Writing Reports
with the
SAS® System's
TABULATE Procedure
or
Big Money Proc Tabulate

Ben Cochran
The Bedford Group
bencochran@nc.rr.com

Writing Reports
with the
SAS® System's TABULATE Procedure

Contents

1. Introduction to Tabular Reports

The TABULATE procedure offers most of the statistics that the MEANS and SUMMARY procedures provide as well as flexible report writing features such as:

- flexible table construction
- multiple dimensions
- use of labels and formats

Generic PROC TABULATE report ⇨

	CODE		
	NEXT DAY	SECOND DAY	THIRD DAY
	SUM	SUM	SUM
	INCOME	INCOME	INCOME
HUB			
FRANKFRT	$5,293.94	0	0
LONDON	$2,742.81	$8,939.39	$2,534.18
NEW YORK	0	$3,857.14	0
SAN FRAN	0	0	$3,742.43
SYDNEY	$18,321.21	$7,511.02	$52,134.25
TOKYO	$2,330.01	0	$7,543.00
Total	$28,687.97	$20,307.54	$65,953.87

PROC TABULATE

The general form of the TABULATE step:

```
PROC TABULATE data= SAS data set  options ;
        CLASS variables;
        VAR variables;
        TABLE expression;
RUN;
```

- **class** variables are declared on the **CLASS** statement,
- **analysis** variables are declared on the **VAR** statement,
- only 'declared' variables are used on the TABLE statement.

Statistic keywords that are available to PROC TABULATE are :

Descriptive: COLPCTN COLPCTSUM NMISS MIN MAX VAR CV MODE KURTOSIS ROWPCTN ROWPCTSUM SUMWGT CSS USS RANGE STD SKEWNESS REPPCTN REPPCTSUM STDERR SUM MEAN STDERR N LCLM UCLM PAGEPCTN PAGEPCTSUM PCTN PCTSUM STD STDDEV

Hypothesis Testing: T PRT | PROBT

Quantile Keywords: P1 P5 P10 P25 P50 P75 P90 P95 P99 Q1 Q3 QRANGE MEDIAN

table ⇨

PROC TABULATE

TABLE Statement

- this is the real workhorse of PROC TABULATE,
- the table format with stats & variables are specified here,
- focus will first be on the 'shape' of the table,
- shape is 'dictated' by the TABLE Statement operators.

Selected operators:

Operator		Task
Comma	,	determines the number of dimensions
Asterisk	*	cross, subgroup or 'within'
Blank		table concatenator
Parentheses	( )	grouping agent
Brackets	< >	specifies denominator definitions
Equal	=	assigns labels or formats

example □

PROC TABULATE

Given the following **Class** variables and values:

Variable:	Values:
REGION	EAST, WEST
QTR	1, 2
YEAR	THIS, LAST

```
proc tabulate;
  class region qtr year ;
  table year , qtr * region;
run;
```

	QTR			
	1		2	
	REGION		REGION	
	EAST	WEST	EAST	WEST
YEAR				
LAST				
THIS				

data ➪

PROC TABULATE

Data for upcoming examples :

Obs	Hub	Country	TYPE	City	INCOME	OVERHEAD	Year
1	London	Australia	MD11	Acton	288.24	230.59	Last
2	London	Australia	MD11	Acton	523.24	418.59	This
3	London	Australia	MD11	Acton	1,500.24	1,200.19	This
4	London	Australia	MD11	Acton	1,660.57	1,328.45	This
5	London	Australia	DC10	Acton	499.24	394.40	Last
6	London	Australia	DC10	Acton	523.24	413.36	This
7	London	Australia	DC10	Acton	804.24	635.35	This
8	London	Australia	DC10	Acton	874.62	690.95	This
9	San Fran	Australia	A300	Acton	198.24	152.64	Last
10	San Fran	Australia	A300	Acton	523.24	402.89	This
11	San Fran	Australia	A300	Acton	1,308.24	1,007.34	This
12	San Fran	Australia	A300	Acton	1,340.82	1,032.43	This
13	New York	Australia	MD11	Melbourne	529.00	322.69	Last
14	New York	Australia	MD11	Melbourne	1,170.00	713.70	This
15	New York	Australia	MD11	Melbourne	1,596.00	973.56	This
16	New York	Australia	MD11	Melbourne	1,876.61	1,144.73	This
17	Sydney	Australia	DC10	Melbourne	628.71	421.24	Last
18	Sydney	Australia	DC10	Melbourne	251.29	168.36	This

