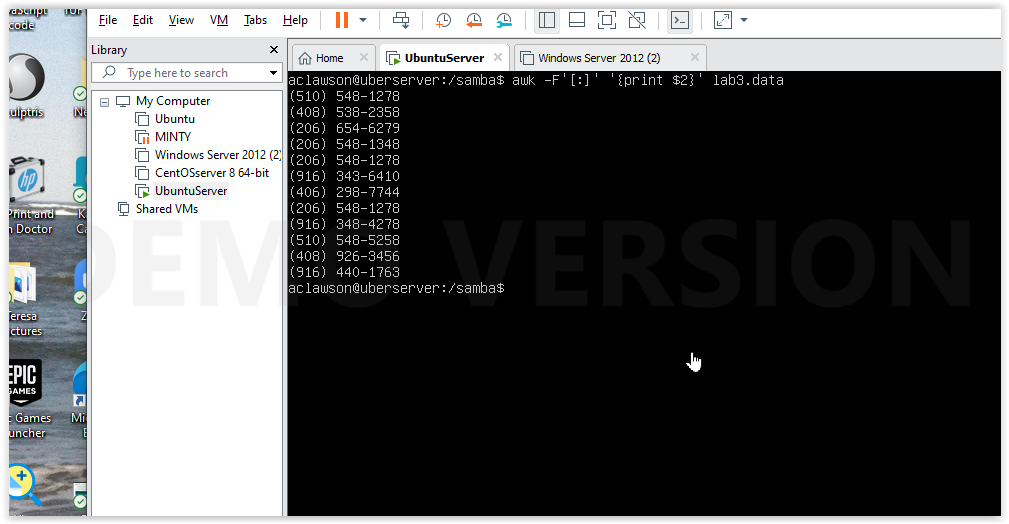
1. Print All phone number’s

Awk -F ‘[:]’ ‘{print $2}’ lab3.data # print variable number 2 which are the phone numbers



1. Print Dan 's phone number.

awk -F '[: ]' '/^Dan/{print $2}' lab3.data # print variable 2 and four on lines (caret) beginning with dan



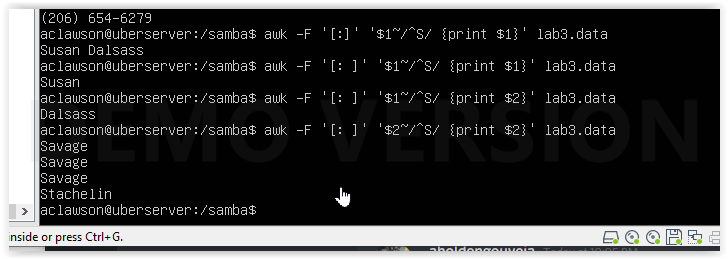
1. Print Susan 's name and phone number.

awk -F '[: ]' '/^Susan/{print $1, $2 }' lab3.data # print first two variables in lines beginning with Susan



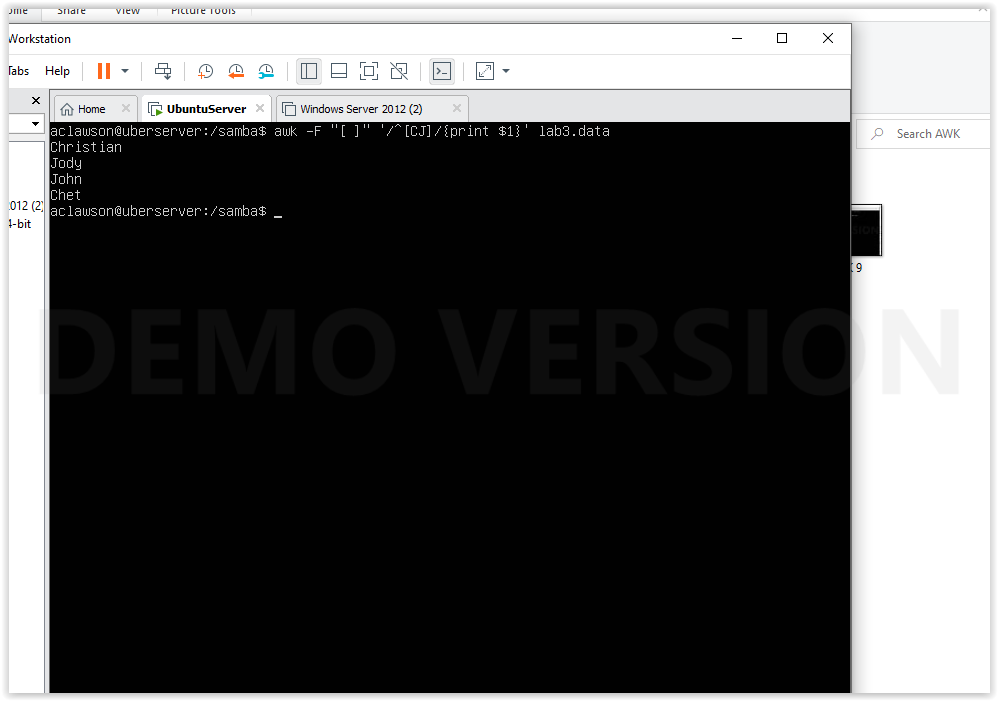
1. Print all last names beginning with S.

