The George Washington University

Department of Statistics

STAT 6197 - Spring 2019

Week 7 – March 1, 2019

Major Topic: Summarizing Data Using PROC Steps, and Creating Reports Using Output Delivery System (ODS) Features

Detailed Topics:

1) Summarizing Data Using PROC Steps
    - PROC MEANS
    - PROC SUMMARY
    - PROC FREQ
    - PROC SURVEYFREQ
    - PROC SURVEYMEANS
    - PROC TABULATE
    - PROC UNIVARIATE

2) Using the ODS Features
    - Select Results Objects from PROC Steps
    - Create New Data Tables
    - Alter Output Appearance
    - Change the Output Format

Readings:

1. Relevant Chapters/Sections from Delwiche L, and Slaughter S. *The Little SAS Book: A Primer*, Fifth Edition Paperback – November 7, 2012

2. Exercises from Relevant Chapters/Sections - Ottesen RA, Delwiche LD, and Slaughter SJ. *Exercises and Projects for The Little SAS Book*, Fifth Edition Paperback – July 1, 2015

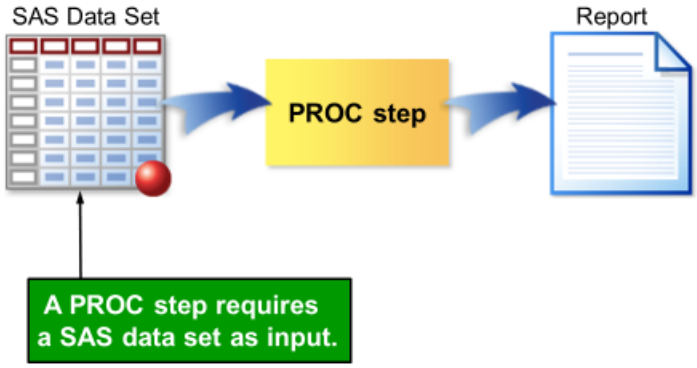
3. Carpenter, A.L. The MEANS/SUMMARY Procedure: Getting Started. SAS Global Forum 2010.

4. SAS/STAT 14.3 User's Guide. The SURVEYMEANS Procedure
5. SAS/STAT 14.3 User's Guide. The SURVEYFREQ Procedure
6. SAS(R) 9.4 Output Delivery System: User's Guide, Fifth Edition

7. Using Procedure Based ODS Data Components in Statistical Reporting by VJ Faber

Planning to Create a Summary Report

To create a summary report in SAS, a PROC step is used.



9

Planning to Create a Summary Report

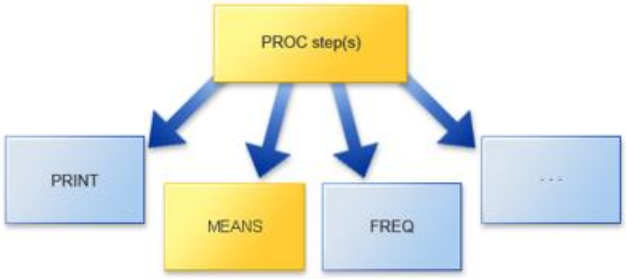
These steps are required for creating a summary report.

Step 1	Name the SAS data set to be referenced.
Step 2	Determine the PROC step to use.
Step 3	Determine the variables to analyze.
Step 4	Determine the grouping variables.

10

Planning to Create a Summary Report

Step 2 Determine the PROC step to use.



- Many other summary procedures are available.

12

MEANS Procedure

The MEANS procedure produces summary reports with descriptive statistics.

```
proc means data=orion.sales;
run;
```

```
PROC MEANS DATA=input-data-set <options statistics>;
   <VAR analysis-variable(s);>
   <CLASS classification-variable(s);>
RUN;
```

- *Analysis variables* are the ***numeric*** variables for which statistics are to be computed.
- *Classification variables* are variables whose values define subgroups for the analysis. They can be character or numeric.

p111d07

33

VAR Statement

The VAR statement identifies the analysis variable (or variables) and their order in the output.

```
proc means data=orion.sales;
   var Salary;
run;
```

VAR *variable(s)*;

The MEANS Procedure

Analysis Variable : Salary

N	Mean	Std Dev	Minimum	Maximum
165	31160.12	20082.67	22710.00	243190.00

35

p111d07

CLASS Statement

The *CLASS statement* specifies one or more variables that the procedure uses to group the data.

Country	Year	N Obs	Variable	Label	N	Mean
CANADA	1993	240	PREDICT	Predicted Sales	240	497.20
			ACTUAL	Actual Sales	240	504.25
	1994	240	PREDICT	Predicted Sales	240	473.71
			ACTUAL	Actual Sales	240	524.88
GERMANY	1993	240	PREDICT	Predicted Sales	240	488.00
			ACTUAL	Actual Sales	240	530.85
	1994	240	PREDICT	Predicted Sales	240	476.81
			ACTUAL	Actual Sales	240	494.14
U.S.A.	1993	2				
	1994	2				

```
proc means data=sashelp.prdsale
           maxdec=2 n mean;
  var predict actual;
  class country year;
run;
```

104

CLASS Statement

The *CLASS statement* identifies variables whose values define subgroups for the analysis.

```
proc means data=orion.sales;
   var Salary;
   class Gender Country;
run;
```

CLASS *classification-variable(s)*;

- Classification variables are character or numeric.
- They typically have few discrete values.
- The data set does ***not*** need to be sorted or indexed by the classification variables.

36

p111d08

Viewing the Output

Statistics are produced for each combination of values of the classification variables.

The MEANS Procedure
Analysis Variable : Salary

Gender	Country	N Obs	N	Mean	Std Dev	Minimum	Maximum
F	AU	27	27	27702.41	1728.23	25185.00	30890.00
	US	41	41	29460.98	8847.03	25390.00	83505.00
M	AU	36	36	32001.39	16592.45	25745.00	108255.00
	US	61	61	33336.15	29592.69	22710.00	243190.00

- *N Obs* – the number of observations with each unique combination of class variables
- *N* – the number of observations with nonmissing values of the analysis variable (or variables)

37

PROC MEANS Statistics

Use options in the PROC MEANS statement to request specific statistics.

```
proc means data=orion.sales nmiss min max sum;
   var Salary;
   class Gender Country;
run;
```

The requested statistics override the default statistics.

39

p111d09

PROC MEANS Statement Options

Options can also be placed in the PROC MEANS statement.

Option	Description
MAXDEC=	Specifies the number of decimal places to display.
NONOBS	Suppresses the N Obs column.

41

MAXDEC= Option

MAXDEC=0

The MEANS Procedure
Analysis Variable : Salary

Country	N Obs	N	Mean	Std Dev	Minimum	Maximum
AU	63	63	30159	12699	25185	108255
US	102	102	31778	23556	22710	243190

MAXDEC=1

The MEANS Procedure
Analysis Variable : Salary

Country	N Obs	N	Mean	Std Dev	Minimum	Maximum
AU	63	63	30159.0	12699.1	25185.0	108255.0
US	102	102	31778.5	23555.8	22710.0	243190.0

p111d10

42

NONOBS Option

N Obs included by default

The MEANS Procedure
Analysis Variable : Salary

Country	N Obs	N	Mean	Std Dev	Minimum	Maximum
AU	63	63	30158.97	12699.14	25185.00	108255.00
US	102	102	31778.48	23555.84	22710.00	243190.00

NONOBS option

The MEANS Procedure
Analysis Variable : Salary

Country	N	Mean	Std Dev	Minimum	Maximum
AU	63	30158.97	12699.14	25185.00	108255.00
US	102	31778.48	23555.84	22710.00	243190.00

43 p111d10

Other PROC MEANS Statistics

Descriptive Statistic Keywords				
CLM	CSS	CV	LCLM	MAX
MEAN	MIN	MODE	N	NMISS
KURTOSIS	RANGE	SKEWNESS	STDDEV	STDERR
SUM	SUMWGT	UCLM	USS	VAR

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

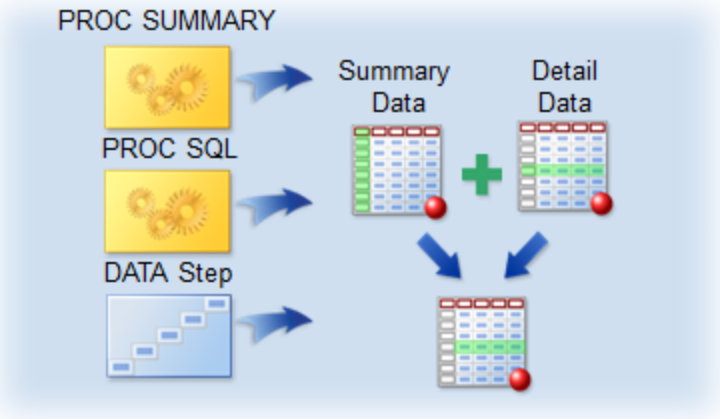
Hypothesis Testing Keywords				
PROBT	T			

44

Acknowledgements: Portions of SAS' copyrighted SAS course content are reproduced here with permission of SAS Institute Inc., Cary, NC, USA. SAS Institute Inc. makes no warranties with respect to these materials and disclaims all liability therefor. Parts of additional materials are adapted from SAS Documentation, and web-based resources. Please let me know of any errors or typos you have found. Do not copy or circulate the handouts.

Introduction

This section explores how to combine summary and detail data using PROC SUMMARY, PROC SQL, and the DATA step.



79

Business Scenario

Use **orion.totalsalaries** to calculate percentages of the total company payroll for each manager. This requires a few steps.

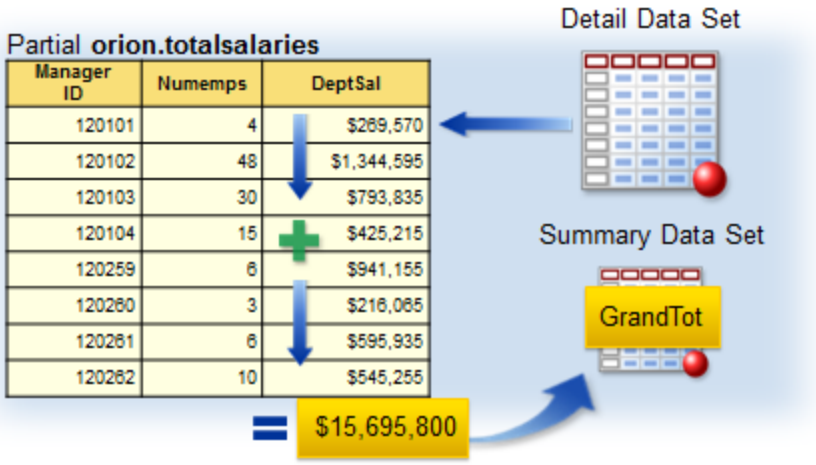
Partial **orion.totalsalaries**

Manager ID	Numemps	Dept Sal	GrandTot	Percent
120101	4	$269,570	$15,695,800	1.72%
120102	48	$1,344,595	$15,695,800	8.57%
120103	30	$793,835	$15,695,800	5.06%
120104	15	$425,215	$15,695,800	2.71%
120259	6	$941,155	$15,695,800	6.00%
120260	3	$216,065	$15,695,800	1.38%
120261	6	$595,935	$15,695,800	3.80%
120262	10	$545,255	$15,695,800	3.47%

81

Business Scenario

Calculate the grand total of **DeptSal** and store it in a summary data set.



