SAS Certification Prep Guide: Base Programming for SAS 9, Third Edition

by SAS Institute

SAS Institute. (c) 2011. Copying Prohibited.

Reprinted for Shane Mc Carthy, Accenture

shane.mc.carthy@accenture.com

Reprinted with permission as a subscription benefit of **Skillport**,
http://skillport.books24x7.com/

All rights reserved. Reproduction and/or distribution in whole or in part in electronic,paper or other forms without written permission is prohibited.

Chapter 8: Producing Descriptive Statistics

Overview

Introduction

As you have seen, one of the many features of PROC REPORT is the ability to summarize large amounts of data by producing descriptive statistics. However, there are SAS procedures that are designed specifically to produce various types of descriptive statistics and to display them in meaningful reports. The type of descriptive statistics that you need and the SAS procedure that you should use depend on whether you need to summarize continuous data values or discrete data values.

If the data values that you want to describe are continuous numeric values (for example, people's ages), then you can use the MEANS procedure or the SUMMARY procedure to calculate statistics such as the mean, sum, minimum, and maximum.

Variable	N	Mean	Std Dev	Minimum	Maximum
Age	20	47	13	15	63
Height	20	67	4	61	75
Weight	20	175	36	102	240
Pulse	70	75	8	65	100
FastGluc	20	299	126	152	568
PostGluc	20	355	126	206	625

Figure 8.1: MEANS Procedure Output

If the data values that you want to describe are discrete (for example, the color of people's eyes), then you can use the FREQ procedure to show the distribution of these values, such as percentages and counts.

Eye Color	Frequency	Percent	Cumulative Frequency	Cumulative Percent
Brown	92	58.60	92	58.60
Blue	65	41.40	157	100.00

Figure 8.2: FREQ Procedure Output

This chapter will show you how to use the MEANS, SUMMARY, and FREQ procedures to describe your data.

Objectives

In this chapter, you learn to

- determine the *n*-count, mean, standard deviation, minimum, and maximum of numeric variables using the MEANS procedure
- control the number of decimal places used in PROC MEANS output
- specify the variables for which to produce statistics
- use the PROC SUMMARY procedure to produce the same results as the PROC MEANS procedure
- describe the difference between the SUMMARY and MEANS procedures
- create one-way frequency tables for categorical data using the FREQ procedure
- create two-way and-way crossed frequency tables
- control the layout and complexity of crossed frequency tables.

Reprinted for OET7P/856094, Accenture SAS Institute, SAS Institute Inc (c) 2011, Copying Prohibited

Computing Statistics Using PROC MEANS

Descriptive statistics such as the mean, minimum, and maximum provide useful information about numeric data. The MEANS procedure provides these and other data summarization tools, as well as helpful options for controlling your output.

Procedure Syntax

The MEANS procedure can include many statements and options for specifying needed statistics. For simplicity, let's consider the procedure in its basic form.

General form, basic MEANS procedure:

```
PROC MEANS <DATA=SAS-data-set>
           <statistic-keyword(s)> <option(s)>;
RUN;
```

where

- *SAS-data-set* is the name of the data set to be used
- *statistic-keyword(s)* specify the statistics to compute
- *option(s)* control the content, analysis, and appearance of output.

In its simplest form, PROC MEANS prints the *n*-count (number of nonmissing values), the mean, the standard deviation, and the minimum and maximum values of every numeric variable in a data set.

```
proc means data=perm.survey;
run;
```

The MEANS Procedure

Variable	N	Mean	Std Dev	Minimum	Maximum
Item1	4	3.7500000	1.2533057	2.0000000	5.0000000
Item2	4	3.0000000	1.6329932	1.0000000	5.0000000
Item3	4	4.2500000	0.5000000	4.0000000	5.0000000
ttem4	4	3.5000000	1.2909944	2.0000000	5.0000000
Item5	4	3.0000000	1.6329932	1.0000000	5.0000000
Item6	4	3.7500000	1.2533057	2.0000000	5.0000000
ttem7	4	3.0000000	1.8257419	1.0000000	5.0000000
Item8	4	2.7500000	1.5000000	1.0000000	4.0000000
Item9	4	3.0000000	1.4142136	2.0000000	5.0000000
Item10	4	3.2500000	1.2533057	2.0000000	5.0000000
Item11	4	3.0000000	1.8257419	1.0000000	5.0000000
Item12	4	2.7500000	0.5000000	2.0000000	3.0000000
Item13	4	2.7500000	1.5000000	1.0000000	4.0000000
Item14	4	3.0000000	1.4142136	2.0000000	5.0000000
Item15	4	3.0000000	1.6329932	1.0000000	5.0000000
Item16	4	2.5000000	1.9148542	1.0000000	5.0000000
Item17	4	3.0000000	1.1547005	2.0000000	4.0000000
Item18	4	3.2500000	1.2533057	2.0000000	5.0000000

Figure 8.3: Default PROC MEANS Output

Reprinted for OET7P/856094, Accenture SAS Institute, SAS Institute Inc (c) 2011, Copying Prohibited

Selecting Statistics

The default statistics that the MEANS procedure produces (*n*-count, mean, standard deviation, minimum, and maximum) are not always the ones that you need. You might prefer to limit output to the mean of the values. Or you might need to compute a different statistic, such as the median or range of the values.

To specify statistics, include statistic keywords as options in the PROC MEANS statement. When you specify a statistic in the PROC MEANS statement, default statistics are not produced. For example, to see the median and range of Perm.Survey numeric values, add the MEDIAN and RANGE keywords as options.

```
proc means data=perm.survey median range;
run;
```

MEANS Procedure Output Displaying Median and Range

The MEANS Procedure

Variable	Median	Range
Item1	4.0000000	3.0000000
Item2	3.0000000	4.0000000
Item3	4.0000000	1.0000000
Item4	3.5000000	3.0000000
Item5	3.0000000	4.0000000
Item6	4.0000000	3.0000000
Item7	3.0000000	4.0000000
Item8	3.0000000	3.0000000
Item9	2.5000000	3.0000000
Item10	3.0000000	3.0000000
Item11	3.0000000	4.0000000
Item12	3.0000000	1.0000000
Item13	3.0000000	3.0000000
Item14	2.5000000	3.0000000
Item15	3.0000000	4.0000000
Item16	2.0000000	4.0000000
Item17	3.0000000	2.0000000
Item18	3.0000000	3.0000000

Figure 8.4: MEANS Procedure Output Displaying Median and Range

The following keywords can be used with PROC MEANS to compute statistics:

Table 8.1: Descriptive Statistics

Keyword	Description
CLM	Two-sided confidence limit for the mean
CSS	Corrected sum of squares
CV	Coefficient of variation
KURTOSIS	Kurtosis
LCLM	One-sided confidence limit below the mean
MAX	Maximum value
MEAN	Average
MODE	Value that occurs most frequently (new in SAS 9.2)