awk -F '[: ]' '$2~/^S/ {print $2}' lab3.data #print second variable that begins with S



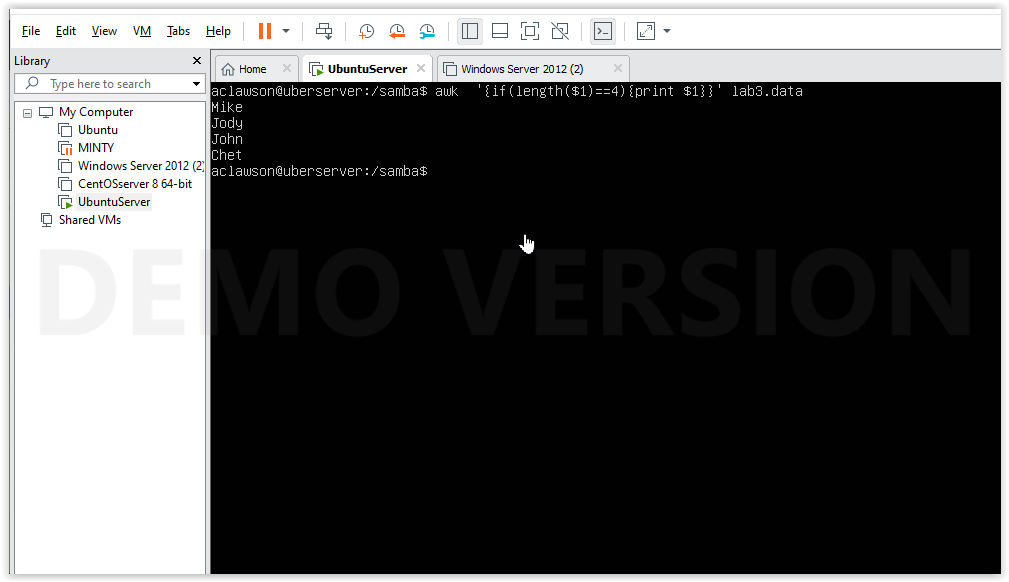
1. Print all first names beginning with either a C or J.

Awk -F “[ ]” '/^[CJ]/{print $1}' lab3.data #print first variable that begin with C or J



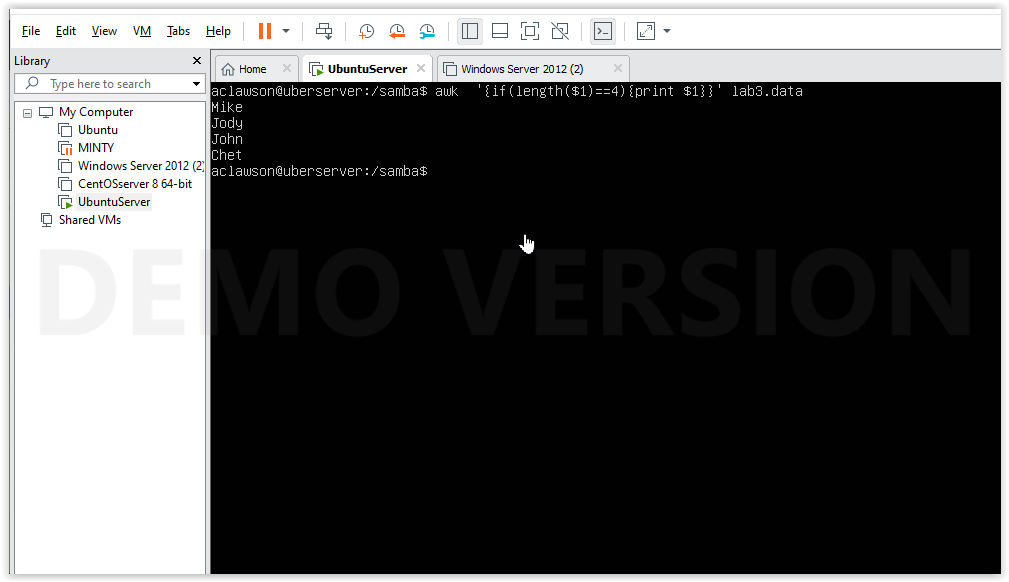
1. Print all first names containing only four characters .