82

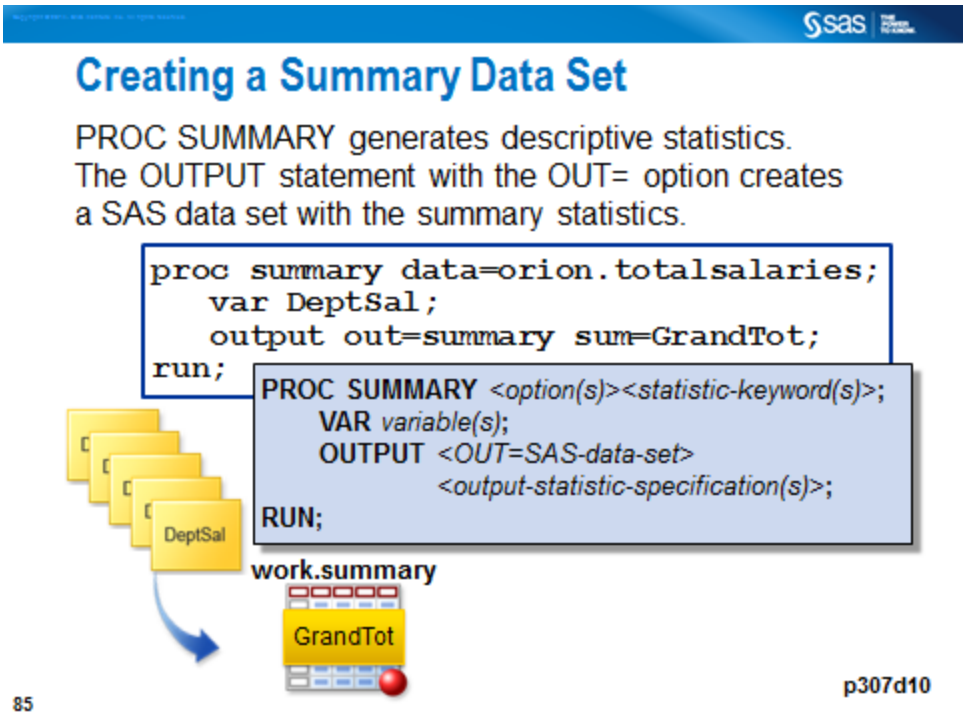
Business Scenario

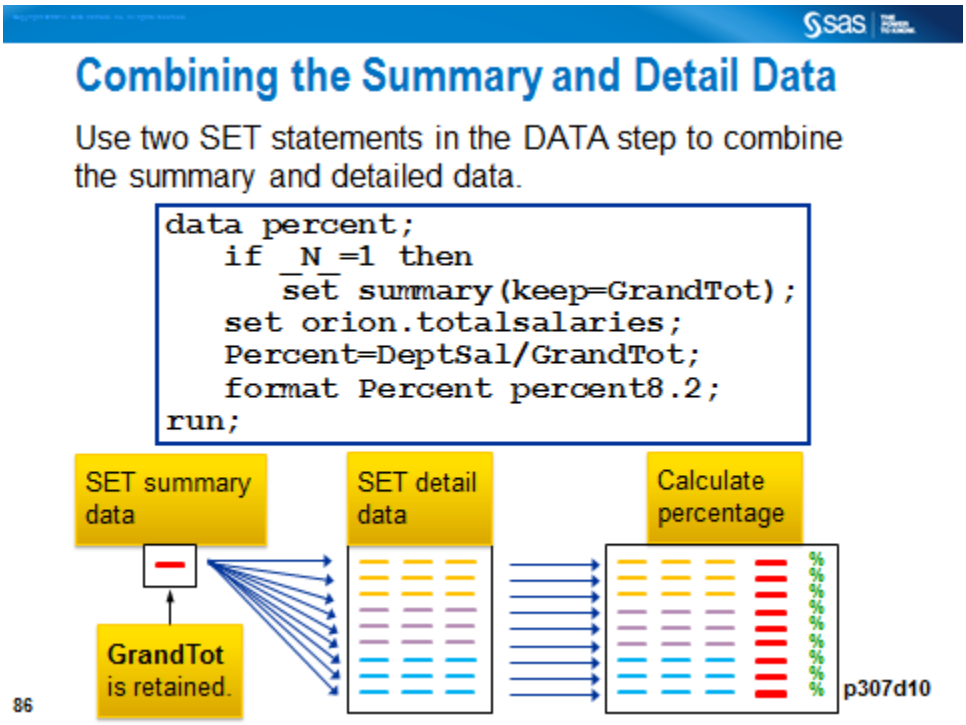
Combine the summary data with each row of the detailed data. Divide each detail amount by the grand total to calculate the percent of payroll by manager.

Partial **orion.totalsalaries**

Manager ID	Numemps	DeptSal	GrandTot	Percent
120101	4	$269,570	$15,695,800	1.72%
120102	48	$1,344,595	$15,695,800	8.57%
120103	30	$793,835	$15,695,800	5.06%
120104	15	$425,215	$15,695,800	2.71%
120259	6	$941,155	$15,695,800	6.00%
120260	3	$216,065	$15,695,800	1.38%
120261	6	$595,935	$15,695,800	3.80%
120262	10	$545,255	$15,695,800	3.47%

83





/Ex1_proc_summary_proc_means_sum.sas

```
1  *Ex1B_proc_summary_sum.sas;
2  options nonumber nodate  ps=58 ls=90;
3  data heart;
4   set sashelp.heart;
5  if status= 'Alive' then death=0;
6  else death=1;
7
8  if status= 'Alive' then survived =1;
9  else survived =0;
10 run;
11 proc summary data=heart;
12     class smoking_status;
13     var death survived;
14     output out=count_data
15         sum(death)=death_count
16         sum(survived)=survived_count;
17 run;
18 proc print data=count_data; run;
```

Line 12: The CLASS statement accepts one or more classification variables. It can be broken into a series of CLASS statements. If a classification variable has a missing value, SAS will eliminate the entire observation from the analysis. With the CLASS statement, you can use various options that include the ORDER= option, DESCENDING option, MISSING option. See SAS® documentation for details.

Line 13: The VAR statement accepts the analysis variable as numeric.

Line 14: The OUTPUT statements accepts the OUT= data set and allows to specify statistics.

Acknowledgements: Portions of SAS' copyrighted SAS course content are reproduced here with permission of SAS Institute Inc., Cary, NC, USA. SAS Institute Inc. makes no warranties with respect to these materials and disclaims all liability therefor. Parts of additional materials are adapted from SAS Documentation, and web-based resources. Please let me know of any errors or typos you have found. Do not copy or circulate the handouts.

Obs	Smoking_Status	_TYPE_	_FREQ_	death_ count	survived_ count
1		0	5173	1971	3202
2	Heavy (16-25)	1	1046	443	603
3	Light (1-5)	1	579	187	392
4	Moderate (6-15)	1	576	213	363
5	Non-smoker	1	2501	891	1610
6	Very Heavy (> 25)	1	471	237	234

TYPE is an automatic numeric variable, which can be used to help us track the level of summarization.

```
20 proc summary data=heart;
21     class sex smoking_status;
22     var weight;
23     output out=stats
24        mean=meanWEIGHT/ levels ways;
25 run;
26 proc print data=stats; run;
```

Line 24: The LEVELS options adds to OUT= data set an automatic numeric variable _LEVEL_. This variable contains a sequential counter of rows within a given value of _TYPE_. The WAYS option adds to OUT= data set an automatic numeric variable _WAY_. Here, this variable indicates a two-way interaction between the SEX and SMOKING_STATUS CLASSIFICATION variables.

Obs	Sex	Smoking_Status	_WAY_	_TYPE_	_LEVEL_	_FREQ_	mean WEIGHT
1			0	0	1	5173	153.088
2		Heavy (16-25)	1	1	1	1046	154.763
3		Light (1-5)	1	1	2	579	146.766
4		Moderate (6-15)	1	1	3	576	144.586
5		Non-smoker	1	1	4	2501	153.742
6		Very Heavy (> 25)	1	1	5	471	164.081
7	Female		1	2	1	2856	141.422
8	Male		1	2	2	2317	167.460
9	Female	Heavy (16-25)	2	3	1	339	136.086
10	Female	Light (1-5)	2	3	2	422	140.610
11	Female	Moderate (6-15)	2	3	3	340	131.649
12	Female	Non-smoker	2	3	4	1682	144.800
13	Female	Very Heavy (> 25)	2	3	5	73	138.431
14	Male	Heavy (16-25)	2	3	6	707	163.719
15	Male	Light (1-5)	2	3	7	157	163.274
16	Male	Moderate (6-15)	2	3	8	236	163.169
17	Male	Non-smoker	2	3	9	819	172.119
18	Male	Very Heavy (> 25)	2	3	10	398	168.733

PROC FREQ

The FREQ procedure can be used for simple frequency distributions, 2-way cross-tabulations, and 3-way tabulations on categorical or ordinal variables.

Ex2_PROC_FREQ_SAS.sas

PROC FREQ – No Option on the TABLES Statement

```
1  *Ex2_PROC_FREQ_SAS;
2  OPTIONS nonumber nodate ps=58 ls=90;
3  title1 'One-Way Table';
4  title2 'No option on the TABLES statement';
5  proc freq data=sashelp.heart;
6  tables sex weight_status bp_status;
7  run;
```

The FREQ Procedure

Sex	Frequency	Percent	Cumulative Frequency	Cumulative Percent
Female	2873	55.15	2873	55.15
Male	2336	44.85	5209	100.00

Weight Status

Weight_ Status	Frequency	Percent	Cumulative Frequency	Cumulative Percent
Normal	1472	28.29	1472	28.29
Overweight	3550	68.23	5022	96.52
Underweight	181	3.48	5203	100.00

Frequency Missing = 6

Blood Pressure Status

BP_Status	Frequency	Percent	Cumulative Frequency	Cumulative Percent
High	2267	43.52	2267	43.52
Normal	2143	41.14	4410	84.66
Optimal	799	15.34	5209	100.00

PROC FREQ – Missing Option on the TABLES Statement

```
 8 title1 'One-Way Table - MISSING Option';
 9 proc freq data=sashelp.heart;
10  tables weight_status / missing;
11 run;
```

The MISSING option in the TABLES statement is used to tell SAS to include missing values in percentage calculations.

One-Way Table - MISSING Option

The FREQ Procedure

Weight Status

Weight_ Status	Frequency	Percent	Cumulative Frequency	Cumulative Percent
	6	0.12	6	0.12
Normal	1472	28.26	1478	28.37
Overweight	3550	68.15	5028	96.53
Underweight	181	3.47	5209	100.00

Acknowledgements: Portions of SAS' copyrighted SAS course content are reproduced here with permission of SAS Institute Inc., Cary, NC, USA. SAS Institute Inc. makes no warranties with respect to these materials and disclaims all liability therefor. Parts of additional materials are adapted from SAS Documentation, and web-based resources. Please let me know of any errors or typos you have found. Do not copy or circulate the handouts.

PROC FREQ – MISSPRINT Option on the TABLES Statement

Line 9: The MISSPRINT option in the TABLES statement tells SAS to treat missing values under a nonmissing category when printing the frequencies but exclude them when calculating statistics.

Options to Suppress Statistics

Use options in the TABLES statement to suppress the display of selected default statistics.

TABLES *variable(s) / options* ;

Option	Description
NOROW	Suppresses the display of the row percentage.
NOCOL	Suppresses the display of the column percentage.
NOPERCENT	Suppresses the percentage display.
NOFREQ	Suppresses the frequency display.

