STAT 425 and STAT 625- Statistical Software

Dr. Monica Jackson, Instructor

Homework #5-Combining Data Sets; Creating Summary Reports

Directions: Please submit one program file, one output file, and one log file for the entire assignment. Use comment statements to separate your answers. For questions that do not require a SAS program use comment statements. For example:

/*

Question #1d: my answer

Question #2a: my answer

*/

/*Question #4b: */

--*SAS program*—

/*Question #5*/

Please make sure the log and output file contain only one run. For example, clear the screen for the log and output file and submit your program one last time before you upload your solutions to ***Blackboard***. See lab 1 for the instructions on how to clear your output and log files.

Part I: Combining SAS Data Sets

1. **Concatenating Data Sets with Variables of Different Lengths and Types**

   a. Open **p110e03**. Submit the PROC CONTENTS steps or explore the data sets interactively to complete the table below. Fill in attribute information for each variable in each data set.

	Code		Company		ContactType	
	Type	Length	Type	Length	Type	Length
orion.charities						
orion.us_suppliers						
orion.consultants						

   b. Write a DATA step to concatenate **orion.charities** and **orion.us_suppliers** and create a temporary data set, **contacts**.

   c. Submit a PROC CONTENTS step to examine **work.contacts**. From which input data set were the variable attributes assigned? ____

   d. Write a DATA step to concatenate **orion.us_suppliers** and **orion.charities** and create a temporary data set, **contacts2**. Notice that these are the same data sets as in the previous program, but they are in reverse order.

   e. Submit a PROC CONTENTS step to examine **work.contacts2**. From which input data set were the variable attributes assigned? ____

   f. Write a DATA step to concatenate **orion.us_suppliers** and **orion.consultants** and create a temporary data set, **contacts3**.

   Why did the DATA step fail? ____

2. **Merging a Sorted Data Set and an Unsorted Data Set in a One-to-Many Merge**

   a. Sort **orion.product_list** by **Product_Level** to create a new data set, **work.product_list**.

b. Merge **orion.product_level** with the sorted data set. Create a new data set, **work.listlevel**, which includes only **Product_ID**, **Product_Name**, **Product_Level**, and **Product_Level_Name**.

c. Create the report below. Include only those observations with **Product Level** equal to *3*. The results should contain 13 observations.

Partial PROC PRINT Output

```
Product_     Product_Level_
   Level          Name            Product_ID    Product_Name

       3    Product Category    210100000000    Children Outdoors
       3    Product Category    210200000000    Children Sports
       3    Product Category    220100000000    Clothes
       3    Product Category    220200000000    Shoes
       3    Product Category    230100000000    Outdoors
```

3. Merging Using the IN= and RENAME= Options

a. Write a PROC SORT step to sort **orion.customer** by **Country** to create a new data set, **work.customer**.

b. Write a DATA step to merge the resulting data set with **orion.lookup_country** by **Country** to create a new data set, **work.allcustomer**.

In the **orion.lookup_country** data set, rename **Start** to **Country** and rename **Label** to **Country_Name**.

Include only four variables: **Customer_ID**, **Country**, **Customer_Name**, and **Country_Name**.

c. Create the report below. The results should contain 308 observations.

Partial PROC PRINT Output

```
Obs     Customer_ID      Country    Customer_Name       Country_Name

  1               .      AD                             Andorra
  2               .      AE                             United Arab Emirates
 ...
 306           3959      ZA         Rita Lotz           South Africa
 307              .      ZM                             Zambia
 308              .      ZW                             Zimbabwe
```

d. Modify the DATA step to store only those observations that contain both customer information and country information. A subsetting IF statement that references the IN= variables in the MERGE statement must be added.

e. Submit the program to create the report below. The results should contain 77 observations.

