Shell Tutorial

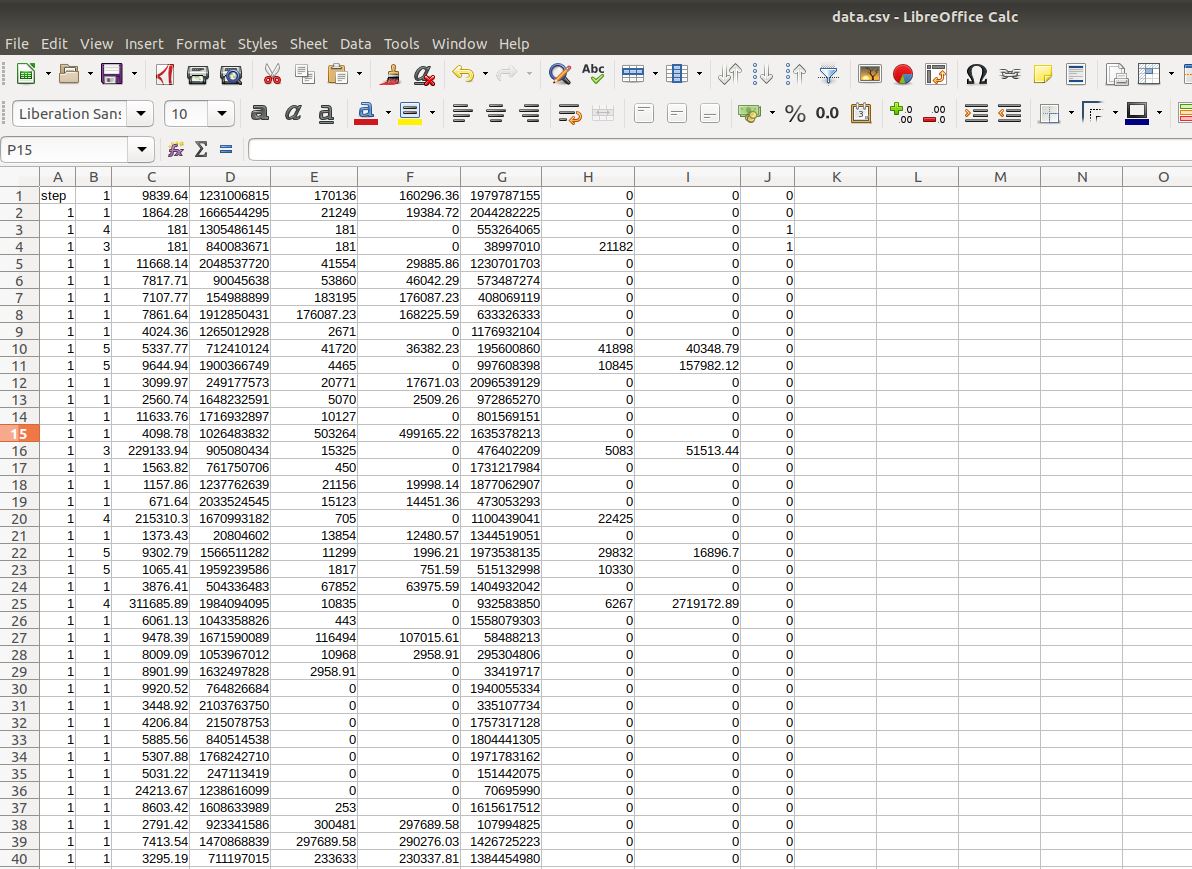
Part-3 (Loops and some other advanced commands)

Before moving to the commands, first let’s have a look at the data files.