Only the first 18 obs are shown.

example □

PROC TABULATE

Trick **1:** Use multiple table statements to generate multiple reports.

```
proc tabulate data = sashelp.pm;
   class hub;
   var   income;
   table hub * n;
   table hub * income * sum;
run;
```

Hub					
Frankfrt	London	New York	San Fran	Sydney	Tokyo
N	N	N	N	N	N
92.00	76.00	96.00	76.00	112.00	60.00

Hub					
Frankfrt	London	New York	San Fran	Sydney	Tokyo
INCOME	INCOME	INCOME	INCOME	INCOME	INCOME
Sum	Sum	Sum	Sum	Sum	Sum
99128.02	161560.84	73164.91	102212.65	238586.14	133330.27

PROC TABULATE

Trick **2:** Use multiple CLASS variables to generate a two dimensional report.

```
proc tabulate data = sashelp.pm format=comma12.2;
   class year type;
   var income;
   where type in('777','747');
   table year,  type * income * sum  ;
run;
```

Notice the comma in the TABLE statement.

Notice the **TYPE** and **Sum** 'labels'. We may decide to suppress them in future reports.

	TYPE	
	747	777
	INCOME	INCOME
	Sum	Sum
Year		
Last	23,999.19	45,720.71
This	131,694.67	177,846.53

PROC TABULATE

Trick **3:** Use multiple statistics and an **OUT =** option**.**

```
proc tabulate data = sashelp.pm  format=comma10.2  out = pm;
   class hub ;
   var income;
   table hub,  income * (n sum mean max mode) ;
run;
```

The **format =** option on the PROC statement has a global effect for all the 'cells' in the report (there is an option that has a 'local' effect that can override the global effect)**.**

Notice the statistics are 'connected' with a blank operator (concatenation)**.**

These statistics are calculated on INCOME (notice the *****) and will make up the columns of the report **.**

This step creates a report as well as an output dataset**.**

More ➪

PROC TABULATE

The report and the WORK.PM data set**.**

	INCOME				
	N	Sum	Mean	Max	Mode
Hub					
Frankfrt	92.00	99,128.02	1,077.48	4,703.86	435.15
London	76.00	161,560.84	2,125.80	19,833.95	523.24
New York	96.00	73,164.91	762.13	3,615.85	556.00
San Fran	76.00	102,212.65	1,344.90	7,100.00	.
Sydney	112.00	238,586.14	2,130.23	14,595.25	428.40
Tokyo	60.00	133,330.27	2,222.17	17,300.80	.

VIEWTABLE: Work.Pm

	Hub	_TYPE_	_PAGE_	_TABLE_	INCOME_N	INCOME_Sum	INCOME_Mean	INCOME_Max	INCOME_Mode
1	Frankfrt	1	1	1	92	99128.018	1077.4785	4703.85668	435.15
2	London	1	1	1	76	161560.84	2125.8005	19833.9489	523.235294
3	New York	1	1	1	96	73164.905	762.13443	3615.84581	556
4	San Fran	1	1	1	76	102212.65	1344.9032	7100.00284	.
5	Sydney	1	1	1	112	238586.14	2130.2334	14595.2537	428.4
6	Tokyo	1	1	1	60	133330.27	2222.1711	17300.7989	.

PROC TABULATE

Trick **4:** Use the **'='** operator to suppress labels.

```
proc tabulate data = sashelp.pm format = comma12.2;
   class hub type;
   var income;
   where type in('777','747');
   table hub, type = ' ' * income * sum = ' ';
run;
```

Notice the absence of the **TYPE** and **Sum** 'labels' **(**see Trick **2).**

Also, notice how wide the Row Title Space is.

	747	777
	INCOME	INCOME
Hub		
Frankfrt	40,993.42	6,290.07
London	14,024.67	26,122.65
New York	14,547.88	15,100.79
San Fran	24,174.57	32,769.05
Sydney	56,210.43	63,354.74
Tokyo	5,742.90	79,929.94

PROC TABULATE

Trick **5:** Enhance the program by including the **f =** , and the **rts** option, and the special **ALL** variable **.**

```
proc tabulate data=sashelp.pm format = comma12.2;
   class hub type;
   var income;
   where type in('747', '777');
   table hub all , type* income * sum =' '
    all = 'Total' * income*sum = ' ' * f = dollar12.2   / rts = 12 ;
run;
```

The 'pink' **all** is in the **row** dimension, while the 'red' **all** is in the **column** dimension**.**

The **f =** option will override the **format =** option.