9

Acknowledgements: Portions of SAS' copyrighted SAS course content are reproduced here with permission of SAS Institute Inc., Cary, NC, USA. SAS Institute Inc. makes no warranties with respect to these materials and disclaims all liability therefor. Parts of additional materials are adapted from SAS Documentation, and web-based resources. Please let me know of any errors or typos you have found. Do not copy or circulate the handouts.

CROSSLIST Option

The *CROSSLIST option* displays two-way tables in column format, instead of cell format.

```
proc freq data=sashelp.orsales ;
   where product_line in ('Outdoors','Sports');
   tables product_line*year / crosslist ;
run;
```

Options in the TABLES statement must come after a forward slash.

88

The FREQ Procedure

Table of Product_Line by Year

Product_ Line	Year	Frequency	Percent	Row Percent	Column Percent
Outdoors	1999	28	6.25	25.00	25.00
	2000	28	6.25	25.00	25.00
	2001	28	6.25	25.00	25.00
	2002	28	6.25	25.00	25.00
	Total	112	25.00	100.00	
Sports	1999	84	18.75	25.00	75.00
	2000	84	18.75	25.00	75.00
	2001	84	18.75	25.00	75.00
	2002	84	18.75	25.00	75.00
	Total	336	75.00	100.00	
Total	1999	112	25.00		100.00
	2000	112	25.00		100.00
	2001	112	25.00		100.00
	2002	112	25.00		100.00
	Total	448	100.00		

89

Acknowledgements: Portions of SAS' copyrighted SAS course content are reproduced here with permission of SAS Institute Inc., Cary, NC, USA. SAS Institute Inc. makes no warranties with respect to these materials and disclaims all liability therefor. Parts of additional materials are adapted from SAS Documentation, and web-based resources. Please let me know of any errors or typos you have found. Do not copy or circulate the handouts.

NLEVELS Option

The *NLEVELS option* displays a table that provides the number of distinct values for each analysis variable.

```
proc freq data=orion.nonsales2 nlevels;
   tables Gender Country / nocum nopercent;
run;
```

```
PROC FREQ DATA=SAS-data-set NLEVELS;
    TABLES variable(s) ;
RUN;
```

16 p111d03

Viewing the Output

PROC FREQ Output

The FREQ Procedure

Number of Variable Levels

Variable	Levels	Missing Levels	Nonmissing Levels
Gender	4	1	3
Country	4	0	4

Gender	Frequency
F	110
G	1
M	123

Frequency Missing = 1

Country	Frequency
AU	33
US	196
au	3
us	3

17

Acknowledgements: Portions of SAS' copyrighted SAS course content are reproduced here with permission of SAS Institute Inc., Cary, NC, USA. SAS Institute Inc. makes no warranties with respect to these materials and disclaims all liability therefor. Parts of additional materials are adapted from SAS Documentation, and web-based resources. Please let me know of any errors or typos you have found. Do not copy or circulate the handouts.

NLEVELS Option

The _ALL_ keyword with the NOPRINT option displays the number of levels for all variables without displaying frequency counts.

```
proc freq data=orion.nonsales2 nlevels;
   tables _all_ / noprint;
run;
```

22 p111d04

Viewing the Output

PROC FREQ Output

The FREQ Procedure
Number of Variable Levels

Variable	Levels	Missing Levels	Nonmissing Levels
Employee_ID	234	1	233
First	204	0	204
Last	228	0	228
Gender	4	1	3
Salary	230	1	229
Job_Title	125	1	124
Country	4	0	4

No frequency tables were displayed.

23

Identifying Observations with Invalid Data

PROC FREQ has uncovered the existence of invalid data values for **Gender**, **Country**, and **Employee_ID**. Use PROC PRINT to display the observations with invalid values.

```
proc print data=orion.nonsales2;
   where Gender not in ('F','M') or
         Country not in ('AU','US') or
         Job_Title is null or
         Employee_ID is missing or
         Employee_ID=120108;
run;
```

24

Viewing the Output

PROC PRINT Output

Obs	Employee_ID	First	Last	Gender	Salary	Job_Title	Country
2	120104	Kareen	Billington	F	46230	Administration Manager	au
6	120108	Gladys	Gromek	F	27660	Warehouse Assistant II	AU
7	120108	Gabriele	Baker	F	26495	Warehouse Assistant I	AU
10	120112	Ellis	Glattback	F	26550		AU
12	120114	Jeannette	Buddery	G	31285	Security Manager	AU
14	.	Austen	Ralston	M	29250	Service Assistant II	AU
84	120695	Trent	Moffat	M	28180	Warehouse Assistant II	au
87	120698	Geoff	Kistanna	M	26160	Warehouse Assistant I	au
101	120723	Deanna	Olsen		33950	Corp. Comm. Specialist II	US
125	120747	Zashia	Farthing	F	43590	Financial Controller I	us
197	120994	Danelle	Sergeant	F	31645	Office Administrator I	us
200	120997	Mary	Donathan	F	27420	Shipping Administrator I	us

original observation numbers

25

Acknowledgements: Portions of SAS' copyrighted SAS course content are reproduced here with permission of SAS Institute Inc., Cary, NC, USA. SAS Institute Inc. makes no warranties with respect to these materials and disclaims all liability therefor. Parts of additional materials are adapted from SAS Documentation, and web-based resources. Please let me know of any errors or typos you have found. Do not copy or circulate the handouts.

Using Formats in PROC FREQ

A FORMAT statement can be used in PROC FREQ to format data values.

```
proc freq data=orion.sales;
   tables Hire_Date / nocum;
   format Hire_Date date9.;
run;
```

Partial PROC FREQ Output

The FREQ Procedure

Hire_Date	Frequency	Percent
01JAN1978	17	10.30
01FEB1978	2	1.21
01APR1978	1	0.61
01JUL1978	1	0.61
01AUG1978	1	0.61

many discrete values, and not what the manager requested

26 p111d05

Using Formats in PROC FREQ

A FORMAT statement can also be used in PROC FREQ to group the data.

```
proc freq data=orion.sales;
   tables Hire_Date / nocum;
   format Hire_Date year4.;
run;
```

Partial PROC FREQ Output

The FREQ Procedure

Hire_Date	Frequency	Percent
1978	23	13.94
1979	2	1.21
1980	4	2.42
1981	3	1.82
1982	7	4.24

fewer discrete values

27 p111d05

PROC SURVEYFREQ - Example

```
1  *Example_surveyfreq.sas;
2  options nonumber nodate ls=132 ps=58;
3  %LET Path=C:\Users\Pradip Muhuri\SASClassGWU\TopicsByWeek\Week9;
4  LIBNAME xnew "&Path";
5  options fmtsearch=(xnew.formats);
6  proc surveyfreq data=xnew.xh163;
7        tables  age_group *AMTOTV_cat13 / cv deff nowt ;
8        strata  varstr;
9        cluster varpsu;
10       weight  perwt13f;
11      title1 "MEPS 2013 Public Use Files";
12 run;
```

MEPS 2013 Public Use Files

The SURVEYFREQ Procedure

Data Summary

Number of Strata	165
Number of Clusters	362
Number of Observations	36940
Number of Observations Used	35068
Number of Obs with Nonpositive Weights	1872
Sum of Weights	315721982

Table of age_group by AMTOTV_cat13

age_group	AMTOTV_ cat13	Frequency	Percent	Std Err of Percent	CV for Percent	Design Effect
<=17 Years	None	2813	5.6647	0.1865	0.0329	2.2835
	1-5	5938	14.1538	0.2931	0.0207	2.4797
	6-9	629	2.0138	0.1214	0.0603	2.6181
	10+	520	1.6172	0.1178	0.0729	3.0610
	Total	9900	23.4494	0.3629	0.0155	2.5733
18-64 Years	None	7679	18.7155	0.3468	0.0185	2.7721
	1-5	8773	26.8489	0.3505	0.0131	2.1933
	6-9	1881	6.1569	0.1754	0.0285	1.8667
	10+	2848	10.1129	0.2448	0.0242	2.3113
	Total	21181	61.8342	0.3931	0.0064	2.2962
65 Years	None	372	1.1913	0.0951	0.0798	2.6953
	1-5	1353	4.4896	0.1904	0.0424	2.9639
	6-9	718	2.5856	0.1353	0.0523	2.5484
	10+	1544	6.4500	0.2752	0.0427	4.4028
	Total	3987	14.7164	0.4205	0.0286	4.9404
Total	None	10864	25.5715	0.4243	0.0166	3.3165
	1-5	16064	45.4922	0.4042	0.0089	2.3103
	6-9	3228	10.7562	0.2572	0.0239	2.4175
	10+	4912	18.1801	0.3662	0.0201	3.1613
	Total	35068	100.000			

Ex8_surveyfreq.sas

Acknowledgements: Portions of SAS' copyrighted SAS course content are reproduced here with permission of SAS Institute Inc., Cary, NC, USA. SAS Institute Inc. makes no warranties with respect to these materials and disclaims all liability therefor. Parts of additional materials are adapted from SAS Documentation, and web-based resources. Please let me know of any errors or typos you have found. Do not copy or circulate the handouts.

PROC SURVEYMEANS

The SURVEYMEANS procedure is used when the analysis involves data that are collected using complex multi-stage probability survey design (US Surveys, such as NHIS, MEPS, and NSDUH). The conventional PROC MEANS is not appropriate to analyze such data. The design characteristics and the survey weights must be specified as shown in the following example.

```
1  *Ex9_Surveymeans_ODS_RTF.sas;
2  OPTIONS nocenter nodate nonumber ps=58 ls=132  ;
3  libname out "D:\A_Data";
4  libname library "D:\A_Data";
5  ods graphics off;
6  proc surveymeans data=out.MEPS_FYC_2014 nobs mean ;
7               stratum varstr;
8               cluster varpsu;
9               weight perwt14f ;
10              var TOTEXP14;
11              domain age_grp sex age_grp*sex('Male');
12              ods output Statistics=out.ALL_FYC14
13                     domain=out.Stat_FY14;
14 run;
```

Line 7: The VAR statement identifies the variables to be analyzed. The CLASS statement (not added to the above code) identifies those numeric variables that are to be analyzed as categorical variables.

Line 8: The DOMAIN statement lists the variables that define domains for subpopulation analysis.

Line 9: The STRATA statement lists the variables that form the strata in a stratified sample design.

Like 10: The CLUSTER statement specifies cluster identification variables in a clustered sample design.

Like 11: The WEIGHT statement names the sampling weight variable.

Acknowledgements: Portions of SAS' copyrighted SAS course content are reproduced here with permission of SAS Institute Inc., Cary, NC, USA. SAS Institute Inc. makes no warranties with respect to these materials and disclaims all liability therefor. Parts of additional materials are adapted from SAS Documentation, and web-based resources. Please let me know of any errors or typos you have found. Do not copy or circulate the handouts.

