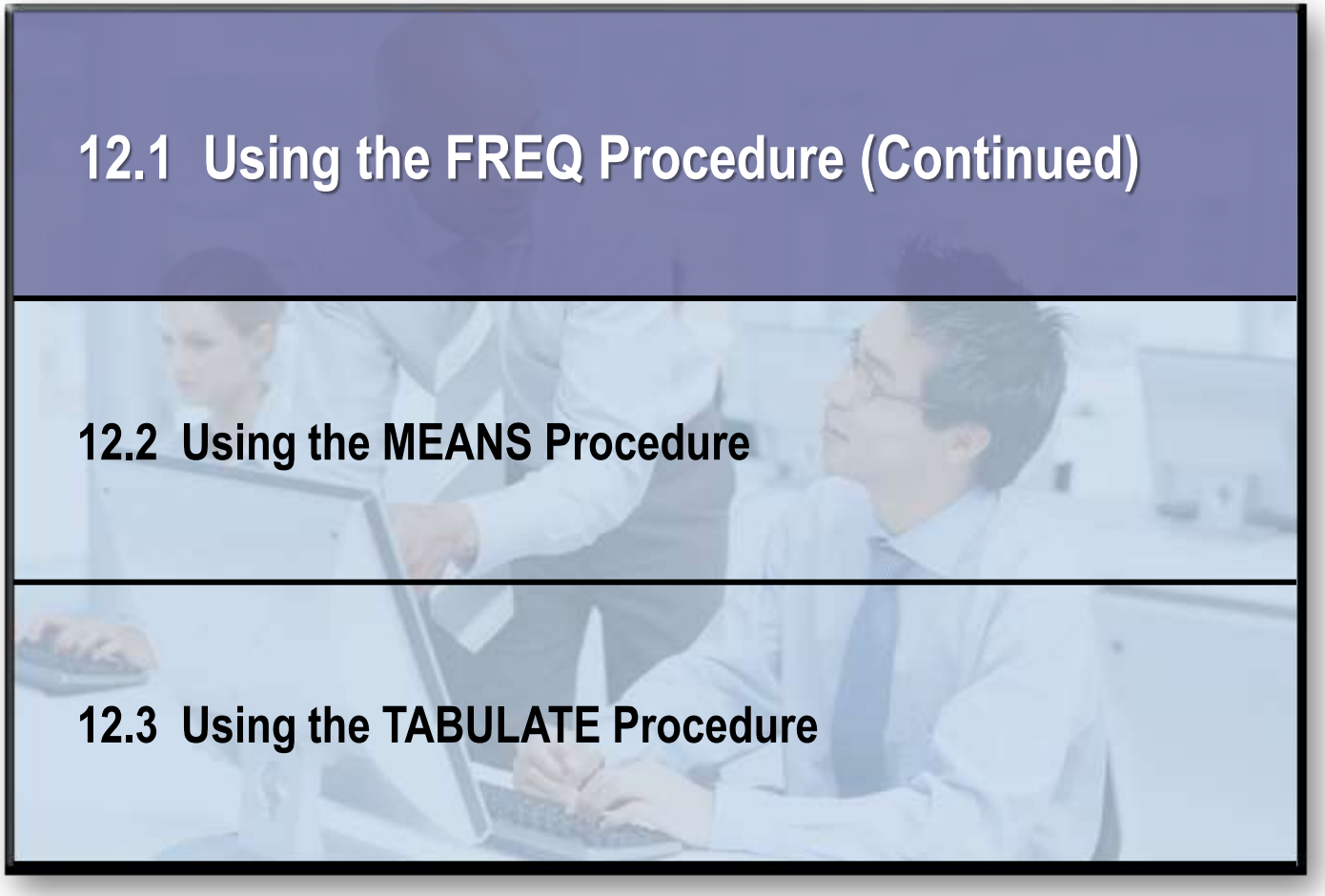
STAT604

Lesson SAS 13

Chapter 12: Producing Summary Reports



Additional TABLES Statement Options

Additional options can be placed in the TABLES statement after a forward slash to control the displayed output.

Option	Description
LIST	displays *n*-way tables in list format.
CROSSLIST	displays *n*-way tables in column format.
~~FORMAT=~~	formats the frequencies in *n*-way tables.

LIST and CROSSLIST Options

Gender	Country	Frequency	Percent	Cumulative Frequency	Cumulative Percent
F	Australia	27	16.36	27	16.36
F	United States	41	24.85	68	41.21
M	Australia	36	21.82	104	63.03
M	United States	61			

```
tables Gender*Country / list;
```

Table of Gender by Country

Gender	Country	Frequency	Percent	Row Percent	Column Percent
F	Australia	27	16.36	39.71	42.86
	United States	41	24.85	60.29	40.20
	Total	68	41.21	100.00	
M	Australia	36	21.82	37.11	57.14
	United States				
	Total				
Total	Australia	63	38.18		100.00
	United States	102	61.82		100.00
	Total	165	100.00		

```
tables Gender*Country / crosslist;
```

p112d01

PROC FREQ Statement Options

Options can also be placed in the PROC FREQ statement.

Option	Description
NLEVELS	displays a table that provides the number of levels for each variable named in the TABLES statement.
PAGE*	displays only one table per page.
COMPRESS*	begins the display of the next one-way frequency table on the same page as the preceding one-way table if there is enough space to begin the table.

*Limited functionality in ODS output.

NLEVELS Option

```
proc freq data=orion.sales nlevels;
   tables Gender Country Employee_ID;
run;
```

Partial PROC FREQ Output

The FREQ Procedure

Number of Variable Levels

Variable	Levels
Gender	2
Country	2
Employee_ID	165

Output Data Sets

PROC FREQ produces output data sets using two different methods.

- The TABLES statement with an OUT= option is used to create a data set with frequencies and percentages.

TABLES *variables* **/ OUT=***SAS-data-set* <*options*>**;**

- The OUTPUT statement with an OUT= option is used to create a data set with specified statistics such as the chi-square statistic.

OUTPUT OUT=*SASdataset* <*options*>**;**

TABLES Statement OUT= Option

The OUT= option in the TABLES statement creates an output data set with the following variables:

- BY variables
- TABLES statement variables
- the automatic variables **COUNT** and **PERCENT**
- other frequency and percentage variables requested with options in the TABLES statement

TABLES *variables* **/ OUT=***SAS-data-set* <*options*>**;**

If more than one table request appears in the TABLES statement, the contents of the data set correspond to the last table request.

TABLES Statement OUT= Option

```
proc freq data=orion.sales noprint;
   tables Gender Country / out=work.freq1;
run;

proc print data=work.freq1;
run;
```

PROC PRINT Output

Obs	Country	COUNT	PERCENT
1	AU	63	38.1818
2	US	102	61.8182

The NOPRINT option suppresses the display of all output.

TABLES Statement OUT= Option

```
proc freq data=orion.sales noprint;
   tables Gender*Country / out=work.freq2;
run;

proc print data=work.freq2;
run;
```

PROC PRINT Output

Obs	Gender	Country	COUNT	PERCENT
1	F	AU	27	16.3636
2	F	US	41	24.8485
3	M	AU	36	21.8182
4	M	US	61	36.9697

p112d02

TABLES Statement OUT= Option

Options can be added to the TABLES statement after the forward slash to control the additional statistics added to the output data set.

Option	Description
OUTCUM	includes the cumulative frequency and cumulative percentage in the output data set for one-way frequency tables.
OUTPCT	includes the percentage of column frequency and row frequency in the output data set for *n*-way frequency tables.

