



FACULTY OF ENGINEERING AND TECHNOLOGY
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

LINUX LABORATORY

ENCS3130

Project No.1: Shell project

Student's name: Sara Issa

Student's ID: 1190673

Teacher's name: Dr. Aziz Qaroush

Section No. 2

Assistant's name: Eng. Shahd Najeeb

Partner's name: Leen Abu Omar

Student's ID: 1190113

Teacher's name: Dr. Mohammad Jubran

Section No. 1

Date: 12.08.2021

Table of Contents:

Introduction about the project.....	2
Example.....	3
The allowed operations in the code	4
The code	7

Introduction about the project:

Initially, the user will be asked to enter the name of the file to read its data, provided that the file extension is **.csv**, also, the file must exist and we can read from it.

While all the previous conditions are true, the available transaction options will be displayed until the user chooses one of them, as the options are:

- D: for dimension.
- C: for computing statistics.
- S: for substitution.
- E: for exit.

NOTE: If an undefined process is entered, it will print an error message.

For Example:

In figure No.1, it shows that the file which the user entered doesn't exist:

```
saraissa@saraissa-VirtualBox:~/Desktop$ ./leen
please enter the name of the file
Book2.csv
Book2.csv doesn't exist
saraissa@saraissa-VirtualBox:~/Desktop$
```

In figure No.2, it shows that the file format which the user entered isn't correct, it must be .csv:

```
saraissa@saraissa-VirtualBox:~/Desktop$ touch book1.cdv
saraissa@saraissa-VirtualBox:~/Desktop$ ./leen
please enter the name of the file
book1.cdv
Error !!! The file format must be .csv !!!
```

In figure No.3, initially we used the command `ls -l` to show the file permission, then we used `chmod 333 filename` to make it unreadable. After that, we ran the shell script for the unreadable file, an error message was displayed as shown in the following figure:

```
saraissa@saraissa-VirtualBox:~/Desktop$ touch Book3.csv
saraissa@saraissa-VirtualBox:~/Desktop$ ls -l
total 36
-rw-rw-r-- 1 saraissa saraissa  0 Aug  7 07:11 book1.cdv
-rw-rw-r-- 1 saraissa saraissa 126 Aug  7 04:05 Book1.csv
-rw-rw-r-- 1 saraissa saraissa  0 Aug  7 07:14 Book3.csv
-rw-rw-r-- 1 saraissa saraissa 115 Aug  1 03:24 'c -l d'
-rwxrwxrwx 1 saraissa saraissa 336 Aug  1 02:33 covid.txt
-rwxrwxr-x 1 saraissa saraissa 155 Aug  1 00:20 d
-rwxrwxr-x 1 saraissa saraissa 1404 Aug  7 07:11 leen
-rwxrwxr-x 1 saraissa saraissa  37 Aug  1 02:55 sc
-rw-rw-r-- 1 saraissa saraissa  17 Aug  1 02:47 size
-rw-rw-r-- 1 saraissa saraissa  0 Aug  1 01:11 test1.txt
-rwx----- 1 saraissa saraissa 115 Aug  1 01:34 test.txt
-rwxrwxr-x 1 saraissa saraissa 955 Aug  1 06:38 TODO
saraissa@saraissa-VirtualBox:~/Desktop$ chmod 333 Book3.csv
saraissa@saraissa-VirtualBox:~/Desktop$ ./leen
please enter the name of the file
Book3.csv
Book3.csv isn't readable
```

Example:

Initially we tried the example which is given in the project, as shown in the following figure:

	A	B	C	D	E	F	G
1	sepal.length	sepal.width	petal.length	petal.width			
2	5.1	3.5	1.4	0.2			
3	4.9	3	1.4	0.2			
4	4.7	3.2	1.3	0.2			
5	4.6	3.1	1.5	0.2			
6	5	3.6	1.4	0.2			
7							
8							
9							
10							
11							
12							
13							
14							
15							
16							

Note: By using the command `pico Book1.csv`, the file was opened and it appeared that the values are separated between each column and another with a comma, because the file extension is .csv

```
GNU nano 4.8 Book1.csv
sepal.length,sepal.width,petal.length,petal.width
5.1,3.5,1.4,0.2
4.9,3,1.4,0.2
4.7,3.2,1.3,0.2
4.6,3.1,1.5,0.2
5,3.6,1.4,0.2
```

The allowed operations in the code:

- **Dimension:** This operation is performed by clicking on the “D” button on the keyboard, which displays the number of rows and columns, provided that the first row -which represents the name of the column- is ignored.