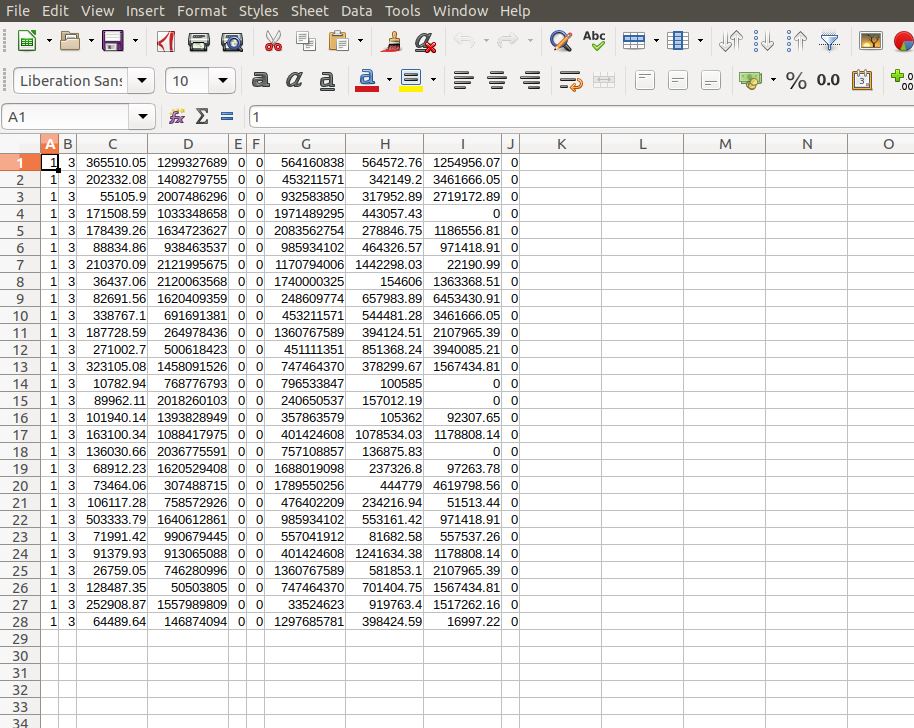
Here we are having two data files “data.csv” and “data1.csv”.

Note - In computing, a comma-separated values (CSV) file is a delimited text file that uses a comma to separate values.

**Data.csv**



**Data1.csv**

****

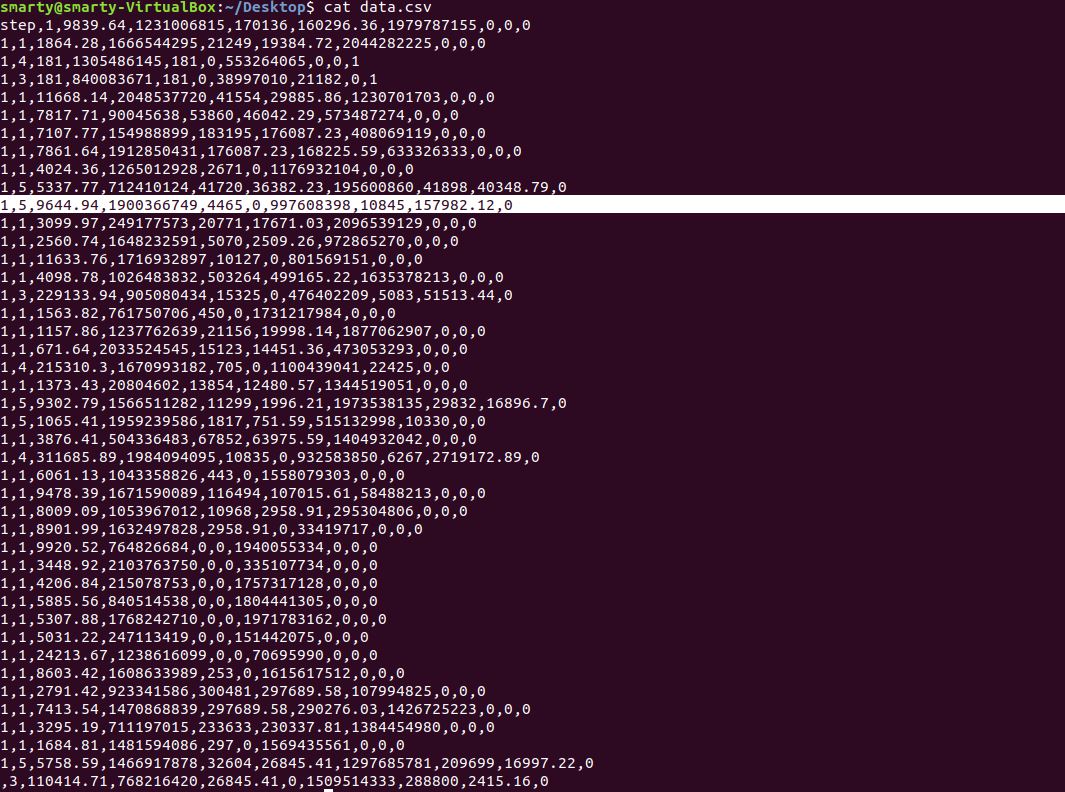
1. **grep –** This command can be used to filter any keyword from a file.

If you want to search for a specific keyword, then you can use this command to filter that out, it will give the whole row which contains that keyword.

Here I want to search for the row containing “1900366749” in file data.csv, so I write a command “grep 190036749 data.csv”

**General Syntax:** grep <keyword> <file\_path>

C:\Users\smarty\Desktop\3\2.JPG



Here you can match the output from the above command and the actual file.

