

Tapping the Power of PROC TABULATE for Analysis Reports Clients Understand

Edward Jensen, Corbett, OR

ABSTRACT

The reporting capabilities of PROC TABULATE are very powerful. In just a few lines of code, a very detailed analysis can be reported. Knowing how to tap into the power of PROC TABULATE begins with an understanding of the fundamental structure of a TABLE command. From that basis, more detailed reports can be presented. This paper follows that approach. It begins with an overview of TABULATE, laying down the basic structure of a TABLE statement and how it relies on the CLASS and VAR statements. It then demonstrates by example how more detail can be added to the report by creative use of TABLE basics and the addition of formats, appearance options, and some tricks to enhance the report presentation. The statistics N, SUM, MEAN, and the sometimes tricky PCTN and PCTSUM are discussed. Once understood, TABULATE can be quickly coded to generate reports that are easily modifiable for future analysis needs.

INTRODUCTION

The TABULATE procedure is most often called upon to use its powerful cross tabulate abilities to compare the relationship of a few variables. This leaves TABULATE to format its report dependent on the nature of the data.

Understanding the way TABULATE works with class and analytical variables and with some imagination, TABULATE can also be a great tool for creating reports formatted to the clients specification. These reports can be imported to spreadsheet products such as Microsoft Excel.

THE TABLE STATEMENT

The TABLE statement defines the report structure. All other statements support the TABLE statement.

The primary function of the TABLE statement is to describe the three dimensions of the report - rows, columns and pages. The basic format is:

```
TABLE <<page expression,> row
expression,> column expression
</table-option-list>;
```

When one expression is presented, a column

expression is activated. When two expressions, the dimension will be row and column. The first dimension in a three dimension is the page expression.

Using the sample data as used in "SAS Guide to TABULATE Processing", pg. 49 (see Figure 1) a single expression TABLE statement as the following will produce a report with a single row as seen in Figure 2:

```
TABLE (GRADE ALL)*(CLASSRM ALL);
```

By simply changing the * to a comma to indicate a dimension rather than a crossing or nesting, we have the two dimensional results seen in Figure 3:

```
TABLE (GRADE ALL),(CLASSRM ALL);
```

Adding a third dimension - for paging - we have the results seen in Figure 4:

```
TABLE (TEAM ALL),(GRADE ALL),
(CLASSRM ALL)/BOX=_PAGE_;
```

Note how the TABLE BOX parameter wrote the page class value within the upper left box of the table. Normally, it is written just above the box.

The complete PROC TABULATE code at this point is:

```
PROC TABULATE DATA=FUNDRAIS
FORMAT=6.;
CLASS GRADE CLASSRM TEAM;
TABLE (TEAM ALL),(GRADE ALL),
(CLASSRM ALL)/BOX=_PAGE_;
```

Crossing Classification Variables

The CLASS statement specifies GRADE, CLASSRM, and TEAM as classification variables. These break the report into various columns, rows, and pages based on their respective values.

In these examples, we have grouped the class variables with the ALL universal class variable to give subtotals and total categories. Categories at the two dimension level (GRADE ALL), (CLASSRM ALL) will be:

```
GRADE*CLASSRM GRADE*ALL
ALL*CLASSRM ALL*ALL
```

Adding the third dimension with (TEAM ALL), the above two lines are each repeated for TEAM and

Coder's Corner

ALL:

```
TEAM*GRADE*CLASSRM    TEAM*GRADE*ALL
TEAM*ALL*CLASSRM       TEAM*ALL*ALL
ALL*GRADE*CLASSRM      ALL*GRADE*ALL
ALL*ALL*CLASSRM        ALL*ALL*ALL
```

TABLE Statistics

In the previous examples, we have relied on the TABULATE statistic default of N - the number of non-missing observations. Now we will go back to the two dimensional report and calculate SUM and MEANS of TOTSALE. To tell TABULATE that TOTSALE is an analysis variable, it is specified in a VAR statement:

```
VAR TOTSALE;

TABLE (GRADE ALL), (CLASSRM
ALL) *TOTSALE*(N SUM MEAN) /RTS=7;
```

The results are displayed in Figure 5. Notice that an RTS parameter was given. This reduced the "row title space" from one-fourth of the LINESIZE value to just 7 which includes the two vertical bar characters on each side for a net of 5 positions. The global FORMAT was also changed from FORMAT=6. to FORMAT=4. to narrow all columns.