MEPS 2013 Public Use Files

The SURVEYMEANS Procedure

Data Summary

Number of Strata	165
Number of Clusters	362
Number of Observations	36940
Number of Observations Used	35068
Number of Obs with Nonpositive Weights	1872
Sum of Weights	315721982

Statistics

Variable	Label	N	Mean	Std Error of Mean	95% CL for Mean	
AMTEXP13	Ambulatory Care Expenses	35068	439.356710	33.843646	372.614365	506.099056

Domain Analysis: Age Group

Age Group	Variable	Label	N	Mean	Std Error of Mean	95% CL for Mean	
<=17 Years	AMTEXP13	Ambulatory Care Expenses	9900	289.499481	121.294552	50.297039	528.701923
18-64 Years	AMTEXP13	Ambulatory Care Expenses	21181	426.297196	27.969472	371.139187	481.455204
65 Years	AMTEXP13	Ambulatory Care Expenses	3987	733.013725	40.411117	653.319809	812.707641

Domain Analysis: Office/Outpatient Provider Visit Category

Office/Outpatient Provider Visit Category	Variable	Label	N	Mean	Std Error of Mean	95% CL for Mean	
None	AMTEXP13	Ambulatory Care Expenses	10864	0	0	0.00000	0.00000
1-5	AMTEXP13	Ambulatory Care Expenses	16064	122.084422	10.553738	101.27162	142.89723
6-9	AMTEXP13	Ambulatory Care Expenses	3228	461.858722	39.619989	383.72497	539.99247
10+	AMTEXP13	Ambulatory Care Expenses	4912	1837.940449	179.606065	1483.74309	2192.13780

PROC TABULATE

The TABULATE can be used to

- display descriptive statistics (e.g., N, SUM, and Mean) in tabular format
- produce tables in up to three dimensions and allows
- report multiple variables one after another hierarchically within each dimension
- label and format the variables as well as the statistics

This procedure is appropriate for summary reports, for example, when the row collapses or summarizes data based on the group or category variables (Zender, 2008). In contrast, the PROC PRINT is appropriate for detail reports in which every observation in the data is listed.

Ex3_proc_tabulate.sas
Ex4_Multilabel_Format.sas

One-Dimensional Table

Multiple TABLE statements can be in one step.

Country	
AU	US
N	N
63	102

```
proc tabulate data=orion.sales;
   class Country Gender;
   table Country;
   table Gender;
run;
```

Gender	
F	M
N	N
68	97

- N is the default statistic when there is only a CLASS statement and no VAR statement.

r102d02

27

Two-Dimensional Table

A two-dimensional table contains a row and a column dimension.

	Country	
	AU	US
	N	N
Gender		
F	27	41
M	36	61

```
proc tabulate data=orion.sales;
   class Country Gender;
   table Gender, Country;
run;
```

r102d02

28

Three-Dimensional Table

A three-dimensional table contains a page, a row, and a column dimension.

Job_Title Sales Rep. I

	Country	
	AU	US
	N	N
Gender		
F	8	13
M		

Job_Title Sales Rep. II

	Country	
	AU	US
	N	N
Gender		
F	10	14

Job_Title Sales Rep. III

	Country	
	AU	US
	N	N
Gender		
F	7	8

```
proc tabulate data=orion.sales;
   class Country Gender Job_Title;
   table Job_Title, Gender, Country;
run;
```

r102d02

29

Analysis Variable

SUM is the default statistic for an analysis variable that is used in the TABLE statement.

Job_Title Sales Rep. I

	Gender	
	F	M
	Salary	Salary
	Sum	Sum
Country		
AU	209390.00	346355.00
US	343730.00	769920.00

Analysis variables are defined in the VAR statement.

```
proc tabulate data=orion.sales;
   class Country Gender Job_Title;
   var Salary;
   table Job_Title, Country, Gender*Salary;
run;
```

r102d02

36

ALL Keyword

The keyword ALL can be included in the TABLE statement to summarize all of the categories for class variables in the given dimension.

	Country		All
	AU	US	
	N	N	N
Gender			
F	27	41	68
M	36	61	97
All	63	102	165

- The keyword ALL does not belong in the CLASS or VAR statement.

```
proc tabulate data=orion.sales;
   class Country Gender;
   table Gender All, Country All;
run;
```

r102d03

45

ALL Keyword

The keyword ALL can be specified in any of the three dimensions.

```
table Job_Title All,
      Gender All,
      Country All;
```

Job_Title Sales Rep. IV

	Country		All
	AU	US	
	N	N	
Gender			
F	2	5	
M	3	6	
All	5	11	

All

	Country		All
	AU	US	
	N	N	N
Gender			
F	27	41	68
M	36	61	97
All	63	102	165

r102d03

46

Concatenated Variables

The keyword ALL can be included with concatenated variables.

	Gender		All	Country		All
	F	M		AU	US	
	N	N	N	N	N	N
Job_Title						
Sales Rep. I	21	42	63	21	42	63
Sales Rep. II	24	22	46	18	28	46

```
proc tabulate data=orion.sales;
   class Country Gender Job_Title;
   table Job_Title, Gender All Country All;
run;
```

r102d03

47

Default Statistics

N is the default statistic when there is only a CLASS statement and no VAR statement.

	Country		All
	AU	US	
	N	N	N
Gender			
F	27	41	68
M	36	61	97
All	63	102	165

```
proc tabulate data=orion.sales;
   class Country Gender;
   table Gender All, Country All;
run;
```

r102d04

57

Default Statistics

SUM is the default statistic when there is a VAR statement with or without a CLASS statement.

		Country		All
		AU	US	
		Salary	Salary	Salary
		Sum	Sum	Sum
Job_Title	Gender			
Sales Rep. I	F	209390.00	343730.00	553120.00
	M	346355.00	769920.00	1116275.00
Sales Rep. II	F	270645.00	387120.00	657765.00
	M	215375.00	388965.00	604340.00
Sales Rep. III	F			
	M			
Sales Rep. IV	F			
	M			
All				

```
proc tabulate data=orion.sales;
   class Country Gender Job_Title;
   var Salary;
   table Job_Title*Gender All,
         Country*Salary All*Salary;
run;
```

r102d04

58

Default Statistics

The default statistics can be specified in the TABLE statement. The asterisk (*) is used to associate a statistic keyword with a variable or the keyword ALL.

```
table Gender All,
      Country*N All*N;
```

```
table Job_Title*Gender All,
      Country*Salary*Sum All*Salary*Sum;
```

r102d04

59

Statistic Keywords

The following descriptive statistics and quantile statistics can be specified in the TABLE statement:

CSS	CV	KURTOSIS	LCLM	MAX
MEAN	MIN	MODE	N	NMISS
RANGE	SKEWNESS	STDDEV	STDERR	SUM
SUMWGT	UCLM	USS	VAR	
PCTN	PCTSUM	REPPCTN	REPPCTSUM	PAGEPCTN
PAGEPCTSUM	ROWPCTN	ROWPCTSUM	COLPCTN	COLPCTSUM
MEDIAN	P1	P5	P10	P25
P75	P90	P95	P99	QRANGE

60

Statistic Keywords

	Country		Salary	Salary
	AU	US		
	N	N	Mean	Max
Job_Title				
Sales Rep. I	21	42	26498.33	32235.00
Sales Rep. II	18	28	27437.07	35990.00
Sales Rep. III	17	17	29446.32	36605.00
Sales Rep. IV	5	11		
All	61	98		

```
proc tabulate data=orion.sales;
   class Country Job_Title;
   var Salary;
   table Job_Title All,
         Country*N
         Salary*Mean Salary*Max;
run;
```

r102d04

61

Parentheses

Multiple statistics can be specified in parentheses.

	Gender		Gender	
	F	M	F	M
	Salary	Salary	Salary	Salary
	Min	Min	Max	Max
Country				
AU	25185.00	25745.00	30890.00	108255.00
US	25390.00	22710.00	83505.00	243190.00
All	25185.00	22710.00	83505.00	243190.00

```
table Country All,
      Gender*Salary*Min
      Gender*Salary*Max;
```

	Gender			
	F		M	
	Salary		Salary	
	Min	Max	Min	Max
Country				
AU	25185.00	30890.00	25745.00	1082
US	25390.00	83505.00	22710.00	2431
All	25185.00	83505.00	22710.00	2431

```
table Country All,
      Gender*Salary*(Min Max);
```

r102d04

64

REPPCTN, ROWPCTN, and COLPCTN

	Country		All	Country		All
	AU	US		AU	US	
	N	N	N	RepPctN	RepPctN	RepPctN
Gender						
F	27	41	68	16.36	24.85	41.21
M	36	61	97	21.82	36.97	58.79
All	63	102	165	38.18	61.82	100.00

```
table Gender All,
      Country*N All*N
      Country*Reppctn
      All*Reppctn;
```

	Country		All	Country		All
	AU	US		AU	US	
	N	N	N	RowPctN	RowPctN	RowPctN
Gender						
F	27	41	68	39.71	60.29	100.00
M	36	61	97	37.11	62.89	100.00
All	63	102	165	38.18	61.82	100.00

```
table Gender All,
      Country*N All*N
      Country*Rowpctn
      All*Rowpctn;
```

	Country		All	Country		All
	AU	US		AU	US	
	N	N	N	ColPctN	ColPctN	ColPctN
Gender						
F	27	41	68	42.86	40.20	41.21
M	36	61	97	57.14	59.80	58.79
All	63	102	165	100.00	100.00	100.00

```
table Gender All,
      Country*N All*N
      Country*Colpctn
      All*Colpctn;
```

71

PCTN and PCTSUM

The denominator within angle brackets (<>) specifies which categories to sum for the denominator.

```
table Gender*(Country All) All,
      Job_Title*(N Pctn<Country All>)
      All*(N Pctn<Country All>);
```

		Job_Title				All	
		Sales Rep. I		Sales Rep. II			
		N	PctN	N	PctN	N	PctN
Gender	Country						
F	AU	8	38.10	10	41.67	18	40.00
	US	13	61.90	14	58.33	27	60.00
	All	21	100.00	24	100.00	45	100.00
M	Country						
	AU	13	30.95	8	36.36	21	32.81
	US	29	69.05	14	63.64	43	67.19
	All	42	100.00	22	100.00	64	100.00
All		63	100.00	46	100.00	109	100.00

Adding the denominator with PCTN and PCTSUM offers a great deal of flexibility.

r102d05

75

Acknowledgements: Portions of SAS' copyrighted SAS course content are reproduced here with permission of SAS Institute Inc., Cary, NC, USA. SAS Institute Inc. makes no warranties with respect to these materials and disclaims all liability therefor. Parts of additional materials are adapted from SAS Documentation, and web-based resources. Please let me know of any errors or typos you have found. Do not copy or circulate the handouts.

```
1  *Example_proc_tabulate.sas;
2  options nonumber nodate ls=132 ps=58 ;
3  PROC TABULATE data=sashelp.heart format=comma7. ;
4    TITLE1 'Two Dimensional TABLE';
5    TITLE2 'N and Row Percentage';
6    CLASS weight_status BP_status;
7    TABLE weight_status all, (BP_Status all)*(N rowpctn*f=6.1);
8  run;
```

Line 6: The CLASS statement should include categorical variables (with a limited number of categories). The addition of numeric variables may result in more pages of output than you intended.

Line 7: The Table statement can have up to three dimension expressions as well as the table options. To identify different dimensions, use commas. The order of the dimensions is page, row, and column. In this example, two dimensions are specified – weight_status variable in the row dimension, and the BP_Status in the column dimension. Options can be added at the end after a '/'. In this example, no options are added. This is essentially a cross-table, and the code below requested two statistics, N and ROWPCTN. See SAS Documentation for the detailed list of statistics.