The RTS option controls the row title space.

output □

PROC TABULATE

	TYPE		
	747	777	Total
	INCOME	INCOME	INCOME
HUB			
FRANKFRT	40,993.42	6,290.07	$47,283.48
LONDON	14,024.67	26,122.65	$40,147.32
NEW YORK	14,547.88	15,100.79	$29,648.68
SAN FRAN	24,174.57	32,769.05	$56,943.61
SYDNEY	56,210.43	63,354.74	$119,565.17
TOKYO	5,742.90	79,929.94	$85,672.84
All	155,693.86	223,567.24	$379,261.10

Let's introduce a little color into this report.

2. Defining STYLES for PROC TABULATE

The **STYLE =** (COMPONENT) = {attribute = value } syntax can also be used to control the appearance of PROC TABULATE output.

STYLE= options are used within **ODS** statements, otherwise they are ignored.

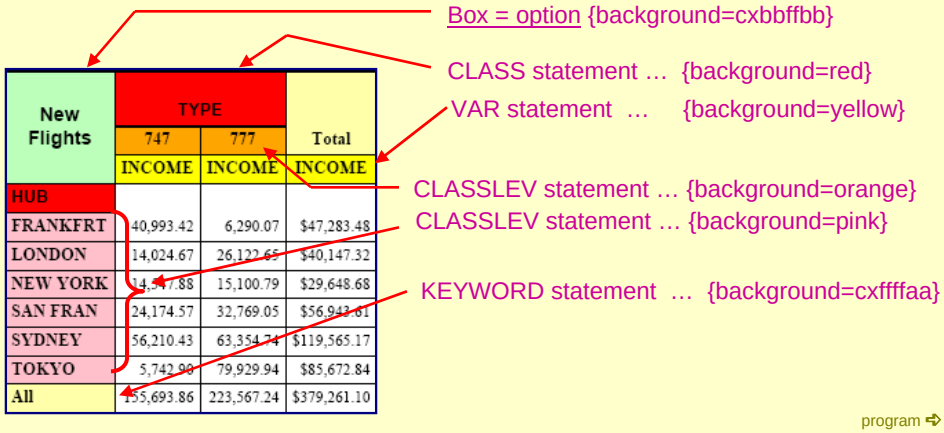
The following table indicates where you can use the **STYLE =** option.

Use the STYLE= option on this statement…	**To affect this part of the report….**
PROC TABULATE	Data cells
CLASS	Class variable name headings
CLASSLEV	Class level value headings
VAR	Analysis Variable name headings
KEYWORD	Keyword headings
TABLE	Table borders, rules and other parts that are not specified elsewhere
TABLE statement, **BOX=** option	Text in upper left box or the report
TABLE statement, **MISSTEXT=** option	Text for missing values in data cells.

Using STYLES to Enhance the Output

The STYLE = (COMPONENT) = {attribute = value } syntax can also be used to control the appearance of the report.

The following 'COMPONENTS' can be controlled by the STYLE = option:



New Flights	TYPE		Total
	747	777	
	INCOME	INCOME	INCOME
HUB			
FRANKFRT	40,993.42	6,290.07	$47,283.48
LONDON	14,024.67	26,122.65	$40,147.32
NEW YORK	14,547.88	15,100.79	$29,648.68
SAN FRAN	24,174.57	32,769.05	$56,943.61
SYDNEY	56,210.43	63,354.74	$119,565.17
TOKYO	5,742.90	79,929.94	$85,672.84
All	155,693.86	223,567.24	$379,261.10

program ➪

Using STYLES to Enhance the Output

Trick **6:** Use the STYLE = options in the PROC TABULATE step.

```
ods pdf file='mypdf.pdf';

proc tabulate data=sas_1.pm format = comma12.2 ;
     class hub type / style={font_face=arial background=red};
     classlev hub   / style={background=pink};
     classlev type  / style={background=orange};
     var income     / style={background=yellow};
     keyword all    / style={background=cxffffaa};
     where type in('747','777');
     table hub all, type * income *sum=' '
           all='Total' * income * sum=' '*f=dollar12.2
           / rts=12 box={label='New Flights'
             style={background=cxbbffbb  font_face=arial font_size=4}};
run;

ods pdf close;
```

output ➪

Using STYLES to Enhance the Output

New Flights	TYPE		Total
	747	777	
	INCOME	INCOME	INCOME
HUB			
FRANKFRT	40,993.42	6,290.07	$47,283.48
LONDON	14,024.67	26,122.65	$40,147.32
NEW YORK	14,547.88	15,100.79	$29,648.68
SAN FRAN	24,174.57	32,769.05	$56,943.61
SYDNEY	56,210.43	63,354.74	$119,565.17
TOKYO	5,742.90	79,929.94	$85,672.84
All	155,693.86	223,567.24	$379,261.10