Grouping Columns by Statistic

Now to see how the report can be modified, we'll change the structure of the TABLE statement. To group the N, SUM, and MEANS values, invert the order of the column expression. Results are given in Figure 6:

```
TABLE (GRADE ALL), (N SUM MEAN)
*TOTSALE *(CLASSRM ALL) /RTS=7;
```

FORMATTING AND LABELING REPORT ELEMENTS

At this point we will make use of some of the format modifications available. Because TOTSALE is a common heading across the top of the report, we will move it to the top of the headings to span all columns. We can also change the title of the keywords N, SUM, MEANS, and ALL.

```
TABLE (GRADE ALL), TOTSALE *(N SUM
MEAN) *(CLASSRM='Class Room'
ALL*F=6.) /RTS=7

KEYLABEL ALL='Total' N='Count';

LABEL TOTSALE='Total Sale';
```

Here the N and ALL statistics were retitled with the KEYLABEL statement. The class variable CLASSRM was labeled in the TABLE statement.

Using another method, TOTSALE was labeled with a LABEL statement. The ALL or "Total" universal class variable was formatted to 6. in the TABLE statement so the "Total" label would not wrap in the column heading. The results are in Figure 7.

Grouping Rows by Statistic

To further show the flexibility in designing a report layout, we will move the statistic specifications to the head of the row dimension. This will group the rows by each statistic. See Figure 8.

```
PROC TABULATE DATA=FUNDRAIS FORMAT=4.
NOSEPS;

TABLE (N SUM MEAN) * (GRADE ALL),
TOTSALE *(CLASSRM='Class Room'
ALL*F=6.) /RTS=15
```

In this example, we also used the NOSEPS option to remove the horizontal divider lines in the report. This reduces vertical space and is very desirable for reports that are imported to a spreadsheet.

We could have just as easily inverted the statistic group (N SUM MEAN) with the row class group (GRADE ALL) and produced a report with rows by statistic grouped by grade.

VAR VS. CLASS VARIABLES

Variables can be used as analysis variables or class variables. As shown earlier, TABULATE uses CLASS variables to develop columns, rows, and pages based on the data values. VAR variables are used to calculate statistics. How the variable is expressed in the TABLE statement does not change it's assignment by the CLASS or VAR statement.

In the following example, variable TOTSALE has been changed to a CLASS variable. The structure of the report as seen in Figure 9 has also been modified to create a report more suited to this representation.

```
PROC TABULATE DATA=FUNDRAIS FORMAT=5.
NOSEPS;

CLASS GRADE CLASSRM TOTSALE;

TABLE TOTSALE ALL, (GRADE ALL)*
(CLASSRM='Class Room' ALL*F=6.)
/RTS=8;
```

A row is written for each unique value of TOTSALE. Since no analysis variables are specified, the N statistic is used. This example shows that great care must be taken when selecting a CLASS variable. If there were more values for TOTSALE, a row would be created for each unique value - to the penny!

USING FORMATS TO GROUP DATA

CLASS variables with many values such as TOTSALE, can readily be grouped by use of a format. To report TOTSALE by ranges, write a format using PROC FORMAT:

```
PROC FORMAT;
VALUE TOTSLGRP 0-<10=' 0-10' 10-
<20='10-20' 20-<30='20-30';
VALUE GRADEID 4='Grade 4' 5='Grade
5';
```

A format for Grade is also given. Now assign the format to TOTSALE in the TABULATE procedure:

```
TABLE TOTSALE ALL, N*(GRADE=' ' ALL)*
(CLASSRM='Class Room' ALL*F=6.)
/RTS=8;

FORMAT TOTSALE TOTSLGRP. GRADE
GRADEID.;
```

As shown in Figure 10, just three unique rows are reported for the three value ranges of TOTSALE. The FORMAT statement only applies to CLASS variables. To format an analysis variable, use the *'f=format'* as shown above.