Reprinted for OET7P/856094, Accenture SAS Institute, SAS Institute Inc (c) 2011, Copying Prohibited

MIN	Minimum value
N	Number of observations with nonmissing values
NMISS	Number of observations with missing values
RANGE	Range
SKEWNESS	Skewness
STDDEV / STD	Standard deviation
STDERR	Standard error of the mean
SUM	Sum
SUMWGT	Sum of the Weight variable values
UCLM	One-sided confidence limit above the mean
USS	Uncorrected sum of squares
VAR	Variance

Table 8.2: Quantile Statistics

Keyword	Description
MEDIAN / P50	Median or 50th percentile
P1	1st percentile
P5	5th percentile
P10	10th percentile
Q1 / P25	Lower quartile or 25th percentile
Q3 / P75	Upper quartile or 75th percentile
P90	90th percentile
P95	95th percentile
P99	99th percentile
QRANGE	Difference between upper and lower quartiles: Q3-Q1

Table 8.3: Hypothesis Testing

Keyword	Description
PROBT	Probability of a greater absolute value for the t value
T	Student's t for testing the hypothesis that the population mean is 0

Limiting Decimal Places

By default, PROC MEANS output automatically uses the BESTw. format to display numeric values in the report.

The BEST *w*. format is the default format that SAS uses for writing numeric values. When there is no format specification, SAS chooses the format that provides the most information about the value according to the available field width. At times, this can result in unnecessary decimal places, making your output hard to read.

```
proc means data=clinic.diabetes min max;
run;
```

The MEANS Procedure		
Variable	**Minimum**	**Maximum**
Age	15.0000000	63.0000000
Height	61.0000000	75.0000000
Weight	102.0000000	240.0000000
Pulse	65.0000000	100.0000000
FastGluc	152.0000000	568.0000000

Reprinted for OET7P/856094, Accenture SAS Institute, SAS Institute Inc (c) 2011, Copying Prohibited

PostGluc	206.0000000	625.0000000

Figure 8.5: Variables Fromatted with BESTw. Format

To limit decimal places, use the MAXDEC= option in the PROC MEANS statement, and set it equal to the length that you prefer.

General form, PROC MEANS statement with MAXDEC= option:

```
PROC MEANS <DATA=SAS-data-set>
                  <statistic-keyword(s)> MAXDEC=n;
```

where *n* specifies the maximum number of decimal places.

```
proc means data=clinic.diabetes min max maxdec=0;
run;
```

The MEANS Procedure		
Variable	**Minimum**	**Maximum**
Age	15	63
Height	61	75
Weight	102	240
Pulse	65	100
FastGluc	152	568
PostGluic	206	625

Figure 8.6: Variables Fromatted Using the MAXDEC= Option

Specifying Variables in PROC MEANS

By default, the MEANS procedure generates statistics for every numeric variable in a data set. But you'll typically want to focus on just a few variables, particularly if the data set is large. It also makes sense to exclude certain types of variables. The values of ID, for example, are unlikely to yield useful statistics.

To specify the variables that PROC MEANS analyzes, add a VAR statement and list the variable names.

General form, VAR statement:

VAR *variable(s)*;

where *variable(s)* lists numeric variables for which to calculate statistics.

```
proc means data=clinic.diabetes min max maxdec=0;
    var age height weight;
run;
```

The MEANS Procedure		
Variable	**Minimum**	**Maximum**
Age	15	63
Height	61	75
Weight	102	240

Figure 8.7: Output with Selected Variables Age, Height, and Weight

Reprinted for OET7P/856094, Accenture SAS Institute, SAS Institute Inc (c) 2011, Copying Prohibited

In addition to listing variables separately, you can use a numbered range of variables.

```
proc means data=perm.survey mean stderr maxdec=2;
     var item1-item5;
run;
```

The MEANS Procedure		
Variable	**Mean**	**Std Error**
Item1	3.75	0.63
Item2	3.00	0.82
Item3	4.25	0.25
Item4	3.50	0.65
Item5	3.00	0.82

Figure 8.8: Output with a Range of Variables Selected

Group Processing Using the CLASS Statement

You will often want statistics for groups of observations, instead of over the entire data set. For example, census numbers are more useful when grouped by region than when viewed as a national total. To produce separate analyses of grouped observations, add a CLASS statement to the MEANS procedure.

General form, CLASS statement:

CLASS *variable(s)***;**

where *variable(s)* specifies category variables for group processing.

CLASS variables are used to categorize data. CLASS variables can be either character or numeric, but they should contain a limited number of discrete values that represent meaningful groupings. If a CLASS statement is used, then the N Obs statistic is calculated. The N Obs statistic is based on the CLASS variables, as shown in the output below.

The output of the program shown below is categorized by values of the variables Survive and Sex. The order of the variables in the CLASS statement determines their order in the output table.

```
proc means data=clinic.heart maxdec=1;
  var arterial heart cardiac urinary;
  class survive sex;
run;
```

The MEANS Procedure								
Survive	**Sex**	**N Obs**	**Variable**	**N**	**Mean**	**Std Dev**	**Minimum**	**Maximum**
DIED	1	4	Arterial	4	92.5	10.5	83.0	103.0
			Heart	4	111.0	53.4	54.0	183.0
			Cardiac	4	176.8	75.2	95.0	260.0
			Urinary	4	98.0	186.1	0.0	377.0
	2	6	Arterial	6	94.2	27.3	72.0	145.0
			Heart	6	103.7	16.7	81.0	130.0
			Cardiac	6	318.3	102.6	156.0	424.0
			Urinary	6	100.3	155.7	0.0	405.0
SURV	1	5	Arterial	5	77 2	12 2	61 0	88 0
			Heart	5	109.0	32.0	77.0	149.0
			Cardiac	5	298.0	139.8	66.0	410.0

Reprinted for OET7P/856094, Accenture SAS Institute, SAS Institute Inc (c) 2011, Copying Prohibited

			Urinary	5	100.8	60.2	44.0	200.0
	2	5	Arterial	5	78.8	6.8	72.0	87.0
			Heart	5	100.0	13.4	84.0	111.0
			Cardiac	5	330.2	87.0	256.0	471.0
			Urinary	5	111.2	152.4	12.0	377.0

Figure 8.9: Output Categorized by Values of the Variables Survive and Sex

Group Processing Using the BY Statement

Like the CLASS statement, the BY statement specifies variables to use for categorizing observations.

General form, BY statement:

BY *variable(s);*

where *variable(s)* specifies category variables for group processing.