TABLES Statement OUT= Option

```
proc freq data=orion.sales noprint;
   tables Gender Country / out=work.freq3
                           outcum;
run;

proc print data=work.freq3;
run;
```

PROC PRINT Output

Obs	Country	COUNT	PERCENT	CUM_FREQ	CUM_PCT
1	AU	63	38.1818	63	38.182
2	US	102	61.8182	165	100.000

TABLES Statement OUT= Option

```
proc freq data=orion.sales noprint;
   tables Gender*Country / out=work.freq4
                           outpct;
run;

proc print data=work.freq4;
run;
```

PROC PRINT Output

Obs	Gender	Country	COUNT	PERCENT	PCT_ROW	PCT_COL
1	F	AU	27	16.3636	39.7059	42.8571
2	F	US	41	24.8485	60.2941	40.1961
3	M	AU	36	21.8182	37.1134	57.1429
4	M	US	61	36.9697	62.8866	59.8039

p112d02

OUTPUT Statement OUT= Option

The OUT= option in the OUTPUT statement creates an output data set with the following variables:

- BY variables
- the variables requested in the TABLES statement
- variables that contain the specified statistics.

OUTPUT OUT=*SASdataset* <*options*>**;**

If more than one table request appears in the TABLES statement, the contents of the data set corresponds to the last table request.

OUTPUT Statement OUT= Option

In order to specify that the output data set contain a particular statistic, you must have PROC FREQ compute the statistic by using the corresponding option in the TABLES statement.

```
proc freq data=orion.sales;
   tables Country / chisq;
   output out=work.freq5 chisq;
run;

proc print data=work.freq5;
run;
```

CHISQ requests chi-square tests and measures of association based on chi-square.

OUTPUT Statement OUT= Option

PROC FREQ Output

The FREQ Procedure

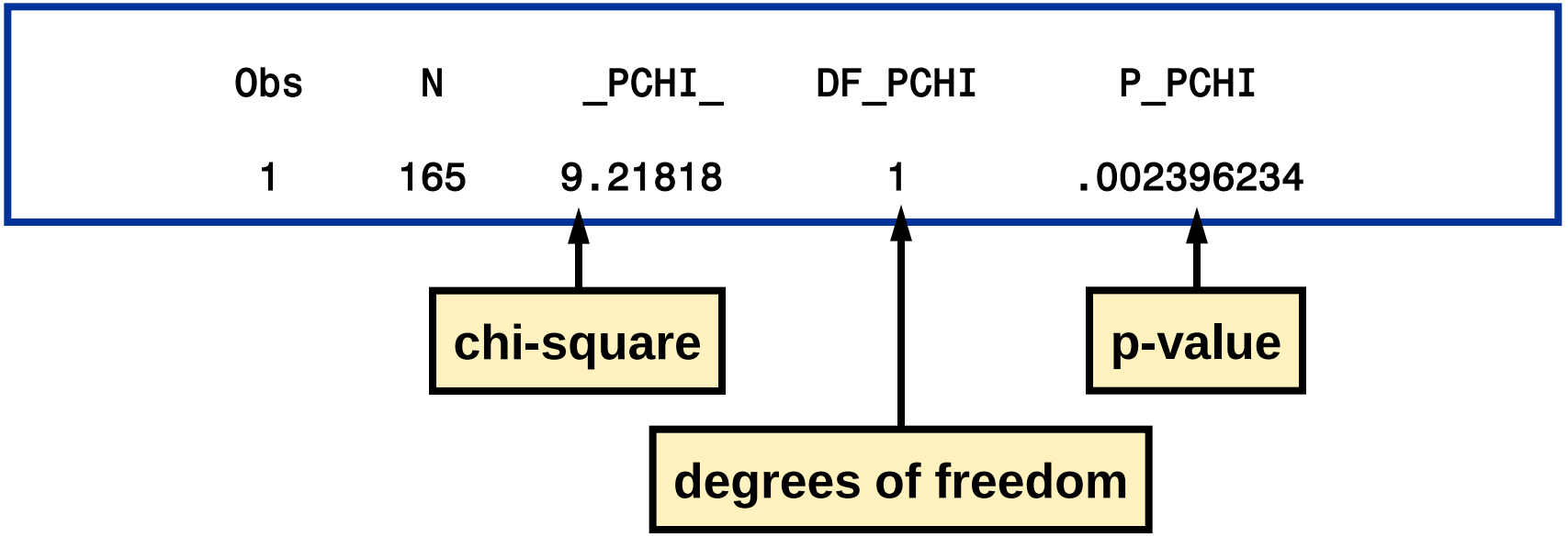
Country	Frequency	Percent	Cumulative Frequency	Cumulative Percent
AU	63	38.18	63	38.18
US	102	61.82	165	100.00

Chi-Square Test for Equal Proportions	
Chi-Square	9.2182
DF	1
Pr > ChiSq	0.0024

Sample Size = 165

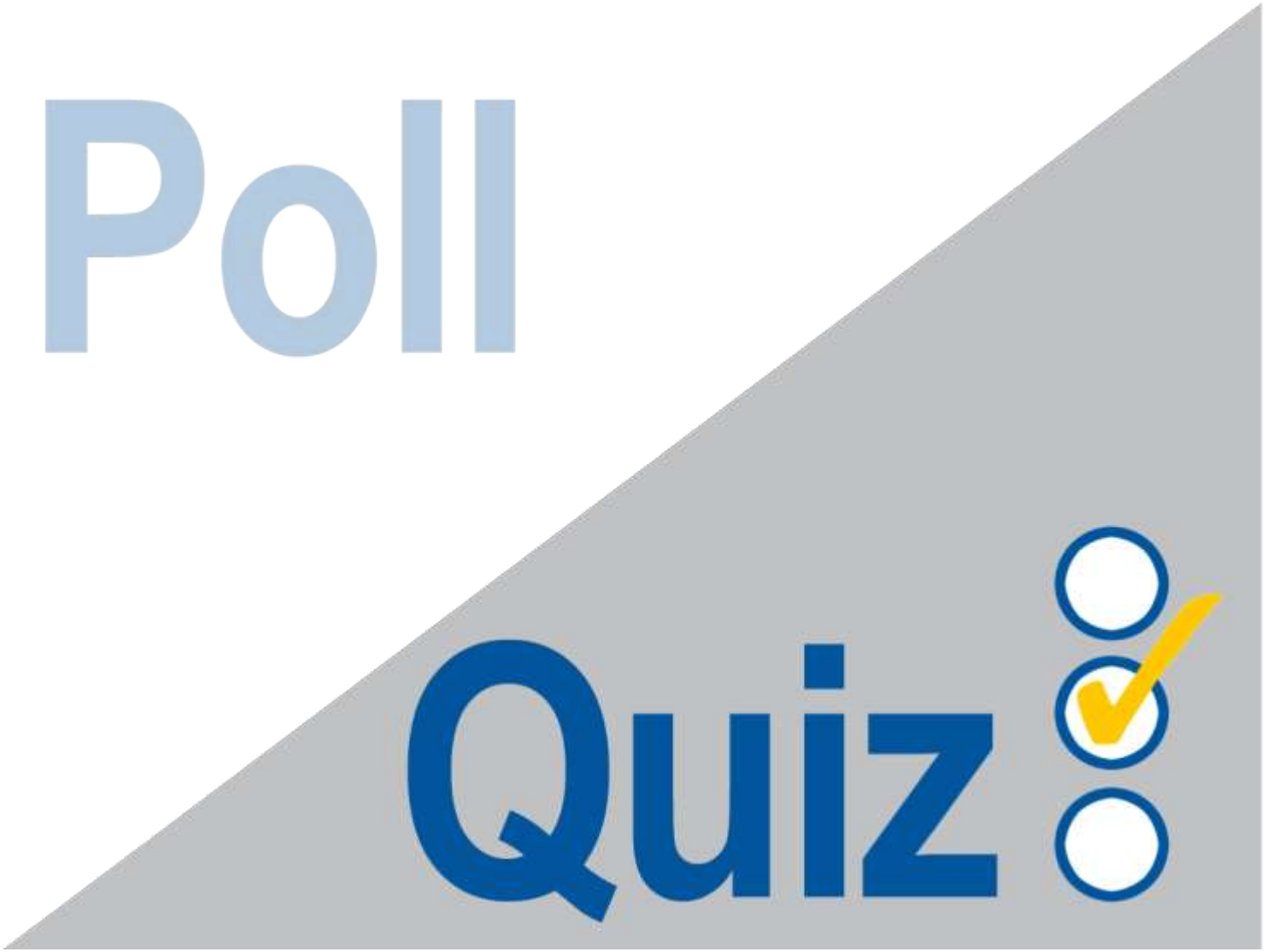
OUTPUT Statement OUT= Option

PROC PRINT Output



When you request a statistic, the OUTPUT data set contains that test statistic plus any associated standard error, confidence limits, *p*-values, and degrees of freedom.