awk ‘’{if(length($1)==4){print $1}}' lab3.data # if length of first variable is 4 characters then print first variable



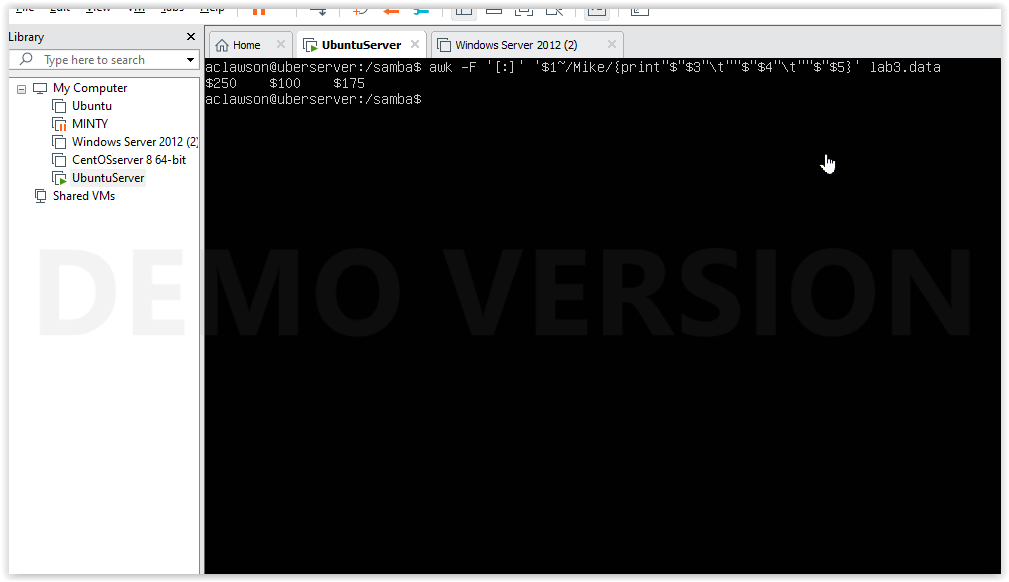
1. Print the first names of all those in the 916 area code.

awk -F '[: ]' '$2~/916/{print $1}' lab3.data # print first variables that have a value of 916 for second variable



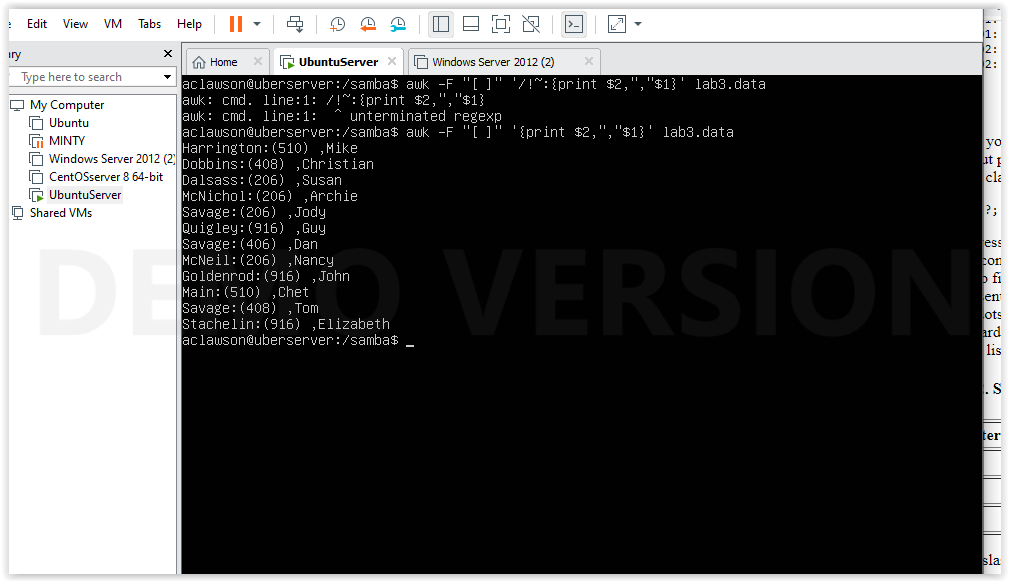
1. Print Mike 's campaign contributions. Each value should be printed with a leading dollar sign; e.g., $250 $100 $175.