But BY and CLASS differ in two key ways:

1. Unlike CLASS processing, BY processing requires that your data already be sorted or indexed in the order of the BY variables. Unless data set observations are already sorted, you will need to run the SORT procedure before using PROC MEANS with any BY group.

   Caution Be careful when sorting data sets to enable group processing. If you don't specify an output data set by using the OUT= option, PROC SORT will overwrite your initial data set with the newly sorted observations.

2. BY group results have a layout that is different from the layout of CLASS group results. Note that the BY statement in the program below creates four small tables; a CLASS statement would produce a single large table.

```
proc sort data=clinic.heart out=work.heartsort;
     by survive sex;
run;
proc means data=work.heartsort maxdec=1;
    var arterial heart cardiac urinary;
    by survive sex;
run;
```

The MEANS Procedure					
Survive=DIED Sex=1					
Variable	**N**	**Mean**	**Std Dev**	**Minimum**	**Maximum**
Arterial	4	92.5	10.5	83.0	103.0
Heart	4	111.0	53.4	54.0	183.0
Cardiac	4	176.8	75.2	95.0	260.0
Urinary	4	98.0	186.1	0.0	377.0
Survive=DIED Sex=2					
Variable	**N**	**Mean**	**Std Dev**	**Minimum**	**Maximum**
Arterial	6	94.2	27.3	72.0	145.0
Heart	6	103.7	16.7	81 0	130.0
Cardiac	6	318.3	102.6	156.0	424.0
Urinary	6	100.3	155.7	0.0	405.0
Survive=SURV Sex=1					
Variable	**N**	**Mean**	**Std Dev**	**Minimum**	**Maximum**

Reprinted for OET7P/856094, Accenture SAS Institute, SAS Institute Inc (c) 2011, Copying Prohibited

Arterial	5	77.2	12.2	61.0	88.0
Heart	5	109.0	32.0	77.0	149.0
Cardiac	5	298.0	139.8	66.0	410.0
Urinary	5	100.8	60.2	44.0	200.0
Survive=SURV Sex=2					
Variable	**N**	**Mean**	**Std Dev**	**Minimum**	**Maximum**
Arterial	5	78.8	6.8	72.0	87.0
Heart	5	100.0	13.4	84.0	111.0
Cardiac	5	330.2	87.0	256.0	471.0
Urinary	5	111.2	152.4	12.0	377.0

Figure 8.10: BY Groups Created by PROC MEANS

Additional Note Because it doesn't require a sorting step, the CLASS statement is easier to use than the BY statement. However, BY group processing can be more efficient when your categories might contain many levels.

Creating a Summarized Data Set Using PROC MEANS

Overview

You might want to create an output SAS data set. You can do this by using the OUTPUT statement in PROC MEANS.

General form, OUTPUT statement:

OUTPUT OUT=*SAS-data-set statistic=variable(s)***;**

where

- **OUT=** specifies the name of the output data set
- *statistic*= specifies the summary statistic written out
- *variable(s)* specifies the names of the variables to create. These variables represent the statistics for the analysis variables that are listed in the VAR statement.

When you use the OUTPUT statement, the summary statistics N, MEAN, STD, MIN, and MAX are produced for all of the numeric variables or for *all* of the variables that are listed in a VAR statement *by default*. To specify which statistics to produce, use the OUT= option in the OUTPUT statement.

Specifying Statistics Using PROC MEANS

You can specify which statistics to produce in the output data set by using the OUT= option in the OUTPUT statement.. To do so, you must specify the statistic and then list all of the variables. The variables must be listed in the same order as in the VAR statement. You can specify more than one statistic in the OUTPUT statement.

The following program creates a typical PROC MEANS report and also creates a summarized output data set.

```
proc means data=clinic.diabetes;
   var age height weight;
   class sex;
   output out=work.sum_gender
      mean=AvgAge AvgHeight AvgWeight
      min=MinAge MinHeight MinWeight;
run;
```

The MEANS Procedure							
Sex	**N Obs**	**Variable**	**N**	**Mean**	**Std Dev**	**Minimum**	**Maximum**

Reprinted for OET7P/856094, Accenture SAS Institute, SAS Institute Inc (c) 2011, Copying Prohibited

F	11	Age	11	48.9090909	13.307550S	16.0000000	63.0000000
		Height	11	63.9090909	2.1191765	61.0000000	68.0000000
		Weight	11	150.4545455	18.4464828	102.0000000	168.0000000
M	9	Age	9	44.0000000	12.3895117	15.0000000	54.0000000
		Height	9	70.6665667	2.6457513	66.0000000	75.0000000
		Weight	9	204.2222222	30.2893454	140.0000000	240.0000000

Figure 8.11: Report Created by PROC MEANS

To see the contents of the output data set, submit the following PROC PRINT step.

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

Obs	Sex	_TYPE_	_FREQ_	AvgAge	AvgHeight	AvgWeight	MinAge	MinHeight	MinWeight
1		0	20	46.7000	66.9500	174.650	15	61	102
2	F	1	11	48.9091	63.9091	150.455	16	61	102
3	M	1	9	44.0000	70.6667	204.222	15	66	140

Figure 8.12: Data Set Created by PROC PRINT

Additional Note You can use the NOPRINT option in the PROC MEANS statement to suppress the default report. For example, the following program creates only the output data set:

```
proc means data=clinic.diabetes noprint;
   var age height weight;
   class sex;
   output out=work.sum_gender
      mean=AvgAge AvgHeight AvgWeight;
run;
```

Additional Note In addition to the variables that you specify, the procedure adds the _TYPE_ and _FREQ_ variables to the output data set. When no statistic keywords are specified, PROC MEANS also adds the variable _STAT_. For more information about these variables, see the SAS documentation for the MEANS procedure.

Creating a Summarized Data Set Using PROC SUMMARY

You can also create a summarized output data set with PROC SUMMARY. When you use PROC SUMMARY, you use the same code to produce the output data set that you would use with PROC MEANS. The difference between the two procedures is that PROC MEANS produces a report by default (remember that you can use the NOPRINT option to suppress the default report). By contrast, to produce a report in PROC SUMMARY, you must include a PRINT option in the PROC SUMMARY statement.