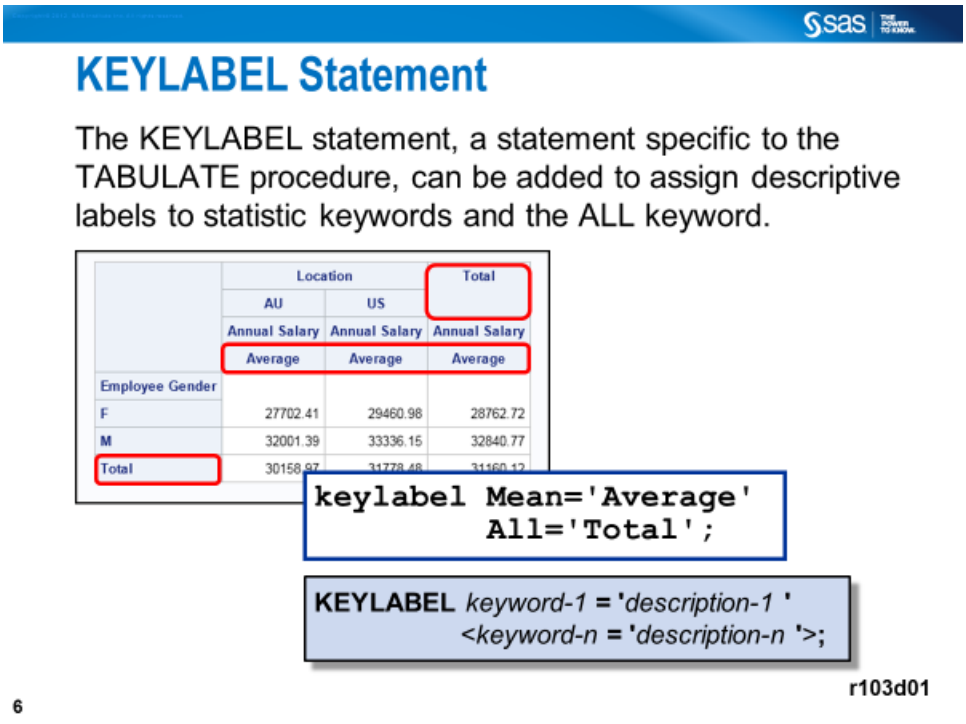
Two Dimensional TABLE
N and Row Percentage

	Blood Pressure Status							
	High		Normal		Optimal		All	
	N	RowPc-tN	N	RowPc-tN	N	RowPc-tN	N	RowPc-tN
Weight Status								
Normal	394	26.8	704	47.8	374	25.4	1,472	100.0
Overweight	1,839	51.8	1,340	37.7	371	10.5	3,550	100.0
Underweight	32	17.7	97	53.6	52	28.7	181	100.0
All	2,265	43.5	2,141	41.1	797	15.3	5,203	100.0

Acknowledgements: Portions of SAS' copyrighted SAS course content are reproduced here with permission of SAS Institute Inc., Cary, NC, USA. SAS Institute Inc. makes no warranties with respect to these materials and disclaims all liability therefor. Parts of additional materials are adapted from SAS Documentation, and web-based resources. Please let me know of any errors or typos you have found. Do not copy or circulate the handouts.

The previous coded has been extended by assigning labels to statistics using the KEYLABEL statement. No output is shown for code in lines 10-17.

```
10 PROC TABULATE data=sashelp.heart format=comma7. ;
11  TITLE1 'Two Dimensional TABLE';
12    TITLE2 'Variable Labels Changed and KEYLABEL Statement Added';
13    CLASS weight_status BP_Status;
14    KEYLABEL N='Total' rowpctn = 'Row %';
15    TABLE weight_status='Body Mass Index Category' all,
16          (BP_Status='Blood Pressure Category' all)*(N rowpctn*f=6.1);
17  run;
```



KEYLABEL Statement

If the KEYLABEL statement specifies a blank value for a statistic, the cell is removed from the table.

	Location		Total
	AU	US	
	Annual Salary	Annual Salary	Annual Salary
	Average	Average	Average
Employee Gender			
F	27702.41	29460.98	2876
M	32001.39	33336.15	3284
Total	30158.97	31778.48	3116

```
keylabel Mean='Average'
         All='Total';
```

	Location		Total
	AU	US	
	Annual Salary	Annual Salary	Annual Salary
Employee Gender			
F	27702.41	29460.98	2876
M	32001.39	33336.15	3284
Total	30158.97	31778.48	3116

```
keylabel Mean=' '
         All='Total';
```

r103d01

9

Labels in the TABLE Statement

If the cell containing a classification or analysis variable is not desired, a blank label for the cell must be specified in the TABLE statement.

	Location		Total
	AU	US	
	Annual Salary	Annual Salary	Annual Salary
Employee Gender			
F	27702.41	29460	
M	32001.39	33336	
Total	30158.97	31778	

```
table Gender All,
      Country*Salary*Mean
      All*Salary*Mean;
```

	Location		Total
	AU	US	
F	27702.41	29460.98	28762.72
M	32001.39	33336.15	32840.77
Total	30158.97	31778.48	31160.12

variable='*label* '

```
table Gender=' ' All,
      Country*Salary=' '*Mean
      All*Salary=' '*Mean;
```

r103d01

10

Labels in the TABLE Statement

A label can also be specified in the TABLE statement to change the heading of a keyword.

	Location		Column Total
	AU	US	
F	27702.41	29460.98	28762.72
M	32001.39	33336.15	32840.77
Row Total	30158.97	31778.48	31160.12

keyword='label '

```
table Gender=' ' All='Row Total',
      Country*Salary=' '*Mean
      All='Column Total'*Salary=' '*Mean;
```

Recommendation: Only put a label in the TABLE statement if the label cannot be accomplished in the LABEL or KEYLABEL statement.

r103d01

11

3.02 Quiz – Correct Answer

When a label for a variable or keyword is specified in the LABEL or KEYLABEL statement ***and*** the TABLE statement, which label is used?

```
proc tabulate data=orion.sales;
   class Gender Country;
   var Salary;
   table Gender='GEN' All='Row',
         Country*Salary='SAL'*Mean
         All='Column'*Salary=' '*Mean;
   label Gender='Employee Gender'
         Salary='Annual Salary';
   keylabel Mean=' '
            All='Total';
run;
```

	Country		Column
	AU	US	
	SAL	SAL	
GEN			
F	27702.41	29460.98	28762.72
M	32001.39	33336.15	32840.77
Row	30158.97	31778.48	31160.12

13

FORMAT Statement

The FORMAT statement can be added to the TABULATE procedure to assign a format to a classification variable.

	Country		All
	Australia	United States	
	Salary	Salary	Salary
	Mean	Mean	Mean
Gender			
Female	27702.41	29460.98	28762.72
Male	32001.39	33336.15	32840.77
All	30158.97	31778.48	31160.12

```
format Gender $genfmt.
       Country $ctryfmt.;
```

FORMAT *variable-1 <variable-n> format*
<variable-2 <variable-n> format>;

r103d02

14

FORMAT= Option

The FORMAT= option can be added to the PROC TABULATE statement to specify a format for the statistical cells.

	Country				All	
	Australia		United States			
	Salary		Salary		Salary	
	N	Mean	N	Mean	N	Mean
Gender						
Female	27	27,702	41	29,461	68	28,763
Male	36	32,001	61	33,336	97	32,841
All	63	30,159	102	31,778	165	31,160

```
proc tabulate data=orion.sales
              format=comma12.;
```

FORMAT=*format*

r103d02

17

Formats in the TABLE Statement

A format modifier can be added to the TABLE statement by using the asterisk (*) to associate a format modifier with an analysis variable or keyword.

	Country				All	
	Australia		United States			
	Salary		Salary		Salary	
	N	Mean	N	Mean	N	Mean
Gender						
Female	27	27,702	41	29,461	68	$28,762.72
Male	36	32,001	61	33,336	97	$32,840.77
All	63	30,159	102	31,778	165	$31,160.12

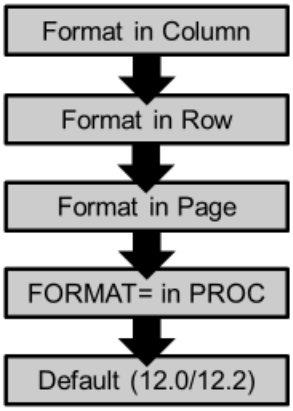
F=*format*

```
table Gender All,
      Country*Salary*(N Mean)
      All*Salary*(N Mean*f=dollar12.2);
```

18

Format Precedence

The following is the order for formats associated with the statistical cells.



By default, the format specified in the column dimension is used over any other format.

If no format is specified in any of the dimensions, the format specified with the FORMAT= option in the PROC statement is used over the default format.

19

3.03 Quiz – Correct Answer

Which format is used for the mean column?

```
proc tabulate data=orion.sales
              format=comma12.;
   class Gender Country;
   var Salary;
   table Gender All*f=dollar12.,
         Country*Salary*(N Mean*f=dollar12.2);
   format Gender $genfmt.
          Country $ctryfmt.;
run;
```

	Country			
	Australia		United States	
	Salary		Salary	
	N	Mean	N	Mean
Gender				
Female	27	$27,702.41	41	$29,460.98
Male	36	$32,001.39	61	$33,336.15
All	$63	$30,158.97	$102	$31,778.48

21

3.04 Quiz – Correct Answer

Why should the PERCENT*w.d* format ***not*** be used in order to create the desired report?

Initial Report

	Job_Title				All	
	Sales Rep. I		Sales Rep. II			
	N	PctN	N	PctN	N	PctN
Gender						
F	21	19.27	24	22.02	45	41.28
M	42	38.53	22	20.18	64	58.72
All	63	57.80	46	42.20	109	100.00

Desired Report

	Job_Title				All	
	Sales Rep. I		Sales Rep. II			
	N	PctN	N	PctN	N	PctN
Gender						
F	21	19.27%	24	22.02%	45	41.28%
M	42	38.53%	22	20.18%	64	58.72%
All	63	57.80%	46	42.20%	109	100.00%

The PERCENT*w.d* format multiplies values by 100 and adds a percent sign (%). The above values do not need to be multiplied by 100.

r103d02

24

Percent Format

A picture format can be created in order to add a percent sign to a percentage value.

```
proc format;
   picture pct(round) low-high='009.99%';
run;
```

PROC FORMAT;
PICTURE *name* <(*format-option(s)*)>
value-or-range-1 <..., *value-or-range-n*>='*picture*'
value-or-range-n <..., *value-or-range-n*>='*picture*';
RUN;

- The PICTURE statement creates a template for printing numbers.
- The ROUND option rounds the value before formatting.

25

continued...

Percent Format

- The nonzero digit selector (2-9) prints a digit.
- The zero-digit selector (0) prints a digit but not a zero.

```
proc format;
   picture pct(round) low-high='009.99%';
run;
```

The above step produces the following formatted values based on the stored values:

Stored value	.5	1.234	25	99.999
Formatted value	0.50%	1.23%	25.00%	100.00%

26

Percent Format

After the picture format is created, the format can be used in the TABULATE procedure.

```
proc format;
   picture pct(round) low-high='009.99%';
run;
proc tabulate data=orion.sales;
   class Gender Job_Title;
   table Gender All,
         Job_Title*(N Pctn*f=pct.)
         All*(N Pctn*f=pct.);
run;
```

	Job_Title				All	
	Sales Rep. I		Sales Rep. II			
	N	PctN	N	PctN	N	PctN
Gender						
F	21	19.27%	24	22.02%	45	41.28%
M	42	38.53%	22	20.18%	64	58.72%
All	63	57.80%	46	42.20%	109	100.00%

r103d02

27

BOX= Option

The BOX= option places text in the box area. This option is specified in the TABLE statement after a slash.

