

Summarizing a Data Table in a Formal Report

Previously the MEANS, UNIVARIATE, and FREQ procedures were used to compute some summary statistics for a data table. There were some very basic forms of tables for displaying this summary information in the output from these procedures. These procedures will be revisited, and the REPORT procedure will be presented to demonstrate how summary information can be displayed in a formal report.

Recall the FREQ procedure was used to "count" the number of distinct levels observed for a variable.

Consider the following data table containing student names, ID numbers, group #, homework (HW) scores and exam scores (EX).

Student	ID#	Group	HW1	HW2	EX1	HW3	HW4	EX2
Total			75	110	100	50	25	100
Greg	101	1	71	88	93	46	23	88
Marsha	381	2	64	96	95	48	25	94
Peter	987	2	68	75	67	35	12	60
Bobby	579	3	55	75	81		17	82
Jan	239	1	70	79	77	38	23	77
Cindy	128	3	67	83	74	42	20	92

Objective 1: Compute the total scores for the homework, exams, and for the course for each student. Missing scores should be counted as a zero for that item.

Objective 2: Assign letter grades for each student based on the grading scale:

Lowest A 90%
 Lowest B 80%
 Lowest C 70%
 Lowest D 60%

Print the data table and examine the results for accuracy.

```
DATA gradebook;
INPUT Student $ ID Group HW1 HW2 EX1 HW3 HW4 EX2 ;
HWTot1 = SUM(HW1, HW2, HW3, HW4) ;
EXTot1 = SUM(EX1, EX2) ;
CourseTL = SUM(HWTot1, EXTot1) ;
IF CourseTL ge 414 THEN Grade = "A" ;
ELSE IF 368 LE CourseTL LE 413 THEN Grade="B" ;
ELSE IF 322 LE CourseTL LE 367 THEN Grade="C" ;
ELSE IF 276 LE CourseTL LE 321 THEN Grade="D" ;
ELSE Grade="F" ;
LABEL HWTot1 = "Homework Point Total"
EXTot1 = "Exam Score Total"
CourseTL= "Point Total for the Course" ;
```

```
DATALINES;
```

Total	.	.	75	110	100	50	25	100
Greg	101	1	71	88	93	46	23	88
Marsha	381	2	64	96	95	48	25	94
Peter	987	2	68	75	67	35	12	60
Bobby	579	3	55	75	81	.	17	82
Jan	239	1	70	79	77	38	23	77
Cindy	128	3	67	83	74	42	20	92

```
;
```

```
PROC PRINT DATA=gradebook LABEL ;
```

```
TITLE 'Objectives 1 and 2';
```

```
TITLE2 'PRINT Procedure with LABEL Option';
```

```
RUN;
```

```
QUIT;
```

Recall: How would the computed *HWTot* and *EXTot* calculations be written if the addition of scores were programmed rather than using the *SUM* function?

How are the *SUM* function and the addition of scores affected by missing values?

Objective 3: Count the frequency of each letter grade and include the number of levels.

CAUTION: Do not include the TOTAL grade book entry as a student in the course.

```
PROC FREQ DATA=gradebook NLEVELS ;
```

```
WHERE student NE "Total" ;
```

```
TABLES Grade ;
```

```
TITLE 'Objective 3';
```

```
TITLE2 'FREQ Procedure with NLEVELS Option';
```

```
TITLE3 'One-way Table';
```

```
RUN;
```

Objective 4: Examine the cross-tabular results of grades earned by each Group.

```
PROC FREQ DATA=gradebook NLEVELS ;
```

```
WHERE student NE "Total" ;
```

```
TABLES Group*Grades ;
```

```
TITLE 'Objective 4';
```

```
TITLE2 'FREQ Procedure with NLEVELS Option';
```

```
TITLE3 'Two-way Table';
```

```
RUN;
```

Recall there are options for the TABLES statement that restricts or modifies the information in each cell of the table.

Objective 5: Display the cross-tabulation results in a listing (vertical) format.

```
PROC FREQ DATA=gradebook NLEVELS ;
WHERE student NE "Total" ;
TABLES Group*Grade / CROSSTAB;
TITLE 'Objective 5';
TITLE2 'FREQ Procedure with NLEVELS Option';
TITLE3 'Two-way Table With CROSSTAB Option';
RUN;
```

The MEANS procedure also allows summary data to be displayed. The sample statistics that are presented can be controlled for the entire sample or for subgroups within the sample.

Objective 6: Using the MEANS procedure, compute the mean, standard deviation and sample size for homework, exam, and course totals for all students.

```
PROC MEANS DATA=gradebook MEAN STD N ;
WHERE student NE "Total" ;
VAR hwtotals extotals courset1 ;
TITLE 'Objective 6';
TITLE2 'MEANS Procedure with no CLASS Statement';
RUN;
```

Objective 7: Redo Objective 6 computing the summary scores for each group in a single table.

```
PROC MEANS DATA=gradebook MEAN STD N ;
WHERE student NE "Total" ;
CLASS group ;
VAR hwtotals extotals courset1 ;
TITLE 'Objective 7';
TITLE2 'Group is identified as the CLASS variable';
RUN;
```

The REPORT Procedure

The REPORT Procedure may be regarded as a more flexible version of the PRINT procedure.