Example

The following example creates an output data set but does not create a report:

```
proc summary data=clinic.diabetes;
   var age height weight;
   class sex;
   output out=work.sum_gender
      mean=AvgAge AvgHeight AvgWeight;
run;
```

If you placed a PRINT option in the PROC SUMMARY statement above, this program would produce the same report as if you replaced the word SUMMARY with MEANS.

```
proc summary data=clinic.diabetes print;
   var age height weight;
   class sex;
   output out=work.sum_gender
      mean=AvgAge AvgHeight AvgWeight;
run;
```


Reprinted for OET7P/856094, Accenture SAS Institute, SAS Institute Inc (c) 2011, Copying Prohibited

The SUMMARY Procedure							
Sex	**N Obs**	**Variable**	**N**	**Mean**	**Std Dev**	**Minimum**	**Maximum**
F	11	Age	11	48.9090909	13.3075508	16.0000000	63.0000000
		Height	11	63.9090909	2.1191765	61.0000000	68.0000000
		Weight	11	150.4545455	18.4464828	102.0000000	168.0000000
M	9	Age	9	44.0000000	12.3895117	15.0000000	54.0000000
		Height	9	70.6666667	2.6457513	66.0000000	75.0000000
		Weight	9	204.2222222	30.2893454	140.0000000	240.0000000

Figure 8.13: Output Created by the SUMMARY Procedure with the PRINT Option Specified

Producing Frequency Tables Using PROC FREQ

The FREQ procedure is a descriptive procedure as well as a statistical procedure. It produces one-way and *n*-way frequency tables, and it concisely describes your data by reporting the distribution of variable values. You can use the FREQ procedure to create crosstabulation tables that summarize data for two or more categorical variables by showing the number of observations for each combination of variable values.

Procedure Syntax

The FREQ procedure can include many statements and options for controlling frequency output. For simplicity, let's consider the procedure in its basic form.

General form, basic FREQ procedure:

PROC FREQ <DATA=*SAS-data-set* >**;**

RUN;

where *SAS-data-set* is the name of the data set to be used.

By default, PROC FREQ creates a one-way table with the frequency, percent, cumulative frequency, and cumulative percent of every value of all variables in a data set. This can produce excessive or inappropriate output. It is recommended that you always use a TABLES statement with PROC FREQ.

Variable	Frequency	Percent	Cumulative Frequency	Cumulative Percent
Value	Number of observations with the value.	Frequency of the value divided by the total number of observations.	Sum of the frequency counts of the value and all other values listed above it in the table.	Cumulative frequency of the value divided by the total number of observations.

Figure 8.14: Default PROC FREQ Output

For example, the following FREQ procedure creates a frequency table for each variable in the data set Finance.Loans. All the unique values are shown for the variables Account and Amount.

```
proc freq data=finance.loans;
run;
```

The SAS System				
The FREQ Procedure				
Account	**Frequency**	**Percent**	**Cumulative Frequency**	**Cumulative Percent**
101-1092	1	11.11	1	11.11
101-1289	1	11.11	2	22.22

Reprinted for OET7P/856094, Accenture SAS Institute, SAS Institute Inc (c) 2011, Copying Prohibited

101-1731	1	11.11	3	33.33
101-3144	1	11.11	4	44.44
103-1135	1	11.11	5	55.56
103-1994	1	11.11	6	66.67
103-2335	1	11.11	7	77.73
103-3864	1	11.11	8	88.89
103-3891	1	11.11	9	100.00
Amount	**Frequency**	**Percent**	**Cumulative Frequency**	**Cumulative Percent**
$3,500	1	11.11	1	11.11
$5,000	1	11.11	2	22.22
$8,700	1	11.11	3	33.33
$10,000	1	11.11	4	44.44
$18,500	1	11.11	5	55.56
$22,000	1	11.11	6	66.67
$30,000	1	11-11	7	77.70
$87,500	1	11.11	8	00.89
$114,000	1	11.11	9	100.00

Figure 8.15: Frequency Table for Account and Amount

Specifying Variables in PROC FREQ

Overview

By default, the FREQ procedure creates frequency tables for every variable in your data set. But this isn't always what you want. A variable that has continuous numeric values—such as DateTime—can result in a lengthy and meaningless table. Likewise, a variable that has a unique value for each observation—such as FullName—is unsuitable for PROC FREQ processing. Frequency distributions work best with variables whose values can be described as categorical, and whose values are best summarized by counts rather than by averages.

To specify the variables to be processed by the FREQ procedure, include a TABLES statement.

General form, TABLES statement:

TABLES *variable(s)***;**

where *variable(s)* lists the variables to include.

Example

The order in which the variables appear in the TABLES statement determines the order in which they are listed in the PROC FREQ report.

Consider the SAS data set Finance.Loans. The variables Rate and Months are best described as categorical variables, so they are the best choices for frequency tables.

Account	Amount	Rate	Months	Payment
101-1092	$22,000	10.00%	60	$467.43
101-1731	$114,000	9.50%	360	$958.57
101-1289	$10,000	10.50%	36	$325.02

Reprinted for OET7P/856094, Accenture SAS Institute, SAS Institute Inc (c) 2011, Copying Prohibited

101-3144	$3,500	10.50%	12	$308.52
103-1135	$8,700	10.50%	24	$403.47
103-1994	$18,500	10.00%	60	$393.07
103-2335	$5,000	10.50%	48	$128.02
103-3864	$87,500	9.50%	360	$735.75
103-3891	$30,000	9.75%	360	$257.75

Figure 8.16: PROC PRINT Report of the Data Set Finance.Loans

```
proc freq data=finance.loans;
  tables rate months;
run;
```

The FREQ Procedure				
Rate	**Frequency**	**Percent**	**Cumulative Frequency**	**Cumulative Percent**
9.50%	2	22.22	2	22.22
9.75%	1	11.11	3	33.33
10.00%	2	22.22	5	55.56
10.50%	4	44.44	9	100.00
Months	**Frequency**	**Percent**	**Cumulative Frequency**	**Cumulative Percent**
12	1	11.11	1	11.11
24	1	11.11	2	22.22
36	1	11.11	3	33.33
48	1	11.11	4	44.44
60	2	22.22	6	66.67
360	3	33.33	9	100.00

Figure 8.17: Frequency Tables for Rate and Months

In addition to listing variables separately, you can use a numbered range of variables.

```
proc freq data=perm.survey;
   tables item1-item3;
run;
```

The FREQ Procedure				
Item1	**Frequency**	**Percent**	**Cumulative Frequency**	**Cumulative Percent**
2	1	25.00	1	25.00
4	2	50.00	3	75.00
5	1	25.00	4	100.00
Item2	**Frequency**	**Percent**	**Cumulative Frequency**	**Cumulative Percent**
1	1	25.00	1	25.00
3	2	50.00	3	75.00
5	1	25.00	4	100.00
Item3	**Frequency**	**Percent**	**Cumulative Frequency**	**Cumulative Percent**
4	3	75.00	3	75.00
5	1	25.00	4	100.00

Figure 8.18: Frequency Tables for Item1-Item3

Adding the NOCUM option to your TABLES statement suppresses the display of cumulative frequencies and cumulative

Reprinted for OET7P/856094, Accenture SAS Institute, SAS Institute Inc (c) 2011, Copying Prohibited

percentages in one-way frequency tables and in list output. The syntax for the NOCUM option is shown below.

```
TABLES variable(s) / NOCUM;
```

Creating Two-Way Tables

So far, you have used the FREQ procedure to create one-way frequency tables. However, it is often helpful to crosstabulate frequencies of two or more variables. For example, census data is typically crosstabulated with a variable that represents geographical regions.