Here is the result of clicking “D” on the screen:

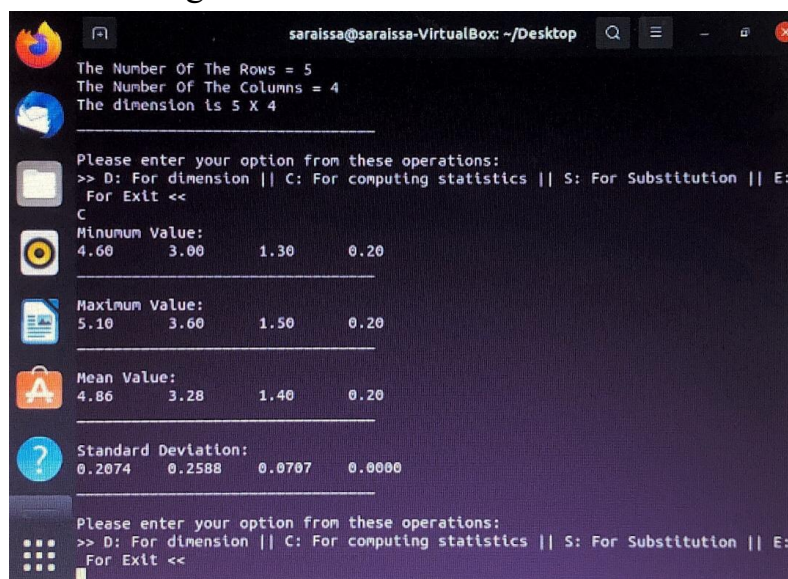
```
saraissa@saraissa-VirtualBox: ~/Desktop$ ./leen
please enter the name of the file
Book1.csv

Please enter your option from these operations:
>> D: For dimension || C: For computing statistics || S: For Substitution || E:
For Exit <<
D
The Number Of The Rows = 5
The Number Of The Columns = 4
The dimension is 5 X 4

Please enter your option from these operations:
>> D: For dimension || C: For computing statistics || S: For Substitution || E:
For Exit <<
█
```

- **Computing Statistics:** This operation allows user to compute the minimum, maximum, mean and standard deviation values of each column. This operation is performed by clicking on the “C” button on the keyboard.

Here is the result of clicking “C” on the screen:



```
saraissa@saraissa-VirtualBox: ~/Desktop
The Number Of The Rows = 5
The Number Of The Columns = 4
The dimension is 5 X 4

Please enter your option from these operations:
>> D: For dimension || C: For computing statistics || S: For Substitution || E:
For Exit <<
C
Minimum Value:
4.60      3.00      1.30      0.20

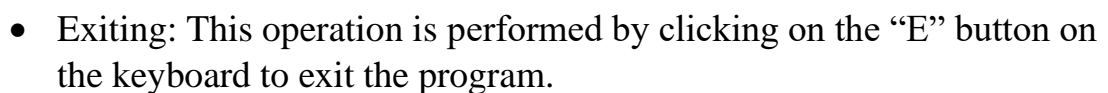
Maximum Value:
5.10      3.60      1.50      0.20

Mean Value:
4.86      3.28      1.40      0.20

Standard Deviation:
0.2074    0.2588    0.0707    0.0000

Please enter your option from these operations:
>> D: For dimension || C: For computing statistics || S: For Substitution || E:
For Exit <<
█
```


- Here is the result of clicking “S” on the screen:



- When the user enters another option, an error message will appear as shown in the following figure:

```
saraissa@saraissa-VirtualBox: ~/Desktop
saraissa@saraissa-VirtualBox:~/Desktop$ ./leen
please enter the name of the file
Book1.csv

Please enter your option from these operations:
>> D: For dimension || C: For computing statistics || S: For Substitution || E:
For Exit <<
F
Sorry !! The operation F is invalid !!

Please enter your option from these operations:
>> D: For dimension || C: For computing statistics || S: For Substitution || E:
For Exit <<

```

As required in the project, we have to test at least two examples, so we used the following example:

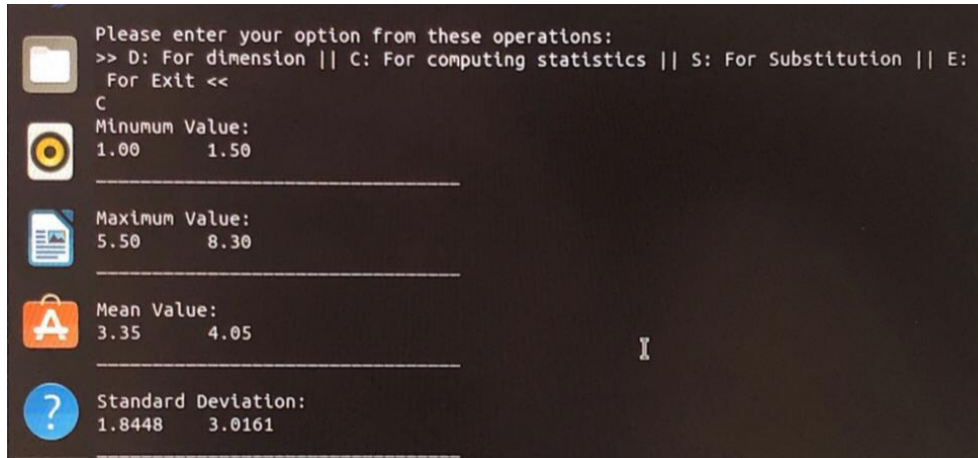
ex2.csv - LibreOffice Calc

	A	B	C	D	E	F	G	H
1	sepal.w	sepal.l						
2	1	4						
3	5.5	2.4						
4	3.6	8.3						
5	3.3	1.5						
6								

Then we tried all the operations, and the results were:

```
saraissa@saraissa-VirtualBox:~/Desktop$ ./leen
please enter the name of the file
ex2.csv

Please enter your option from these operations:
>> D: For dimension || C: For computing statistics || S: For Substitution || E:
For Exit <<
D
The Number Of The Rows = 4
The Number Of The Columns = 2
The dimension is 4 X 2
```



