/\* Problem 7.10 #3

/\*Using the Sales data set, list the observations for employee numbers

(EmpID) 9888 and 0177. Do this two ways, one using OR operators and the

other using the IN operator. Note: EmpID is a character variable.\*/

/\*Nilupul Rathgama for BAN500\*/

/\*Problem 7.10 #3\* WITH "OR"/

FILENAME REFFILE '/home/nilunra0/my\_courses/sue.mcdaniel/Wks 5\_6/Sales.xls';

/\* access to Sales file to the program \*/

**PROC** **IMPORT** DATAFILE=REFFILE /\* Importing the Sales file \*/

DBMS=XLS /\* Determining what type of file is being used \*/

OUT=WORK.IMPORT1 REPLACE; /\* Naming Said referenced file for SAS \*/

GETNAMES=YES; /\* Using the Xls names as file type \*/

**RUN**;

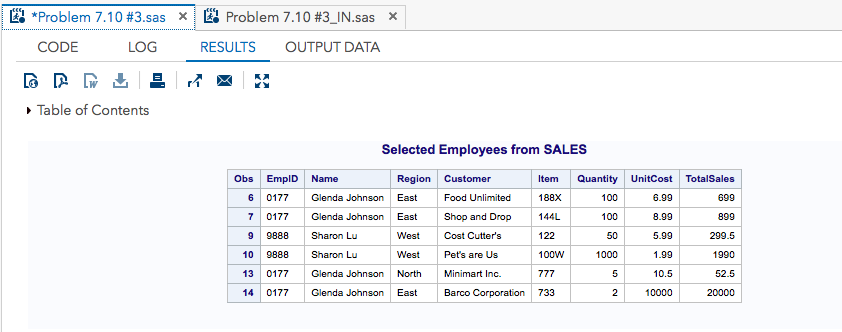
**proc** **print** data=WORK.IMPORT1; /\*Selecting Variables to Print\*/

title "Selected Employees from SALES";

WHERE EmpID = '9888' OR EmpID = '0177';

/\*This code writes the data structure to the log\*/

**run**;



/\*Nilupul Rathgama for BAN500\*/

/\*Problem 7.10 #3\* WITH "IN"/

FILENAME REFILE '/home/nilunra0/my\_courses/sue.mcdaniel/Wks 5\_6/Sales.xls';

/\* access to Sales file to the program \*/

**PROC** **IMPORT** DATAFILE=REFILE /\* Importing the Sales file \*/

DBMS=XLS /\* Determining what type of file is being used \*/

OUT=WORK.IMPORT1 REPLACE; /\* Naming Said referenced file for SAS \*/

GETNAMES=YES; /\* Using the Xls names as file type \*/

**RUN**;

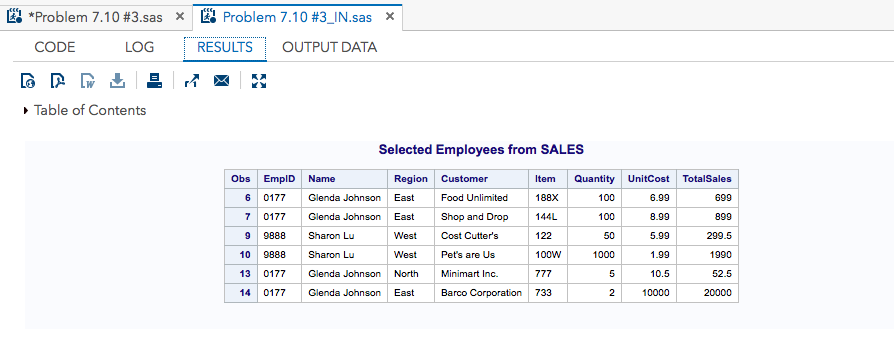
**proc** **print** data=WORK.IMPORT1; /\*Selecting Variables to Print\*/

title "Selected Employees from SALES";

WHERE EmpID IN ('9888','0177');

/\*This code writes the data structure to the log\*/

**run**;



/\*Problem 7.10 #4. Using the Sales data set, create a new, temporary SAS data set

containing Region and TotalSales plus a new variable called Weight

with values of 1.5 for the North Region, 1.7 for the South Region, and 2.0

for the West and East Regions. Use a SELECT statement to do this\*/

/\*Nilupul Rathgama for BAN500\*/

/\*Problem 7.10 #4\*/

FILENAME SALES '/home/nilunra0/my\_courses/sue.mcdaniel/Wks 5\_6/Sales.xls';