In this example, GRADE was also formatted with a more descriptive title so one of the heading boxes could be dropped by giving a temporary blank label to GRADE. The placement of statistic N in the heading rows was controlled by locating "N*" before GRADE.

VARIABLES USED FOR BOTH CLASS AND ANALYSIS

Now that TOTSALE has been grouped by amounts, we want to see the sum and average of TOTSALE within each group. This requires TOTSALE to be considered as both a class and analysis variable. Since a variable must be either class or analysis, we create a second variable TOTSALEV equal to TOTSALE.

```
DATA FUNDRAIS;
SET FUNDRAIS;
TOTSALEV=TOTSALE;
```

Changing the TABLE statement to add TOTSALEV as an analysis variable:

```
VAR TOTSALEV;

TABLE TOTSALE* (GRADE ALL) ALL,
TOTSALEV * (CLASSRM='Class Room'
ALL*F=6.)* (SUM*F=3. MEAN*F=4.)
/RTS=15;
```

We see the results in Figure 11.

PCTN AND PCTSUM STATISTICS

The TABLE statement has many more statistics available than the SUM and MEANS used above. Of them PCTN and PCTSUM are worth noting. PCTN calculates the percentage a frequency (N) represents of another, default is the complete report PCTSUM is similar but is based on the SUM. Figure 12 shows how another column was added using the PCTSUM. Notice value 100 in the lower right cell.

```
TABLE TOTSALE* (GRADE ALL) ALL,
TOTSALEV * (CLASSRM=' ' ALL*F=6.)*
(SUM*F=3. PCTSUM*F=6.1 MEAN*F=4.)
/RTS=14;
```

The basis for a denominator is very flexible. In the following example, each column is reported as 100% so the crossings representing all columns (TOTSALE*GRADE and TOTSALE) are given. Since ALL is reported, it must also be specified. Figure 13 contains the results.

```
TABLE TOTSALE* (GRADE ALL) ALL,
TOTSALEV * (CLASSRM=' ' ALL*F=6.)*
(SUM*F=3. PCTSUM<TOTSALE*GRADE
TOTSALE ALL>*F=6.1 MEAN*F=4.)
/RTS=14;
```

Beware! Cells that result from a cross with ALL will report value 100 as a default if the cross representing that cell is not given in the denominator. Be sure to check all results. A 100 is calculated independently of the subordinate cells.

CONCATENATING VARIABLES

In the previous examples, rows and columns were developed by crossing variables. Rows and columns can be added by simply concatenating variables. Just list variables side-by-side in the TABLE statement as shown in the following:

```
options missing=' ';

TABLE ALL='====Total====' TEAM GRADE
CLASSRM TOTSALE, TOTSALEV *
(CLASSRM=' ' ALL*F=6.)* (SUM*F=3.
PCTSUM< ALL*CLASSRM TEAM*CLASSRM
TEAM*ALL GRADE*CLASSRM GRADE*ALL
CLASSRM*CLASSRM CLASSRM*ALL
TOTSALE*CLASSRM TOTSALE*ALL ALL
>*F=6.1 MEAN*F=4.) /RTS=14;

label team    ='====Team====';
label grade   ='====Grade====';
label classrm='====Class====';
label totslev='====Total Sales====';
```

Special labels were provided for easier reading and the character for printing missing values was set to a blank so a period is not written for missing values. The large denominator is due to the number of variables reported. Each crossing must be represented. The report is in Figure 14.

SAME CATEGORY DENOMINATOR

Rather than listing the appropriate crossings of class variables in the denominator, the denominator can be a VAR variable. If a single variable is given as in Figure 15, the PCTSUM is based on that variable's value in the same category.

```
TABLE ALL='====Total====' TEAM GRADE ,
(PCTSUM< TOTSALEV >*F=7.1 ) *
(CLASSRM=' ' ALL*F=7.1) * (PENCILS
TABLETS ) /RTS=14;
```

ADDITIONAL FORMATTING OPTIONS

The previous examples have shown us the wide variety of options and techniques available for making reports meaningful to the client. Several more formatting options are worth considering.