1. **>(Greater Than symbol)** **<filename>**– This command is used to store the output of a command to file.

If you want to store the output of a command in a new file, then you can use ‘> filename’ to do so.

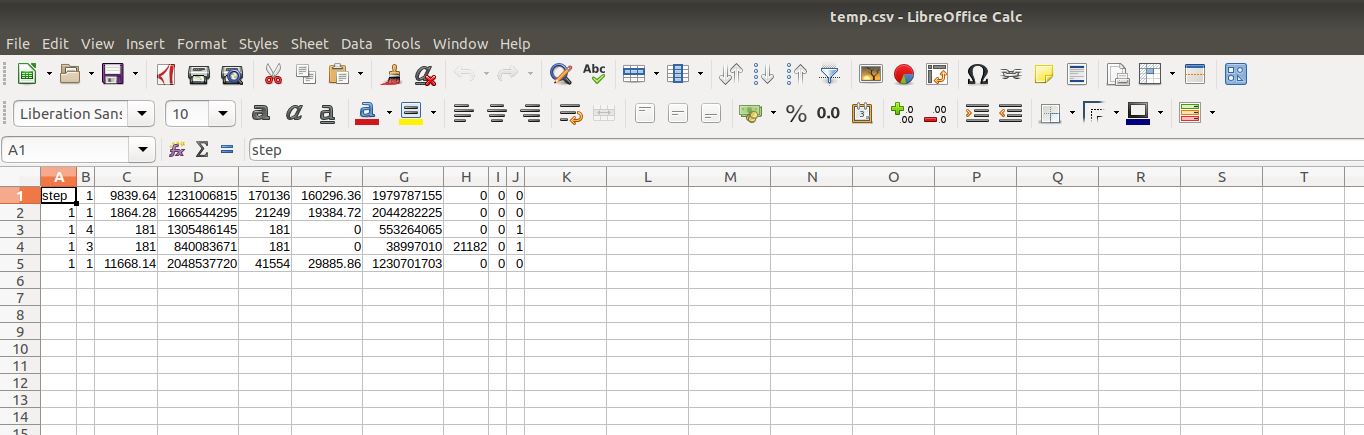
Like here, I want to store the first 5 rows of data.csv file in a new file named “temp.csv”

C:\Users\smarty\Desktop\3\3.JPG

Now, see a new file is automatically created on the desktop named “temp.csv”

C:\Users\smarty\Desktop\3\4.JPG

Here is the content of the new file, which is the top 5 rows of the data.csv file.



1. **| (Pipeline Symbol)** – This symbol can combine two or more commands.

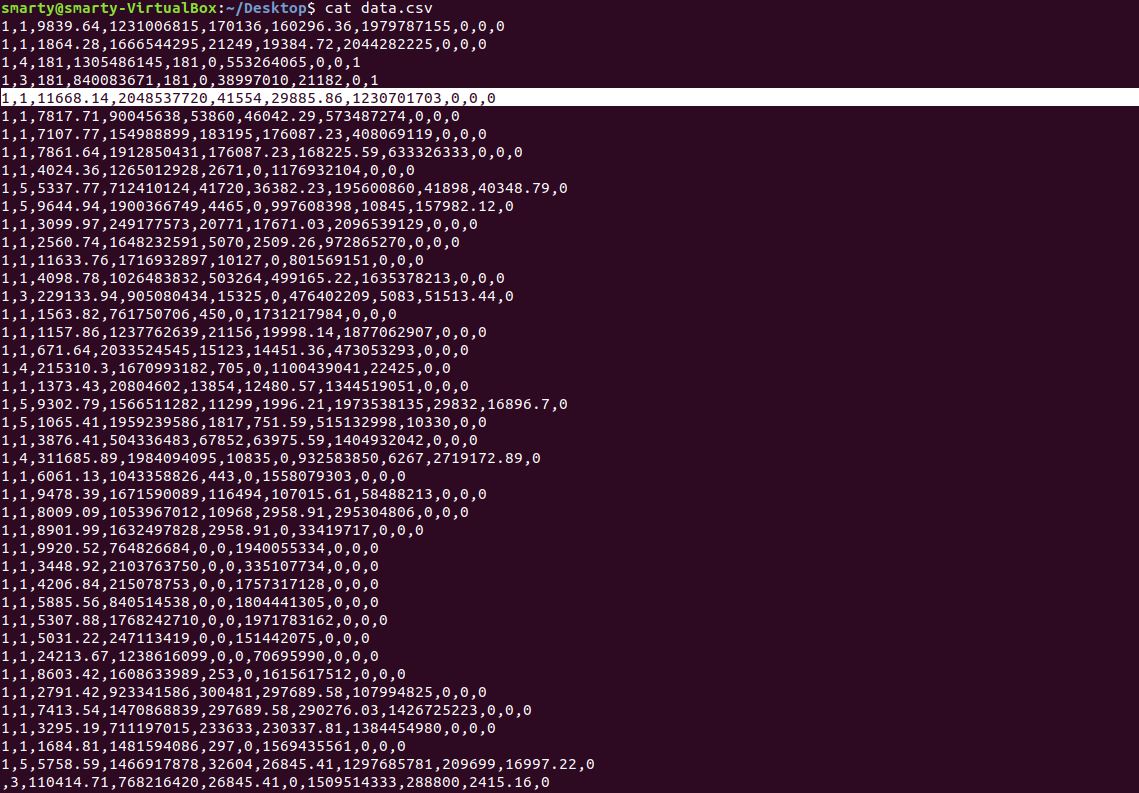
If you want to run two or more commands together, then you can do so using this pipeline symbol.