Warehouse Title	Gender	
	F	M
Warehouse Assistant I	5	
Warehouse Assistant II	3	
Warehouse Assistant III	1	
Warehouse Assistant IV	1	
Warehouse Manager		1

```
table Job_Title=' ', Gender*N=' '
      / box='Warehouse Title';
```

BOX = '*string*'

Job Title	Gender	
	F	
Warehouse Assistant I	5	
Warehouse Assistant II	3	
Warehouse Assistant III	1	
Warehouse Assistant IV	1	
Warehouse Manager		1

```
label Job_Title='Job Title';
table Job_Title=' ', Gender*N=' '
      / box=Job_Title;
```

BOX=*variable*

34

continued...

BOX= Option

The _PAGE_ value places the page dimension text in the box area.

Country AU	Gender	
	F	M
Job Title		
Warehouse Assistant I	2	2
Warehouse Assistant II	1	1
Warehouse Assistant III		1
Warehouse Assistant IV	1	

```
table Country, Job_Title,
      Gender*N=' '
      / box=_page_;
```

BOX=_PAGE_

Country US	Gender	
	F	M
Job Title		
Warehouse Assistant I	3	1
Warehouse Assistant II	2	
Warehouse Assistant III	1	
Warehouse Manager		1

r103d03

35

Missing Analysis Values

By default, missing analysis values are displayed with a period.

Job Title	Gender	
	F	M
Warehouse Assistant I	5	3
Warehouse Assistant II	3	1
Warehouse Assistant III	1	1
Warehouse Assistant IV	1	.
Warehouse Manager	.	1

36

MISSTEXT= Option

The MISSTEXT= option specifies a text string for missing analysis values. This option is specified in the TABLE statement after a slash.

Job Title	Gender	
	F	M
Warehouse Assistant I	5	3
Warehouse Assistant II	3	1
Warehouse Assistant III	1	1
Warehouse Assistant IV	1	None
Warehouse Manager	None	1

```
table Job_Title=' ', Gender*N=' '
      / box='Job Title'
        misstext='None';
```

MISSTEXT='*string*'

r103d03

37

MISSING= Option

An alternative to the MISSTEXT= option is the MISSING= option in the OPTIONS statement.

MISSING=*character*

```
options missing=0;

proc tabulate data=orion.warehouse;
   class Gender Job_Title;
   table Job_Title=' ', Gender*N=' '
         / box='Job Title';
run;

options missing=.;
```

resets back to the default

The MISSING= option specifies the character to be printed for missing numeric values but is limited to one character.

r103d03

38

MISSING Option

The MISSING option includes observations with missing classification values. If the option is placed in the PROC statement, all missing classification values are included.

```
proc tabulate data=orion.warehouse missing;
   class Gender Job_Title;
   table Job_Title=' ', Gender*N=' '
         / box='Job Title';
run;
```

Job Title	Gender		
		F	M
	.	1	.
Warehouse Assistant I	1	5	3
Warehouse Assistant II	.	3	1
Warehouse Assistant III	.	1	1
Warehouse Assistant IV	.	1	.
Warehouse Manager	.	.	1

r103d03

42

MISSING Option

The MISSING option can be placed in the CLASS statement to control which missing classification values are included.

```
proc tabulate data=orion.warehouse;
   class Gender / missing;
   class Job_Title;
   table Job_Title=' ', Gender*N=' '
         / box='Job Title';
run;
```

Job Title	Gender	
	F	M
Warehouse Assistant I	5	3
Warehouse Assistant II	3	1
Warehouse Assistant III	1	1
Warehouse Assistant IV	1	.
Warehouse Manager	.	1

The MISSING option is specified in the CLASS statement after a slash.

43 r103d03

PRINTMISS Option

The PRINTMISS option creates row and column headings that are the same for all pages of a table. This option is specified in the TABLE statement after a slash.

```
table Country, Job_Title, Gender*N=' '
      / box=_page_ printmiss;
```

Country AU	Gender	
Job_Title	F	M
Warehouse Assistant I	2	2
Warehouse Assistant II	1	1
Warehouse Assistant III	.	1
Warehouse Assistant IV	1	.
Warehouse Manager	.	

Country US	Gender	
Job_Title	F	M
Warehouse Assistant I	3	1
Warehouse Assistant II	2	.
Warehouse Assistant III	1	.
Warehouse Assistant IV	.	.
Warehouse Manager	.	1

47 r103d03

Trafficlighting

The STYLE= option can be added to the TABULATE procedure in order to incorporate trafficlighting.

Job Title	Gender			
	Female		Male	
	Salary		Salary	
	N	Mean	N	Mean
Chief Marketing Officer	1	$207,885	.	.
Marketing Assistant I	1	$27,380	.	.
Marketing Assistant II	2	$29,203	3	$29,772
Marketing Assistant III	.	.	1	$31,630
Marketing Assistant IV	.	.	1	$34,925
Marketing Manager	1	$63,640	2	$63,165
Senior Marketing Manager	1	$87,420	.	.

Trafficlighting is the act of highlighting individual cells based on the cell's value.

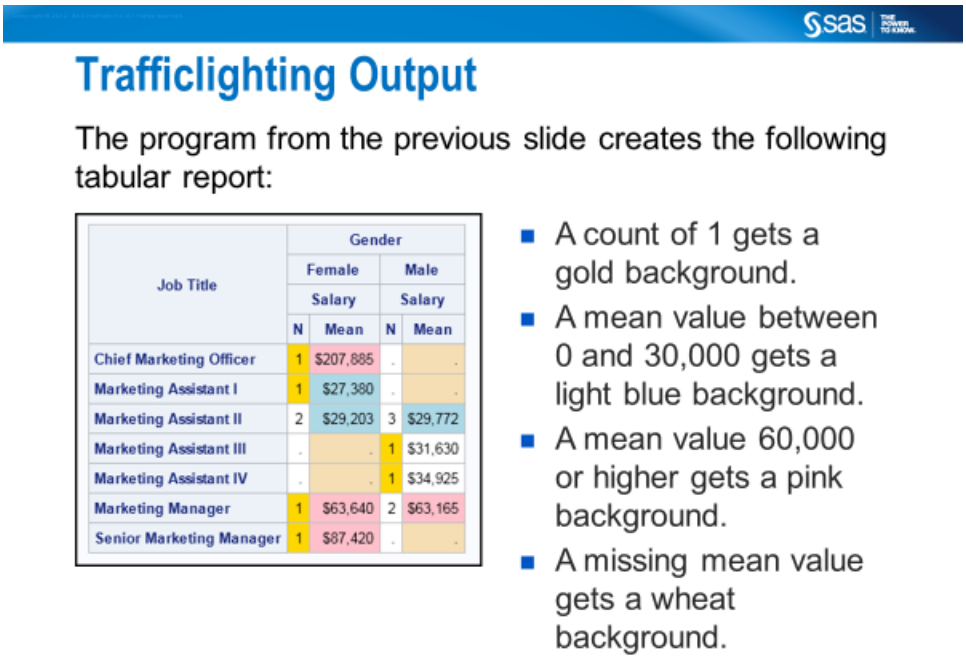
r103d04

60

```
proc format;
   value ncell 1 = 'Gold';
   value meancell 0-30000 = 'LightBlue'
                  60000-high = 'Pink'
                  . = 'Wheat';
run;

proc tabulate data=orion.marketing;
   class Gender Job_Title;
   var Salary;
   table Job_Title=' ',
         Gender*Salary*
         (N*{style={background=ncell.}}
          Mean*f=dollar12.*
          {style={background=meancell.}})
         / box=Job_Title;
   label Job_Title='Job Title';
   format Gender $genfmt.;
run;
```

64



Below is an example of a 3-dimensional table (Lines 20-29).

Line 24: The VAR statement includes a numeric variable (i.e., weight) for which the mean is calculated for cross-classification of variables that are listed in the CLASS statement. Note the requested statistics of N and MEAN in the TABLE statement (Lines 26-28).

```
20 PROC TABULATE data=sashelp.heart format=comma7. ;
21 TITLE1 'Three Dimensional TABLE';
22   TITLE2 'Mean Weight';
23   CLASS weight_status BP_Status sex;
24   VAR  weight;
25   KEYLABEL N='Total'  mean = 'Mean (lbs)';
26   TABLE (sex all), weight_status='Body Mass Index Category' all,
27         (BP_Status='Blood Pressure Category' all)
28         *(N weight*mean);
29 run;
```

Ex3_proc_tabulate.sas
Ex4_Multilabel_Format.sas

Three Dimensional TABLE
Mean Weight

Sex Female

	Blood Pressure Category						All	
	High		Normal		Optimal			
	Total	Weight Mean (lbs)	Total	Weight Mean (lbs)	Total	Weight Mean (lbs)	Total	Weight Mean (lbs)
Body Mass Index Category								
Normal	207	122	387	121	252	120	846	121
Overweight	961	158	716	149	230	142	1,907	153
Underweight	18	100	61	104	37	105	116	104
All	1,186	151	1,164	137	519	129	2,869	141

Sex Male

	Blood Pressure Category						All	
	High		Normal		Optimal			
	Total	Weight Mean (lbs)	Total	Weight Mean (lbs)	Total	Weight Mean (lbs)	Total	Weight Mean (lbs)
Body Mass Index Category								
Normal	187	144	317	146	122	145	626	145
Overweight	878	181	624	174	141	172	1,643	178
Underweight	14	128	36	123	15	122	65	124
All	1,079	174	977	163	278	158	2,334	167

Three Dimensional TABLE
Mean Weight

All

	Blood Pressure Category						All	
	High		Normal		Optimal			
	Total	Weight Mean (lbs)	Total	Weight Mean (lbs)	Total	Weight Mean (lbs)	Total	Weight Mean (lbs)
Body Mass Index Category								
Normal	394	132	704	132	374	128	1,472	131
Overweight	1,839	169	1,340	161	371	154	3,550	164
Underweight	32	113	97	111	52	110	181	111
All	2,265	162	2,141	149	797	139	5,203	153

Acknowledgements: Portions of SAS' copyrighted SAS course content are reproduced here with permission of SAS Institute Inc., Cary, NC, USA. SAS Institute Inc. makes no warranties with respect to these materials and disclaims all liability therefor. Parts of additional materials are adapted from SAS Documentation, and web-based resources. Please let me know of any errors or typos you have found. Do not copy or circulate the handouts.

The above code has been modified to generate a 2-dimensional table by concatenating page and row dimensions.

```
31 PROC TABULATE data=sashelp.heart format=comma7.;
32  TITLE1 'Concatenated Rows - Two Dimensional TABLES';
33    TITLE2 'Mean Weight';
34    CLASS weight_status sex bp_status;
35    VAR  weight;
36    KEYLABEL N='Total'  mean = 'Mean (lbs)';
37    TABLE (sex all)*weight_status='Body Mass Index Category' all,
38           (bp_status='Blood Pressure Category' all)
39               *(N weight*mean);
40    run;
```

Concatenated Rows - Two Dimensional TABLES
Mean Weight

		Blood Pressure Category							
		High		Normal		Optimal		All	
		Total	Weight Mean (lbs)	Total	Weight Mean (lbs)	Total	Weight Mean (lbs)	Total	Weight Mean (lbs)
Sex	Body Mass Index Category								
Female	Normal	207	122	387	121	252	120	846	121
	Overweight	961	158	716	149	230	142	1,907	153
	Underweight	18	100	61	104	37	105	116	104
Male	Normal	187	144	317	146	122	145	626	145
	Overweight	878	181	624	174	141	172	1,643	178
	Underweight	14	128	36	123	15	122	65	124
All	Normal	394	132	704	132	374	128	1,472	131
	Overweight	1,839	169	1,340	161	371	154	3,550	164
	Underweight	32	113	97	111	52	110	181	111
All		2,265	162	2,141	149	797	139	5,203	153

MULTILABEL Option in PROC TABULATE

```
1  *Ex4_Multilabel_Format.sas ;
2  options nodate nonumber;
3  proc format ;
4   value m_agefmt (multilabel notsorted)
5                  low-34 = '25-34 Years'
6                  35-44 = '35-44 Years'
7                  45-54 = '45-54 Years'
8                  55-64 = '55-64 Years'
9                  low-49= '25-49 Years'
10                 50-64 ='50-64 Years';
11
12          value m_agefmt_x (multilabel)
13                 low-34 = '25-34 Years'
14                 35-44 = '35-44 Years'
15                 45-54 = '45-54 Years'
16                 55-64 = '55-64 Years'
17                 low-49= '25-49 Years'
18                 50-64 ='50-64 Years';
```

The option MULTILABEL allows the assignment of multiple labels to the values from the AgeATStart variable in SASHELP.HEART data file. It also allows the assignment of multiple labels to the overlapped subtotal categories.