The FORMCHAR TABULATE option specifies characters for table outlines and dividers. The default is: FORMCHAR='|---|+|---'. Setting all 11 characters to blanks will print the report without the TABULATE look. However, the '|' divider character is very useful as a delimiter when importing to other tools.

To control the sequence of columns and rows, consider the ORDER= TABULATE option. FORMATTED sets the order by the class value's formatted result. FREQ, sets the order by descending frequency count. INTERNAL, the default, orders by sort value.

The TABLE statement option CONDENSE will print as many logical pages on a single physical page as possible reducing the amount of physical pages printed. Very useful for tables with limited rows.

If the table is wrapping its columns onto the next page, try using a page dimension. Breaking the table by using multiple TABLE statements in the same TABULATE step will print multiple tables. Using the CONDENSE option may help by printing the wrapped portion on the same page.

The BY statement will write a new table for each BY group. The file must be sorted or NOTSORTED specified.

Beware of Missing Class Values

If the observation contains missing values for any variable listed in the CLASS statement, the observation is not included in the table unless you specify the MISSING option in the PROC TABULATE statement. The MISSING option will report missing values as a valid level for the class variable. When reviewing your report, confirm it is reporting the expected population.

CONCLUSION

The TABULATE procedure is a very flexible report generator as well as a very capable tool for analysis. Understanding the underlying statistic of the TABLE statement and its supporting statements, a wide range of report formats become available. Because of the ease in making changes to the report, Proc TABULATE becomes an ideal tool for responding to rapidly changing analysis requirements.

REFERENCES

SAS Guide to TABULATE Processing, Second Edition, SAS Institute Inc., Cary, NC, USA, 1990.

TRADEMARKS

SAS is a registered trademark or trademark of SAS Institute Inc. in the USA and other countries.

Other brand and product names are registered trademarks or trademarks of their respective companies.

AUTHOR

Ed Jensen has been programming with SAS Software since 1981 when he was at Consolidated Freightways as project lead on the Workers' Compensation System. Since then, he has applied SAS Software at Freightliner Corporation, Federal Reserve Bank of San Francisco, and currently, Wells Fargo Bank, San Francisco. He has a BA in Chemistry
Address: PO Box 283, Corbett, OR, 97019
E-Mail: 74617.1435@compuserve.com
Phone: (503) 880-5463

Figure 1. Sample Dataset

(Source: SAS Guide to Tabulate Processing, pg. 49)

OBS	TEAM	GRADE	CLASSRM	NAME	PENCILS	TABLETS	TOTSALE
1	BLUE	4	A	ANN	4	8	12
2	RED	4	A	MARY	5	10	15
3	BLUE	4	A	JOHN	6	4	10
4	RED	4	A	BOB	2	3	5
5	BLUE	4	B	FRED	6	8	14
6	RED	4	B	LOUISE	12	2	14
7	BLUE	4	B	ANNETTE	.	9	.
8	RED	4	B	HENRY	8	10	18
9	BLUE	4	C	KATHY	4	7	11
10	BLUE	4	C	JIM	3	9	12
11	RED	4	C	STEVE	13	7	20
12	BLUE	5	A	ANDREW	3	5	8
13	RED	5	A	SAMUEL	12	10	22
14	BLUE	5	A	LINDA	7	12	19
15	RED	5	A	SARA	4	.	.
16	BLUE	5	B	MARTIN	9	13	22
17	RED	5	B	MATTHEW	7	6	13
18	BLUE	5	B	BETH	15	10	25

Figure 2. One Dimension Report

TABLE (GRADE ALL) * (CLASSRM ALL);

GRADE											
4				5			ALL				
CLASSRM				CLASSRM			CLASSRM				
A	B	C	ALL	A	B	ALL	A	B	C	ALL	
N	N	N	N	N	N	N	N	N	N	N	N
4	4	3	11	4	4	8	8	8	3	19	

