



## Activities Week 3

If you restarted your SAS session since the last exercise, open and submit the **libname.sas** program .

### 1. Displaying orion.order\_fact with the PRINT Procedure

- Retrieve the starter program **L3\_A1.sas**. Run the program and view the output. Observe that there are 617 observations. Observations might be displayed over two lines, depending on output settings.
- Add a **SUM** statement to display the sum of **Total\_Retail\_Price**. The last several lines of the report are shown below.

Obs	Order_ Type	Product_ID	Quantity	Total_Retail_ Price	CostPrice_ Per_Unit	Discount
610	1	240700100007	2	\$45.70	\$9.30	.
611	1	240700100017	2	\$19.98	\$11.40	40%
612	1	240700400003	2	\$24.80	\$5.60	.
613	1	240800100042	3	\$760.80	\$105.30	.
614	1	240500200016	3	\$95.10	\$14.50	.
615	1	240500200122	2	\$48.20	\$11.50	.
616	1	240700200018	4	\$75.20	\$10.30	.
617	1	220101400130	2	\$33.80	\$5.70	.
				=====		
				\$100,077.46		

- Add a **WHERE** statement to select only the observations with **Total\_Retail\_Price** more than 500. Submit the program. Verify that 35 observations are displayed.

What do you notice about the Obs column? \_\_\_\_\_

Did the sum of **Total\_Retail\_Price** change to reflect only the subset? \_\_\_\_\_

- Add an option to suppress the Obs column. Verify that there are 35 observations in the results.

How can you verify the number of observations in the results? \_\_\_\_\_

- Add an **ID** statement to use **Customer\_ID** as the identifying variable. Submit the program. The results contain 35 observations.

How did the output change? \_\_\_\_\_

- Add a **VAR** statement to display **Customer\_ID**, **Order\_ID**, **Order\_Type**, **Quantity**, and **Total\_Retail\_Price**.

What do you notice about **Customer\_ID**? \_\_\_\_\_

- Modify the **VAR** statement to address the issue with **Customer\_ID**.

## 2. Displaying orion.customer\_dim with the PRINT Procedure

- Write a PRINT step to display **orion.customer\_dim**.
- Modify the program to display a subset of **orion.customer\_dim** by selecting only the observations for customers between the ages of 30 and 40. Also, suppress the Obs column. The resulting report should contain 17 observations.
- Add a statement to use **Customer\_ID** instead of Obs as the identifying column. Submit the program and verify the results.
- Add a statement to limit the variables to those shown in the report below.

Customer_			
Customer_ID	Customer_Name	Age	Customer_Type
4	James Kvarniq	33	Orion Club members low activity
9	Cornelia Krah1	33	Orion Club Gold members medium activity
11	Elke Wallstab	33	Orion Club members high activity
...	...	...	...
54655	Lauren Marx	38	Internet/Catalog Customers
70201	Angel Borwick	38	Orion Club Gold members low activity

## 3. Producing a Default Listing Report of orion.order\_fact



This exercise assumes that you are creating LISTING output in the SAS windowing environment.

- Produce a default listing report of **orion.order\_fact**. The output might wrap onto a second line.
- Investigate the use of the LINESIZE= SAS system option to adjust the width of the lines. What are the minimum and maximum values for the LINESIZE= option? \_\_\_\_\_

Submit an OPTIONS statement with LINESIZE= set to the highest allowed value. Resubmit the step, and observe the horizontal scroll bar, if it is displayed.

Reset the line size to 96 when you are finished.

- Another way to create compact output is to request vertical headings. Investigate the HEADING= option in the PROC PRINT statement, and then experiment with it to generate vertical headings and then horizontal headings.

How do you specify vertical headings? \_\_\_\_\_

How do you specify horizontal headings? \_\_\_\_\_

## 4. Producing a Default Listing Report of orion.product\_dim



This exercise assumes that you are creating LISTING output in the SAS windowing environment.

- Produce a default listing report to display **orion.product\_dim**. Notice that the column width varies from one page to the next, depending on the width of the values that are displayed on each page.

- b. Investigate the WIDTH= option of the PROC PRINT statement, and modify the program to use this option with a value of UNIFORM.

How are the results different when you use the WIDTH=UNIFORM option? \_\_\_\_\_

- c. Why might the procedure run more slowly with this option? \_\_\_\_\_

- d. How can you save computer resources and still display columns consistently across pages? \_\_\_\_\_

## 5. Sorting orion.employee\_payroll and Displaying the New Data Set

- a. Open **L3\_A5.sas**. Add a PROC SORT step before the PROC PRINT step to sort **orion.employee\_payroll** by **Salary**. Place the sorted observations into a temporary data set named **sort\_salary**.
- b. Modify the PROC PRINT step to display the new data set. Verify that your output matches the report below.

Obs	Employee_ID	Employee_ Gender	Salary	Birth_ Date	Employee_ Hire_Date	Employee_ Term_Date	Marital_ Status	Dependents
1	121084	M	22710	3150	12784	.	M	3
2	120191	F	24015	1112	17167	17347	S	0
...								
422	120261	M	243190	4800	11535	.	O	1
423	120262	M	268455	5042	11932	.	M	2
424	120259	M	433800	2946	12297	.	M	1

## 6. Sorting orion.employee\_payroll and Displaying Grouped Observations

- a. Open **p104e06**. Add a PROC SORT step before the PROC PRINT step to sort **orion.employee\_payroll** by **Employee\_Gender**, and within gender, by **Salary** in descending order. Place the sorted observations into a temporary data set named **sort\_salary2**.
- b. Modify the PROC PRINT step to display the new data set with the observations grouped by **Employee\_Gender**.