Poll



12.03 Quiz

- Retrieve and submit program **p112a01**.

```
proc freq data=orion.sales;
   tables Gender / chisq out=freq6 outcum;
   output out=freq7 chisq;
run;
proc print data=freq6;
run;
proc print data=freq7;
run;
```

- Review the PROC FREQ output.
- Review the PROC PRINT output from the TABLES statement OUT= option.
- Review the PROC PRINT output from the OUTPUT statement OUT= option.

12.03 Quiz – Correct Answer

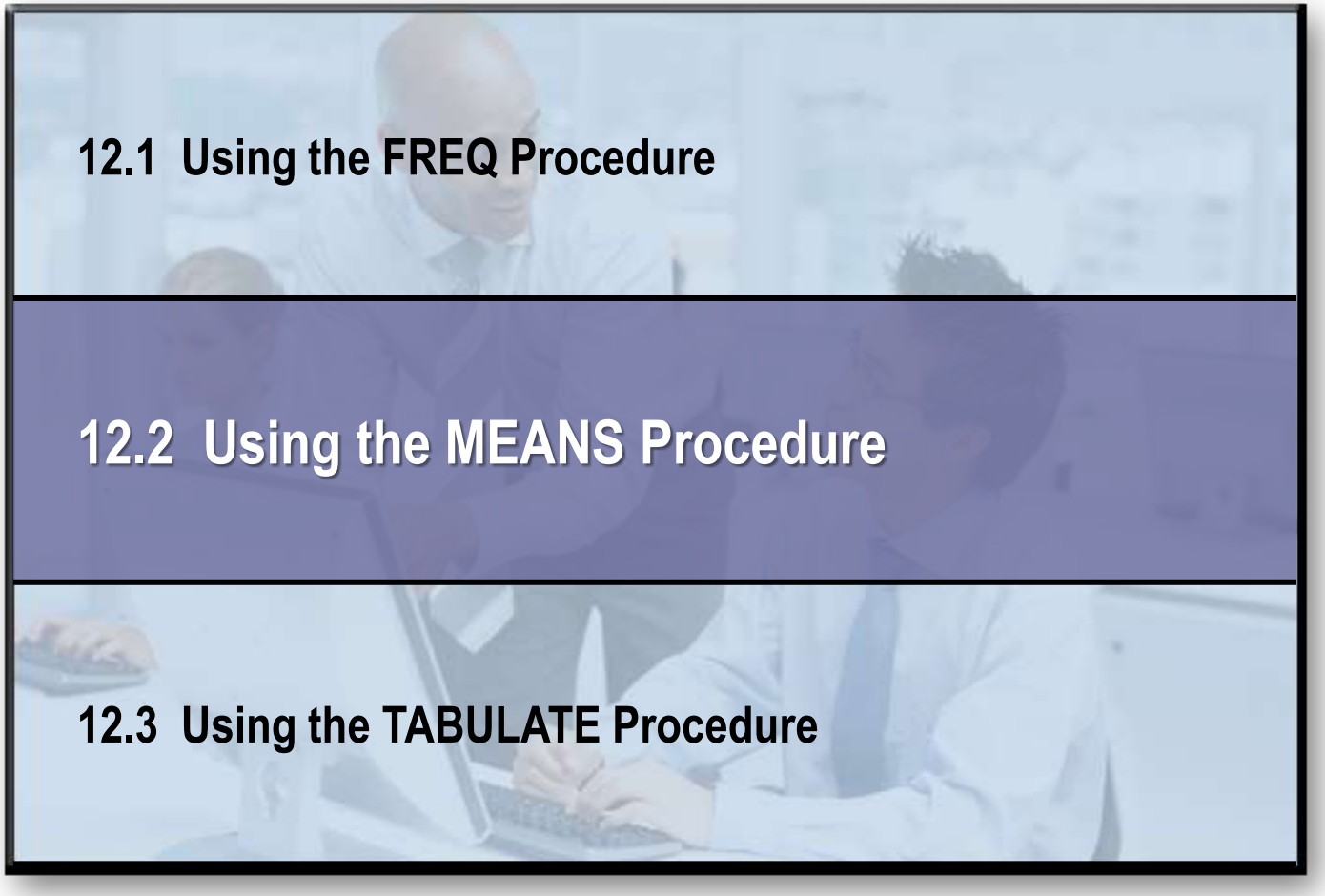
The first part of the PROC FREQ output is in the SAS data set that was created with the TABLES statement.

Obs	Gender	COUNT	PERCENT	CUM_FREQ	CUM_PCT
1	F	68	41.2121	68	41.212
2	M	97	58.7879	165	100.000

The second part of the PROC FREQ output is in the SAS data set that was created with the OUTPUT statement.

Obs	N	_PCHI_	DF_PCHI	P_PCHI
1	165	5.09697	1	0.023968

Chapter 12: Producing Summary Reports



Objectives

- Calculate summary statistics and multilevel summaries with the MEANS procedure.
- Enhance summary tables with options.
- Produce output data sets by using the OUT= option in the OUTPUT statement.
- Compare the SUMMARY procedure to the MEANS procedure.

The MEANS Procedure

The *MEANS procedure* provides data summarization tools to compute descriptive statistics for variables across all observations and within groups of observations.

General form of the MEANS procedure:

```
PROC MEANS DATA=SASdataset <statistic(s)> <option(s)>;
    VAR analysis-variable(s);
    CLASS classification-variable(s);
RUN;
```

The MEANS Procedure

By default, the MEANS procedure reports the number of nonmissing observations, the mean, the standard deviation, the minimum value, and the maximum value of all numeric variables.

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

The MEANS Procedure

Variable	N	Mean	Std Dev	Minimum	Maximum
Employee_ID	165	120713.90	450.0866939	120102.00	121145.00
Salary	165	31160.12	20082.67	22710.00	243190.00
Birth_Date	165	3622.58	5456.29	-5842.00	10490.00
Hire_Date	165	12054.28	4619.94	5114.00	17167.00

p112d05

The VAR Statement

The *VAR statement* identifies the analysis variables and their order in the results.