/\* access to Sales file to the program \*/

**PROC** **IMPORT** DATAFILE=Sales/\* Importing the Sales file \*/

DBMS=XLS /\* Determining what type of file is being used \*/

OUT=work.import1; /\* Naming Said referenced file for SAS \*/

GETNAMES=YES; /\* Using the Xls names as file type \*/

**RUN**;

**DATA** sales (keep = Region TotalSales weight);

/\*the DATA step automatically names each successive data set \*/

set work.import1; /\*SET stmt reads an observation from an existing SAS data set\*/

select;/\*helps conditionally execute stmt based on the value

of a single categorical variable\*/

when (Region = 'North') Weight = **1.5**;

when (Region = 'South') Weight = **1.7**;

when (Region = 'East') Weight = **2.0**;

when (Region = 'West') Weight = **2.0**;

otherwise;

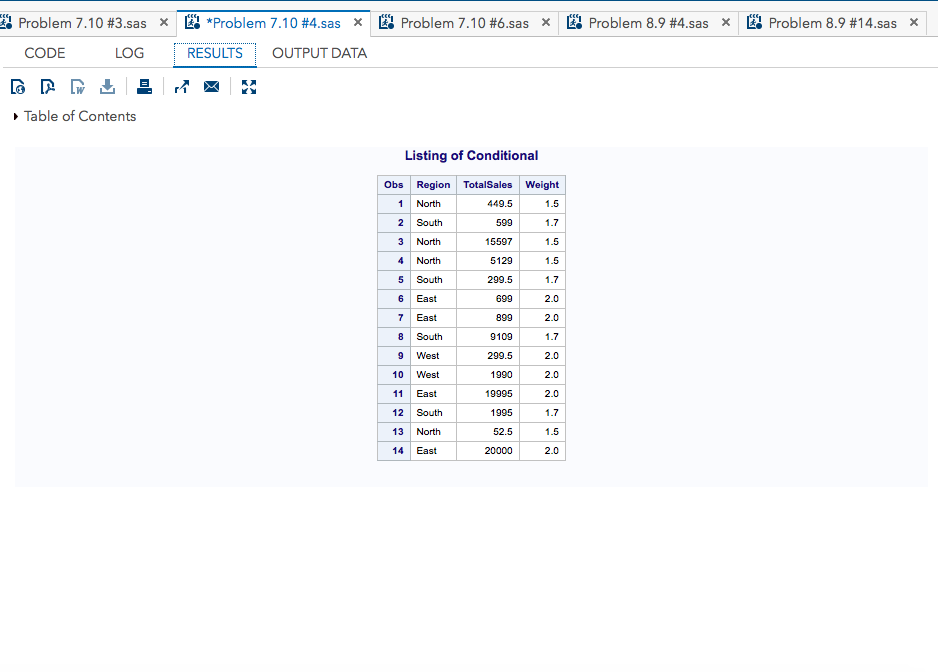
end;

**run**;

**proc** **print** data=Sales; /\*Selecting Variables to Print\*/

title 'Listing of Conditional';

**run**;



/\*Problem 7.10 #6. Using the Sales data set, list all the observations observations

where Region is North and Quantity is less than 60. Include in this

list any observations where the customer name (Customer) is Pet’s are Us.\*/

/\*Nilupul Rathgama for BAN500\*/

/\*Problem 7.10 #6\*/

FILENAME SALES '/home/nilunra0/my\_courses/sue.mcdaniel/Wks 5\_6/Sales.xls';

/\* access to Sales file to the program \*/

**PROC** **IMPORT** DATAFILE=SALES /\* Importing the Sales file \*/

DBMS=XLS /\* Determining what type of file is being used \*/

OUT=WORK.IMPORT1; /\* Naming Said referenced file for SAS \*/

GETNAMES=YES; /\* Using the Xls names as file type \*/

**RUN**;

**data** WORK.IMPORT1;

/\*the DATA step automatically names each successive data set \*/

set WORK.IMPORT1 (keep = Region Quantity Customer);