awk -F '[: ]' '$1~/Mike/{print "$"$3”\t”"$"$4”\t”"$"$5}' lab3.data # if the first variable value is Mike print variables 3, 4 and 5 “$” adds the dollar sign to the beginning and the \t adds a tab



1. Print last names followed by a comma and the first name.

awk -F '[ ]' '{print $2 ”,” $1}' lab3.data #print variable 2 then a , then variable 1



1. Write an awk script called facts that

# !/bin/bash

# Facts Awk Script

Awk –F “[:]” ‘$1 ~/Savage/{print $1, $2}’ lab3.data | Awk –F “[:]” ‘$1~/Chet/{print $3, $4, $5}’ lab3.data | Awk –F “:” ‘$3 ~/50/{print $1}’ lab3.data

I gave permissions to facts.sh using $ sudo chmod +x facts.sh and then ran it with $ ./facts.sh

* 1. Prints full names and phone numbers for the Savages .

Awk –F “[:]” ‘$1 ~/Savage/ {print $1, $2}’lab3.data # variables with the value “Savage” will have variables 1 and 2

* 1. Prints Chet 's contributions.

Awk –F “[: ]” ‘$1~/Chet/ {print $3, $4, $5}’ lab3.data # first variables with the value “Chet” will have variables 3, 4 and 5 printed

* 1. Prints all those who contributed $50 the first month.

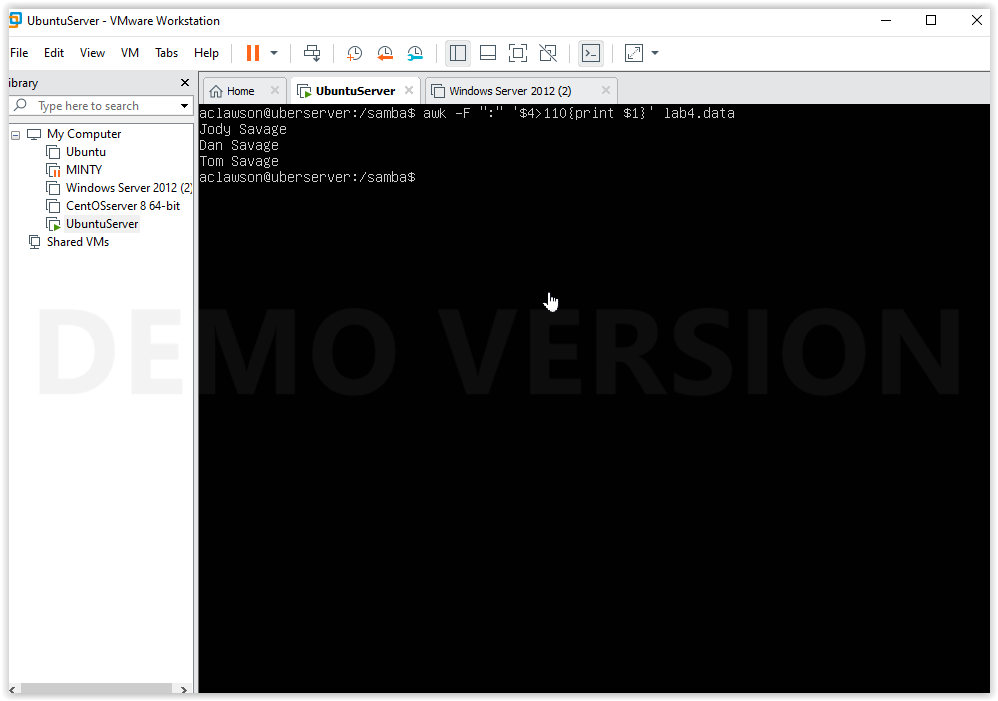
Awk –F “:” ‘$3 ~/50/{print $1}’ lab3.data

All of these commands work on their own but for some reason when I put them together part A and C work as the screenshot shows. I double checked part B by running the command solo and it works, I double checked to make sure the command was the same in my script and it is but part B is the only part that doesn’t work.