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

The MEANS Procedure

Analysis Variable : Salary

N	Mean	Std Dev	Minimum	Maximum
165	31160.12	20082.67	22710.00	243190.00

p112d05

The 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;
```

The MEANS Procedure

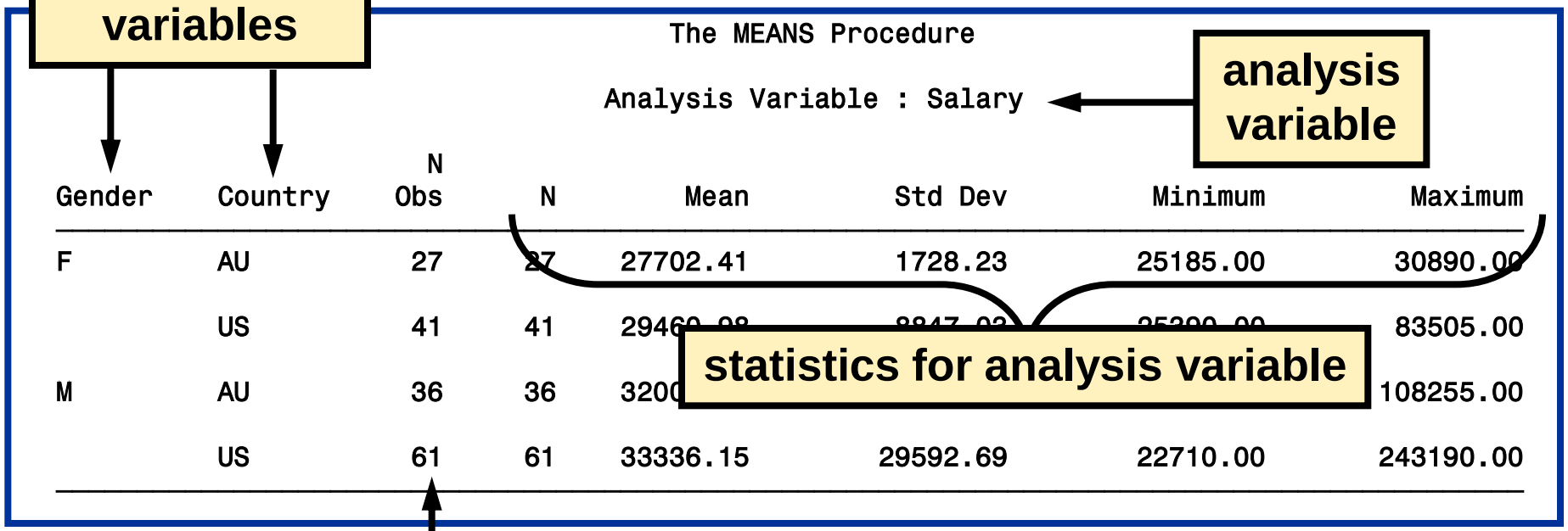
Analysis Variable : Salary

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

p112d05

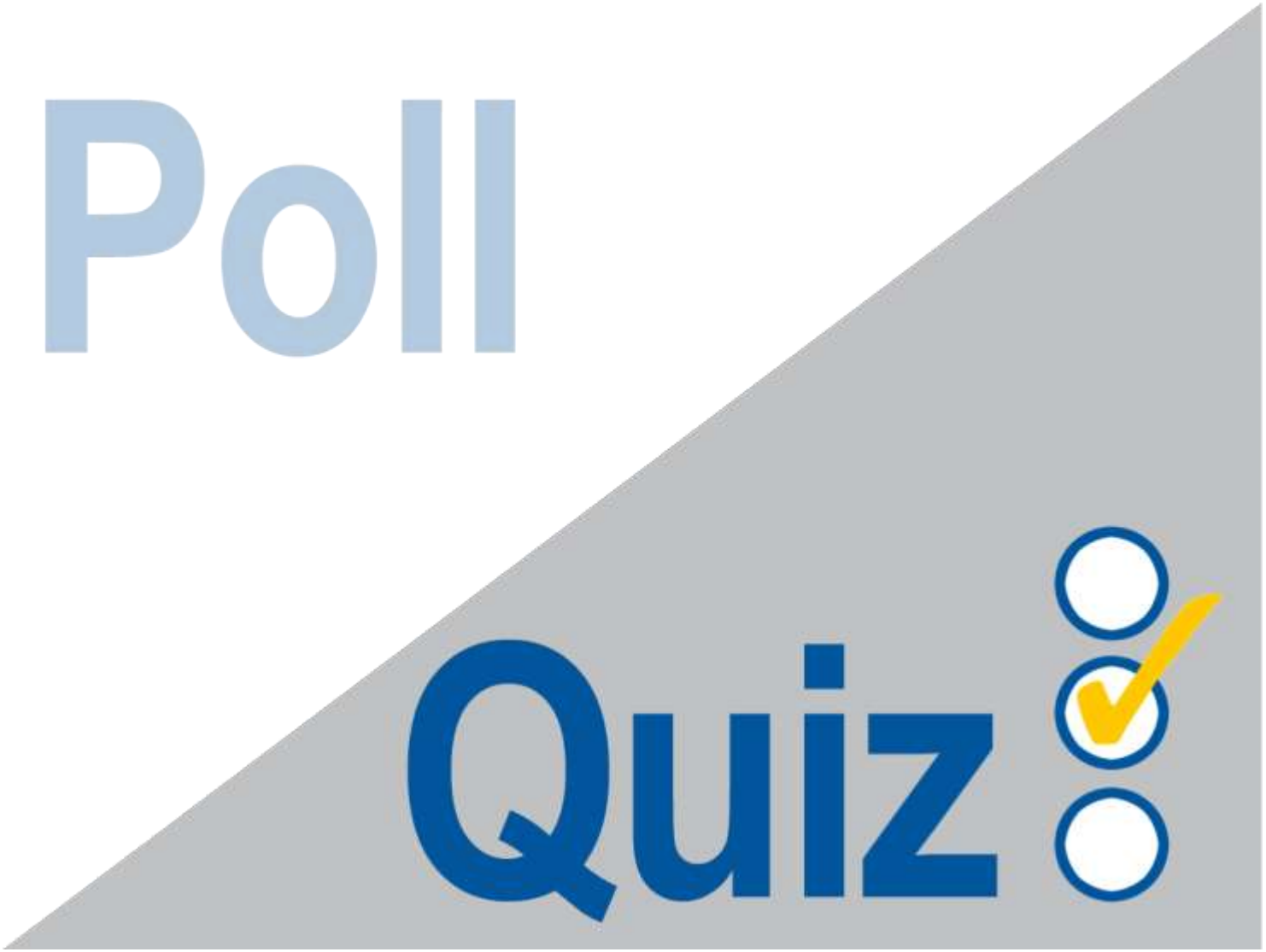
The CLASS Statement

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



The CLASS statement adds the N Obs column, which is the number of observations for each unique combination of the class variables.

Poll



12.04 Quiz

For a given data set, there are 63 observations with a **Country** value of AU. Of those 63 observations, only 61 observations have a value for **Salary**.

Which output is correct?

a.

Analysis Variable : Salary

Country	N Obs	N
AU	63	61

b.

Analysis Variable : Salary

Country	N Obs	N
AU	61	63

12.04 Quiz – Correct Answer

For a given data set, there are 63 observations with a **Country** value of AU. Of those 63 observations, only 61 observations have a value for **Salary**.

Which output is correct?

a.

Analysis Variable : Salary

Country	N Obs	N
AU	63	61

b.

Analysis Variable : Salary

Country	N Obs	N
AU	61	63

Additional SAS Statements

Additional statements can be added to enhance the reports.

```
proc format;
   value $ctryfmt 'AU'='Australia'
                  'US'='United States';
run;

options nodate pageno=1;
ods html file='p112d05.html';
proc means data=orion.sales;
   var Salary;
   class Gender Country;
   where Job_Title contains 'Rep';
   format Country $ctryfmt.;
   title 'Sales Rep Summary Report';
run;
ods html close;
```

p112d05

Additional SAS Statements

HTML Output

Sales Rep Summary Report

The MEANS Procedure

Analysis Variable : Salary							
Gender	**Country**	**N Obs**	**N**	**Mean**	**Std Dev**	**Minimum**	**Maximum**
F	Australia	27	27	27702.41	1728.23	25185.00	30890.00
	United States	40	40	28109.88	1874.39	25390.00	32985.00
M	Australia	34	34	28112.35	2295.81	25745.00	36605.00
	United States	58	58	27775.26	2311.91	22710.00	35990.00

PROC MEANS Statistics

The statistics to compute and the order to display them can be specified in the PROC MEANS statement.