/\*SET stmt reads an observation from an existing SAS data set\*/

select; /\*helps conditionally execute stmt based on the value

of a single categorical variable\*/

where (Region eq "North" and Quantity le **60**) and Customer eq "Pet's are Us";

/\*This code writes the data structure to the log\*/

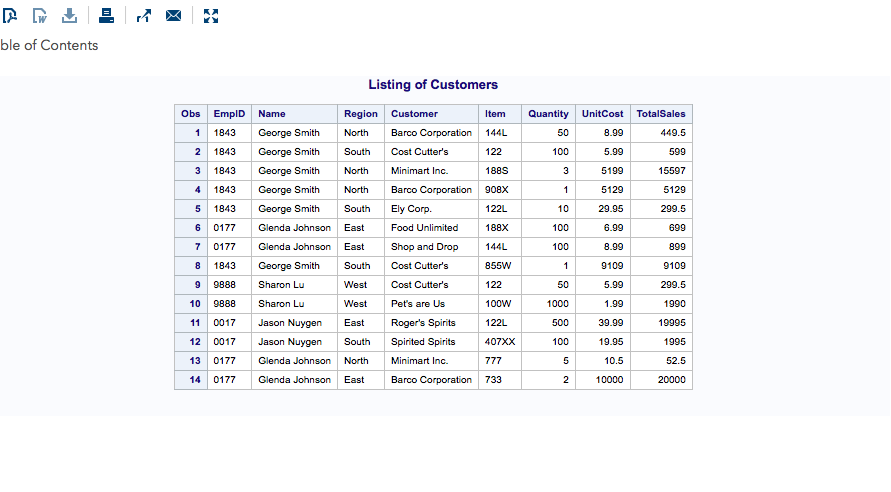
end;

**run**;

**proc** **print** data=WORK.IMPORT1;/\*Selecting Variables to Print\*/

title "Listing of Customers";

**run**;



/\*Problem 8.9 #4.

/\*4. Count the number of missing values for the variables A, B, and C

in the Missing data set.

Add the cumulative number of missing values to each observation

(use variable names MissA, MissB, and MissC). Use the MISSING function

to test for the missing values.\*/

/\*Nilupul Rathgama for BAN500\*/

/\*Problem 8.9 #4\*/

FILENAME SALES '/home/nilunra0/my\_courses/sue.mcdaniel/Wks 5\_6/missing.txt';

/\* access to Sales file to the program \*/

**PROC** **IMPORT** DATAFILE=missing /\* Importing the Sales file \*/

DBMS=txt /\* Determining what type of file is being used \*/

OUT=WORK.IMPORT1; /\* Naming Said referenced file for SAS \*/

GETNAMES=YES; /\* Using the Xls names as file type \*/

**RUN**;

**data** WORK.IMPORT1;

/\*the DATA step automatically names each successive data set \*/

input X $ Y Z A;/\*brings an input data record into the

input buffer without creating any SAS variables\*/

/\*if-then statement can be used to create a new variable

for a selected subset of the observations\*/

if missing(X) then misscounterX+**1**;

if missing(Y) then misscounterY+**1**;

if missing(Z) then misscounterZ+**1**;

if missing(A) then misscounterA+**1**;

/\*DATALINES statement with an INPUT statement to read data that you

enter directly in the program, rather than data stored in an external file.\*/

datalines;

M 56 68 89

F 33 60 71

M 45 91 .

F 35 35 68

M . 71 81

M 50 68 71

. 23 60 46

M 65 72 103

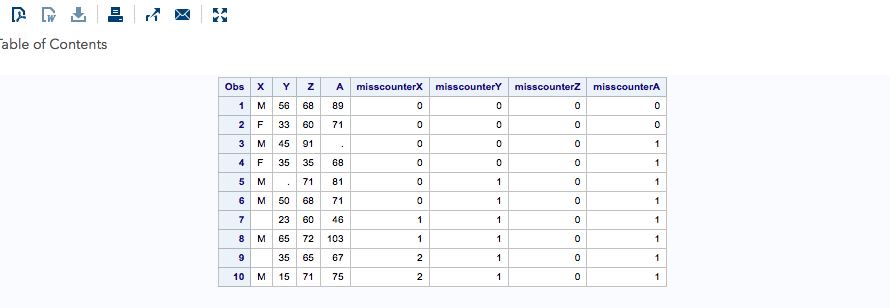
. 35 65 67

M 15 71 75

;

**proc** **print** data=WORK.IMPORT1;/\*Selecting Variables to Print\*/

**run**;



/\*Problem 8.9 #14.