----- Employee_Gender=F -----							
Obs	Employee_ID	Salary	Birth_ Date	Employee_ Hire_Date	Employee_ Term_Date	Marital_ Status	Dependents
1	120260	207885	3258	10532	.	M	2
2	120719	87420	4770	14641	.	M	1
3	120661	85495	-400	10227	17347	M	3
...							
190	120196	24025	10257	17167	17347	S	0
191	120191	24015	1112	17167	17347	S	0
----- Employee_Gender=M -----							

Obs	Employee_ID	Salary	Birth_ Date	Employee_ Hire_Date	Employee_ Term_Date	Marital_ Status	Dependents
192	120259	433800	2946	12297	.	M	1
193	120262	268455	5042	11932	.	M	2
...							
423	120190	24100	10566	17837	18017	M	2
424	121084	22710	3150	12784	.	M	3

## 7. Sorting orion.employee\_payroll and Displaying a Subset of the New Data Set

- Sort **orion.employee\_payroll** by **Employee\_Gender**, and by descending **Salary** within gender. Place the sorted observations into a temporary data set named **sort\_sal**.
- Print a subset of the **sort\_sal** data set. Select only the observations for active employees (those without a value for **Employee\_Term\_Date**) who earn more than \$65,000. Group the report by **Employee\_Gender**, and include a total and subtotals for **Salary**. Suppress the Obs column. Display only **Employee\_ID**, **Salary**, and **Marital\_Status**. The results contain 18 observations.

----- Employee_Gender=F -----			
	Employee_ID	Salary	Marital_ Status
	120260	207885	M
	120719	87420	M
	...		
	120677	65555	M
	-----	-----	
	Employee_Gender	605190	
----- Employee_Gender=M -----			
	Employee_ID	Salary	Marital_ Status
	120259	433800	M
	120262	268455	M
	...		
	120268	76105	S
	-----	-----	
	Employee_Gender	2072410	
		=====	
		2677600	

## 8. Retaining the First Observation of Each BY Group

- Sort **orion.orders** by **Customer\_ID**. Place the sorted observations in a temporary data set.
- Display the sorted data set. The resulting report should contain 490 observations. **Customer\_ID** is listed multiple times for customers that placed more than one order.

- c. Investigate an option that causes PROC SORT to retain only the first observation in each BY group.
- d. Add the appropriate option to the PROC SORT step to retain only the first observation in each BY group. The results contain 75 observations with no duplicate values for **Customer\_ID**.
- e. Explore the DUPOUT= option to write duplicate observations to a separate output data set.

## 9. Displaying Titles and Footnotes in a Detail Report

- a. Open and submit **L3\_A9.sas** to display all observations for Australian Sales Rep IVs
- b. Add a VAR statement to display only the variables shown in the report below.
- c. Add TITLE and FOOTNOTE statements to include the titles and footnotes shown in the report below.
- d. Submit the program and verify the output. The results contain five observations as shown below.
- e. Submit a null TITLE and null FOOTNOTE statement to clear all titles and footnotes.

Australian Sales Employees Senior Sales Representatives					
Obs	Employee_ID	First_ Name	Last_Name	Gender	Salary
7	120125	Fong	Hofmeister	M	32040
10	120128	Monica	Kletschkus	F	30890
17	120135	Alexei	Platts	M	32490
41	120159	Lynelle	Phoumirath	F	30765
48	120166	Fadi	Nowd	M	30660
Job_Title: Sales Rep. IV					

## 10. Displaying Column Headings in a Detail Report

- a. Open and submit **L3\_A10.sas** . Modify the program to define and use the following labels:

Variable	Label
<b>Employee_ID</b>	Employee ID
<b>First_Name</b>	First Name
<b>Last_Name</b>	Last Name
<b>Salary</b>	Annual Salary

Submit the program and verify the output.

Entry-level Sales Representatives				
Employee ID	First Name	Last Name	Gender	Annual Salary
121023	Shawn	Fuller	M	26010
121028	William	Smades	M	26585

121029	Kuo-Chung	Mcelwee	M	27225
...				
121138	Hershell	Tolley	M	27265
121140	Saunders	Briggi	M	26335
Job_Title: Sales Rep. I				

- b. Modify the program to use a blank space as the SPLIT= character to generate two-line column headings. Submit the modified program and verify that two-line column labels are displayed.

Entry-level Sales Representatives				
Employee ID	First Name	Last Name	Gender	Annual Salary
121023	Shawn	Fuller	M	26010
121028	William	Smades	M	26585
121029	Kuo-Chung	Mcelwee	M	27225
...				
121138	Hershell	Tolley	M	27265
121140	Saunders	Briggi	M	26335
Job_Title: Sales Rep. I				

## 11. Writing an Enhanced Detail Report

- a. Write a program to display a subset of **orion.employee\_addresses** as shown below. The program should sort the observations by **State**, **City**, and **Employee\_Name** and then display the sorted observations grouped by **State**. The resulting report should contain 311 observations.

US Employees by State			
----- State=CA -----			
Employee ID	Name	City	Zip Code
120656	Amos, Salley	San Diego	92116
120759	Apr, Nishan	San Diego	92071
121017	Arizmendi, Gilbert	San Diego	91950
121062	Armant, Debra	San Diego	92025
121049	Bataineh, Perrior	San Diego	92126
...			