```
proc means data=orion.sales sum mean range;
   var Salary;
   class Country;
run;
```

The MEANS Procedure

Analysis Variable : Salary

Country	N Obs	Sum	Mean	Range
AU	63	1900015.00	30158.97	83070.00
US	102	3241405.00	31778.48	220480.00

p112d05

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			

PROC MEANS Statement Options

Options can also be placed in the PROC MEANS statement.

Option	Description
MAXDEC=	specifies the number of decimal places to use in printing the statistics.
FW=	specifies the field width to use in displaying the statistics.
NONOBS	suppresses reporting the total number of observations for each unique combination of the class variables.

MAXDEC= Option

```
proc means data=orion.sales maxdec=0;
```

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

```
proc means data=orion.sales maxdec=1;
```

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

p112d05

FW= Option

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

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

```
proc means data=orion.sales fw=15;
```

Analysis Variable : Salary

Country	N Obs	N	Mean	Std Dev	Minimum	Maximum
AU	63	63	30158.96825397	12699.13932690	25185.00000000	108255
US	102	102	31778.48039216	23555.84171928	22710.00000000	243190

p112d05

NONOBS Option

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

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

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

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

p112d05

Output Data Sets

PROC MEANS produces output data sets using the following method:

OUTPUT OUT=*SASdataset <options>***;**

The output data set contains the following variables:

- BY variables
- class variables
- the automatic variables **_TYPE_** and **_FREQ_**
- the variables requested in the OUTPUT statement

OUTPUT Statement OUT= Option

The statistics in the PROC statement impact only the MEANS report, not the data set.

```
proc means data=orion.sales sum mean range;
   var Salary;
   class Gender Country;
   output out=work.means1;
run;

proc print data=work.means1;
run;
```

OUTPUT Statement OUT= Option

Partial PROC PRINT Output

Obs	Gender	Country	_TYPE_	_FREQ_	_STAT_	Salary
1			0	165	N	165.00
2			0	165	MIN	22710.00
3			0	165	MAX	243190.00
4			0	165	MEAN	31160.12
5			0	165	STD	20082.67
6		AU	1	63	N	63.00
7					MIN	25185.00
8					MAX	108255.00
9		AU	1	63	MEAN	30158.97
10		AU	1	63	STD	12699.14
11		US	1	102	N	102.00
12		US	1	102	MIN	22710.00
13		US	1	102	MAX	243190.00
14		US	1	102	MEAN	31778.48
15		US	1	102	STD	23555.84
16	F		2	68	N	68.00
17	F		2	68	MIN	25185.00
18	F		2	68	MAX	83505.00
19	F		2	68	MEAN	28762.72
20	F		2	68	STD	6974.15

default statistics

OUTPUT Statement OUT= Option

The OUTPUT statement can also do the following:

- specify the statistics for the output data set
- select and name variables

```
proc means data=orion.sales noprint;
   var Salary;
   class Gender Country;
   output out=work.means2
          min=minSalary max=maxSalary
          sum=sumSalary mean=aveSalary;
run;

