Homework 2

Due: Tuesday, June 28th (before 11:59 pm) on GauchoSpace.

Instructions: For the General Knowledge Questions, answer as succinctly as possible. For the Programming Assignment, show your code as well as a partial screenshot of your output.

General Knowledge Questions

1. Write 2 statements each of which assign the SAS date value corresponding to June 28th, 2022 to the variable 'due_date'.
2. Which statement, in the PRINT procedure, is used to specify the variable used to create page breaks?
3. Name four different input styles.
4. Which statement is used in the DATASETS procedure to change the name of a variable?
5. Which statements can be used to control the variables being written to the output data set?
6. List the two symbols used for line pointer control.
7. In which step(s) can IF statements be used?
8. Which statement is used in the DATA step to direct SAS to read from an existing SAS dataset?
9. Name the function that is used to convert all letters in a string to lowercase.
10. Which input style is used to read in data that has a variable name assigned to each value?
11. Which statement is used to direct SAS to raw data within the program?
12. Which statement is used in the DATASETS procedure to change the format of a variable?
13. Without the use of DO and END statements, how many executable statements are allowed in an IF-THEN or ELSE statement?
14. Which procedure can be used to display the data portion of a SAS data set?
15. In the DATA step, can a WHERE statement be used when evaluating a calculated variable?

Programming Assignment

1. Use the **`CA_Educational_Data.xlsx`** file found on GauchoSpace.
   a) Use the IMPORT procedure to create a SAS data set named **`CA_GradRates`** (in the work library) from the Excel file. Replace any data sets with the same name. Import the worksheet named **`FourYearGradRates`**.
   b) Create a report displaying the data portion of the **`CA_GradRates`** data set.
   c) Create a report displaying the descriptor portion of the **`CA_GradRates`** data set.

2. Use the **military** data set from the *data1* folder on GauchoSpace.
   a) Sort the data set by **Type** (reverse alphabetical order), **State** (alphabetical order), and **City** (alphabetical order), and output the sorted data set into **`work.military_s`**.
   b) Create a report that
      i. Has an appropriate title.
      ii. Displays the variables **Type**, **State**, **City**, and **Airport** (in this order).

iii. Displays the subgroups of **Type** on separate pages.
iv. Uses the **Type** as an identification variable.
v. Displays the total number of observations in each subgroup.
vi. Assign the label 'Military Branch' to **Type**, 'State Abbreviation' to **State**, and 'Airport Name' to **airport**.

3. Use the **`crew`** data set found in the data1 folder.
   a) Create a temporary SAS data set named **`SalaryStatement`**:
      i. Create a variable named **TaxWithheld** that contains the tax withholdings for each employee. Calculate the **TaxWithheld** values as shown below (note that in real life the calculation is different):
         - 12% of Salary when **Salary** $\leq$ 40,525
         - 22% of Salary when 40,526 $\leq$ **Salary** $\leq$ 86,375
         - 24% of Salary when 86,376 $\leq$ **Salary** $\leq$ 164,925
         - 32% of Salary when **Salary** $\geq$ 164,926
      ii. Create a variable named **NetPay** that contains the after tax salary. The **`SalaryStatement`** data set should contain only the variables **EmpID, Salary,** and **NetPay**.
   b) Now create a report using the **`SalaryStatement`** data set.
      i. Use an appropriate title.
      ii. Suppress observation numbers.
      iii. Display the values of **Salary** and **NetPay** with dollar signs, commas, and no decimal places.

4. Use the **`crew`** data set found in the data1 folder.
   a) Create a temporary SAS data set named **`AnnualBonus:`**
      i. Create a variable named **Bonus** that contains an annual bonus amount for each employee calculated as 15% of **Salary**.
      ii. Create a (numeric) variable named **HireWeekday** that contains the day of the week each employee was hired. Hint: Determine the weekday portion of the employee's hire date (**HireDate**).
      iii. Assign the label 'Last Name' to the **LastName** variable and 'Day of the Week Hired' to the **HireWeekday** variable.
      iv. The **`AnnualBonus`** data set should contain only the variables **LastName**, **HireWeekday**, and **Bonus**.
   b) Now create a report using the **`AnnualBonus`** data set:
      i. Display the values of **Bonus** with dollar signs, commas, and two decimal places.
      ii. Use an appropriate title.
      iii. Displays label(s).
      iv. Suppress observation numbers.
      v. Display the variables in the following order: **LastName**, **HireWeekday**, and **Bonus**.