The simplest crosstabulation is a two-way table. To create a two-way table, join two variables with an asterisk (*) in the TABLES statement of a PROC FREQ step.

General form, TABLES statement for crosstabulation:

TABLES *variable-1 *variable-2 <* … variable-n>***;**

where (for two-way tables)

- *variable-1* specifies table rows
- *variable-2* specifies table columns.

When crosstabulations are specified, PROC FREQ produces tables with cells that contain

- cell frequency
- cell percentage of total frequency
- cell percentage of row frequency
- cell percentage of column frequency.

For example, the following program creates the two-way table shown below.

```
proc format;
   value wtfmt low-139='< 140'
               140-180='140-180'
               181-high='> 180';
   value htfmt low-64='< 5''5"'
               65-70='5''5-10"'
               71-high='> 5''10"';
run;
proc freq data=clinic.diabetes;
   tables weight*height;
   format weight wtfmt. height htfmt.;
run;
```


Reprinted for OET7P/856094, Accenture SAS Institute, SAS Institute Inc (c) 2011, Copying Prohibited

The FREQ Procedure

Frequency Percent Row Pct Col Pct	Table of Weight by Height			
	Height			
Weight	< 5'5"	5'5-10"	> 5'10"	Total
< 140	2	0	0	2
	10.00	0.00	0.00	10.00
	100.00	0.00	0.00	
	28.57	0.00	0.00	
140-180	5	5	0	10
	25.00	25.00	0.00	50.00
	50.00	50.00	0.00	
	71.43	62.50	0.00	
> 180	0	3	5	8
	0.00	15.00	25.00	40.00
	0.00	37.50	62.50	
	0.00	37.50	100.00	
Total	7	8	5	20
	35.00	40.00	25.00	100.00

Figure 8.19: Two-Way Table Created by PROC FREQ

Note that the first variable, Weight, forms the table rows, and the second variable, Height, forms the columns; reversing the order of the variables in the TABLES statement would reverse their positions in the table. Note also that the statistics are listed in the legend box.

Creating *N*-Way Tables

Overview

For a frequency analysis of more than two variables, use PROC FREQ to create *n*-way crosstabulations. A series of two-way tables is produced, with a table for each level of the other variables.

For example, suppose you want to add the variable Sex to your crosstabulation of Weight and Height in the data set Clinic.Diabetes. Add Sex to the TABLES statement, joined to the other variables with an asterisk (*).

```
tables sex*weight*height;
```

Determining the Table Layout

The order of the variables is important. In *n*-way tables, the last two variables of the TABLES statement become the two-way rows and columns. Variables that precede the last two variables in the TABLES statement stratify the crosstabulation tables.

```
       levels
       ↓
tables sex*weight*height;
       ↑   ↑
       rows + columns = two-way tables
```

Notice the structure of the output that is produced by the program shown below.

```
proc format;
   value wtfmt low-139='< 140'
               140-180='140-180'
               181-high='> 180';
   value htfmt low-64='< 5''5"'
               65-70='5''5-10"'
               71-high='> 5''10"';
run;
```


Reprinted for OET7P/856094, Accenture SAS Institute, SAS Institute Inc (c) 2011, Copying Prohibited

```
proc freq data=clinic.diabetes;
   tables sex*weight*height;
   format weight wtfmt. height htfmt.;
run;
```

The FREQ Procedure

Frequency Percent Row Pct Col Pct	Table 1 of Weight by Height Controlling for Sex=F			
	Height			
Weight	< 5'5"	5'5-10"	> 5'10"	Total
< 140	2 18.18 100.00 28.57	0 0.00 0.00 0.00	0 0.00 0.00 .	2 18.18
140-180	5 45.45 55.56 71.43	4 36.36 44.44 100.00	0 0.00 0.00 .	9 81.82
> 180	0 0.00 . 0.00	0 0.00 . 0.00	0 0.00 . .	0 0.00
Total	7 63.64	4 36.36	0 0.00	11 100.00

Frequency Percent Row Pct Col Pct	Table 2 of Weight by Height Controlling for Sex=M			
	Height			
Weight	< 5'5"	5'5-10"	> 5'10"	Total
< 140	0 0.00 . .	0 0.00 . 0.00	0 0.00 . 0.00	0 0.00
140-180	0 0.00 0.00 .	1 11.11 100.00 25.00	0 0.00 0.00 0.00	1 11.11
> 180	0 0.00 0.00 .	3 33.33 37.50 75.00	5 55.56 62.50 100.00	8 88.89
Total	0 0.00	4 44.44	5 55.56	9 100.00

Figure 8.20: Creating N-Way Tables

Creating Tables in List Format

Overview

When three or more variables are specified, the multiple levels of *n*-way tables can produce considerable output. Such bulky, often complex crosstabulations are often easier to read as a continuous list. Although this eliminates row and column frequencies and percents, the results are compact and clear.

Additional Note The LIST option is not available when you also specify statistical options.

To generate list output for crosstabulations, add a slash (/) and the LIST option to the TABLES statement in your PROC FREQ step.

TABLES *variable-1 *variable-2 <* … variable-n>* **/ LIST;**

Example

Adding the LIST option to our Clinic.Diabetes program puts its frequencies in a simple, short table.

```
proc format;
   value wtfmt low-139='< 140'
               140-180='140-180'
               181-high='> 180';
value htfmt low-64='< 5''5"'
               65-70='5''5-10"'
               71-high='> 5''10"';
run;
proc freq data=clinic.diabetes;
   tables sex*weight*height / list;
   format weight wtfmt. height htfmt.;
run;
```


Reprinted for OET7P/856094, Accenture SAS Institute, SAS Institute Inc (c) 2011, Copying Prohibited

The FREQ Procedure						
Sex	Weight	Height	Frequency	Percent	Cumulative Frequency	Cumulative Percent
F	<110	<5'5"	2	10.00	2	10.00
F	140-180	< 5'5M	5	25.00	7	35.00
F	140-180	5'5-10M	4	20.00	11	55.00
M	140-180	5'5-10"	1	5.00	12	60.00
M	>180	5'5-10"	3	15.00	15	75.00
M	>180	>5'10"	5	25.00	20	100.00

Figure 8.21: Table Created by Using the LIST Option

Changing the Table Format

Beginning in SAS®9, adding the CROSSLIST option to your TABLES statement displays crosstabulation tables in ODS column format. This option creates a table that has a table definition that you can customize by using the TEMPLATE procedure.