Figure 3. Two Dimension Report

TABLE (GRADE ALL), (CLASSRM ALL);

GRADE	CLASSRM			
	A	B	C	ALL
	N	N	N	N
4	4	4	3	11
5	4	4	.	8
ALL	8	8	3	19

Figure 4. Three Dimension Report

TABLE (TEAM ALL), (GRADE ALL), (CLASSRM ALL)/BOX=_PAGE_;

TEAM BLUE	CLASSRM			ALL
	A	B	C	
	N	N	N	
GRADE				
4	2	2	2	6
5	2	2	.	4
ALL	4	4	2	10

< PAGE >

TEAM RED	CLASSRM			ALL
	A	B	C	
	N	N	N	
GRADE				
4	2	2	1	5
5	2	2	.	4
ALL	4	4	1	9

< PAGE >

ALL	CLASSRM			ALL
	A	B	C	
	N	N	N	
GRADE				
4	4	4	3	11
5	4	4	.	8
ALL	8	8	3	19

Figure 5. Using Analysis Variable Statistics

TABLE (GRADE ALL), (CLASSRM ALL)*TOTSALE*(N SUM MEAN) /RTS=7;

	CLASSRM											
	A			B			C			ALL		
	TOTSALE			TOTSALE			TOTSALE			TOTSALE		
	N	SUM	MEAN	N	SUM	MEAN	N	SUM	MEAN	N	SUM	MEAN
GRADE												
4	4	42	11	3	46	15	3	43	14	10	131	13
5	3	49	16	4	67	17	.	.	.	7	116	17
ALL	7	91	13	7	113	16	3	43	14	17	247	15

Figure 6. Grouping Columns by Statistic

TABLE (GRADE ALL), (N SUM MEAN) *TOTSALE *(CLASSRM ALL) /RTS=7;

	N				SUM				MEAN			
	TOTSALE				TOTSALE				TOTSALE			
	CLASSRM				CLASSRM				CLASSRM			
	A	B	C	ALL	A	B	C	ALL	A	B	C	ALL
GRADE												
4	4	3	3	10	42	46	43	131	11	15	14	13
5	3	4	.	7	49	67	.	116	16	17	.	17
ALL	7	7	3	17	91	113	43	247	13	16	14	15

Figure 7. Labeling and Selective Formatting

TABLE (GRADE ALL), TOTSALE *(N SUM MEAN) *(CLASSRM='Class Room' ALL*F=6.) /RTS=7

	Total Sale											
	Count				SUM				MEAN			
	Class Room				Class Room				Class Room			
	A	B	C	Total	A	B	C	Total	A	B	C	Total
GRADE												
4	4	3	3	10	42	46	43	131	11	15	14	13
5	3	4	.	7	49	67	.	116	16	17	.	17
Total	7	7	3	17	91	113	43	247	13	16	14	15

Figure 8. Grouping Rows by Statistic

PROC TABULATE DATA=FUNDRAIS FORMAT=4. NOSEPS;
TABLE (N SUM MEAN)* (GRADE ALL), TOTSALE *(CLASSRM='Class Room' ALL*F=6.)/RTS=15;

		Total Sale			
		Class Room			
		A	B	C	Total
Count	GRADE				
	4	4	3	3	10
	5	3	4	.	7
SUM	Total	7	7	3	17
	GRADE				
	4	42	46	43	131
MEAN	5	49	67	.	116
	Total	91	113	43	247
	GRADE				
4		11	15	14	13
	5	16	17	.	17
	Total	13	16	14	15

Figure 9. Var vs. Class Variables

TABLE TOTSALE ALL, (GRADE ALL)* (CLASSRM='Class Room' ALL*F=6.) /RTS=8;