Partial PROC PRINT Output

```
                                                          Country_
            Obs     Customer_ID    Country    Customer_Name         Name

              1              29      AU       Candy Kinsey          Australia
              2              41      AU       Wendell Summersby     Australia
              3              53      AU       Dericka Pockran       Australia
              4             111      AU       Karolina Dokter       Australia
              5             171      AU       Robert Bowerman       Australia
```

Part II: Creating Summary Reports

1. Producing Frequency Reports with PROC FREQ

a. Retrieve the starter program **p111e03**.

b. Add statements to the PROC FREQ step to produce three frequency reports.

1) Number of orders in each year: Apply a format to the **Order_Date** variable to combine all orders within the same year.

2) Number of orders of each order type: Apply the ORDERTYPES. format that is defined in the starter program to the **Order_Type** variable. Suppress the cumulative frequency and percentages.

3) Number of orders for each combination of year and order type: Suppress all percentages that normally appear in each cell of a two-way table.

c. Submit the program to produce the following output:

PROC FREQ Output

```
                              Order Summary by Year and Type

                                    The FREQ Procedure

                              Date Order was placed by Customer

                                                   Cumulative    Cumulative
          Order_Date    Frequency     Percent     Frequency       Percent
          ƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒ
                2007          104       21.22           104         21.22
                2008           87       17.76           191         38.98
                2009           70       14.29           261         53.27
                2010          113       23.06           374         76.33
                2011          116       23.67           490        100.00


                                         Order Type

                         Order_
                           Type    Frequency     Percent
                       ƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒ
                       Retail           260       53.06
                       Catalog          132       26.94
                       Internet          98       20.00
                              Table of Order_Date by Order_Type

                        Order_Date(Date Order was placed by Customer)
                                   Order_Type(Order Type)

                    Frequency,Retail  ,Catalog ,Internet,  Total
                    ƒƒƒƒƒƒƒƒƒˆƒƒƒƒƒƒƒƒˆƒƒƒƒƒƒƒƒˆƒƒƒƒƒƒƒƒˆ
                        2007 ,     45 ,     41 ,     18 ,    104
                    ƒƒƒƒƒƒƒƒƒˆƒƒƒƒƒƒƒƒˆƒƒƒƒƒƒƒƒˆƒƒƒƒƒƒƒƒˆ
                        2008 ,     51 ,     20 ,     16 ,     87
                    ƒƒƒƒƒƒƒƒƒˆƒƒƒƒƒƒƒƒˆƒƒƒƒƒƒƒƒˆƒƒƒƒƒƒƒƒˆ
                        2009 ,     27 ,     23 ,     20 ,     70
                    ƒƒƒƒƒƒƒƒƒˆƒƒƒƒƒƒƒƒˆƒƒƒƒƒƒƒƒˆƒƒƒƒƒƒƒƒˆ
                        2010 ,     67 ,     33 ,     13 ,    113
                    ƒƒƒƒƒƒƒƒƒˆƒƒƒƒƒƒƒƒˆƒƒƒƒƒƒƒƒˆƒƒƒƒƒƒƒƒˆ
                        2011 ,     70 ,     15 ,     31 ,    116
                    ƒƒƒƒƒƒƒƒƒˆƒƒƒƒƒƒƒƒˆƒƒƒƒƒƒƒƒˆƒƒƒƒƒƒƒƒˆ
                    Total         260      132       98      490
```

2. Validating orion.qtr2_2011 with PROC FREQ

Write a PROC FREQ step to validate the data in **orion.qtr2_2011**.

a. Create frequency tables for **Order_ID** and **Order_Type**. Include the **Number of Variable Levels** table.

b. Submit the program.

The data in **orion.qtr2_2011** should meet the following requirements:

- **Order_ID** must be unique (36 distinct values) and not missing.
- **Order_Type** must have a value of *1*, *2*, or *3*.