Using STYLES to Enhance the Output

Putting Hyperlinks in the Report

Trick **7:** Modify the program by using different style attributes. Define a URL to be used as a hyperlink in the **BOX =** option. Make it link to a spreadsheet.

```
ods rtf file= 'c:\flight.rtf';

proc tabulate data=sas_1.pm f=15.2 s={font_face=arial } ;
     var income;
     class hub type;
     where type in('747', '777');
     keyword all;
     table hub all, type * income * sum=' '
           all='Total' * income * sum = ' ' * f=dollar12.2 /
           rts=12 box={label='New Flights' style=
           { url='c:\newflights.xls' background=light yellow} } ;
run;
ods rtf close;
```

output ➪

Noticed the label in the BOX = option.

Using STYLES to Enhance the Output

When you click on 'New Flights'…

… the 'newflights.xls opens

New Flights	TYPE		Total
	747	777	
	INCOME	INCOME	INCOME
HUB			
FRANKFRT	40993.42	6290.07	$47,283.48
LONDON	14024.67	26122.65	$40,147.32
NEW YORK	14547.88	15100.79	$29,648.68
SAN FRAN	24174.57	32769.05	$56,943.61
SYDNEY	56210.43	63354.74	$119,565.17
TOKYO	5742.90	79929.94	$85,672.84
All	155693.86	223567.24	$379,261.10

Results Viewer - newflights

H11 =

	A	B	C	D
1	Hub	Type	Location	Time
2	Frankfort	Departing	Paris	8:30
3	Frankfort	Departing	Krakow	9:30
4	Frankfort	Arriving	Paris	14:30
5	Frankfort	Arriving	Krakow	16:45
6	Tokyo	Departing	Sydney	6:30
7	Tokyo	Departing	Osaka	7:30
8	Tokyo	Arriving	Sydney	11:30
9	Tokyo	Arriving	Osaka	13:15
10	London	Departing	Marrakesh	10:22
11	London	Departing	Dublin	11:15
12	London	Arriving	Marrakesh	15:22
13	London	Arriving	Dublin	19:45
14				

new

3. Calculating Percentages

The TABULATE procedure can calculate the following statistics:

- **ROWPCTN & ROWPCTSUM - calculate the percentage of the value in a cell in relation to the total of the values in the ROW.**
- **COLPCTN & COLPCTSUM - calculate the percentage of the value in a cell in relation to the total of the values in the COLUMN.**
- **PAGEPCTN & PAGEPCTSUM - calculate the percentage of the value in a cell in relation to the total of the values in the PAGE.**
- **REPPCTN & REPPCTSUM - calculate the percentage of the value in a cell in relation to the total of the values in the REPORT.**
- **PCTN & PCTSUM - these statistics can calculate these same percentages. They enable you to manually define denominators. These statistics print the percentage of the value in a single table cell in relation to the value (used in the denominator of the calculation of the %) in another table cell or to the total of the values in a group of cells.**

The N statistics are calculated as a 'frequency count' and does NOT use an analysis variable. The SUM statistics are calculated with an analysis variable.

Calculating Percentages with PROC TABULATE

Trick **8 –** Ilustrate the N, SUM, PCTN, and PCTSUM statistics.
(Program is run in EG).

```
proc tabulate data = sashelp.class format=comma12.2;
     class sex age;
     var   height ;
     table age all='Total' * ( [style = [ background=yellow] ] ),
           height * (n sum pctn pctsum);
run;
```

	Height			
	N	Sum	PctN	PctSum
Age				
11	2	108.80	10.53	9.19
12	5	297.20	26.32	25.09
13	3	184.30	15.79	15.56
14	4	259.60	21.05	21.92
15	4	262.50	21.05	22.16
16	1	72.00	5.26	6.08
Total	19	1,184.40	100.00	100.00

The TOTAL number of observations (**N**) is 19.
PCTN = N / 19.