	GRADE										
	4				5			Total			
	Class Room			Total	Class Room		Total	Class Room			Total
	A	B	C		A	B		A	B	C	
	Count	Count	Count	Count	Count	Count	Count	Count	Count	Count	Count
Total Sale											
5											
7	1	.	.	1	.	.	.	1	.	.	1
8	.	.	.	.	1	1	1	1	1	.	1
10	1	.	.	1	.	.	1	1	.	.	1
11	.	.	1	1	.	.	.	.	.	1	1
12	1	.	1	2	.	.	.	1	.	1	2
13	.	.	.	.	.	1	1	.	1	.	1
14	.	2	.	2	.	.	.	.	2	.	2
15	1	.	.	1	.	.	.	1	.	.	1
18	.	1	.	1	.	.	.	.	1	.	1
19	.	.	.	.	1	.	1	1	.	.	1
20	.	.	1	1	.	.	.	.	.	1	1
22	.	.	.	.	1	1	2	1	1	.	2
25	.	.	.	.	.	1	1	.	1	.	1
Total	4	3	3	10	3	4	7	7	7	3	17

Figure 10. Var vs. Class Variables

TABLE TOTSALE ALL, N*(GRADE=' ' ALL)* (CLASSRM='Class Room' ALL*F=6.) /RTS=8;
FORMAT TOTSALE TOTSLGRP. GRADE GRADEID.;

	Counts											
	Grade 4				Grade 5				Total			
	Class Room			Total	Class Room		Total	Class Room			Total	
	A	B	C		A	B		A	B	C		
Total Sale												
0-10	1	.	.	1	1	1	2	2	1	.	3	
10-20	3	3	2	8	1	1	2	4	4	2	10	
20-30	.	.	1	1	1	2	3	1	2	1	4	
Total	4	3	3	10	3	4	7	7	7	3	17	

Figure 11. Variables Used for Both CLASS and Analysis

TABLE TOTSALE* (GRADE ALL) ALL, TOTSALEV * (CLASSRM='Class Room' ALL*F=6 * (SUM*F=3.
MEAN*F=4.) /RTS=15;

		Total Sales							
		Class Room						Total	
		A		B		C			
		SUM	MEAN	SUM	MEAN	SUM	MEAN	SUM	MEAN
Total Sales	GRADE								
0-10	4	5	5	7	7	7	7	5	5
	5	8	8	7	7	7	7	15	8
	Total	13	7	7	7	7	7	20	7
10-20	GRADE								
	4	37	12	46	15	23	12	106	13
	5	19	19	13	13	13	12	32	16
	Total	56	14	59	15	23	12	138	14
20-30	GRADE								
	4	22	22	47	24	20	20	20	20
	5	22	22	47	24	20	20	69	23
	Total	22	22	47	24	20	20	89	22
Total		91	13	113	16	43	14	247	15

Figure 12. PCTN and PCTSUM Statistics

TABLE TOTSALE* (GRADE ALL) ALL, TOTSALEV * (CLASSRM=' ' ALL*F=6.) * (SUM*F=3. PCTSUM*F=6.1
MEAN*F=4.) /RTS=14;

		Total Sales											
		Room A			Room B			Room C			Total		
		SUM	PCTSUM	MEAN	SUM	PCTSUM	MEAN	SUM	PCTSUM	MEAN	SUM	PCTSUM	MEAN
Total Sales	GRADE												
	0-10	4	5	2.0	5	7	2.8	7	7	9.3	12	5	2.0
10-20	5	8	3.2	8	7	2.8	7	7	9.3	12	15	6.1	8
	Total	13	5.3	7	7	2.8	7	7	9.3	12	20	8.1	7
	GRADE												
	4	37	15.0	12	46	18.6	15	23	9.3	12	106	42.9	13
20-30	5	19	7.7	19	13	5.3	13	23	9.3	12	32	13.0	16
	Total	56	22.7	14	59	23.9	15	23	9.3	12	138	55.9	14
	GRADE												
	4	22	8.9	22	47	19.0	24	20	8.1	20	20	8.1	20
Total	5	22	8.9	22	47	19.0	24	20	8.1	20	69	27.9	23
	Total	22	8.9	22	47	19.0	24	20	8.1	20	89	36.0	22
Total		91	36.8	13	113	45.7	16	43	17.4	14	247	100.0	15