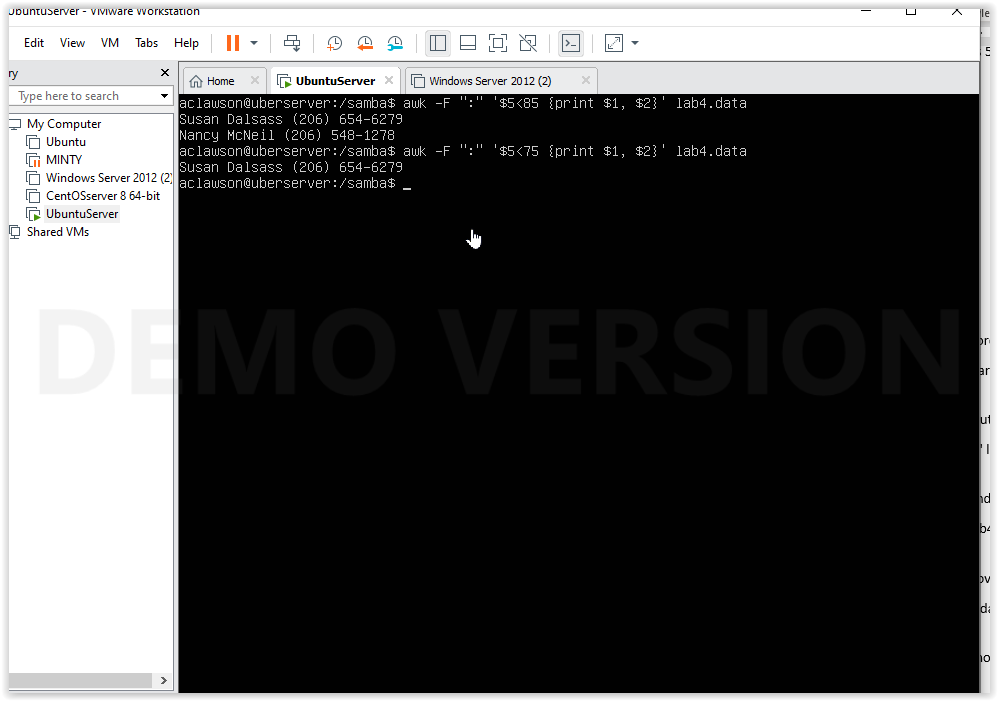
1. Print the first and last names of those who contributed more than $110 in the second month.

Awk –F “:” ‘$4 >110 {print $1}’ lab4.data #print variable 1 if variable 4 is greater than 110



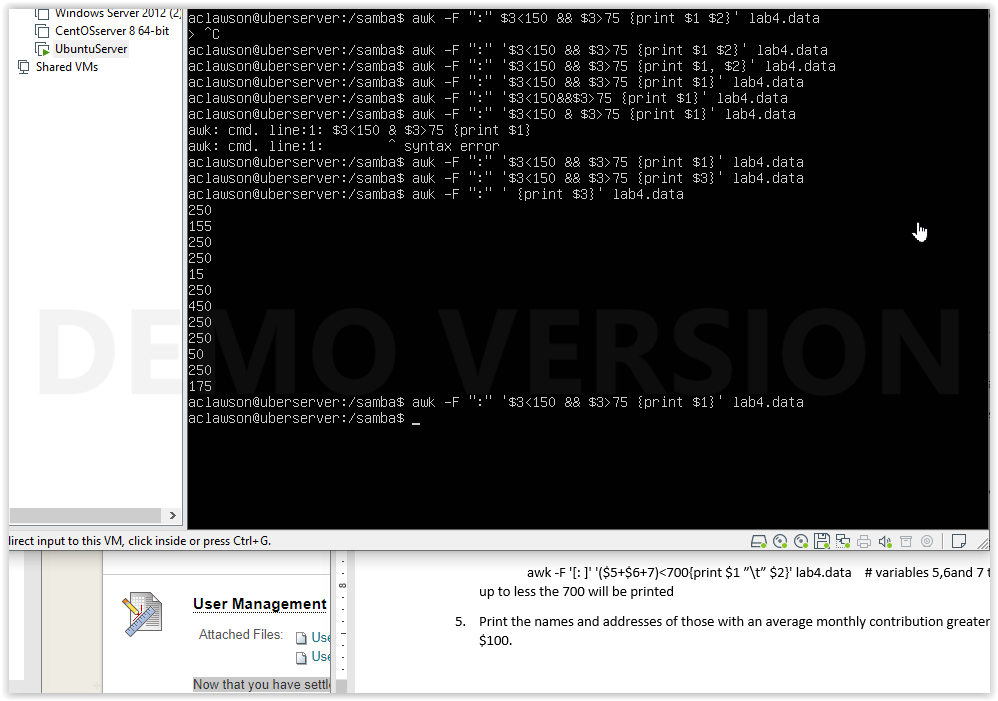
1. Print the names and phone numbers of those who contributed less than $75 in the last month.

