Assignment 3 – Due on Thursday 19th before 5.00 pm (PDT)

Note: For each question, produce separate PDF files for SAS code and output. Those PDF files should be uploaded to the Crowdmark under each question.

Q1) The following attached data files need to be combined into a single SAS data set.

- AO Division.dat
- CORP Division.dat
- EmpData.dat
- FINANCE IT Division.dat
- FLIGHT OPS Division.dat
- HR Division.dat
- Personal.dat
- SALES Division.dat

a) Create a single data set in SAS using the six Division files and name it as *DIVISION*. You may have to use FIRSTOBS option to read datasets.

b) Combine *DIVISION* data with *EmpData* to create *EmpData_div_merged* dataset. Then merge *EmpData_div_merged* dataset with *Personal* dataset to create the complete dataset and name it as *Air_Emps_Full*.

I. Ensure that the *Air_Emps_Full* dataset does not include the *PHONE* variable and update the salary variable to be named *CurrentSalary*. Include an appropriate format for this variable.

II. Assign descriptive labels to the variables using LABEL statement (Hint: Read the online document on LABEL statement available at https://documentation.sas.com, or any other related SAS online document). Comment on the advantage of labelling variables.

c) Use the *Air_Emps_Full* data set to create a subset of data that only includes records for employees that were hired prior to January 1, 1990, and whose current salary is below $45,000. Name this dataset as *Air_Emps_Underpaid.*

d) Produce a PDF showing the first 10 observations with all variables in *Air_Emps_Underpaid.*

Q2) Using the *Air_Emps_Full* data set from Q1, you need to create a new data set named *Air_Emps_Raises* that contains a new salary variable named *New_Salary* that is computed as follows. (Note that the percentage increase should not be affected by the flat increase. Also, consider the years of service of an employee from hire date to 01/01/2019.)

- For employees with at least 35 years of service, the new salary includes:
  - A $3,500 at increase
  - A 2.5% increase for anyone whose job is at level 2 or level 3 (The number in a job type indicates the level of that job type. E.g., Job type, FLTAT3 is at level 3)
  - A 1.5% raise for anyone whose job is at level 1 (e.g. MECH01)
  - A 2.0% raise for all other job levels
- For employees with at least 25 years of service (but not 35 or more), the new salary includes:
  - A $2,000 flat increase
  - A 2.0% increase for level 2 or 3 jobs
  - A 1.0% increase for level 1 jobs
  - A 1.75% increase for all other jobs
- For all others, the new salary includes:
  - No flat increase
  - A 2.50% increase for level 2 or 3 jobs
  - A 1.25% increase for level 1 jobs
  - A 0.75% increase for all other jobs
- If years of service cannot be computed for an employee
  - New Salary should be missing
  - A note should be printed to the log, indicating the employee ID that needs to be investigated

a) Ensure the final data set is sorted by EMPID and that your two new variables (New Salary and Years of Service) have appropriate labels and formats.

b) Without recreating the *Air_Emps_Underpaid* dataset in Q1, we want to subset it to only include individuals whose new salary would still classify them as underpaid. To do this, using the above conditions, create the *New_Salary* variable in *Air_Emps_Underpaid* dataset as well, but only keep records that would still be classified as underpaid even after they got a raise (New salary is below $45,000).

c) Write both datasets to a single PDF using titles and footnotes to differentiate the two data sets (10 observations from each dataset).

Q3) Consider the attached three SAS datasets: `aprtarget.sas7bdat, maytarget.sas7bdat, and junetarget.sas7bdat`

a) Write a data step to concatenate these three SAS data sets and create a new data set called **work.q3vienna**. Use the RENAME= option to rename any variables necessary to combine the datasets.

b) Browse the SAS log. There should be no warning or error messages. How many observations are written to the new data set? Comment on the code.

c) Now modify the DATA step to create two new variables: **TotalTar** and **TotalRev**.
   - **TotalTar** is the total targeted number of economy and first class passengers.
   - **TotalRev** is the total revenue expected from economy and first class passengers.

   Keep only the variables **FlightID, Destination, Date, TotalTar**, and **TotalRev**.
   Produce a PDF to print first 10 observations to verify the data.

Q4) The weather in Birmingham, Alabama on December 15, 1999, might have caused some customers to alter their shipping plans. Investigate how much cargo revenue was lost on all flights out of Birmingham by comparing the targeted revenue with the actual revenue.

a) Sort the SAS data set '`target121999`' into a temporary SAS data set called **work.sort_b** (Use OUT option in PROC SORT). Sort by the variable FlightID. Create a subset for Birmingham (BHM) on December 15, 1999.
b) Sort the SAS data set '`sales121999`' into a temporary data set called **work.sort_s**. Sort by the variable FlightID. Create a subset for Birmingham (BHM) on December 15,1999.
c) Create a new temporary data set called *compare* by merging the **sort_b** and **sort_s** data sets by the variable **FlightID**. Subtract **CargoRev** from **CargoTarRev** to create a new variable called **LostCargoRev**.
d) Produce a PDF to print the merged data set compare (print only the variables **CargoTarRev**, **CargoRev**, and **LostCargoRev**) and label the **LostCargoRev** variable. Format the **LostCargoRev** variable with a dollar sign and two decimal digits.

Q5) Produce separate PDF documents for each output in part a-c.
a) Use an iterative DO loop to plot the following equation:

$$y = 3x^2 - 5x + 10$$

Use values of x from 0 to 10, with an increment of 0.10 .

b) You have daily temperatures for each hour of the day for two cities (Dallas and Houston). The 48 temperature values are strung out in several lines like this:

80 81 82 83 84 84 87 88 89 89
91 93 93 95 96 97 99 95 92 90 88
86 84 80 78 76 77 78
80 81 82 82 86
88 90 92 92 93 96 94 92 90
88 84 82 78 76 74

The first 24 values represent temperatures from Hour 1 to Hour 24 for Dallas and the next 24 values represent temperatures for Hour 1 to Hour 24 for Austin. Using the appropriate DO loops, create a dataset (Temperature) with 48 observations, each observation containing the variables City, Hour, and Temp.

c) You invest $1,000 a year at 4.25% interest, compounded quarterly. How many years will it take to reach $30,000?. Use DO WHILE or DO UNTIL statements to solve this problem.