Lines 4-10: The MULTILABEL option is used to create a 'subtotal' for ages 25-49 and another ‘subtotal’ for ages 50-64. Note that these two subtotals overlap the four age groups. This method can be applied to any format when subtotals are required. With this format, the TABULATE procedure would provide the ‘count’ and ‘statistics’ for individual categories as well the subtotal categories. The NOTSORTED option is used to retain the specified order required for reporting.

Line 12-18: Only the MULTILABEL option is specified; the NOTSORTED option is added to the VALUE statement.

Ex4_Multilabel_Format.sas

```
20 proc tabulate data=sashelp.heart;
21    class AgeAtStart/mlf preloadfmt order=data;
22    var AgeAtdeath;
23    table AgeAtStart all,
24      n*format=5.0 (AgeAtdeath)*mean*format=4.1;
25    Format AgeAtStart m_agefmt.;
26    title1 'Value m_agefmt (multilabel notsorted)';
27    title2 ' Class AgeAtStart/mlf preloadfmt order=data';
28 run;
```

Line 21: The MLF option is added to activate the MULTILABEL option that was used in the VALUE statement in PROC FORMAT.

" …The option ORDER=DATA tells the tabulate to group the class variables by their order in the input data-set. Next the PRELOADFMT option indicates that the formats for the class variables are already loaded. This is used together with the ORDER=DATA option to ensure that the variables are displayed in the order the format has created. The MLF option indicates that the format is a multi-label format and allows for overlapping ranges to be displayed in the table." Morris (2011)

Value m_agefmt (multilabel notsorted)
Class AgeAtStart/mlf preloadfmt order=data

		All
		Age at Dea-th
	N	Mean
Age at Start		
25-34 Years	850	55.2
35-44 Years	1960	62.6
45-54 Years	1614	71.7
55-64 Years	785	78.6
25-49 Years	3618	64.1
50-64 Years	1591	76.3
All	5209	70.5

Acknowledgements: Portions of SAS' copyrighted SAS course content are reproduced here with permission of SAS Institute Inc., Cary, NC, USA. SAS Institute Inc. makes no warranties with respect to these materials and disclaims all liability therefor. Parts of additional materials are adapted from SAS Documentation, and web-based resources. Please let me know of any errors or typos you have found. Do not copy or circulate the handouts.

Ex4_Multilabel_Format.sas

```
30 proc tabulate data=sashelp.heart;
31    class AgeAtStart/mlf;
32    var AgeAtdeath;
33    table AgeAtStart all,
34      n*format=5.0 (AgeAtdeath)*mean*format=4.1;
35    Format AgeAtStart m_agefmt_x.;
36    title1 'Value m_agefmt (multilabel)';
37    title2 'Class AgeAtStart/mlf';
38 run;
```

Line 31: Note the addition of the MLF option to the CLASS statement (but not the PRELOADFMT and the ORDER=DATA options).

Value m_agefmt (multilabel)
Class AgeAtStart/mlf

		All
		Age at Death
	N	Mean
Age at Start		
25-34 Years	850	55.2
25-49 Years	3618	64.1
35-44 Years	1960	62.6
45-54 Years	1614	71.7
50-64 Years	1591	76.3
55-64 Years	785	78.6
All	5209	70.5

UNIVARIATE Procedure

PROC UNIVARIATE displays extreme observations, missing values, and other statistics for the variables included in the VAR statement.

```
proc univariate data=orion.nonsales2;
   var Salary;
run;
```

PROC UNIVARIATE DATA=*SAS-data-set*;
    <**VAR** *variable(s)*;>
RUN;

If the VAR statement is omitted, PROC UNIVARIATE analyzes all numeric variables in the data set.

46 p111d11

Viewing the Output: Extreme Observations

The *Extreme Observations* section includes the five lowest and five highest values for the analysis variable and the corresponding observation numbers.

Partial PROC UNIVARIATE Output

Extreme Observations

-----Lowest----		-----Highest----	
Value	Obs	Value	Obs
2401	20	163040	1
2650	13	194885	231
24025	25	207885	28
24100	19	268455	29
24390	228	433800	27

Obs is the observation number, not the count of observations with that value.

47

NEXTROBS= Option

The *NEXTROBS= option* specifies the number of extreme observations to display.

```
proc univariate data=orion.nonsales2
                nextrobs=3;
   var Salary;
run;
```

Partial PROC UNIVARIATE Output

```
        The UNIVARIATE Procedure
            Variable:  Salary

          Extreme Observations

-----Lowest----        -----Highest----

  Value      Obs          Value      Obs

   2401       20         207885       28
   2650       13         268455       29
  24025       25         433800       27
```

48

p111d11

ID Statement

The ID statement displays the value of the identifying variable (or variables) in addition to the observation number.

```
proc univariate data=orion.nonsales2;
   var Salary;
   id Employee_ID;
run;
```

ID *variable(s)*;

49

p111d11

Output Delivery System (ODS)

Use the Output Delivery System to perform the following tasks:

- alter the appearance of the output
  - style, color, font, and so on
- change the destination of the output
- select output
- create new data tables from PROC step output

`Ex10_Freq_ods_output_crosstabs.sas`

- Use ODS SELECT to request specific output.
- Use ODS OUTPUT to create a new SAS data table.
- Use the OUTPUT statement to customize the creation of a new SAS data table.

Ex7_ODS_Files.sas;

```
28 ODS html;
29 ODS OUTPUT ExtremeObs = Extreme_Obs_x;
30 ODS SELECT ExtremeObs;
31 proc univariate data=sashelp.class;
32 var weight;
33 run;
34 ODS html CLOSE;
```

The UNIVARIATE Procedure
Variable: Weight

Extreme Observations			
Lowest		Highest	
Value	Obs	Value	Obs
50.5	11	112.5	1
77.0	13	112.5	8
83.0	6	128.0	16
84.0	9	133.0	17
84.0	2	150.0	15

```
17 ods listing;
18 ods output basicmeasures=measures;
19 ods select basicmeasures;
20 proc univariate data=sashelp.class noprint;
21 var weight;
22 run;
23 proc print data=measures; run;
```

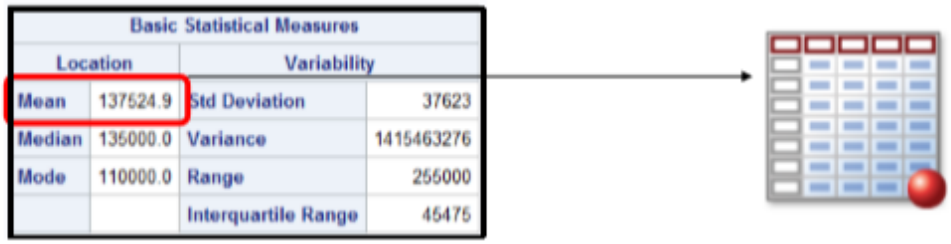
Obs	Var Name	Loc Measure	LocValue	VarMeasure	VarValue
1	Weight	Mean	100.0263	Std Deviation	22.77393
2	Weight	Median	99.5000	Variance	518.65205
3	Weight	Mode	84.0000	Range	99.50000
4	Weight		_	Interquartile Range	28.50000

Output Delivery System (ODS)

Use the OUTPUT statement to customize the new data table.

```
proc univariate data=ameshousing;
   var saleprice;
   output out=stats mean=sp_mean;
run;
```

OUTPUT OUT=*new-data-set-name*
keyword-1 = variable-name-1 … keyword-n = variable-name-n;



Basic Statistical Measures

Location		Variability	
Mean	137524.9	Std Deviation	37623
Median	135000.0	Variance	1415463276
Mode	110000.0	Range	255000
		Interquartile Range	45475

***Ex7_ODS_Files.sas;**

```
34 proc univariate data=sashelp.class noprint;
35    var weight;
36    output out=stats mean=mean_weight
37                     median=median_weight;
38 run;
39 proc print data=stats; run;
```

Obs	mean_ weight	median_ weight
1	100.026	99.5

You can specify multiple OUTPUT statements to create multiple data tables for individual values to place in the new tables [SAS Documentation].

ODS provides various formatting options that are not available when using individual procedures or the DATA step without ODS enhancements.

Destinations Used with Excel

- The CSVALL destination creates a CSV (comma-separated value) file.
- The EXCELXP destination creates an XML (Extensible Markup Language) file.
- The MSOFFICE2K destination creates an HTML (HyperText Markup Language) file.

```
ods csvall file='shoes.csv';
ods tagsets.excelxp file='shoes.xml';
ods msoffice2k file='shoes.html';

proc print data=sashelp.shoes;
 ...
run;

ods _all_ close;
```

127

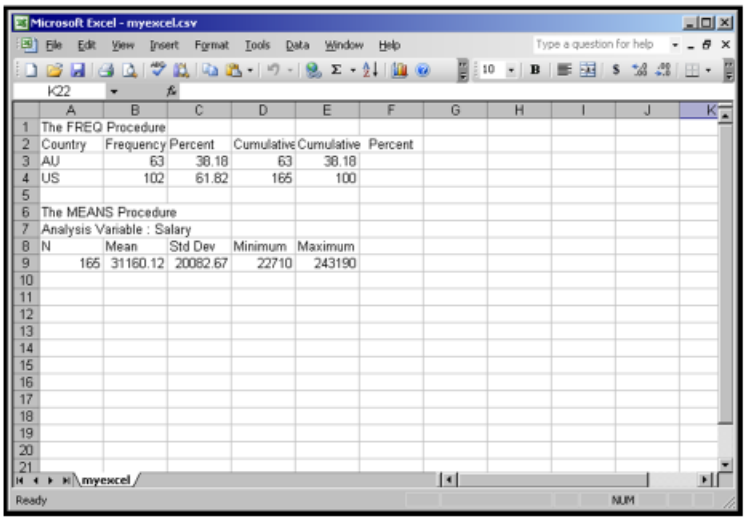
Destinations Used with Excel

Destination	Type of File	Extension	Viewed In
CSVALL	Comma-Separated Value	csv	Editor or Microsoft Excel
MSOFFICE2K	Hypertext Markup Language	html	Web browser or Microsoft Word or Microsoft Excel
TAGSETS. EXCELXP	Extensible Markup Language	xml	Microsoft Excel

70

CSVALL Destination

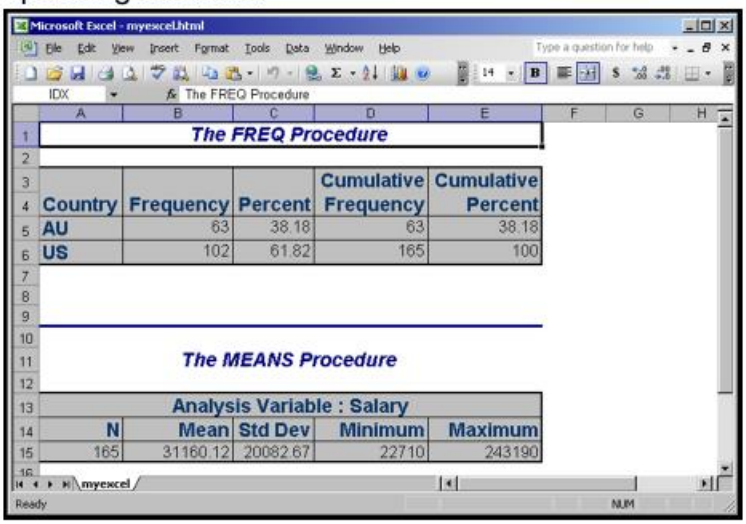
CSVALL does not include any style information.



