Homework 13 Data Step and Macro Interface

1. **Creating Macro Variables with the SYMPUTX Routine** 
   1. Copy the program shown below into the Editor window. Submit the program and examine the output that it creates.

**%macro emporders(idnum=121044);**

**proc print data=orion.orders noobs;**

**var Order\_ID Order\_Type Order\_Date Delivery\_Date;**

**where Employee\_ID=&idnum;**

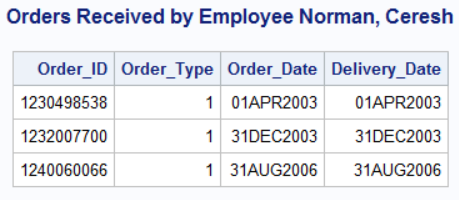
**title "Orders Received by Employee &idnum";**

**run;**

**%mend emporders;**

**%emporders()**

* 1. Modify the macro to include a DATA step that creates a macro variable named NAME based on the variable **Employee\_Name** found in the **orion.employee\_addresses** data set.
  2. Modify the TITLE statement to display the name of the employee instead of the employee’s ID number. Call the macro using the default parameter value.
  3. Call the macro again, but with a parameter value of 121066.



1. **Creating Macro Variables with the SYMPUTX Routine** 
   1. Copy the program shown below into the Editor window. This program creates a summary data set named **customer\_sum** that summarizes **Total\_Retail\_Price** by **Customer\_ID** and sorts the data set by descending **CustTotalPurchase**. Submit the program and examine the output that it creates.

**proc means data=orion.order\_fact nway noprint;**

**var Total\_Retail\_Price;**

**class Customer\_ID;**

**output out=customer\_sum sum=CustTotalPurchase;**

**run;**

**proc sort data=customer\_sum;**

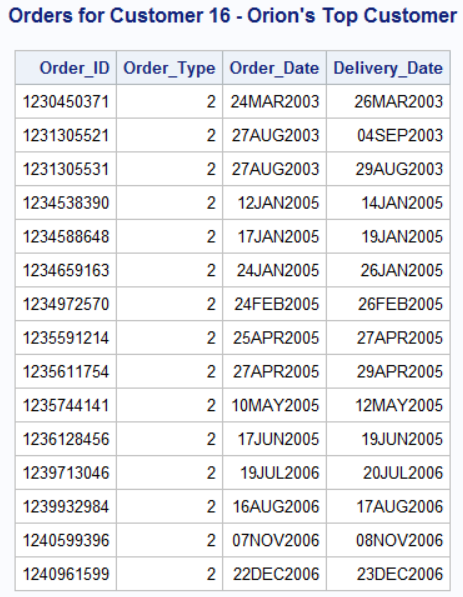
**by descending CustTotalPurchase;**

**run;**

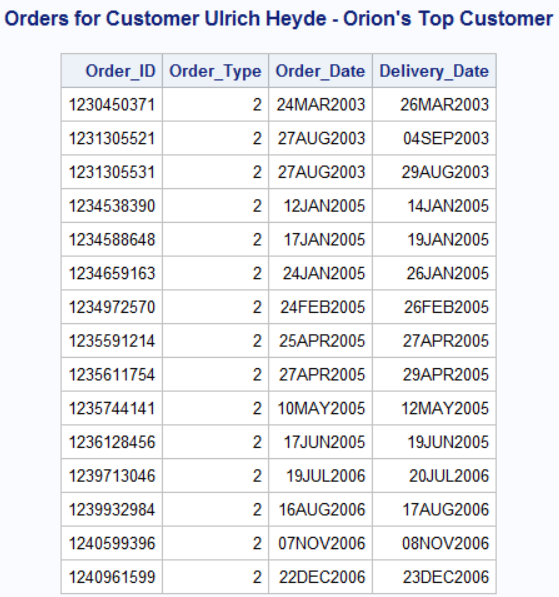
**proc print data=customer\_sum(drop=\_type\_);**

**run;**

* 1. Create a macro variable named TOP that contains the ID number for the top customer. Then modify the program to print only the orders for Orion’s top customer.



* 1. Modify the program to print the customer’s name instead of the customer’s ID in the TITLE statement. Customer names are found in the **orion.customer\_dim** data set.



1. **Creating Macro Variables with the SYMPUTX Routine** 
   1. Copy the program shown below into the Editor window. Submit the program and examine the output that it creates.

**proc means data=orion.order\_fact nway noprint;**

**var Total\_Retail\_Price;**

**class Customer\_ID;**

**output out=customer\_sum sum=CustTotalPurchase;**

**run;**

**proc sort data=customer\_sum ;**

**by descending CustTotalPurchase;**

**run;**

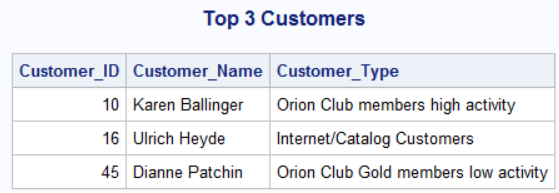
**proc print data=customer\_sum(drop=\_type\_);**

**run;**

* 1. Using the **customer\_sum** data set, create a single macro variable, TOP3, that contains the customer IDs of the top three customers by revenue.