Like here I want to filter the Fifth row from data.csv, so for this I type the command “head –n 5 data.csv | tail –n 1”

Now “head –n 5 data.csv” will give the first five rows from the data.csv file and “tail –n 1” will give the last of those five rows i.e 5th row from the dataset.

C:\Users\smarty\Desktop\3\6.JPG

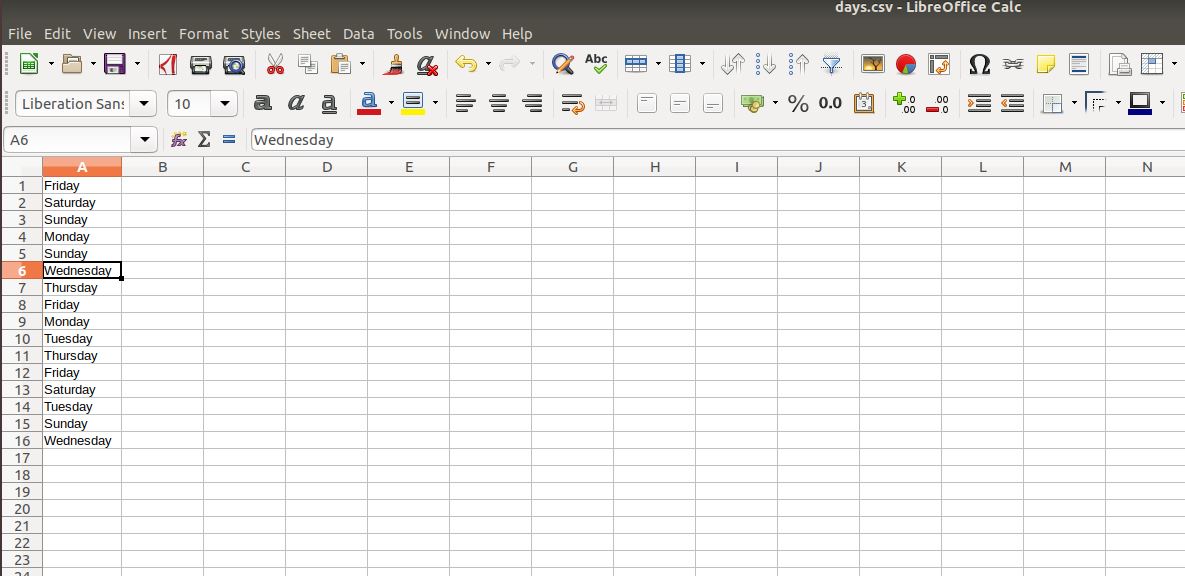
You can verify it from the actual dataset, like here the highlighted row is the 5th row of the dataset.



1. **sort –** This command is used to sort the content of the file.

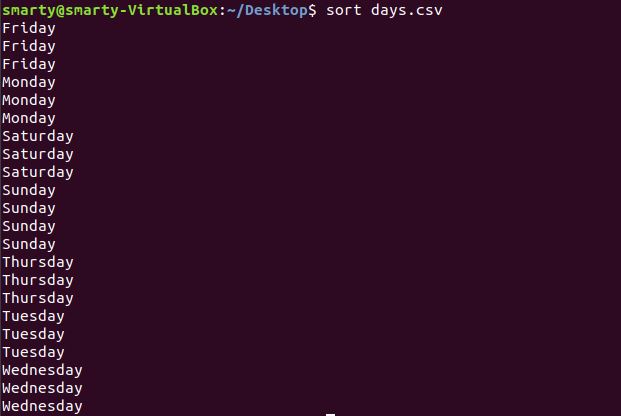
If you want to see the content of the file in a sorted order then you can use this command.