The **SUM** of all the student's HEIGHT is 1,184.40.
PCTSUM = SUM / 1184

Calculating Percentages with PROC TABULATE

Trick **9 –** Ilustrate the N, SUM, ROWPCTN, and ROWPCTSUM statistics.
Add SEX to the TABLE statement to make a 2D report.

```
proc tabulate data = sashelp.class format=comma12.2;
     class sex age;
     var   height ;
     table age, sex * height * (n sum rowpctn rowpctsum);
run;
```

	Sex							
	F				M			
	Height				Height			
	N	Sum	RowPctN	RowPctSum	N	Sum	RowPctN	RowPctSum
Age								
11	1	51.30	50.00	47.15	1	57.50	50.00	52.85
12	2	116.10	40.00	39.06	3	181.10	60.00	60.94
13	2	121.80	66.67	66.09	1	62.50	33.33	33.91
14	2	127.10	50.00	48.96	2	132.50	50.00	51.04
15	2	129.00	50.00	49.14	2	133.50	50.00	50.86
16	.	.	.	.	1	72.00	100.00	100.00

When the **RowPctN** columns are added, they total 100 for each row. The same is true with **RowPctSum.**

Note: Program was run in EG.

Calculating Percentages with PROC TABULATE

Trick **10:** Use the denominator definitions as an **alternative way** to create percentages that add to 100 across the row.

```
proc format;
      picture pfmt low-hight = '   009.99%';
proc tabulate data = sashelp.class format=comma12.2;
     class sex age;
     var   height ;
     table age, sex * height * (n sum pctn<sex> pctsum<sex>*f=pfmt.);
run;
```

	Sex							
	F				M			
	Height				Height			
	N	Sum	PctN	PctSum	N	Sum	PctN	PctSum
Age								
11	1	51.30	50.00	47.15%	1	57.50	50.00	52.84%
12	2	116.10	40.00	39.06%	3	181.10	60.00	60.93%
13	2	121.80	66.67	66.08%	1	62.50	33.33	33.91%
14	2	127.10	50.00	48.95%	2	132.50	50.00	51.04%
15	2	129.00	50.00	49.14%	2	133.50	50.00	50.85%
16	.	.	.	.	1	72.00	100.00	100.00%

Note: PCTSUM is **formatted** while PCTN is not.

The output is basically the same as the previous report. Before ROWPCT was developed, this was the statistical method.

Calculating Percentages with PROC TABULATE

Trick 11 – Modify the previous program by adding **STYLE =** options and the Keyword ALL to the Column dimension. (Style= can be abbreviated as S=).

```
proc tabulate data = sashelp.class format=comma12.2;
     class sex age;
     var   height ;
     table age, sex * height * (n sum rowpctn*[s=[background=pink] ]
                                rowpctsum*[style=[background=yellow] ] )
                all * height * (n sum rowpctn='Percent'*[s=[background=orange] ] );
     keyword rowpctn   / style=[background=red];
run;
```

	Sex								All	
	F				M					
	Height				Height				Height	
	N	Sum	RowPctN	RowPctSum	N	Sum	RowPctN	RowPctSum	N	Sum	Percent
Age											
11	1	51.30	50.00	47.15	1	57.50	50.00	52.85	2	108.80	100.00
12	2	116.10	40.00	39.06	3	181.10	60.00	60.94	5	297.20	100.00
13	2	121.80	66.67	66.09	1	62.50	33.33	33.91	3	184.30	100.00
14	2	127.10	50.00	48.96	2	132.50	50.00	51.04	4	259.60	100.00
15	2	129.00	50.00	49.14	2	133.50	50.00	50.86	4	262.50	100.00
16	.	.	.	.	1	72.00	100.00	100.00	1	72.00	100.00

Calculating Percentages with PROC TABULATE

Trick **12** – Illustrate the N, SUM, COLPCTN, and COLPCTSUM statistics.

```
proc tabulate data = sashelp.class format=comma12.2;
     class sex age;
     var   height ;
     table age all * ( [style = [background=yellow]] ),
           sex=' ' * height * (n sum colpctn colpctsum);
     keyword all  / style=[background=cyan];
run;
```

	F				M			
	Height				Height			
	N	Sum	ColPctN	ColPctSum	N	Sum	ColPctN	ColPctSum
Age								
11	1	51.30	11.11	9.41	1	57.50	10.00	9.00
12	2	116.10	22.22	21.29	3	181.10	30.00	28.34
13	2	121.80	22.22	22.34	1	62.50	10.00	9.78
14	2	127.10	22.22	23.31	2	132.50	20.00	20.73
15	2	129.00	22.22	23.66	2	133.50	20.00	20.89
16	.	.	.	.	1	72.00	10.00	11.27
All	9	545.30	100.00	100.00	10	639.10	100.00	100.00