Note that **Customer\_ID** is a numeric variable.

* 1. Using the **orion.customer\_dim** data set, print a listing of the top three customers.



1. **Creating Multiple Macro Variables with the SYMPUTX Routine**
   1. Copy the program shown below into the Editor window.

**%macro memberlist(id=1020);**

**%put \_user\_;**

**title "A List of &id";**

**proc print data=orion.customer;**

**var Customer\_Name Customer\_ID Gender;**

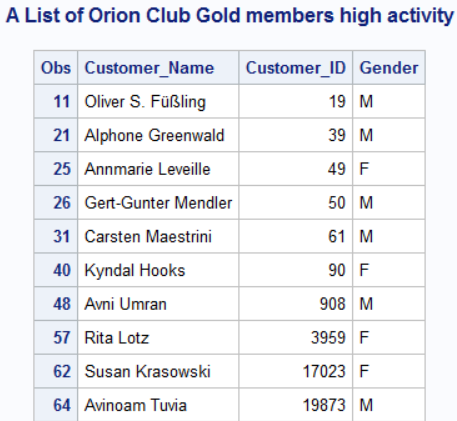
**where Customer\_Type\_ID=&id;**

**run;**

**%mend memberlist;**

**%memberlist()**

* 1. The **orion.customer\_type** data set contains the variable **Customer\_Type\_ID**, which uniquely identifies the customer membership level and activity level. Modify the macro to include a DATA step to create a series of macro variables named TYPE*xxxx*, where *xxxx* is the value of **Customer\_Type\_ID**. The value of each TYPE macro variable should be the value of **Customer\_Type**.
  2. Modify the TITLE statement so that it displays the appropriate customer type. Use an indirect macro variable reference to one of the TYPE variables based on the current value of ID. Submit the modified program.
  3. Call the macro again, but with a parameter value of 2030.



1. **Using Indirect References in a Macro Call** 
   1. Copy the program shown below into the Editor window. Submit the program and examine the results.

**data \_null\_;**

**set orion.customer\_type;**

**call symputx('type'||left(Customer\_Type\_ID), Customer\_Type);**

**run;**

**%put \_user\_;**

**%macro memberlist(custtype);**

**proc print data=orion.customer\_dim;**

**var Customer\_Name Customer\_ID Customer\_Age\_Group;**

**where Customer\_Type="&custtype";**

**title "A List of &custtype";**

**run;**

**%mend memberlist;**

* 1. Create a macro variable named NUM with the value of 2010. Execute the macro so that the value of CUSTTYPE resolves to Orion Club members low activity in the macro call.



1. **Using a Table Lookup Application** 
   1. Using **orion.country**, create a series of macro variables in which the name of the macro variable is the country abbreviation (**Country**) and the value of the macro variable is the country name (**Country\_Name**). Use a %put statement to display the macro variables.
   2. Copy the program shown below into the Editor window.

**%let code=AU;**

**proc print data=Orion.Employee\_Addresses;**

**var Employee\_Name City;**

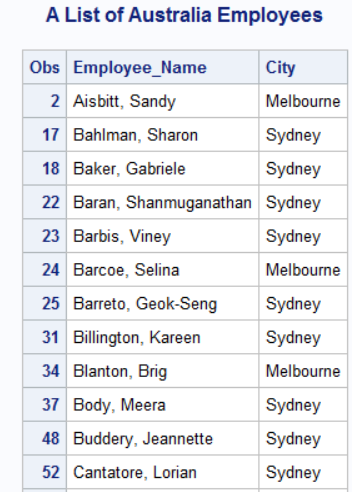
**where Country="&code";**

**title "A List of *xxxxx* Employees";**

**run;**

* 1. Use indirect macro variable referencing to replace the ***xxxxx*** with the appropriate country name.

(Partial output):



1. **Resolving Macro Variables with the SYMGET Function**
   1. Copy the program shown below into the Editor window to create a series of macro variables containing the customer type. Submit the first DATA step and the %PUT statement   
      and examine the results.

**data \_null\_;**

**set orion.customer\_type;**

**call symputx('type'||left(Customer\_Type\_ID), Customer\_Type);**

**run;**

**%put \_user\_;**

**data us;**

**set orion.customer;**

**where Country="US";**

**keep Customer\_ID Customer\_Name Customer\_Type\_ID;**

**run;**

**proc print data=us noobs;**

**title "US Customers";**

**run;**

* 1. Modify the second DATA step to create a new variable named **CustType** that contains the value of the macro variable TYPE*xxxx* created in part **a**. Add the new variable to the KEEP statement.