STAT 425 and STAT 625- Statistical Software

Dr. William Howell, Professor

# Homework #3-Reading SAS Data; Reading Spreadsheet and Database Data

**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-Reading SAS data sets

1. **Creating a SAS Data Set**
   - **a.** Write a DATA step to create a new data set named **work.assistant**. Use the data set **orion.staff** as input.
   - **b.** The **work.assistant** data set should contain only the observations where **Job_Title** contains *Assistant* and **Salary** is less than *$26,000*.
   - **c.** Create two new variables, **Increase** and **New_Salary**.
     - **Increase** is **Salary** multiplied by 0.10.
     - **New_Salary** is **Salary** added to **Increase**.
   - **d.** Generate a detail listing report as shown below. Display **Employee_ID** as the identifier in place of the Obs column. The results should contain five observations.

| Employee_ID | Job_Title | Salary | Increase | New_Salary |
|---|---|---|---|---|
| 120685 | Warehouse Assistant I | $25,130.00 | $2,513.00 | $27,643.00 |
| 120688 | Warehouse Assistant I | $25,905.00 | $2,590.50 | $28,495.50 |
| 120690 | Warehouse Assistant I | $25,185.00 | $2,518.50 | $27,703.50 |
| 121010 | Service Assistant I | $25,195.00 | $2,519.50 | $27,714.50 |
| 121011 | Service Assistant I | $25,735.00 | $2,573.50 | $28,308.50 |

2. **Subsetting Observations Based on Three Conditions**
   - **a.** Write a DATA step to create **work.delays**. Use **orion.orders** as input.
   - **b.** Create a new variable, **Order_Month**, and set it to the month of **Order_Date**.

     Hint: Use the MONTH function.

**c.** Use a WHERE statement and a subsetting IF statement to select only the observations that meet all of the following conditions:

- **Delivery_Date** values that are more than four days beyond **Order_Date**
- **Employee_ID** values that are equal to *99999999*
- **Order_Month** values occurring in *August*

**d.** The new data set should include only **Employee_ID**, **Customer_ID**, **Order_Date**, **Delivery_Date**, and **Order_Month**.

**e.** Add permanent labels for **Order_Date**, **Delivery_Date**, and **Order_Month** as shown below.

**f.** Add permanent formats to display **Order_Date** and **Delivery_Date** as MM/DD/YYYY.

**g.** Add a PROC CONTENTS step to verify that the labels and formats were stored permanently.

```
           Alphabetic List of Variables and Attributes

    #    Variable         Type    Len    Format       Label

    2    Customer_ID      Num       8    12.          Customer ID
    4    Delivery_Date    Num       8    MMDDYY10.    Date Delivered
    1    Employee_ID      Num       8    12.          Employee ID
    3    Order_Date       Num       8    MMDDYY10.    Date Ordered
    5    Order_Month      Num       8                 Month Ordered
```

**h.** Write a PROC PRINT step to create the report below. Results should contain nine observations.

| Obs | Employee_ID | Customer_ID | Order_Date | Delivery_ Date | Order_ Month |
|---|---|---|---|---|---|
| 1 | 99999999 | 70187 | 08/13/2007 | 08/18/2007 | 8 |
| 2 | 99999999 | 52 | 08/20/2007 | 08/26/2007 | 8 |
| 3 | 99999999 | 16 | 08/27/2007 | 09/04/2007 | 8 |
| 4 | 99999999 | 61 | 08/29/2007 | 09/03/2007 | 8 |
| 5 | 99999999 | 2550 | 08/10/2008 | 08/15/2008 | 8 |
| 6 | 99999999 | 70201 | 08/15/2008 | 08/20/2008 | 8 |
| 7 | 99999999 | 9 | 08/10/2009 | 08/15/2009 | 8 |
| 8 | 99999999 | 71 | 08/30/2010 | 09/05/2010 | 8 |
| 9 | 99999999 | 70201 | 08/24/2011 | 08/29/2011 | 8 |

