Print

Summary of Lesson 4: Producing Detail Reports

This summary contains topic summaries, syntax, and sample programs.

Topic Summaries

To go to the movie where you learned a task or concept, select a link.

Subsetting Report Data

You can use the VAR statement in a PROC PRINT step to subset the variables in a report. You specify the variables to include and list them in the order in which they are to be displayed.

You can use the SUM statement in a PROC PRINT step to calculate and display report totals for the requested numeric variables.

```
PROC PRINT DATA=SAS-data-set;
      VAR variable(s);
      SUM variable(s);
RUN;
```

The WHERE statement in a PROC PRINT step subsets the observations in a report. When you use a WHERE statement, the output contains only the observations that meet the conditions specified in the WHERE expression. This expression is a sequence of operands and operators that form a set of instructions that define the condition. The operands can be constants or variables. Remember that variable operands must be defined in the input data set. Operators include comparison, arithmetic, logical, and special WHERE operators.

```
WHERE where-expression;
```

You can use the ID statement in a PROC PRINT step to specify a variable to print at the beginning of the row instead of an observation number. The variable that you specify replaces the Obs column.

```
ID variable(s);
```

Sorting and Grouping Report Data

The SORT procedure sorts the observations in a data set. You can sort on one variable or multiple variables, sort on character or numeric variables, and sort in ascending or descending order. By default, SAS replaces the original SAS data set unless you use the OUT= option to specify an output data set. PROC SORT does not generate printed output.

Every PROC SORT step must include a BY statement to specify one or more BY variables. These are variables in the input data set whose values are used to sort the data. By default, SAS sorts in ascending order, but you can use the keyword DESCENDING to specify that the values of a variable are to be sorted in descending order. When your SORT step has multiple BY variables, some variables can be in ascending and others in descending order.

You can also use a BY statement in PROC PRINT to display observations grouped by a particular variable or variables. The groups are referred to as BY groups. Remember that the input data set must be sorted on the variables specified in the BY statement.

```
PROC SORT DATA=input-SAS-data-set
                  <OUT=ouput-SAS-data-set>;
       BY <DESCENDING> by-variable(s);
RUN;
```

Enhancing Reports

You can enhance a report by adding titles, footnotes, and column labels. Use the global TITLE statement to define up to 10 lines of titles to be displayed at the top of the output from each procedure. Use the global FOOTNOTE statement to define up to 10 lines of footnotes to be displayed at the bottom of the output from each procedure.

TITLE*n* '*text*';
FOOTNOTE*n* '*text*';

Titles and footnotes remain in effect until you change or cancel them, or until you end your SAS session. Use a null TITLE statement to cancel all titles, and a null FOOTNOTE statement to cancel all footnotes.

Use the LABEL statement in a PROC PRINT step to define temporary labels to display in the report instead of variable names. Labels can be up to 256 characters in length. Most procedures use labels automatically, but PROC PRINT does not. Use the LABEL option in the PROC PRINT statement to tell SAS to display the labels. Alternatively, the SPLIT= option tells PROC PRINT to use the labels and also specifies a split character to control line breaks in column headings.

```
PROC PRINT DATA=SAS-data-set LABEL;
      LABEL variable='label'
                variable='label'
               ... ;
RUN;
```

SPLIT='*split-character*';

Sample Programs

Subsetting Your Report

```
proc print data=orion.sales;
   var Last_Name First_Name Salary;
   sum Salary;
run;
```

Selecting Observations

```
proc print data=orion.sales noobs;
   var Last_Name First_Name Salary Country;
   where Country='AU' and Salary<25500;
run;
```

Using the CONTAINS Operator

```
proc print data=orion.sales noobs;
   var Last_Name First_Name Country Job_Title;
   where Country='AU' and Job_Title contains 'Rep';
run;
```

Subsetting Observations and Replacing the Obs Column

```
proc print data=orion.customer_dim;
   where Customer_Age=21;
   id Customer_ID;
   var Customer_Name
       Customer_Gender Customer_Country
       Customer_Group Customer_Age_Group
       Customer_Type;
run;
```

Sorting a Data Set

```
proc sort data=orion.sales
          out=work.sales_sort;
   by Salary;
run;

proc print data=work.sales_sort;
run;
```

Sorting a Data Set by Multiple Variables

```
proc sort data=orion.sales
          out=work.sales2;
   by Country descending Salary;
run;

proc print data=work.sales2;
run;
```

Grouping Observations in Reports

```
proc sort data=orion.sales
          out=work.sales2;
   by Country descending Salary;
run;

proc print data=work.sales2;
   by Country;
run;
```

Displaying Titles and Footnotes in a Report

```
title1 'Orion Star Sales Staff';
title2 'Salary Report';
footnote1 'Confidential';

proc print data=orion.sales;
   var Employee_ID Last_Name Salary;
run;

proc print data=orion.sales;
   var Employee_ID First_Name Last_Name Job_Title Hire_Date;
run;
```

Changing and Canceling Titles and Footnotes

```
title1 'Orion Star Sales Staff';
title2 'Salary Report';
footnote1 'Confidential';

proc print data=orion.sales;
   var Employee_ID Last_Name Salary;
run;

title1 'Employee Information';
proc print data=orion.sales;
   var Employee_ID First_Name Last_Name Job_Title Hire_Date;
run;
```

Displaying Labels in a Report

```
title1 'Orion Star Sales Staff';
title2 'Salary Report';
footnote1 'Confidential';

proc print data=orion.sales label;
   var Employee_ID Last_Name Salary;
   label Employee_ID = 'Sales ID'
         Last_Name = 'Last Name'
         Salary = 'Annual Salary';
run;
title;
footnote;
```

SAS Programming 1: Essentials

Close