Awk –F “:” ‘$5<75 {print $1, $2}’ lab4.data #print variable 5 if it is less than 75



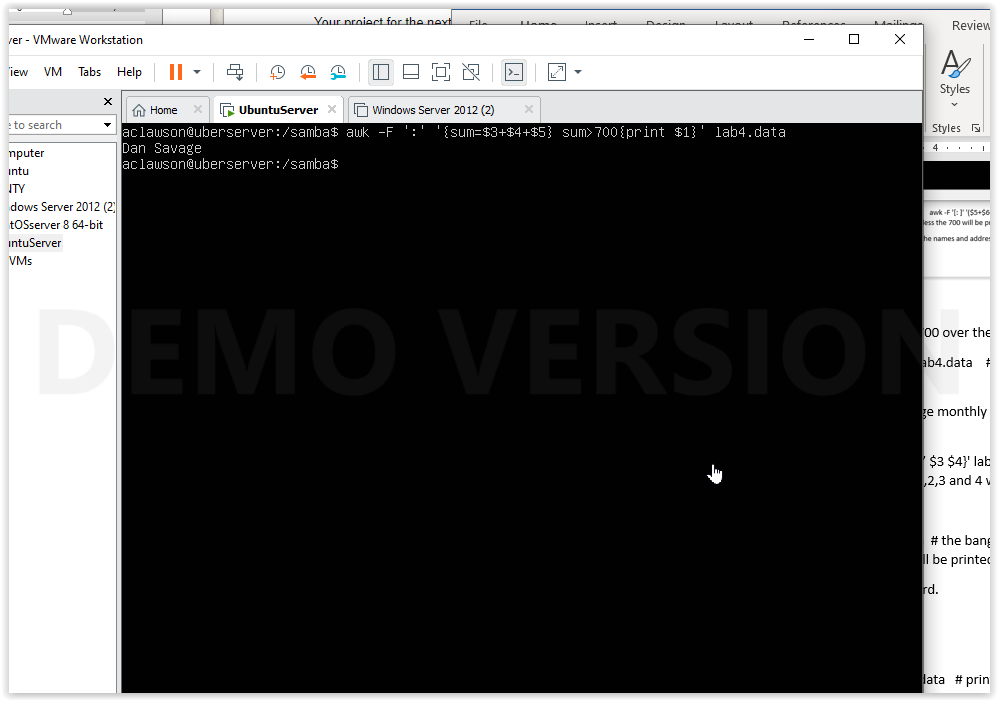
1. Print the names of those who contributed between $75 and $150 in the first month.

Awk –F “:” ‘$3<150 && $3>75 {print $1}’ lab4.data #print variable three if it falls between 75 and 150



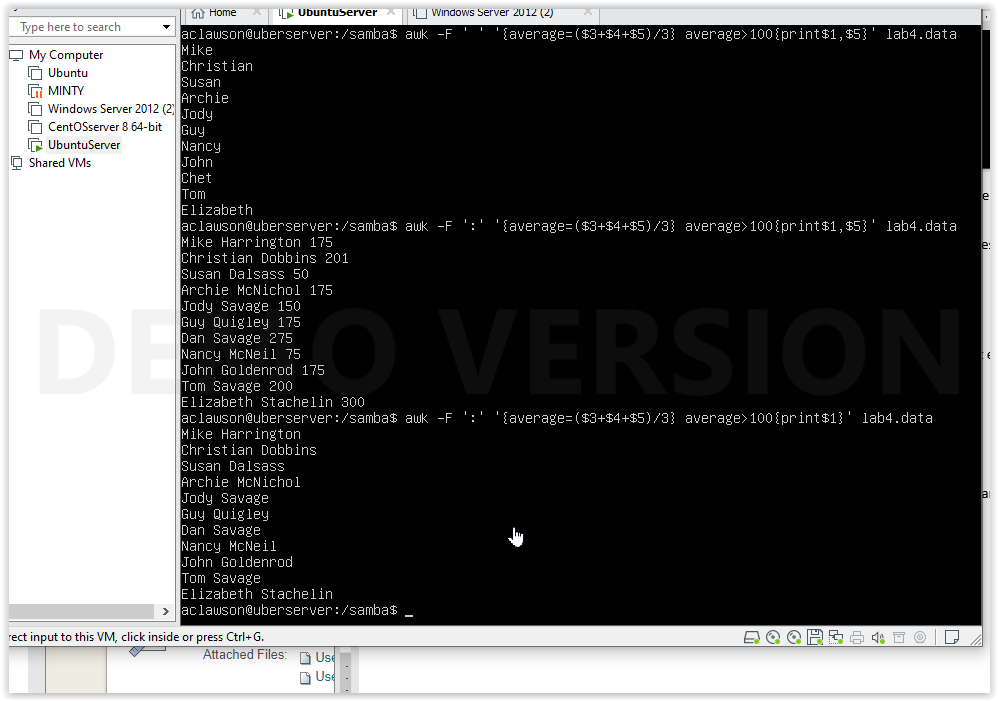
1. Print the names of those who contributed less than $700 over the three-month period