/\*14. Generate a table of integers and squares starting at 1

and ending when the square value is greater than 100. Use either a

DO UNTIL or DO WHILE statement to accomplish this.\*/

/\*Nilupul Rathgama for BAN500\*/

/\*Problem 8.9 #14\*/

FILENAME SQAURES '/home/nilunra0/my\_courses/sue.mcdaniel/Wks 5\_6/Sales.xls';

/\* access to Sales file to the program \*/

**PROC** **IMPORT** DATAFILE=squares/\* Importing the Sales file \*/

DBMS=txt /\* Determining what type of file is being used \*/

OUT=work.import1; /\* Naming Said referenced file for SAS \*/

GETNAMES=YES; /\* Using the Xls names as file type \*/

**RUN**;

**data** work.import1;

/\*the DATA step automatically names each successive data set \*/

do Integers = **1** to **100** until (squares ge **100**);

/\*using do until taking values from 1 to 100 and

specifying the condition for squares variable to

stop the loop when it reaches 100\*/

/\*The DO statement specifies that the statements following the DO statement

be executed as a group until a matching END statement appears.\*/

Squares = Integers \* integers;

Squares = Integers \* integers;

output;

/\*OUTPUT without arguments causes the current observation to

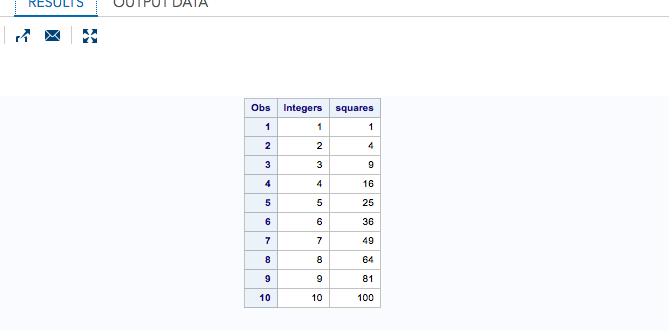
be written to all data sets that are named in the DATA statement.\*/

end;

**run**;

**proc** **print** data=work.import1;/\*Selecting Variables to Print\*/

**run**;



/\*Problem 8.9 #14\*/

/\*using IF STMT\*/

\*Nilupul Rathgama for BAN500\*/

/\*Problem 8.9 #14\*/

FILENAME SQAURES '/home/nilunra0/my\_courses/sue.mcdaniel/Wks 5\_6/Sales.xls';

/\* access to Sales file to the program \*/

**PROC** **IMPORT** DATAFILE=squares/\* Importing the Sales file \*/

DBMS=txt /\* Determining what type of file is being used \*/

OUT=work.import1; /\* Naming Said referenced file for SAS \*/

GETNAMES=YES; /\* Using the Xls names as file type \*/

**RUN**;

/\*using IF STMT\*/

**data** work.import1;

/\*the DATA step automatically names each successive data set \*/

do Integers = **1** to **100** by **1**;

/\*The DO statement specifies that the statements following the DO statement

be executed as a group until a matching END statement appears.\*/

Squares = Integers \* integers;

if Squares gt **100** then leave;

/\*if-then statement can be used to create a new variable

for a selected subset of the observations\*/

output;

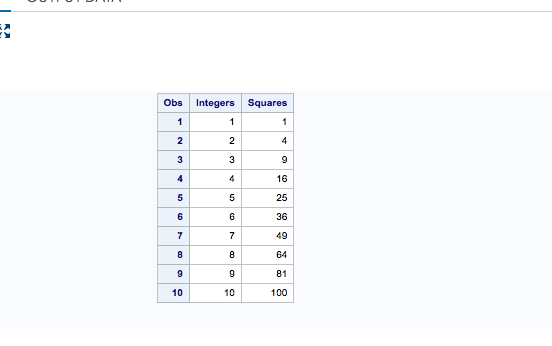
/\*OUTPUT without arguments causes the current observation to

be written to all data sets that are named in the DATA statement.\*/

end;

**run**;

**proc** **print** data=work.import1;/\*Selecting Variables to Print\*/

**run;**