72

MSOFFICE2K Destination

MSOFFICE2K keeps the style information, including spanning headers.



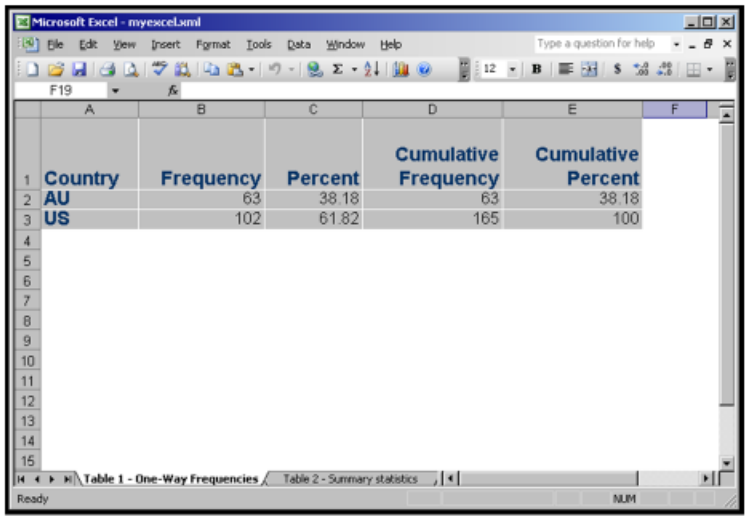
73

Ex11_to_Excel.sas

Acknowledgements: Portions of SAS' copyrighted SAS course content are reproduced here with permission of SAS Institute Inc., Cary, NC, USA. SAS Institute Inc. makes no warranties with respect to these materials and disclaims all liability therefor. Parts of additional materials are adapted from SAS Documentation, and web-based resources. Please let me know of any errors or typos you have found. Do not copy or circulate the handouts.

EXCELXP Destination

EXCELXP keeps the style information. Output from each procedure is on a separate sheet.



Country	Frequency	Percent	Cumulative Frequency	Cumulative Percent
AU	63	38.18	63	38.18
US	102	61.82	165	100

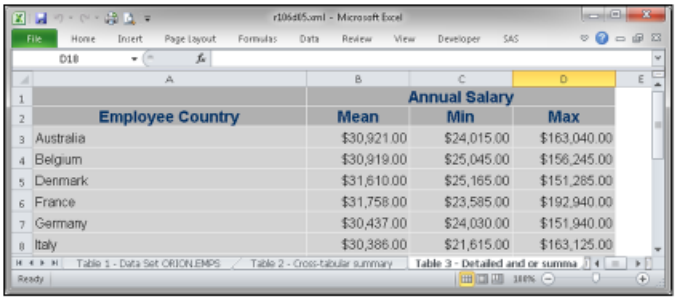
74

TAGSETS.EXCELXP Destination

The EXCELXP destination is based on SpreadsheetML, which is an XML dialect developed by Microsoft to represent information in an Excel workbook.

```
ods tagsets.excelxp file="&path2\r106d05.xml";
```

When the XML file is opened in Excel, each worksheet represents a different output table.



Employee Country	Annual Salary Mean	Min	Max
Australia	$30,921.00	$24,015.00	$163,040.00
Belgium	$30,919.00	$25,045.00	$156,245.00
Denmark	$31,610.00	$25,165.00	$151,285.00
France	$31,758.00	$23,585.00	$192,940.00
Germany	$30,437.00	$24,030.00	$151,940.00
Italy	$30,386.00	$21,615.00	$163,125.00

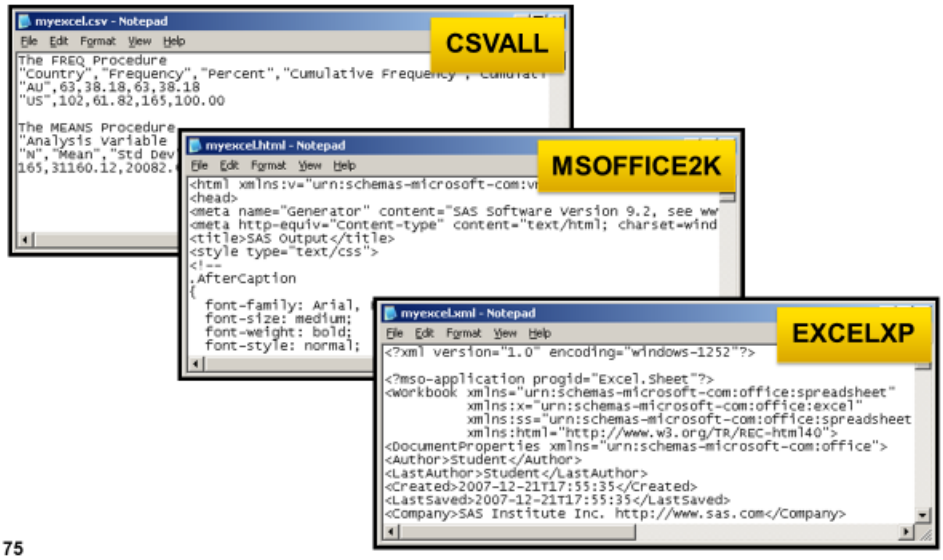
37

r106d05

Acknowledgements: Portions of SAS' copyrighted SAS course content are reproduced here with permission of SAS Institute Inc., Cary, NC, USA. SAS Institute Inc. makes no warranties with respect to these materials and disclaims all liability therefor. Parts of additional materials are adapted from SAS Documentation, and web-based resources. Please let me know of any errors or typos you have found. Do not copy or circulate the handouts.

Keep in Mind

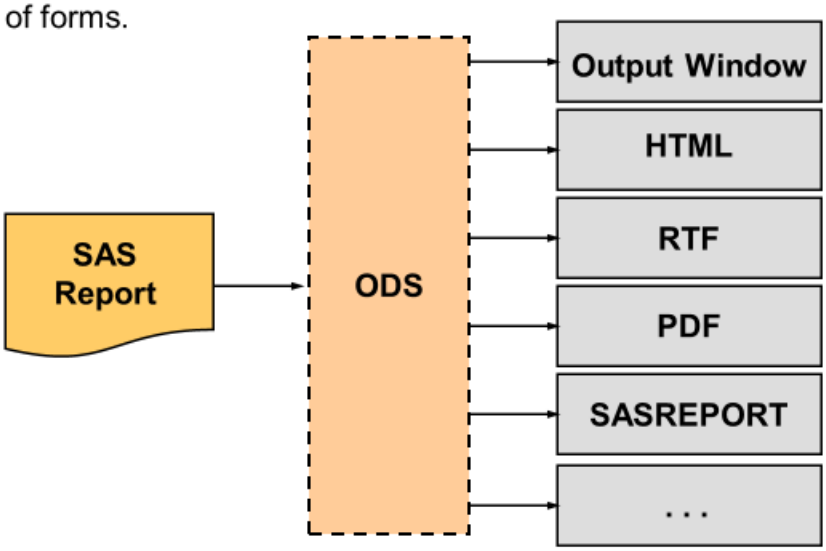
The file you are creating is not an Excel file.



75

Output Delivery System (ODS)

ODS statements enable you to create output in a variety of forms.



113

ODS RTF and PDF Statements

- The *ODS RTF statement* opens or closes the RTF destination, which produces Rich Text Format files that are viewable with a word processor.
- The *ODS PDF statement* opens or closes the PDF destination, which produces Portable Document Format files that are viewable with an Adobe product.

```
ods rtf file='shoes.rtf';
ods pdf file='shoes.pdf';

proc freq data=sashelp.shoes;
   ...
run;

ods rtf close;
ods pdf close;
```

120

Multiple ODS Destinations

The ODS _ALL_ CLOSE statement closes ***all*** open destinations including the LISTING destination.

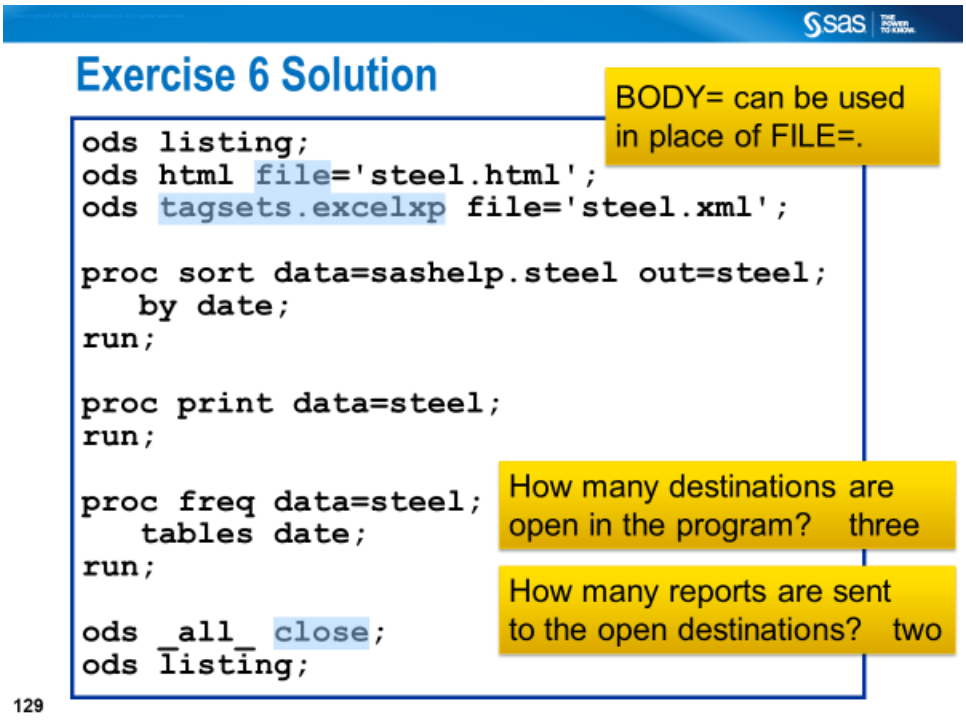
```
ods listing;
ods html file='shoes.html';
ods rtf file='shoes.rtf';
ods pdf file='shoes.pdf';

proc print data=sashelp.shoes;
  ...
run;

ods _all_ close;
ods listing;
```

124

Ex5_ODS_RTF_PDF_EXCEL.sas"
Ex6_ODS_Multiple_sheets.sas"



Ex_Week_7_List_of_Files.sas