Figure 13. PCTN and PCTSUM Statistics - Column Denominator

TABLE TOTSALE* (GRADE ALL) ALL, TOTSALEV * (CLASSRM=' ' ALL*F=6.)* (SUM*F=3.
PCTSUM<TOTSALE*GRADE TOTSALE ALL>*F=6.1 MEAN*F=4.) /RTS=14;

		Total Sales											
		Room A			Room B			Room C			Total		
		SUM	PCTSUM	MEAN	SUM	PCTSUM	MEAN	SUM	PCTSUM	MEAN	SUM	PCTSUM	MEAN
Total Sales	GRADE												
0-10	4	5	5.5	5	7	6.2	7	.	.	.	5	2.0	5
	5	8	8.8	8	7	6.2	7	.	.	.	15	6.1	8
	Total	13	14.3	7	7	6.2	7	.	.	.	20	8.1	7
10-20	GRADE												
	4	37	40.7	12	46	40.7	15	23	53.5	12	106	42.9	13
	5	19	20.9	19	13	11.5	13	.	.	.	32	13.0	16
	Total	56	61.5	14	59	52.2	15	23	53.5	12	138	55.9	14
20-30	GRADE												
	4	22	24.2	22	47	41.6	24	20	46.5	20	20	8.1	20
	5	22	24.2	22	47	41.6	24	20	46.5	20	69	27.9	23
	Total	44	48.4	22	94	83.2	48	40	93.0	40	89	36.0	22
Total	Total	91	100.0	13	113	100.0	16	43	100.0	14	247	100.0	15

Figure 14. Concatenating Variables

TABLE ALL='===Total=== ' TEAM GRADE CLASSRM TOTSALE, TOTSALEV * (CLASSRM=' ' ALL*F=6.)*
(SUM*F=3. PCTSUM< ALL *CLASSRM ALL *ALL TEAM *CLASSRM TEAM *ALL GRADE*CLASSRM GRADE*ALL
CLASSRM* CLASSRM CLASSRM*ALL TOTSALE *CLASSRM TOTSALE *ALL ALL >*F=6.1 MEAN*F=4.) /RTS=14;

		Total Sales											
		Room A			Room B			Room C			Total		
		SUM	PCTSUM	MEAN	SUM	PCTSUM	MEAN	SUM	PCTSUM	MEAN	SUM	PCTSUM	MEAN
===Total===		91	36.8	13	113	45.7	16	43	17.4	14	247	100.0	15
===Team===													
BLUE		49	19.8	12	61	24.7	20	23	9.3	12	133	53.8	15
RED		42	17.0	14	52	21.1	13	20	8.1	20	114	46.2	14
===Grade===													
4		42	17.0	11	46	18.6	15	43	17.4	14	131	53.0	13
5		49	19.8	16	67	27.1	17				116	47.0	17
===Class===													
Room A		91	36.8	13							91	36.8	13
Room B					113	45.7	16				113	45.7	16
Room C								43	17.4	14	43	17.4	14
=Total Sales													
0-10		13	5.3	7	7	2.8	7				20	8.1	7
10-20		56	22.7	14	59	23.9	15	23	9.3	12	138	55.9	14
20-30		22	8.9	22	47	19.0	24	20	8.1	20	89	36.0	22

Figure 15. Same Category Denominator

TABLE ALL='===Total=== ' TEAM GRADE , (PCTSUM< TOTSALEV >*F=7.1) * (CLASSRM=' ' ALL*F=7.1)* (PENCILS TABLETS) /RTS=14;

		Percentage of Total Sales									
		Room A		Room B		Room C		Total			
		PENCILS	TABLETS	PENCILS	TABLETS	PENCILS	TABLETS	PENCILS	TABLETS		
===Total===		42.9	57.1	54.0	46.0	46.5	53.5	48.6	51.4		
===Team===											
BLUE		40.8	59.2	49.2	50.8	30.4	69.6	42.9	57.1		
RED		45.2	54.8	59.6	40.4	65.0	35.0	55.3	44.7		
===Grade===											
4		40.5	59.5	56.5	43.5	46.5	53.5	48.1	51.9		
5		44.9	55.1	52.2	47.8			49.1	50.9		