Calculating Percentages with PROC TABULATE

Trick **13** – Illustrate the N, SUM, REPPCTN, and REPPCTSUM statistics.
(Program was run in EG).

```
proc tabulate data = sashelp.class format=comma12.2;
     class sex age;
     var   height ;
     table age all, sex * height * (n sum reppctn reppctsum);
run;
```

	Sex							
	F				M			
	Height				Height			
	N	Sum	RepPctN	RepPctSum	N	Sum	RepPctN	RepPctSum
Age								
11	1	51.30	5.26	4.33	1	57.50	5.26	4.85
12	2	116.10	10.53	9.80	3	181.10	15.79	15.29
13	2	121.80	10.53	10.28	1	62.50	5.26	5.28
14	2	127.10	10.53	10.73	2	132.50	10.53	11.19
15	2	129.00	10.53	10.89	2	133.50	10.53	11.27
16	.	.			1	72.00	5.26	6.08
All	9	545.30	47.37	46.04	10	639.10	52.63	53.96

Calculating Percentages with PROC TABULATE

Trick **14** – Illustrate traffic lighting using PROC FORMAT. (Program run in EG).

```
proc format;
      value colorfmt low-60   ='cxff88aa'
                     120-130  ='yellow'
                     151-high ='light green';
 run;
proc tabulate data = sashelp.class format=comma12.2;
     class sex age;
     var   height ;
     table age all,
           sex=' ' * height * sum * [style=[background=colorfmt.] ];
 run;
```

	F	M
	Height	Height
	Sum	Sum
Age		
11	51.30	57.50
12	116.10	181.10
13	121.80	62.50
14	127.10	132.50
15	129.00	133.50
16	.	72.00
All	545.30	639.10

Notice the VALUE statement in the FORMAT procedure.

Also notice the STYLE= option in the TABULATE procedure.

Calculating Percentages with PROC TABULATE

Trick **15** – Insert a bitmap into the title area. Send the report to a pdf file.

```
ods pdf file='c:\SGF.pdf';
ods escapechar = '^';
   title1 f=arial c=blue height=8 'New Report';
   title3 j=c "^S={preimage='c:\ben\UserGroups\partner.bmp' } ";
   options nodate;

proc tabulate data = sashelp.class format=comma12.2;
   class sex age;
   var  height ;
   table age all, sex=' ' * height * ( n sum  reppctn reppctsum );
run;

ods pdf close;
```

Notice the TITLE statements.

Calculating Percentages with PROC TABULATE

Trick **15** output.

New Report

	F				M			
	Height				Height			
	N	Sum	RepPctN	RepPctSum	N	Sum	RepPctN	RepPctSum
Age								
11	1	51.30	5.26	4.33	1	57.50	5.26	4.85
12	2	116.10	10.53	9.80	3	181.10	15.79	15.29
13	2	121.80	10.53	10.28	1	62.50	5.26	5.28
14	2	127.10	10.53	10.73	2	132.50	10.53	11.19
15	2	129.00	10.53	10.89	2	133.50	10.53	11.27
16	.	.	.	.	1	72.00	5.26	6.08
All	9	545.30	47.37	46.04	10	639.10	52.63	53.96

If the logo is too big…

Logo / Image in Titles

Task: Put the logo and the first title on the same line and control the logo size.

```
ods escapechar = '^';
ods pdf  file = 'c:\Lisbon.pdf ' ;
title1  justify = center f = arial  c = blue  height = 18pt  bold  'Revenue after 2004'
        justify = right '^S = {preimage = "d:\ben\partner.bmp?height=0.75in&width=2in "}';
title2;
 options nodate;

proc print data = sas_1.mdv;
     where  city =: 'SAN FRAN';
     var month / style = {foreground=red};
     sum sales2005  sales2006;
 run;

ods pdf close;
```

The **HEIGHT=** and **WIDTH =** options are new undocumented features in **SAS9.3.**

Revenue after 2004

Obs	MONTH	Sales2005	Sales2006
56	2	894.00	907.74
57	4	2,081.00	2,345.65
58	7	4,919.00	5,442.21
59	12	2,346.00	2,670.02
		10,240.00	11,365.62

About the Speaker

Speaker	Ben Cochran
Office Location	The Bedford Group 3224 Bedford Ave. Raleigh, NC 27607
Telephone	(919) 741.0370
E-Mail	bencochran@nc.rr.com