Like here I am having a file days.csv having weekdays in an unsorted manner.



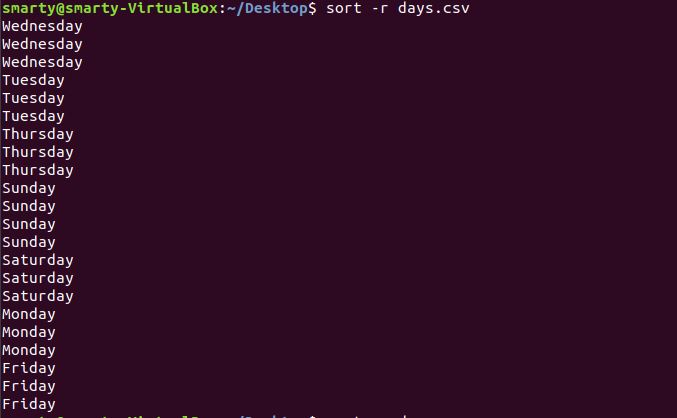
But now if you want to see it in sorted order then you can use this “sort <filename>” command.

It will sort the file as per the ascending order.



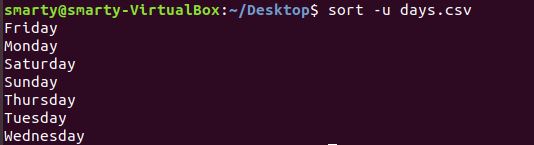
If you want to get the output in reverse order then you can update the command,

“sort -r days.csv“



Now what if you want to remove all the duplicate values, for you can use the command

“sort –u days.csv”



1. **wc –** This command will give the number of lines, no. of words, no. of bytes dataset is having.

If you want to know the no. of rows in a dataset then you can use this command “wc(word count)”.

Here I am having a dataset named data.csv and I want to get all the rows of it, so for this, I have used to command

wc - l data.csv

C:\Users\smarty\Desktop\3\14.JPG

Here it shows that the file is having 43 rows.

1. **echo –** It is used to print the content of a variable.

If you have stores some value in a variable and want to see that value, that you can simply use this command echo $<variable> and it will print the content.

Like here, as I have set the value of the variable day=Sunday, and when I use the command “echo day” it prints the value stored in it.

**C:\Users\smarty\Desktop\3\15.JPG**

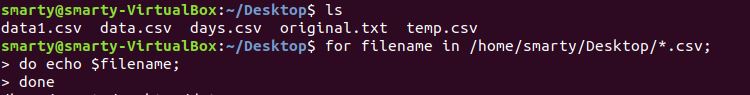
1. **For Loop –** If you want to repeat a command to ‘n’ number of times then you can use the concept of loops.

E.g. here I have used the for loop for all the files present on desktop like here there are 5 files present (4 .csv files and 1 .txt file). Now I make a for loop that prints all the csv filename’s present on the desktop.

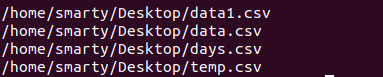
“for filename in /home/smarty/Desktop/\*.csv;

do echo $filename;

done”



Output:

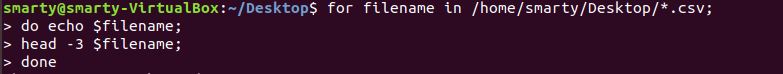


What if you want to run more than one command in a single loop?

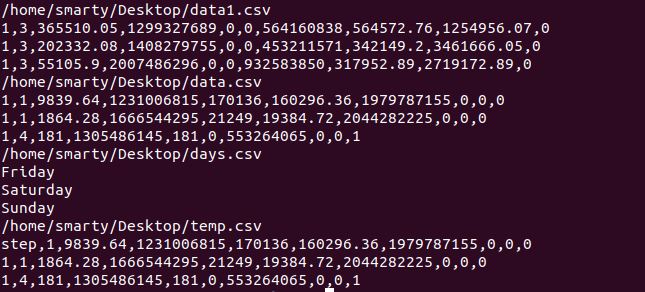
As here I have given two commands inside the for loop i.e.

“do echo $filename; (print the file name)

Head -3 $filename; (print the first three lines of the files)”



Output (here it first print the filename followed by its first three rows, and repeat the same process for all the files on the desktop)

****

created by Vidit