Here is the code we wrote in shell script:

```
#!/bin/bash
```

```
#an automatic message appears to ask the user to enter the file name for checking it
```

```
echo " please enter the name of the file"
```

```
read file #read command here is used to read the content of a line into a variable
```

```
#if statement to check if the entered file exists or not, if not, an error message will  
appear and then will exit the program
```

```
if [ ! -e $file ]
```

```
then
```

```
echo " $file doesn't exist"
```

```
exit 0 #to exit
```

```
fi #end of if statement
```

```
#if statement to check if the entered file is readable or not, if not, an error message  
will appear and then will exit the program
```

```
if [ ! -r $file ]
```

```
then
```

```
echo " $file isn't readable "
```



```
exit 0 #to exit
```

```
fi #end of if statement
```

```
#if statement to check if the format of the entered file is .csv or not, if not, an error message will appear and then will exit the program
```

```
if ! [[ "${file: -4}" == ".csv" ]] #the last four characters of the file name must be = .csv
```

```
then
```

```
echo "Error !!! The file format must be .csv!!!"
```

```
exit 0 #to exit
```

```
fi #end of if statement
```

```
#infinite loop only exits if the user enters "E", infinite loop to allow the user to choose all other possible operations
```

```
while [ 1 ]
```

```
do
```

```
echo "_____"
```

```
echo " "
```

```
#Menu to ask user to enter the operation needed
```

```
echo "Please enter your option from these operations: "
```

```
echo ">> D: For dimension || C: For computing statistics || S: For Substitution || E: For Exit <<"
```

```
read option #to read the option which the user enters
```

```
case $option in #case statement takes the option entered and checks it.
```

```
"D") #the first operation is the dimensions of the data
```

```
NumOfRows=$(cat $file | wc -l ) #to count the number of rows without the first row which implements the properties of each column
```

```
echo "The Number Of The Rows = $((NumOfRows - 1))" #to print the number of rows
```

```
NumOfColumns=$( head -n1 $file | grep -o "," | wc -l ) #to count the number of columns
```

```
echo "The Number Of The Columns = $((NumOfColumns + 1))" #to print the number of columns
```

```
echo "The dimension is $((NumOfRows - 1)) X $((NumOfColumns + 1))" ;; #to print the dimension
```

```
"C") #the second operation is to compute the max, min, mean and standard deviation values
```

```
NumOfRows=$(( $(cat $file | wc -l ) - 1 ))
```

```
NumOfColumns=$(( $( head -n1 $file | grep -o "," | wc -l ) + 1 ))
```

```
echo "Minimun Value: "
```

```
for i in $(seq 1 $NumOfColumns); #for loop to find the minimum value for each column starting with column 1 and ending with the number of columns
```

```
do
```

```
Min=`cut -d',' -f$i $file | grep -v '[A-Za-z]' | sort -n | head -n1` #to calculate the minimum value for the $i column
```

```
printf "%-10.2f" $Min #to print the minimum value of the column horizontally
```

```
done
```

```
echo " "
```

```
echo " _____ "  
echo " "
```

```
echo "Maximum Value: "
```

```
for i in $(seq 1 $NumOfColumns); #for loop to find the maximum value for each  
column starting with column 1 and ending with the number of columns
```

```
do
```

```
Max=`cut -d',' -f$i $file | grep -v '[A-Za-z]' | sort -n | tail -n1` #to calculate the  
maximum value for the $i column
```

```
printf "%-10.2f" $Max #to print the maximum value of the column horizontally
```

```
done
```

```
echo " "
```

```
echo " _____ "
```

```
echo " "
```

```
echo "Mean Value:"
```

```
for i in $(seq 1 $NumOfColumns); #for loop to find the mean value for each  
column starting with column 1 and ending with the number of columns
```

```
do
```

```
Sum=$(cut -d',' -f$i $file | awk '{total += $i } END { print total }' ) #to calculate  
the summation of all values for each column
```

```
Mean=`echo $Sum/$NumOfRows|bc -l` #to calculate the average for each column
```

```
printf "%-10.2f" $Mean #to print the mean value of the column horizontally
```

```
echo $Mean >> MeanFile
```

```
done
```

```
echo " "
```

```
echo " _____ "
```

```
echo " "
```

```
echo "Standard deviation:"
```

```
for i in $(seq 1 $NumOfColumns); #for loop to find the standard deviation value  
for each column starting with column 1 and ending with the number of columns  
do
```

```
Sum=$(cut -d',' -f$i $file | awk '{total += $i } END { print total }' ) #to calculate  
the summation of all values for each column
```

```
Mean=`echo $Sum/$NumOfRows|bc -l` #to calculate the average for each column  
to use them for finding the standard deviation value
```

```
STDEV=$(awk -F ',' -v result=0 'NR!=1 {result+=(($"i" - "$Mean")^2)}  
END{print sqrt(result/(NR-2))}' $file) #to calculate the standard deviation value  
for each column
```

```
printf "%-10.4f" $STDEV #to print the standard deviation value of the column  
horizontantly
```

```
done
```

```
echo " "
```

```
;;
```

```
"S") #the third operation is to substitute missing values by the mean of its column
```

```
NumOfRows=$(( $(cat $file | wc -l ) -1 ))
```

```
NumOfColumns=$(( $( head -n1 $file | grep -o ',' | wc -l ) + 1 ))
```

```
cat $file | sed '1d' > copyfile
```

```
i=1
```

```
for i in $(seq 1 $NumOfColumns);
```

```
do
```

```

cut -d',' -f$i copyfile > ColumnFile$i
done

#copying the MeanFile data into mean
cp MeanFile mean
for i in $(seq 1 $NumOfColumns);
do
sed 's/^\$/$(head -1 mean)"/g' ColumnFile$i > temp
mv temp ColumnFile$i
sed '1d' mean > temp
mv temp mean
done
for x in $(seq 1 $NumOfRows);
do
for y in $(seq 1 $NumOfColumns);
do
printf "%s\t" $(cat ColumnFile$y | head -1 )
sed '1d' ColumnFile$y > temp
mv temp ColumnFile$y
done
printf "\n"
done
;;

"E") #the last operation to exit the program

```



```
break ;;
```

*) #when user enters any other operation, this error message will appear

```
echo " Sorry !! The operation $option is invalid !!"
```

```
;;
```

```
esac
```

```
done
```