Notice the structure of the output that is produced by the program shown below.

```
proc format;
    value wtfmt low-139='< 140'
                140-180='140-180'
                181-high='> 180';
    value htfmt low-64='< 5''5"'
                65-'5''5-10"'
                71-high='> 5''10"';
run;
proc freq data=clinic.diabetes;
   tables sex*weight*height / crosslist;
   format weight wtfmt. height htfmt.;
run;
```

The FREQ Procedure					
Table of Weight by Height					
Controlling for Sex=F					
Weight	Height	Frequency	Percent	Row Percent	Column Percent
<140	< 5'5"	2	18.18	100.00	28.57
	5'5-10"	0	0.00	0.00	0.00
	>5'10"	0	0.00	0.00	.
	Total	2	1810	100 00	.
140-180	< 5'5"	5	45.45	55.56	71.43
	5'5-10"	4	36.36	44.44	100.00
	> 5'10"	0	0.00	0.00	.
	Total	9	81.82	100.00	.
>180	< 5'5"	0	0.00	.	0.00
	5'5-10"	0	0.00	.	0.00
	> 5'10"	0	0.00	.	.
	Total	0	0.00	.	
Total	<5'5"	7	63.64		100.00
	5'5-10"	4	36.36		100.00
	> 5'10"	0	0.00		.

Reprinted for OET7P/856094, Accenture SAS Institute, SAS Institute Inc (c) 2011, Copying Prohibited

	Total	11	100.00		

Figure 8.22: Table Created by Using CROSSLIST Option: Sex=F

Table of Weight by Height					
Controlling for Sex=M					
Weight	**Height**	**Frequency**	**Percent**	**Row Perce nt**	**Column Percent**
>140	**< 5'5"**	0	0.00	.	.
	5'5-10"	0	0.00	.	0.00
	>5'10"	0	0.00	.	0.00
	Total	0	0.00	.	
140-180	**< 5'5"**	0	0.00	0.00	.
	5'5-10"	1	11.11	100.00	25.00
	>5'10"	0	0.00	0.00	0.00
	Total	1	11.11	100.00	
>180	**< 5'5"**	0	0.00	0.00	.
	5'5-10"	3	33.33	37.50	75.00
	> 5'10"	5	55.56	62.50	100.00
	Total	8	88.89	100.00	
Total	**< 5'5"**	0	0.00		.
	5'5-10"	4	44.44		100.00
	>5'10"	5	55.56		100.00
	Total	9	100.00		

Figure 8.23: Table Created by Using CROSSLIST Option: Sex=M

Suppressing Table Information

Another way to control the format of crosstabulations is to limit the output of the FREQ procedure to a few specific statistics. Remember that when crosstabulations are run, PROC FREQ produces tables with cells that contain:

- cell frequency
- cell percentage of total frequency
- cell percentage of row frequency
- cell percentage of column frequency.

You can use options to suppress any of these statistics. To control the depth of crosstabulation results, add any combination of the following options to the TABLES statement:

- NOFREQ suppresses cell frequencies
- NOPERCENT suppresses cell percentages
- NOROW suppresses row percentages
- NOCOL suppresses column percentages.

Example

Suppose you want to use only the percentages of Sex and Weight combinations in the data set Clinic.Diabetes. To block frequency counts and row and column percentages, add the NOFREQ, NOROW, and NOCOL options to your program's

Reprinted for OET7P/856094, Accenture SAS Institute, SAS Institute Inc (c) 2011, Copying Prohibited

TABLES statement.

```
proc format;
    value wtfmt low-139='< 140'
                140-180='140-180'
                181-high='> 180';
run;
proc freq data=clinic.diabetes;
   tables sex*weight / nofreq norow nocol;
   format weight wtfmt.;
run;
```

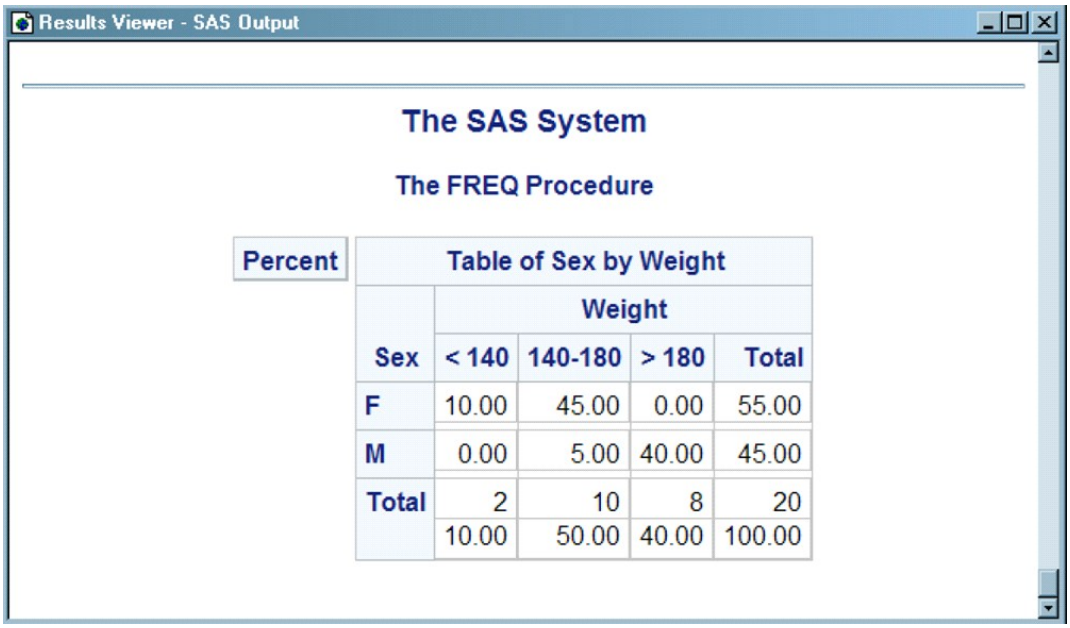

Figure 8.24: Suppressing Table Information

Notice that Percent is the only statistic that remains in the table's legend box.

Chapter Summary

Text Summary

Purpose of PROC MEANS

The MEANS procedure provides an easy way to compute descriptive statistics. Descriptive statistics such as the mean, minimum, and maximum provide useful information about numeric data.

Specifying Statistics

By default, PROC MEANS computes the *n*-count (the number of non-missing values), the mean, the standard deviation, and the minimum and maximum values for variables. To specify statistics, list their keywords in the PROC MEANS statement.

Limiting Decimal Places

Because PROC MEANS uses the BEST. format by default, procedure output can contain unnecessary decimal places. To limit decimal places, use the MAXDEC= option and set it equal to the length that you prefer.

