for substitute for the phrase Westley Pirate with As you wish and using the > we put the results in a file called wishpirate

Text

Description automatically generated

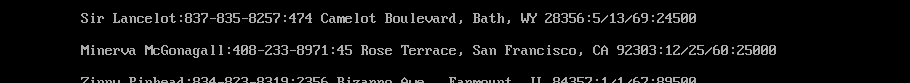
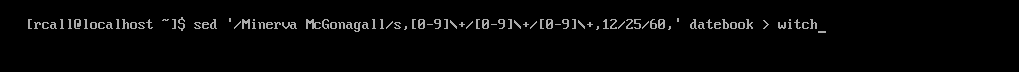
Text

Description automatically generated

8. Change Minerva McGonagall's birthday to 12/25/1960 . Assume you don't know Minerva's original birthday. Use a regular expression to search for it.

Code: sed ‘/Minerva McGonagall/s,[0-9]\+/[0-9]\+/[0-9]\+,12/25/60,’ datebook > witch

Explanation: we use sed with Minerva McGonagall to find the line we want to change. Using s for substitute we specify we want to find a date in that line and change it to what we want and then output results to a file named witch. We also use , and + as delimiters instead of / just to make it less confusing when trying to find a date format in the line.



9. Delete all blank lines.

Code: sed ‘/^$/d ‘ datebook

Explanation: using the sed command we us ^$ to specify lines that begin and end with blank then use the d in the command to delete them out of datebook.

Text

Description automatically generated

10. Write a sed script that will (actual sed script, NOT just the commands on the command line)

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

b. Remove the duplicate lines

c. Print the contents of the file with the first name then the phone number.

d. Append at the end of the file "Happily Ever after. The End"

Code:

**sed -e '17d ;**

**/^$/d ;**

**/^/s/:[0-9]\* .\*// ;**

**s/..[a-z]\*:/ /;**

**1i\ Great Literary Characters;**

**$a\ Happily Ever after. The End' $answer > results**

**cat results**

Explanation: #The commands below are as follows in order

# sed -e '; ; ; ; ;' $answer > results will allow us to run multiple patterns in one command we also use ' from the first pattern to the last pattern and split them up with ; this will allow us to run all of the commands till we send the output to a file we create or update with our output.

# 17d will delete the duplicate line Wesley

# /^$/d will remove blank lines from our list

# /^/s/:[0-9]\* .\*// this will remove everything but the first half of the line which contains most# of what we need I found not adding a space after the first \* would not give me the right information after testing putting a space in seems to give us the first name last name and phone number we still need to cut this down I wasn’t able to combine the next command yet so i had it as a# second command. This command works by starting from the beginning of the line with the ^ character then using :[0-9]\* .\*/ to look for the second colon and start there the space between the \* .\* is important because this way it will match the second colon and the numbers afterwards with a space. the following part of the command // is to replace everything from the beginning of this string to the end of the line with a blank space to erase it. leaving only the first and last name also the phone number.

#s/..[a-z]\*:/ / this command will remove the last name by first starting at ..[a-z]\*: which grabs the last name because the .. is any character followed by letters [a-z] then using \* to match any number of characters so we don’t have to set a limit. That string will end with : grabbing the last name and then like the last command we use / / a space to delete it and separate the first name and phone number.

#1i\ Great Literary Characters This command will start by specifying 1 which is the line we want to i insert the text above this line with following text.

#$a\ Happily Ever after. The End This will append below the end of the last line by using the $a command which means go to the last line $ and using a for append it adds a new line with the following text.

#$anwser > results will take what file you specified and send it to results file which will be displayed automatically by the next command.

#cat results will display the file results which will have the output from this script.



Citations:

I used our weekly material:

Read the chapter from <https://flylib.com/books/en/4.356.1.40/1/>

Adrianna's video on how to use sed: <https://youtu.be/YMqOocY0ovs>

I also used following sites to get a better understanding or Ideas:

https://unix.stackexchange.com/questions/95939/how-exactly-do-i-create-a-sed-script-and-use-it-to-edit-a-file

https://www.linuxquestions.org/questions/programming-9/removing-duplicate-lines-with-sed-276169/ http://www.yourownlinux.com/2015/04/sed-command-in-linux-append-and-insert-lines-to-file.html

https://riptutorial.com/sed/example/12280/using-different-delimiters