proc print data=work.means2;
run;
```

The NOPRINT option suppresses the display of all output.

p112d06

OUTPUT Statement OUT= Option

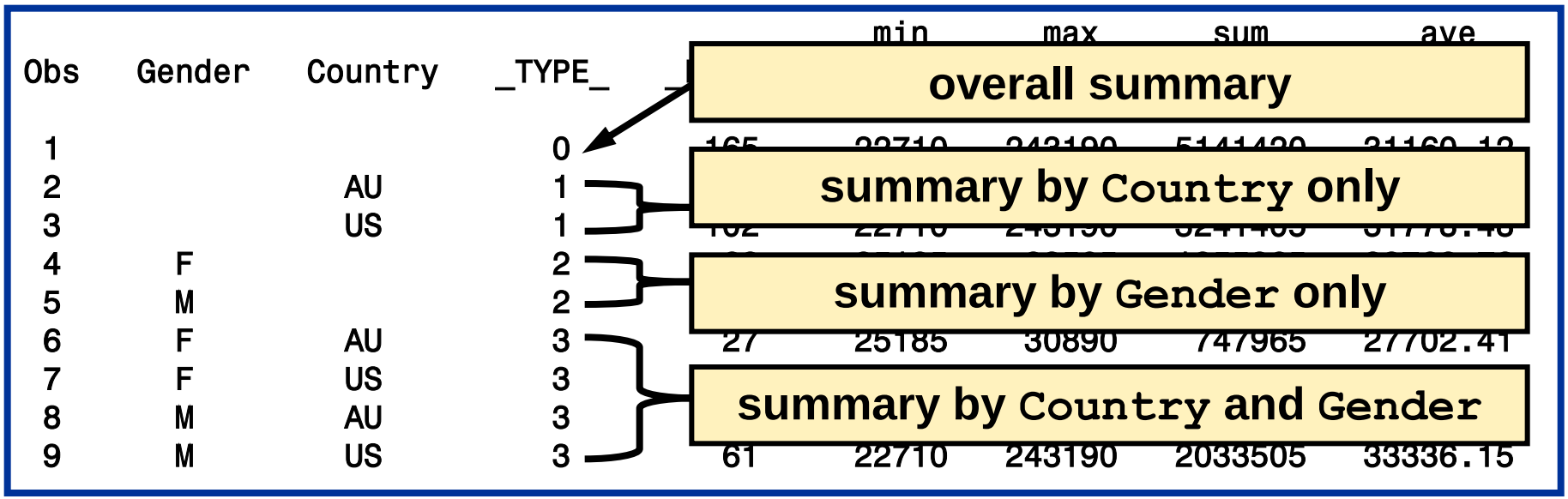
PROC PRINT Output

Obs	Gender	Country	_TYPE_	_FREQ_	min Salary	max Salary	sum Salary	ave Salary
1			0	165	22710	243190	5141420	31160.12
2		AU	1	63	25185	108255	1900015	30158.97
3		US	1	102	22710	243190	3241405	31778.48
4	F		2	68	25185	83505	1955865	28762.72
5	M		2	97	22710	243190	3185555	32840.77
6	F	AU	3	27	25185	30890	747965	27702.41
7	F	US	3	41	25390	83505	1207900	29460.98
8	M	AU	3	36	25745	108255	1152050	32001.39
9	M	US	3	61	22710	243190	2033505	33336.15

OUTPUT Statement OUT= Option

TYPE is a numeric variable that shows which combination of class variables produced the summary statistics in that observation.

PROC PRINT Output



OUTPUT Statement OUT= Option

Obs	Gender	Country	_TYPE_	_FREQ_	min Salary	max Salary	sum Salary	ave Salary
1			0	165	22710	243190	5141420	31160.12
2		AU	1	63	25185	108255	1900015	30158.97
3		US	1	102	22710	243190	3241405	31778.48
4	F		2	68	25185	83505	1955865	28762.72
5	M		2	97	22710	243190	3185555	32840.77
6	F	AU	3	27	25185	30890	747965	27702.41
7	F	US	3	41	25390	83505	1207900	29460.98
8	M	AU	3	36	25745	108255	1152050	32001.39
9	M	US	3	61	22710	243190	2033505	33336.15

TYPE	Type of Summary	_FREQ_
0	overall summary	165
1	summary by **Country** only	63 AU + 102 AU = 165
2	summary by **Gender** only	68 F + 97 M = 165
3	summary by **Country** and **Gender**	27 F AU + 41 F US + 36 M AU + 61 M US = 165

OUTPUT Statement OUT= Option

Options can be added to the PROC MEANS statement to control the output data set.

Option	Description
NWAY	specifies that the output data set contain only statistics for the observations with the highest **_TYPE_** value.
DESCENDTYPES	orders the output data set by descending **_TYPE_** value.
CHARTYPE	specifies that the **_TYPE_** variable in the output data set is a character representation of the binary value of **_TYPE_**.

OUTPUT Statement OUT= Option

without options

Obs	Gender	Country	_TYPE_	_FREQ_	min Salary	max Salary	sum Salary	ave Salary
1			0	165	22710	243190	5141420	31160.12
2		AU	1	63	25185	108255	1900015	30158.97
3		US	1	102	22710	243190	3241405	31778.48
4	F		2	68	25185	83505	1955865	28762.72
5	M		2	97	22710	243190	3185555	32840.77
6	F	AU	3	27	25185	30890	747965	27702.41
7	F	US	3	41	25390	83505	1207900	29460.98
8	M	AU	3	36	25745	108255	1152050	32001.39
9	M	US	3	61	22710	243190	2033505	33336.15

with NWAY

Obs	Gender	Country	_TYPE_	_FREQ_	min Salary	max Salary	sum Salary	ave Salary
1	F	AU	3	27	25185	30890	747965	27702.41
2	F	US	3	41	25390	83505	1207900	29460.98
3	M	AU	3	36	25745	108255	1152050	32001.39
4	M	US	3	61	22710	243190	2033505	33336.15

p112d06

OUTPUT Statement OUT= Option

with DESCENDTYPES

Obs	Gender	Country	_TYPE_	_FREQ_	min Salary	max Salary	sum Salary	ave Salary
1	F	AU	3	27	25185	30890	747965	27702.41
2	F	US	3	41	25390	83505	1207900	29460.98
3	M	AU	3	36	25745	108255	1152050	32001.39
4	M	US	3	61	22710	243190	2033505	33336.15
5	F		2	68	25185	83505	1955865	28762.72
6	M		2	97	22710	243190	3185555	32840.77
7		AU	1	63	25185	108255	1900015	30158.97
8		US	1	102	22710	243190	3241405	31778.48
9			0	165	22710	243190	5141420	31160.12

p112d06

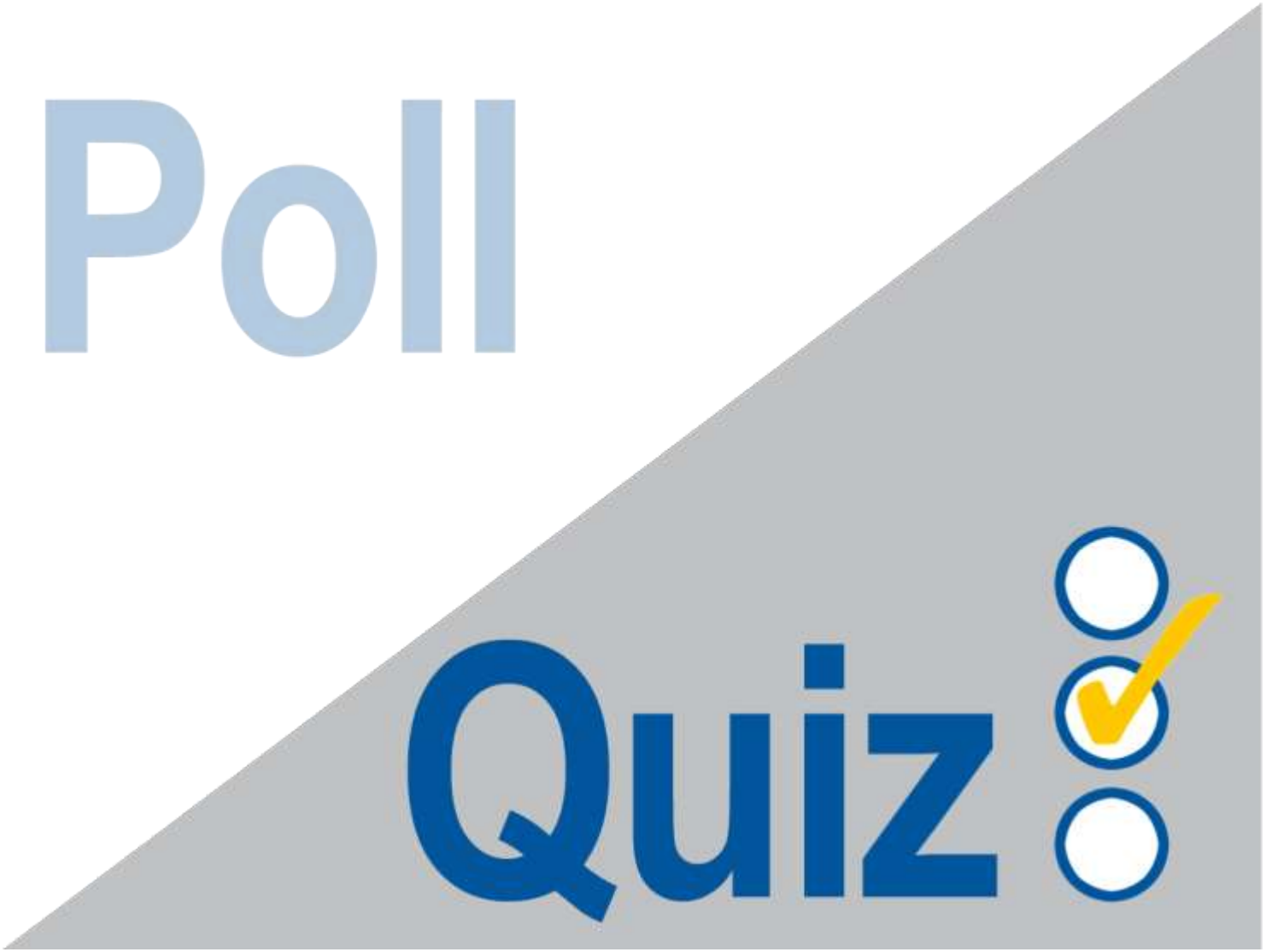
OUTPUT Statement OUT= Option

with CHARTYPE

Obs	Gender	Country	_TYPE_	_FREQ_	min Salary	max Salary	sum Salary	ave Salary
1			00	165	22710	243190	5141420	31160.12
2		AU	01	63	25185	108255	1900015	30158.97
3		US	01	102	22710	243190	3241405	31778.48
4	F		10	68	25185	83505	1955865	28762.72
5	M		10	97	22710	243190	3185555	32840.77
6	F	AU	11	27	25185	30890	747965	27702.41
7	F	US	11	41	25390	83505	1207900	29460.98
8	M	AU	11	36	25745	108255	1152050	32001.39
9	M	US	11	61	22710	243190	2033505	33336.15

p112d06

Poll



12.05 Quiz

- Retrieve and submit program **p112a02**.
- Review the PROC PRINT output.
- Add a WHERE statement to the PROC PRINT step to subset **`_TYPE_`** for observations summarized by **`Gender`** only.
- Submit the program and verify the results.

12.05 Quiz – Correct Answer