Specifying Variables in PROC MEANS

By default, PROC MEANS computes statistics for all numeric variables. To specify the variables to include in PROC MEANS output, list them in a VAR statement.

Group Processing Using the CLASS Statement

Reprinted for OET7P/856094, Accenture SAS Institute, SAS Institute Inc (c) 2011, Copying Prohibited

Include a CLASS statement, specifying variable names, to group PROC MEANS output by variable values of classes.

Group Processing Using the BY Statement

Include a BY statement, specifying variable names, to group PROC MEANS output by variable values. Your data must be sorted according to those variables. Statistics are not computed for the BY variables.

Creating a Summarized Data Set Using PROC MEANS

You can create an output data set that contains summarized variables by using the OUTPUT statement in PROC MEANS. When you use the OUTPUT statement without specifying the statistic-keyword= option, the summary statistics N, MEAN, STD, MIN, and MAX are produced for all of the numeric variables or for all of the variables that are listed in a VAR statement.

Creating a Summarized Data Set Using PROC SUMMARY

You can also create a summarized output data set by using PROC SUMMARY. The PROC SUMMARY code for producing an output data set is exactly the same as the code for producing an output data set with PROC MEANS. The difference between the two procedures is that PROC MEANS produces a report by default, whereas PROC SUMMARY produces an output data set by default.

The FREQ Procedure

The FREQ Procedure is a descriptive procedure as well as a statistical procedure that produces one-way and n-way frequency tables. It describes your data by reporting the distribution of variable values.

Specifying Variables

By default, the FREQ procedure creates frequency tables for every variable in your data set. To specify the variables to analyze, include them in a TABLES statement.

Creating Two-Way Tables

When a TABLES statement contains two variables joined by an asterisk (*), PROC FREQ produces crosstabulations. The resulting table displays values for

- cell frequency
- cell percentage of total frequency
- cell percentage of row frequency
- cell percentage of column frequency.

Creating N-Way Tables

Crosstabulations can include more than two variables. When three or more variables are joined in a TABLES statement, the result is a series of two-way tables that are grouped by the values of the first variables listed.

Creating Tables in List Format

To reduce the bulk of n-way table output, add a slash (/) and the LIST option to the end of the TABLES statement. PROC FREQ then prints compact, multi-column lists instead of a series of tables. Beginning in SAS®9, you can use the CROSSLIST option to format your tables in ODS column format.

Suppressing Table Information

You can suppress the display of specific statistics by adding one or more options to the TABLES statement:

- NOFREQ suppresses cell frequencies
- NOPERCENT suppresses cell percentages
- NOROW suppresses row percentages

Reprinted for OET7P/856094, Accenture SAS Institute, SAS Institute Inc (c) 2011, Copying Prohibited

- NOCOL suppresses column percentages.

Syntax

```
PROC MEANS <DATA=SAS-data-set>
           <statistic-keyword(s)> <option(s)>;
           <VARvariable(s)>;
           <CLASSvariable(s)>;
           <BYvariable(s)>;
           <OUTPUTout=SAS-data-set statistic=variable(s)>;
RUN;

PROC SUMMARY <DATA=SAS-data-set>
            <statistic-keyword(s)> <option(s)>;
            <VARvariable(s)>;
            <CLASSvariable(s)>;
            <BYvariable(s)>;
            <OUTPUTout=SAS-data-set>;
RUN;
PROC FREQ <DATA=SAS-data-set>;
            TABLESvariable-1 *variable-2 <*...variable-n>
            / <NOFREQ|NOPERCENT|NOROW|NOCOL|CROSSLIST>;
RUN;
```

Sample Programs

```
 proc means data=clinic.heart min max maxdec=1;
    var arterial heart cardiac urinary;
    class survive sex;
run;

proc summary data=clinic.diabetes;
   var age height weight;
   class sex;
   output out=work.sum_gender
      mean=AvgAge AvgHeight AvgWeight;
run;

proc freq data=clinic.heart;
   tables sex*survive*shock / nopercent list;
run;
```

Points to Remember

- In PROC MEANS, use a VAR statement to limit output to relevant variables. Exclude statistics for variables such as dates.
- By default, PROC MEANS prints the full width of each numeric variable. Use the MAXDEC= option to limit decimal places and improve legibility.
- Data must be sorted for BY group processing. You might need to run PROC SORT before using PROC MEANS with a BY statement.
- PROC MEANS and PROC SUMMARY produce the same results; however, the default output is different. PROC MEANS produces a report, whereas PROC SUMMARY produces an output data set.
- If you do not include a TABLES statement, PROC FREQ produces statistics for every variable in the data set. Variables that have continuous numeric values can create a large amount of output. Use a TABLES statement to exclude such variables, or group their values by applying a FORMAT statement.

Chapter Quiz

Select the best answer for each question. After completing the quiz, check your answers using the answer key in the appendix.

1. The default statistics produced by the MEANS procedure are *n*-count, mean, minimum, maximum, and… ?

Reprinted for OET7P/856094, Accenture SAS Institute, SAS Institute Inc (c) 2011, Copying Prohibited

a. median

b. range

c. standard deviation

d. standard error of the mean.

2. Which statement will limit a PROC MEANS analysis to the variables Boarded, Transfer, and Deplane? ?

a. `by boarded transfer deplane;`

b. `class boarded transfer deplane;`

c. `output boarded transfer deplane;`

d. `var boarded transfer deplane;`

3. The data set Survey.Health includes the following variables. Which is a poor candidate for PROC MEANS analysis? ?

a. `IDnum`

b. `Age`

c. `Height`

d. `Weight`

4. Which of the following statements is true regarding BY group processing? ?

a. BY variables must be either indexed or sorted.

b. Summary statistics are computed for BY variables.

c. BY group processing is preferred when you are categorizing data that contains few variables.

d. BY group processing overwrites your data set with the newly grouped observations.