Awk –F “:” ‘{sum=$3+$4+$5} sum>800{print $1} ‘ lab4.data # variables 3,4and 5 that add up to less the 700 will be printed



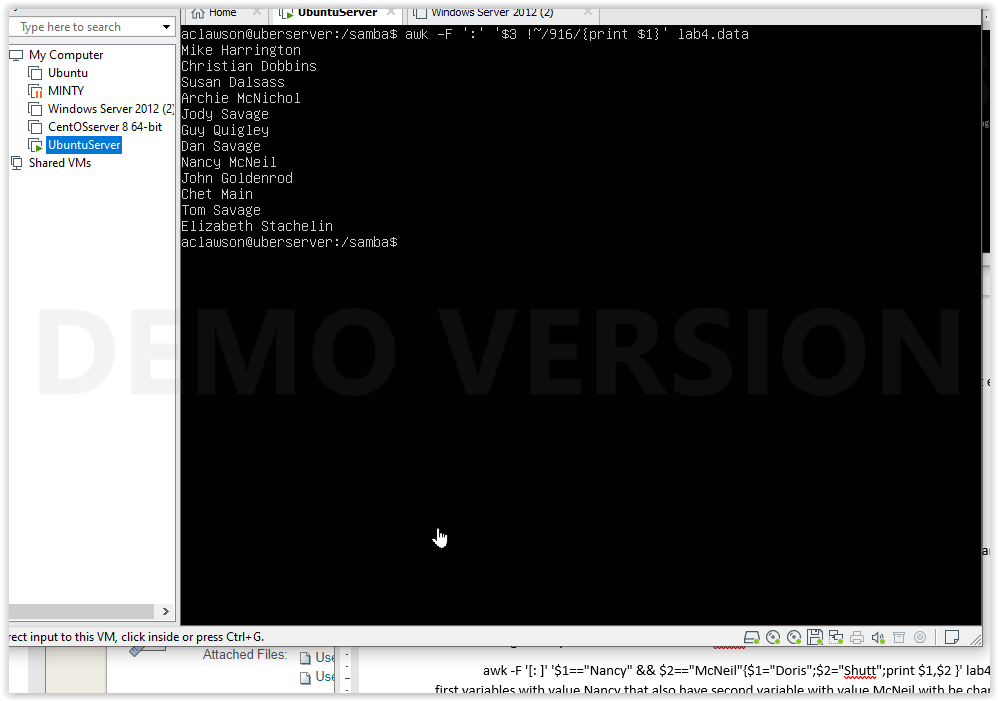
1. Print the names and addresses of those with an average monthly contribution greater than $100.

Awk –F “:” ‘{average=($3+$4+$5)/3} average>100{print $1} homework1.txt # if variables 3, 4 and 5 added together is greater than 100 variable1 will be printed could not find the addresses