What invalid data exists for **Order_ID** and **Order_Type**? ______

3. Analyzing Missing Numeric Values with PROC MEANS

a. Retrieve the starter program **p111e08**.

b. Display the number of missing values and the number of nonmissing values present in the **Birth_Date**, **Emp_Hire_Date**, and **Emp_Term_Date** variables.

c. Add a CLASS statement to display separate statistics for each value of **Gender**.

d. Suppress the column that displays the total number of observations in each classification group.

e. Submit the program to produce the following report:

PROC MEANS Output

```
                        Number of Missing and Non-Missing Date Values

                                    The MEANS Procedure

        Employee                                                     N
        Gender      Variable         Label                         Miss      N
        ƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒ
        F           Birth_Date       Employee Birth Date              0    191
                    Emp_Hire_Date    Employee Hire Date               0    191
                    Emp_Term_Date    Employee Termination Date      139     52

        M           Birth_Date       Employee Birth Date              0    233
                    Emp_Hire_Date    Employee Hire Date               0    233
                    Emp_Term_Date    Employee Termination Date      169     64
        ƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒ
```

4. Validating orion.shoes_tracker with the UNIVARIATE Procedure

a. Write a PROC UNIVARIATE step to validate **Product_ID** of **orion.shoes_tracker**. A valid **Product_ID** value must have exactly 12 digits.

b. Submit the program and view the Extreme Observations output.

How many values of **Product_ID** are too small? ____

How many values of **Product_ID** are too large? ____

5. Directing Output to the EXCELXP Destination

a. Retrieve the starter program **p111e13**.

b. Add ODS statements to create an XML output with the following naming convention:

Windows	"&path\p111e13.xls"
UNIX	"&path/p111e13.xls"

c. Add the STYLE= option to the ODS statement to use the Listing style template.

d. Submit the program and view the XML output in Microsoft Excel. Select **Yes** to open the files when you are prompted about the file format and extension not matching.

Part III- Supplemental exercises for STAT 625 and Honors credit

1. **Merging and Creating Output in Multiple Data Sets**

   a. Write a PROC SORT step to sort **orion.orders** by **Employee_ID** to create a new data set, **work.orders**.

   b. Write a DATA step to merge **orion.staff** and **work.orders** by **Employee_ID** and create two new data sets: **work.allorders** and **work.noorders**.

      - The **work.allorders** data set should include all observations from **work.orders**, regardless of matches or nonmatches from the **orion.staff** data set.
      - The **work.noorders** data set should include only those observations from **orion.staff** that do not have a match in **work.orders**.
      - Both new data sets should include only **Employee_ID**, **Job_Title**, **Gender**, **Order_ID**, **Order_Type**, and **Order_Date**.

   c. Submit the program and confirm that **work.allorders** was created with 490 observations and six variables and **work.noorders** was created with 324 observations and six variables.

   d. Create a detailed listing report for each new data set with an appropriate title.

2. **Creating an Output Data Set with PROC FREQ**

   Write a program to perform a frequency analysis on **Product_ID** in **orion.order_fact**.

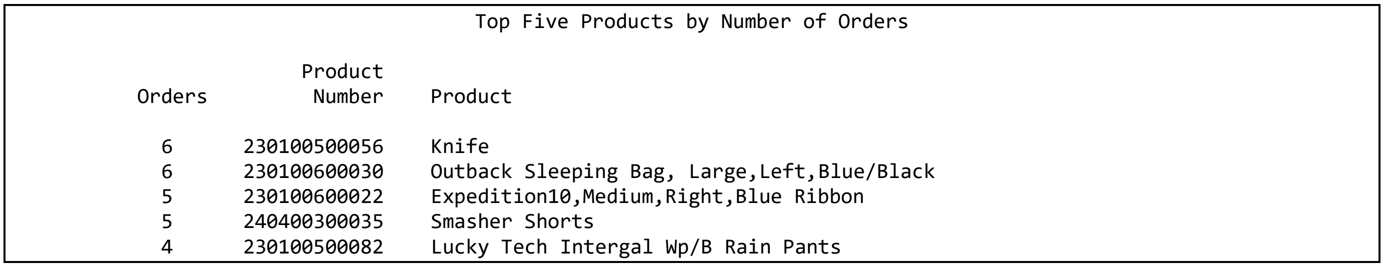
   a. Create an output data set that contains the frequency counts based on **Product_ID**. Explore the SAS Help Facility or online documentation for information about creating an output data set of counts from PROC FREQ results.