```
proc means data=orion.sales noprint chartype;
   var Salary;
   class Gender Country;
   output out=work.means2
          min=minSalary max=maxSalary
          sum=sumSalary mean=aveSalary;
run;
proc print data=work.means2;
   where _type_= '10';
run;
```

Obs	Gender	Country	_TYPE_	_FREQ_	min Salary	max Salary	sum Salary	ave Salary
4	F		10	68	25185	83505	1955865	28762.72
5	M		10	97	22710	243190	3185555	32840.77

p112a02s

OUTPUT Statement OUT= Option

Program **p112d07** is an example of merging a PROC MEANS output data set with a detail data set to create the following partial report.

Obs	First_ Name	Last_Name	Salary	Comparison to Country Salary Average	Comparison to Gender Salary Average
1	Tom	Zhou	108255	Above Average	Above Average
2	Wilson	Dawes	87975	Above Average	Above Average
3	Irenie	Elvish	26600	Below Average	Below Average
4	Christina	Ngan	27475	Below Average	Below Average
5	Kimiko	Hotstone	26190	Below Average	Below Average
6	Lucian	Daymond	26480	Below Average	Below Average
7	Fong	Hofmeister	32040	Above Average	Below Average
8	Satyakam	Denny	26780	Below Average	Below Average
9	Sharryn	Clarkson	28100	Below Average	Below Average
10	Monica	Kletschkus	30890	Above Average	Above Average

detail data

detail data compared to summary data

The SUMMARY Procedure

The SUMMARY procedure provides data summarization tools to compute descriptive statistics for variables across all observations and within groups of observations.

General form of the SUMMARY procedure:

```
PROC SUMMARY DATA=SASdataset <statistic(s)>
                                    <option(s)>;
     VAR analysis-variable(s);
     CLASS classification-variable(s);
RUN;
```

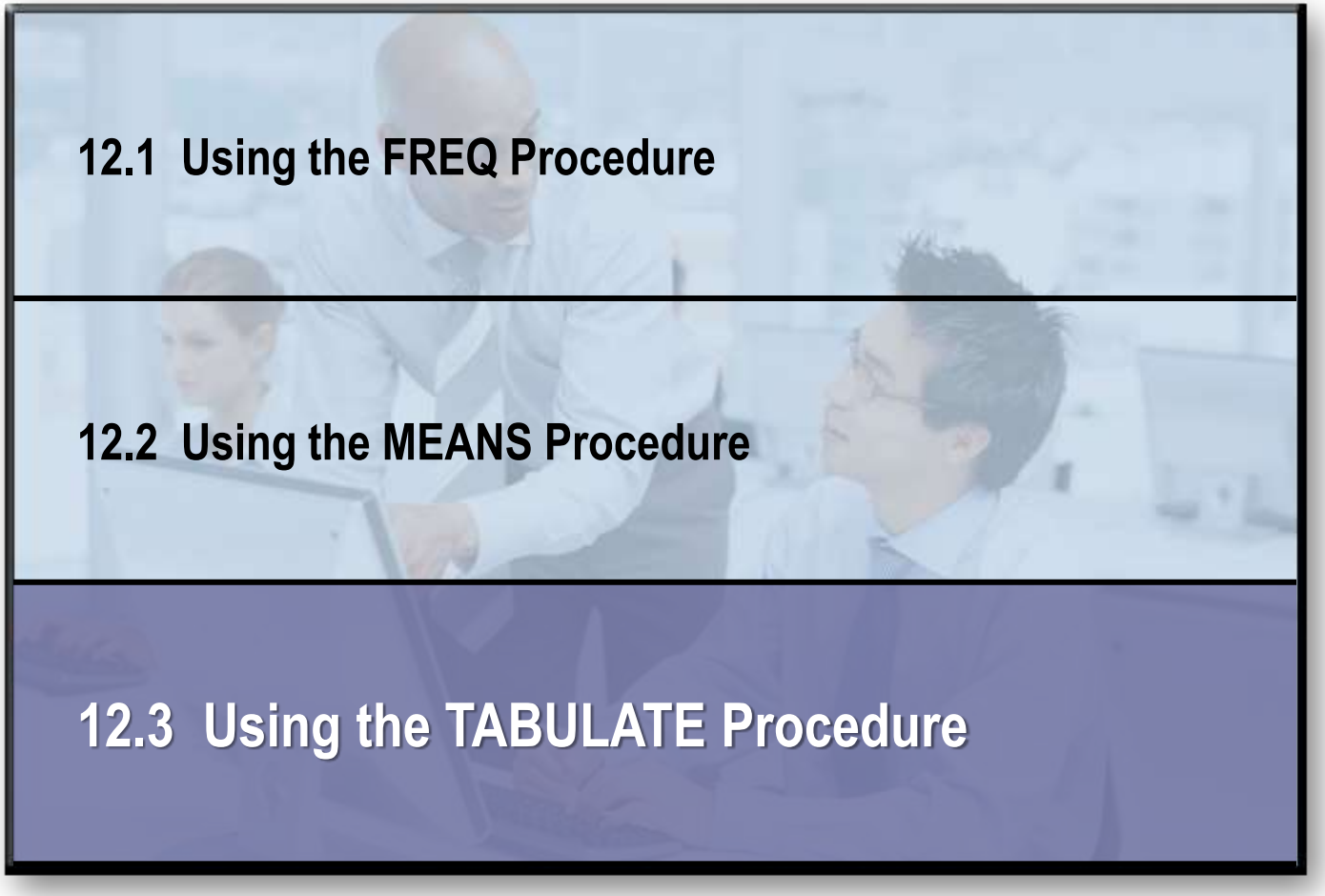
The SUMMARY Procedure

The SUMMARY procedure uses the same syntax as the MEANS procedure.

The only differences to the two procedures are the following:

PROC MEANS	PROC SUMMARY
The PRINT option is set by default, which displays output.	The NOPRINT option is set by default, which displays no output.
Omitting the VAR statement analyzes all the numeric variables.	Omitting the VAR statement produces a simple count of observations.

Chapter 12: Producing Summary Reports



Objectives

- Create one-, two-, and three-dimensional tabular reports using the TABULATE procedure.
- Produce output data sets by using the OUT= option in the PROC statement.

The TABULATE Procedure

The TABULATE procedure displays descriptive statistics in tabular format.

General form of the TABULATE procedure:

```
PROC TABULATE DATA=SASdataset <options>;
      CLASS classificationvariable(s);
      VAR analysis-variable(s);
      TABLE  pageexpression,
             rowexpression,
             columnexpression </ option(s)>;
RUN;
```

Dimensional Tables

The TABULATE procedure produces one-, two-, or three-dimensional tables.

	page dimension	row dimension	column dimension
one-dimensional			✓
two-dimensional		✓	✓
three-dimensional	✓	✓	✓

One-Dimensional Table

Country	
AU	US
N	N
63.00	102.00

- **Country** is in the column dimension.

Two-Dimensional Table

	Country	
	AU	US
	N	N
Gender		
F	27.00	41.00
M	36.00	61.00

- **Country** is in the column dimension.
- **Gender** is in the row dimension.

Three-Dimensional Table

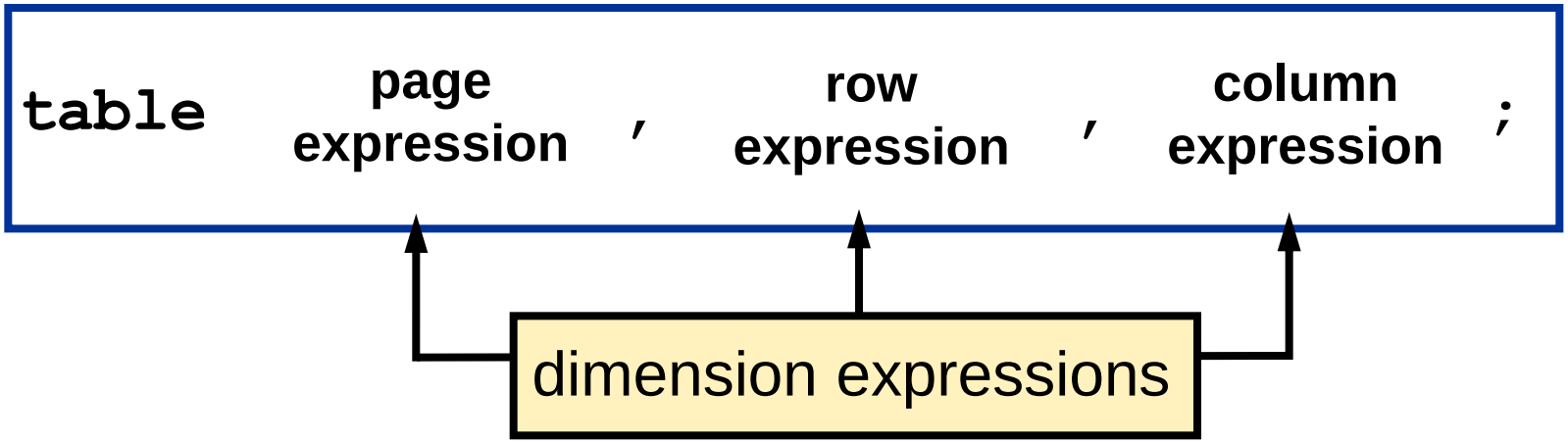
Job_Title Sales Rep. I

	Country	
	AU	US
	N	N
Gender		
F	8.00	13.00
M	13.00	29.00

- **Country** is in the column dimension.
- **Gender** is in the row dimension.
- **Job_Title** is in the page dimension.

The TABLE Statement

The TABLE statement describes the structure of the table.



- Commas separate the dimension expressions.
- Every variable that is part of a dimension expression must be specified as a classification variable (CLASS statement) or an analysis variable (VAR statement).

The TABLE Statement

PROC TABULATE

age w olumn

The TABLE Statement