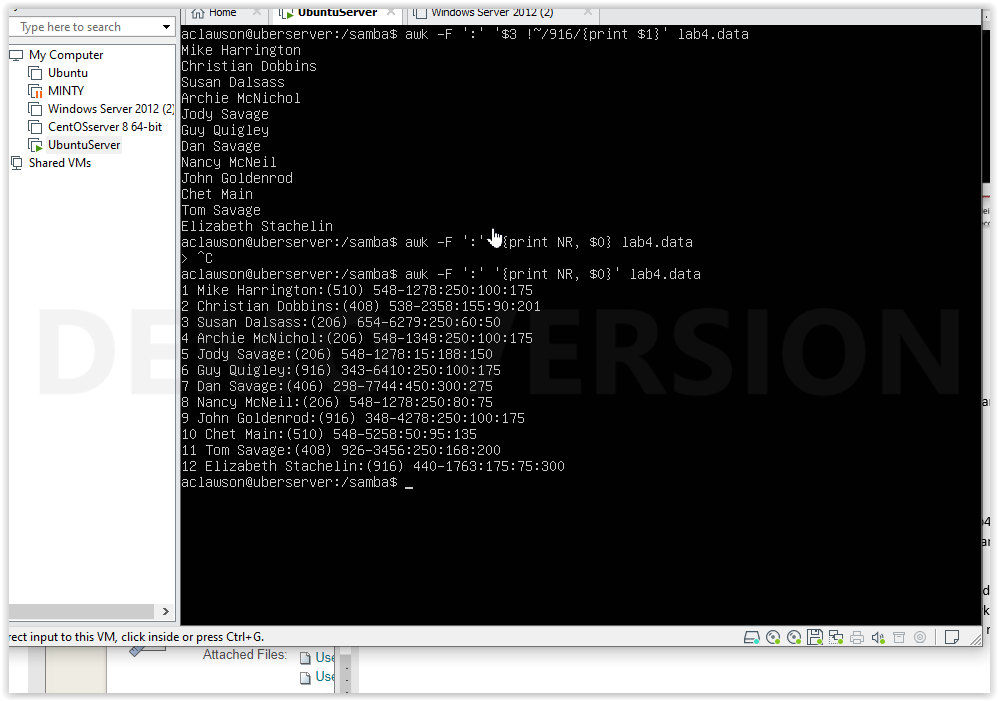
1. Print the first name of those not in the 916 area code.

Awk –F “[: ]” ‘$3 !~/916/{print $1} lab4.data # the bang makes the ~ not equal for variable 3, any first variable that meet that will be printed



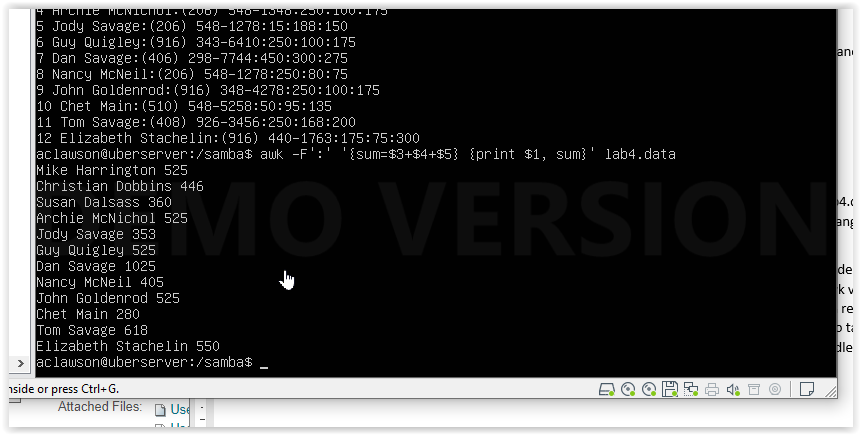
1. Print each record preceded by the number of the record.

awk '{print NR,$0}' lab4.data



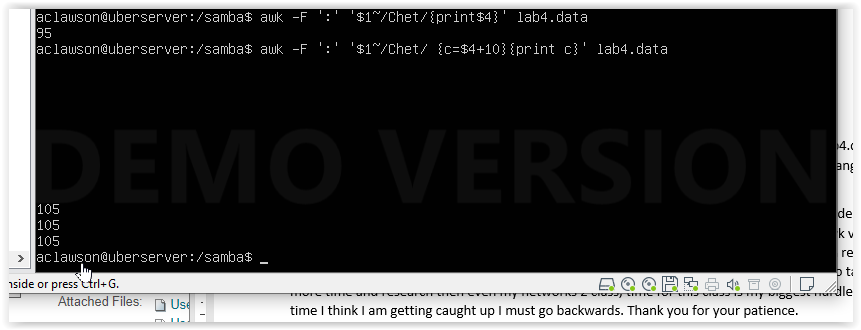
1. Print the name and total contribution of each person.

awk –F “:” ‘{sum=$3+$4+$5} {print $1, sum}’ lab4.data # print variables 1, and sum of $3 $4 and $6



1. Add $10 to Chet 's second contribution.

Awk –F “:” ‘$1 ~/Chet/ {c=$4+10} {print c}’ lab4.data



10. Change Nancy McNeil 's name to Doris Shutt.

awk ‘{gsub(/Nancy McNeil/, “Louise McInnes”);} {print $1}’ lab4.data 