   b. Combine the output data set with **orion.product_list** to obtain the **Product_Name** value for each **Product_ID** code. Output only those products that were ordered.

   c. Sort the combined data so that the most frequently ordered products appear first in the resulting data set. Print the first five observations—that is, those that represent the five products ordered most often. Use the OBS= data set option to limit the number of observations that are displayed.

   d. Submit the program to produce the following report:

   PROC PRINT Output



```
                                              Top Five Products by Number of Orders

                        Product
            Orders       Number    Product

              6      230100500056    Knife
              6      230100600030    Outback Sleeping Bag, Large,Left,Blue/Black
              5      230100600022    Expedition10,Medium,Right,Blue Ribbon
              5      240400300035    Smasher Shorts
              4      230100500082    Lucky Tech Intergal Wp/B Rain Pants
```

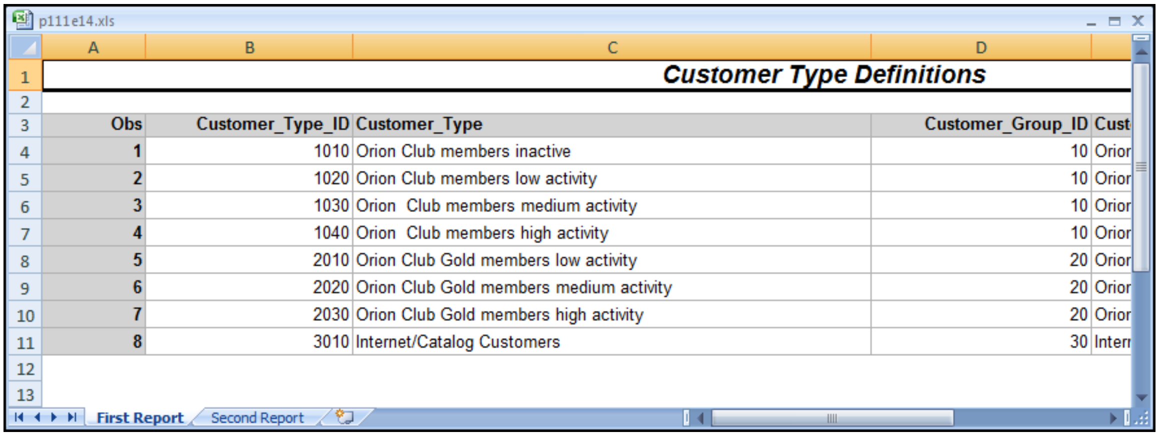
3. **Adding Options to the EXCELXP Destination**

   a. Retrieve the starter program **p111e14**.

   b. Submit the program and view the XML output in Microsoft Excel. Select **Yes** to open the files when you are prompted about the file format and extension not matching.

   c. In the log, view the documentation for the EXCELXP destination.

   d. Add EMBEDDED_TITLES= and SHEET_NAME= options to create the following output:



Customer Type Definitions

Obs	Customer_Type_ID	Customer_Type	Customer_Group_ID	Cust
1	1010	Orion Club members inactive	10	Orion
2	1020	Orion Club members low activity	10	Orion
3	1030	Orion Club members medium activity	10	Orion
4	1040	Orion Club members high activity	10	Orion
5	2010	Orion Club Gold members low activity	20	Orion
6	2020	Orion Club Gold members medium activity	20	Orion
7	2030	Orion Club Gold members high activity	20	Orion
8	3010	Internet/Catalog Customers	30	Intern