# Part II-Reading Spreadsheet and Database Data

**3. Accessing an Excel Worksheet**

**a.** Open a new program and write an OPTIONS statement to set VALIDVARNAME to **V7**.

**b.** Write a LIBNAME statement to create a libref named **prod** that references the Excel workbook **products.xlsx**.

**c.** Write a PROC CONTENTS step to view all of the contents of **prod**.

**d.** Submit the program to determine the names of the four worksheets in **products.xlsx**.

**e.** Write a DATA step that reads the worksheet containing sports data and creates a new data set named **work.golf**.

**f.** The data set **work.golf** should have the following characteristics:

- include only the observations where **Category** is equal to *Golf*
- not include the **Category** variable
- include a label of **Golf Products** for the **Name** variable

**f.** Write a LIBNAME statement to clear the **prod** libref.

**g.** Write a PROC PRINT step to create the report below. Results should contain 56 observations.

Partial PROC PRINT Output

| Obs | Golf Products |
|---|---|
| 1 | Ball Bag |
| 2 | Red/White/Black Staff 9 Bag |
| 3 | Tee Holder |
| 4 | Bb Softspikes - Xp 22-pack |
| 5 | Bretagne Performance Tg Men's Golf Shoes L. |

# Part III- Supplemental exercises for STAT 625 and Honors credit

**4. Using the SOUNDS-LIKE Operator to Select Observations**

**a.** Write a DATA step to create a new data set named **work.tony**. Use **orion.customer_dim** as input.

**b.** Include a WHERE statement in the DATA step to select observations in which the **Customer_FirstName** value sounds like *Tony*.

Documentation about the SOUNDS-LIKE operator can be found in the SAS Help Facility or product documentation by searching for "sounds-like operator."

**c.** Write a PROC PRINT step to create the following report:

| Obs | FirstName | Customer_ LastName | Customer_ |
|---|---|---|---|
| 1 | Tonie | Asmussen | |
| 2 | Tommy | Mcdonald | |

**5. Using an IF-THEN/DELETE Statement to Subset Observations**

**a.** Write a DATA step to create **work.bigdonations**. Use **orion.employee_donations** as input.

**b.** Use the SUM function to create a new variable, **Total**, which holds the sum of the four quarterly donations.

**c.** Use the N function to create a new variable, **NumQtrs**, which holds the count of nonmissing values in **Qtr1**, **Qtr2**, **Qtr3**, and **Qtr4**. Explore the N function in the SAS Help Facility or online documentation.

**d.** The new data set should ***not*** include the charities or method of payment.

**e.** The final data set should contain only observations that meet the following two conditions:

- **Total** values greater than or equal to 50
- **NumQtrs** value equal to 4

Use an IF-THEN/DELETE statement to eliminate the observations where the conditions are not met. Explore the use of IF-THEN/DELETE in the SAS Help Facility or online documentation.

**f.** Store permanent labels in the new data set as shown in the report below.

**g.** Create the following report to verify that the labels were stored:

Alphabetic List of Variables and Attributes

| # | Variable | Type | Len | Format | Label |
|---|---|---|---|---|---|
| 1 | Employee_ID | Num | 8 | 12. | Employee ID |
| 7 | NumQtrs | Num | 8 | | |
| 2 | Qtr1 | Num | 8 | | First Quarter |
| 3 | Qtr2 | Num | 8 | | Second Quarter |
| 4 | Qtr3 | Num | 8 | | Third Quarter |
| 5 | Qtr4 | Num | 8 | | Fourth Quarter |
| 6 | Total | Num | 8 | | |

**h.** Create the report below. The results should contain 50 observations.

| Employee ID | First Quarter | Second Quarter | Third Quarter | Fourth Quarter | Total | Num Qtrs |
|---|---|---|---|---|---|---|
| 120267 | 15 | 15 | 15 | 15 | 60 | 4 |
| 120269 | 20 | 20 | 20 | 20 | 80 | 4 |
| 120271 | 20 | 20 | 20 | 20 | 80 | 4 |
| 120275 | 15 | 15 | 15 | 15 | 60 | 4 |
| 120660 | 25 | 25 | 25 | 25 | 100 | 4 |