5. Which group processing statement produced the PROC MEANS output shown below? ?

The MEANS Procedure

Survive	Sex	NObs	Variable	N	Mean	Std Dev	Minimum	Maximum
DIED	1	4	Arterial	4	92.5000000	10.4721854	83.0000000	103.0000000
			Heart	4	111.0000000	53.4103610	54.0000000	183.0000000
			Cardiac	4	178.7500000	75.2257713	95.0000000	260.0000000
			Urinary	4	98.0000000	186.1343601	0	377.0000000
	2	6	Arterial	6	94.1066867	27.3160514	72 0000000	145.0000000
			Heart	6	103.6666667	16.6573307	81 0000000	130.0000000
			Cardiac	6	318.3333333	102.6034437	156.0000000	424.0000000
			Urinary	6	100.3333333	155.7134120	0	405.0000000
SURV	1	5	Arterial	5	77.2000000	12.1942609	61.0000000	88.0000000
			Heart	5	109.0000000	31.9687347	77.0000000	149.0000000
			Cardiac	5	298.0000000	139.8499196	66 0000000	410.0000000
			Urinary	5	100 8000000	60.1722527	44 0000000	200.0000000
	2	5	Arterial	5	78.8000000	6.8337398	72.0000000	87.0000000
			Heart	5	100.0000000	13.3790882	84.0000000	111.0000000
			Cardiac	5	330.2000000	86.9839066	256.0000000	471.0000000
			Urinary	5	111.2000000	152.4096454	12.0000000	377.0000000

Figure 8.25: PROC MEANS Output

a. `class sex survive;`

Reprinted for OET7P/856094, Accenture SAS Institute, SAS Institute Inc (c) 2011, Copying Prohibited

b. `class survive sex;`

c. `by sex survive;`

d. `by survive sex;`

6. Which program can be used to create the following output? ?

Sex	NObs	Variable	N	Mean	Std Dev	Minimum	Maximum
F	11	Age	11	48.9090909	13.3075508	16.0000000	63.0000000
		Height	11	63.9090909	2.1191765	61.0000000	68.0000000
		Weight	11	150.4545455	18.4464828	102.0000000	168.0000000
M	9	Age	9	44.0000000	12.3895117	15.0000000	54.0000000
		Height	9	70.6666667	2.6457513	66.0000000	75.0000000
		Weight	9	204.2222222	30.2893454	140.0000000	240.0000000

Figure 8.26: Output

a.
```
proc means data=clinic.diabetes;
   var age height weight;
   class sex;
   output out=work.sum_gender
      mean=AvgAge AvgHeight AvgWeight;
run;
```

b.
```
proc summary data=clinic.diabetes print;
   var age height weight; class sex;
   output out=work.sum_gender
      mean=AvgAge AvgHeight AvgWeight;
run;
```

c.
```
proc means data=clinic.diabetes noprint;
   var age height weight;
   class sex;
   output out=work.sum_gender
      mean=AvgAge AvgHeight AvgWeight;
run;
```

d. Both a and b.

7. By default, PROC FREQ creates a table of frequencies and percentages for which data set variables? ?

a. character variables

b. numeric variables

c. both character and numeric variables

d. none: variables must always be specified

8. Frequency distributions work best with variables that contain ?

a. continuous values.

b. numeric values.

c. categorical values.

d. unique values.

9. Which PROC FREQ step produced this two-way table? ?

Reprinted for OET7P/856094, Accenture SAS Institute, SAS Institute Inc (c) 2011, Copying Prohibited

The FREQ Procedure

Frequency Percent Row Pct Col Pct	Table of Weight by Height			
	Height			
Weight	< 5'5"	5'5-10"	> 5'10"	Total
< 140	2	0	0	2
	10.00	0.00	0.00	10.00
	100.00	0.00	0.00	
	28.57	0.00	0.00	
140-180	5	5	0	10
	25.00	25.00	0.00	50.00
	50.00	50.00	0.00	
	71.43	62.50	0.00	
> 180	0	3	5	8
	0.00	15.00	25.00	40.00
	0.00	37.50	62.50	
	0.00	37.50	100.00	
Total	7	8	5	20
	35.00	40.00	25.00	100.00

Figure 8.27: Two-Way Table

a.
```
proc freq data=clinic.diabetes;
   tables height weight;
   format height htfmt. weight wtfmt.;
run;
```

b.
```
proc freq data=clinic.diabetes;
   tables weight height;
   format weight wtfmt. height htfmt.;
run;
```

c.
```
proc freq data=clinic.diabetes;
   tables height*weight;
   format height htfmt. weight wtfmt.
run;
```

d.
```
proc freq data=clinic.diabetes;
   tables weight*height;
   format weight wtfmt. height htfmt.;
run;
```

10. Which PROC FREQ step produced this table? ?

Reprinted for OET7P/856094, Accenture SAS Institute, SAS Institute Inc (c) 2011, Copying Prohibited

The FREQ Procedure

Percent	Table of Sex by Weight			
	Weight			
Sex	< 140	140-180	> 180	Total
F	10.00	45.00	0.00	55.00
M	0.00	5.00	40.00	45.00
Total	2	10	8	20
	10.00	50.00	40.00	100.00

Figure 8.28: PROC FREQ Table

a.
```
proc freq data=clinic.diabetes;
   tables sex weight / list;
   format weight wtfmt.;
run;
```

b.
```
proc freq data=clinic.diabetes;
   tables sex*weight / nocol;
   format weight wtfmt.;
run;
```

c.
```
proc freq data=clinic.diabetes;
   tables sex weight / norow nocol;
   format weight wtfmt.;
run;
```

d.
```
proc freq data=clinic.diabetes;
   tables sex*weight / nofreq norow nocol;
   format weight wtfmt.;
run;
```

Answers

1. Correct answer: c

   By default, the MEANS procedure produces the n, mean, minimum, maximum, and standard deviation.

2. Correct answer: d

   To specify the variables that PROC MEANS analyzes, add a VAR statement and list the variable names.

3. Correct answer: a

   Unlike Age, Height, or Weight, the values of IDnum are unlikely to yield any useful statistics.

Reprinted for OET7P/856094, Accenture SAS Institute, SAS Institute Inc (c) 2011, Copying Prohibited

4. Correct answer: a

Unlike CLASS processing, BY group processing requires that your data already be indexed or sorted in the order of the BY variables. You might need to run the SORT procedure before using PROC MEANS with a BY group.

5. Correct answer: b

A CLASS statement produces a single large table, whereas BY group processing creates a series of small tables. The order of the variables in the CLASS statement determines their order in the output table.

6. Correct answer: d

You can use either PROC MEANS or PROC SUMMARY to create the table. Adding a PRINT option to the PROC SUMMARY statement produces the same report as if you used PROC MEANS.

7. Correct answer: c

By default, PROC FREQ creates a table for all variables in a data set.

8. Correct answer: c

Both continuous values and unique values can result in lengthy, meaningless tables. Frequency distributions work best with categorical values.

9. Correct answer: d

An asterisk is used to join the variables in a two-way TABLES statement. The first variable forms the table rows. The second variable forms the table columns.

10. Correct answer: d

An asterisk is used to join the variables in crosstabulation tables. The only results shown in this table are cell percentages. The NOFREQ option suppresses cell frequencies, the NOROW option suppresses row percentages, and the NOCOL option suppresses column percentages.

Reprinted for OET7P/856094, Accenture SAS Institute, SAS Institute Inc (c) 2011, Copying Prohibited