For exercises 5-7, use the **`CarAccidents.dat`** file posted on GauchoSpace. This raw data file contains information on car accidents in the months of January and February of 2014. The table below provides the complete record layout of the raw data file.

Variable Name	Columns	Data Type
Reference	1-7	Character
EastRef	10-15	Numeric
NorthRef	17-22	Numeric
nVehicles	24	Numeric
nCasualties	26	Numeric
Day	29-38	Character (mm/dd/yyyy)
Time	40-43	Numeric
RoadClass	45	Numeric
RoadSurface	47	Numeric
Lighting	49	Numeric
Weather	51	Numeric
Class	53	Numeric
Severity	55	Numeric
Gender	57	Numeric
Age	59-60	Numeric
VehicleType	62-63	Numeric

5. Reading Raw Data Using Formatted Input
   a) Create a SAS data set named **`work.CarAccidents`** by writing a DATA step that only uses formatted input to read in and create the following variables: **`Reference`**, **`Day`**, **`Time`**, **`Weather`**, **`nVehicles`**, and **`nCasualties`**. Use pointer control **AND** an appropriate SAS date informat to store the values of **`Day`**.
   b) What type of variable is **`Day`**?
   c) Read the log and answer the following questions:
      i. How many records were read from the raw data file?
      ii. How many observations does the resulting (output) SAS data set contain?
      iii. How many variables does the resulting (output) SAS data set contain?
   d) Create a *listing report* called *CarAccidents.lst*.
      i. Use PROC PRINT to display the data portion of the data set.
      ii. Assign an appropriate title.
      iii. Display the date and time, and display page numbers.
      iv. Set the page size to 105. Set the line size to 85.
      v. Display the values of the **Day** variable in the form of 06/28/2022.

6. Changing Variable Attributes
   a) Modify the DATA Step of the previous program to assign the following attributes:
      i. Assign the label 'Reference Number' to the **Reference** variable.
      ii. Assign the label 'Date of Accident' to the **Day** variable.
      iii. Assign the label 'Weather Condition' to the **Weather** variable.
      iv. Assign the label 'Number of Vehicles' to the **nVehicles** variable.
      v. Assign the label 'Number of Casualties' to the **nCasualties** variable.
      vi. Run PROC CONTENTS to verify that the changes were made.
   b) Use PROC DATASETS to change the following attributes of the **Reference** and **Day** variables:
      i. Assign a format to the **Day** variable such that the dates are displayed in the form of 28JUN2022.
      ii. Assign the label 'Reference ID' to the **Reference** variable.
      iii. Run PROC CONTENTS to verify that the changes were made.

7. Reading Raw Data Using Column Input
   a) Create a SAS data set named **work.CarAccidents2**, by writing a DATA step that only uses column input, and reads in the following variables: **Reference**, **Day**, **Time**, **Severity**, **nVehicles**, and **VehicleType**.
   b) What type of variable is **Day**?
   c) Read the log to answer the following questions:
      i. How many records were read from the raw data file?
      ii. How many observations does the resulting SAS data set contain?
      iii. How many variables does the resulting SAS data set contain?
   d) Use PROC CONTENTS to display the descriptor portion of the data set.
   e) Create a *listing report* called *CarAccidents2.lst*. Use PROC PRINT to display the data portion of the data set. *Do not* display page numbers, but do display the date and time. Set the line size to 100. Display a title of 'Car Accident Data' in blue italic text, justified to the left.
      i. What is the color of the title in the Results window?
      ii. What is the color of the title in the listing report?
      iii. Are the date and time displayed in the Results window?