To-do #1

Shell script

Islam 1191375

Ibrahim 1190747

The idea of our project is to calculate every Colom alone by splitting them into separated files and calculate the min, max, mean, standard deviation for every single file. Then substitute the mean in the empty cells and after this to merge the files again together in the mother file.



The file dose not exist error



Wrong extinction error

 

Format is wrong error

 

Number of cells not equal error

 

All the options of the shell script showing the dimension of the file data, then the max, min, mean and standard deviation of every Colom alone in a table. Then the substitution option which change the empty cell in every Colom with the Colom mean value. Then the exit option which exit the script and delete the files that used to calculate everything

Another example

 

#!/bin/bash

echo "enter the file name with it's extintion"

read file

if [ ! -f "$file" ]

then

echo "file dose not exist"

exit 1

fi

if [ ! $( echo $file | grep '\.csv$' ) ]

then

echo "wrong file format (the extintion is not .csv)"

exit 1

fi

rows=$(cat $file | wc -l)

rows=$(( rows - 1 ))

#calculate the number of coloms by calcuate the first line comas

coloms=$(sed -n '1p' $file | awk -F, '{print NF-1}')

coloms=$(( coloms + 1 ))

#send the first line string into another file

touch titels

touch nums

cat $file | head -1 > titels

#send the numbers into file

cat $file | tail -$rows > nums

mv nums $file

while IFS= read line

do

#erorr messege if the file contain somtheng other than numbers or commas

if [ $(echo "$line" | grep [A-Za-z?@#+'&'\*$%^/!\-]) ]

then

echo file format is wrong fix it please

exit 1

fi

lineCo=$(echo $line | awk -F, '{print NF-1}')

lineCo=$(( lineCo + 1 ))

if [ $lineCo -ne $coloms ]

then

echo "Error: number of cells are not equal in all rows."

exit 1

fi

done < "$file"

for (( i=1; i<=$coloms; i++ ))

do

touch $i

done

#seperate every colom into a file

while IFS= read line

do

for (( i=1; i<=$coloms; i++ ))

do

num=$(echo $line | cut -d ',' -f$i)

echo $num >> $i

done

done < "$file"

#infinte loop to repeat the orders

while [ true ]

do

echo "please chose the character of the action below"

echo " D : for Dimension"

echo " C : for computing statistics"

echo " S : for subsitution"

echo " X : for exit"

read input

#the dimention option

if [ "$input" = D -o "$input" = d ]

then

echo "the dimention is $rows X $coloms"

#the calculate statistics option max, min, mean, sdev

elif [ "$input" = C -o "$input" = c ]

then

printf "max "

for (( f=1; f<=$coloms; f++ ))

do

#it grep the lines and sort them then take the first line as the greatest number

max=$(cat "$f" | grep ^[-0-9] | sort -gr | head -1)

printf "%f " $max

done

echo

printf "min "

for (( f=1; f<=$coloms; f++ ))

do

#it grep the lines and sort them in reverse order then take the first line as the minimum number

min=$(cat "$f" | grep ^[-0-9] | sort -g | head -1)

printf "%f " $min

done

echo

#print the mean after calculate it for every file

printf "mean "

for (( f=1; f<=$coloms; f++ ))

do

mean=0

sum=0

count=$(cat $f | wc -l)

while IFS= read line

do

if [ -z "$line" ]

then

count=$(( count - 1 ))

elif [ $(echo $line | grep '[0-9]') ]

then

sum=$(calc $line + $sum )

fi

done < "$f"

#take the first4 digits after the point

mean=$(echo "scale=4; $sum / $count" | bc)

printf "%f " $mean

done

echo

# calculte the stander diveation

printf "sdev "

for (( f=1; f<=$coloms; f++ ))

do

mean=0

sum=0

sd=0

sum2=0

# ignore the empty lines

count=$(cat $f | grep '[0-9]' | wc -l)

while IFS= read line

do

sum=$(calc $line + $sum )

done < "$f"

mean=$(calc $sum / $count)

while IFS= read line

do

lm=$(calc $line - $mean)

lm2=$( echo $lm \\* $lm | bc )

sum2=$(calc $sum2 + $lm2)

done < "$f"

bf=$(echo "scale=2; $sum2 / $count" | bc)

sd=$(echo "$bf" | awk '{print sqrt($1)}')

printf "%f " $sd

done

echo

elif [ "$input" = S -o "$input" = s ]

then

for (( i=1; i<=$coloms; i++ ))

do

mean=0

sum=0

count=$(cat $i | wc -l)

while IFS= read line

do

if [ -z "$line" ]

then

count=$(( count - 1 ))

else

sum=$(calc $line + $sum )

fi

done < "$i"

mean=$(echo "scale=2; $sum / $count" | bc)

while IFS= read line

do

#subsitute the mean in the empty cells

if [ -z $line ]

then

echo $mean >> temp

else

echo $line >> temp

fi

done < "$i"

mv temp $i

done

#marege the seperated files to the mother file again

x=1

for (( i=2; i<=$coloms; i++ ))

do

x="$x $i"

done

paste -d ',' $x > $file

cat "$file" >> titels

mv titels "$file"

elif [ "$input" = X -o "$input" = x ]

then

echo thank you

#delete the used files for calculations

x=1

for (( i=2; i<=$coloms; i++ ))

do

x="$x $i"

done

rm $x

exit 1

else

echo wrong character try again

fi

done