A simple syntax of the REPORT procedure appears below. There are some statements and options available that are not shown here. This is merely an introduction to the procedure.

```
PROC REPORT <options> ;
BY <DESCENDING> variable1 . . . <DESCENDING> variablen;           SORT first!
COLUMN specify column variables – order is important ;
DEFINE report-item / <usage>      <attribute(s)> <option(s)>  <justification>
                        <COLOR=color>      <'column-header-1' <... 'column-header-n'>> <style>;
RBREAK  BEFORE|AFTER  / <options> ;
```

where the options on the PROC REPORT statement include:

NOWINDOWS or NOWD	displays the resulting output from this procedure in the Output window. To open REPORT in an interactive window, the option WINDOWS or WD can be used. WINDOWS or WD is the default setting.
COLUMN statement	This statement selects the variables to be included in the report. The order of the variables specified in this statement determines the order the variables appear in the report.
DEFINE statement	<p>This statements modifies the basic column reporting by allowing summary statistics to be computed for numeric variables, specify column headers and the width of columns in the report, and change the order of the rows in the report.</p> <p>ORDER – orders the rows in the output listing</p> <p>WIDTH = # - specifies the width of the column for the report-item</p> <p>GROUP – defines the report-item as a grouping</p> <p>"column-header" – labels the columns with the specified text utilizing the specified width</p> <p>SUM, MEAN, N, MAX, MIN are selected statistics. Choose only one per variable to summarize across the group or total list. SUM is the default.</p>
RBREAK statement	<p>This statement will include a grand total at the top (BEFORE) or at the bottom (AFTER) of the report. Options include:</p> <p>SUMMARIZE – requests the total to be printed</p> <p>OL – prints a single line <i>over</i> the total</p> <p>DOL – prints a double line <i>over</i> the total</p> <p>UL – prints a single line <i>under</i> the total</p> <p>DUL – prints a double line <i>under</i> the total</p>

Objective 8: Produce a report containing the totals of the homework and exams, the course total and the letter grade for each student in the course. Include a grand total for each of the numeric variables at the top of the list. Compare the output with a PRINT procedure used to accomplish the same task.

```
PROC REPORT DATA=gradebook NOWD ;
WHERE student NE "Total" ;
COLUMN student hwttotal exttotal courset1 grade ;
RBREAK BEFORE / SUMMARIZE UL ;
TITLE 'Objective 8';
TITLE2 'Default Output for REPORT Procedure';
```

```
PROC PRINT DATA=gradebook ;
WHERE student NE "Total" ;
VAR student hwttotal exttotal courset1 grade ;
SUM hwttotal exttotal courset1 ;
TITLE2 'Default Output for PRINT Procedure';
RUN;
```

Objective 9: Modify the report in Objective 8 so that the grades are listed from highest to lowest; that is, A's first, B's second, and so on. Subtotal the numeric variables for each letter grade. Again compare REPORT versus PRINT.

```
PROC SORT DATA=gradebook; BY grade DESCENDING courset1 ;
PROC REPORT DATA=gradebook NOWD ;
WHERE student NE "Total" ;
BY grade ;
COLUMN student hwttotal exttotal courset1 grade ;
RBREAK AFTER / SUMMARIZE DOL ;
TITLE 'Objective 9';
RUN;
```

```
PROC PRINT DATA=gradebook ;
WHERE student NE "Total" ;
BY grade ;
VAR student hwttotal exttotal courset1 grade ;
SUM hwttotal exttotal courset1 ;
RUN;
```

Objective 10: Print a report of the homework scores and homework total. Label all columns. HW1 and HW2 were not labeled in the initial DATA step. Note how they can be labeled in DEFINE statement.

```
PROC REPORT DATA=gradebook NOWD;
WHERE student NE "Total" ;
```

```

COLUMN student hw1 hw2 hwtotal ;
DEFINE hw1 / WIDTH = 12 'Homework 1' ;
DEFINE hw2 / "Homework 2" ;
DEFINE hwtotal / WIDTH=6 ;
TITLE 'Objective 10';
RUN;

```

Objective 11: Redo the task in Objective 10. Change the third DEFINE statement to:

```

DEFINE hwtotal / ORDER WIDTH=6 ;

```

This will put the observations in order from low to high according to the values in the HWTOTAL variable.

Objective 12: Create a list report of students where the students names, ID#'s, and grades are listed according to project group variable.

```

PROC REPORT DATA=gradebook NOWD;
WHERE student NE "Total" ;
COLUMN group student id grade;
DEFINE group / GROUP WIDTH = 12 'Project Group' ;
DEFINE id / "ID #" ;
DEFINE grade / WIDTH = 6 ;
TITLE 'Objective 12';
RUN;

```

The uppercase GROUP in the DEFINE statement is an option for that statement. "group" (lowercase) is a variable in the grade book data table.

Notice if you leave the variable group out of the COLUMN statement, the procedure will not run since group has not been requested for the report.

Objective 13: Rerun Objective 12 with a different ordering of variables in the COLUMN statement, such as: COLUMN student id grade group ;

Objective 14: Create a report of the average course total score for each project group.

```

PROC REPORT DATA=gradebook NOWD;
WHERE student NE "Total" ;
COLUMN group courset1;
DEFINE group / GROUP WIDTH = 12 'Project Group' ;
DEFINE courset1 / "Course Total" MEAN ;
TITLE 'Objective 14';
RUN;

```

Note: If student or ID# or any other variable were included in the COLUMN statement that would require each row of the file to be printed, the MEAN option would not compute the MEAN for each project group.

Variations to Objective 14:

1. Delete MEAN from the second DEFINE statement and resubmit. Note the sum for each group is printed.
2. Delete the group variable from the COLUMN statement and delete the DEFINE statement for this variable. Note the requested statistic in the remaining DEFINE statement is computed across all students.

The NOWD option on the PROC REPORT statements allows the output to go to the output window rather than a separate Report window. One can get tables in an HTML format by putting ODS HTML; in a program prior to any REPORT procedure. Remember that this is a global command, and that HTML output will be generated for all procedures until an ODS HTML CLOSE; statement is processed or the SAS session is closed.