```
table  page expression , row expression , column expression ;
```

Examples:

```
table Country;
```

```
table Gender , Country;
```

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

The CLASS Statement

The CLASS statement identifies variables to be used as classification, or grouping, variables.

General form of the CLASS statement:

CLASS *classificationvariable(s)*;

- N, the number of nonmissing values, is the default statistic for classification variables.
- Examples of classification variables:

  `Job_Title`, **`Gender`**, and **`Country`**

The VAR Statement

The VAR statement identifies the numeric variables for which statistics are calculated.

General form of the VAR statement:

VAR *analysisvariable(s)*;

- SUM is the default statistic for analysis variables.
- Examples of analysis variables:

  `Salary` and **`Bonus`**

One-Dimensional Table

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

Country	
AU	US
N	N
63.00	102.00

p112d08

Two-Dimensional Table

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

	Country	
	AU	US
	N	N
Gender		
F	27.00	41.00
M	36.00	61.00

p112d08

Three-Dimensional Table

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

Three-Dimensional Table

Partial PROC TABULATE Output

Job_Title Sales Rep. I

	Country	
	AU	US

Job_Title Sales Rep. II

	Country	
	AU	US
	N	N
Gender		
F	10.00	14.00
M	8.00	14.00

Dimension Expression

Elements that can be used in a dimension expression:

- classification variables
- analysis variables
- the universal class variable ALL
- keywords for statistics

Operators that can be used in a dimension expression:

- blank, which concatenates table information
- asterisk *, which crosses table information
- parentheses (), which group elements

Dimension Expression

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

	Country	
	AU	US
	Salary	Salary
	Sum	Sum
Gender		
F	747965.00	1207900.00
M	1152050.00	2033505.00
All	1900015.00	3241405.00

p112d08

PROC TABULATE Statistics

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

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

Hypothesis Testing Keywords				
PROBT	T			

PROC TABULATE Statistics

```
proc tabulate data=orion.sales;
   class Gender Country;
   var Salary;
   table Gender all, Country*Salary*(min max);
run;
```

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

Additional SAS Statements

Additional statements can be added to enhance the report.

```
proc format;
   value $ctryfmt 'AU'='Australia'
                  'US'='United States';
run;

options nodate pageno=1;

ods html file='p112d08.html';
proc tabulate data=orion.sales;
   class Gender Country;
   var Salary;
   table Gender all, Country*Salary*(min max);
   where Job_Title contains 'Rep';
   label Salary='Annual Salary';
   format Country $ctryfmt.;
   title 'Sales Rep Tabular Report';
run;
ods html close;
```

p112d08

Additional SAS Statements

HTML Output

Sales Rep Tabular Report

	Country			
	Australia		United States	
	Annual Salary		Annual Salary	
	Min	Max	Min	Max
Gender				
F	25185.00	30890.00	25390.00	32985.00
M	25745.00	36605.00	22710.00	35990.00
All	25185.00	36605.00	22710.00	35990.00

Output Data Sets

PROC TABULATE produces output data sets using the following method:

PROC TABULATE DATA=*SAS-data-set*
OUT=*SAS-data-set* <*options*>**;**

The output data set contains the following variables:

- BY variables
- class variables
- the automatic variables **_TYPE_**, **_PAGE_**, and **_TABLE_**
- calculated statistics

PROC Statement OUT= Option

```
proc tabulate data=orion.sales
              out=work.tabulate;
   where Job_Title contains 'Rep';
   class Job_Title Gender Country;
   table Country;
   table Gender, Country;
   table Job_Title, Gender, Country;
run;

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

PROC Statement OUT= Option

Partial PROC PRINT Output

Obs	Job_Title	Gender	Country	_TYPE_	_PAGE_	_TABLE_	N
1			AU	001	1	1	61
2			US	001	1	1	98
3		F	AU	011	1	2	27
4		F	US	011	1	2	40
5		M	AU	011	1	2	34
6		M	US	011	1	2	58
7	Sales Rep. I	F	AU	111	1	3	8
8	Sales Rep. I	F	US	111	1	3	13
9	Sales Rep. I	M	AU	111	1	3	13
10	Sales Rep. I	M	US	111	1	3	29
11	Sales Rep. II	F	AU	111	2	3	10
12	Sales Rep. II	F	US	111	2	3	14
13	Sales Rep. II	M	AU	111	2	3	8
14	Sales Rep. II	M	US	111	2	3	14
15	Sales Rep. III	F	AU	111	3	3	7
16	Sales Rep. III	F	US	111	3	3	8
17	Sales Rep. III	M	AU	111	3	3	10
18	Sales Rep. III	M	US	111	3	3	9

PROC Statement OUT= Option

TYPE is a character variable that shows which combination of class variables produced the summary statistics in that observation.

Partial PROC PRINT Output

Obs	Job_Title	Gender	Country	_TYPE_	_PAGE_	_TABLE_	N
1			AU	001	1	1	61
2			US	001	1	1	98
3		F	AU	011	1	2	27
4		F	US	011			
5		M	AU	011			
6		M	US	011			

0 for **Job_Title**,
1 for **Gender**, and
1 for **Country**

PROC Statement OUT= Option

PAGE is a numeric variable that shows the logical page number that contains that observation.

Partial PROC PRINT Output

Obs	Job_Title	Gender	Country	_TYPE_	_PAGE_	_TABLE_	N
7	Sales Rep. I	F	AU	111	1		
8	Sales Rep. I	F	US	111	1		
9	Sales Rep. I	M	AU	111	1		
10	Sales Rep. I	M	US	111	1		
11	Sales Rep. II	F	AU	111	2		
12	Sales Rep. II	F	US	111	2		
13	Sales Rep. II	M	AU	111	2		
14	Sales Rep. II	M	US	111	2		
15	Sales Rep. III	F	AU	111	3		
16	Sales Rep. III	F	US	111	3		
17	Sales Rep. III	M	AU	111	3		
18	Sales Rep. III	M	US	111	3		

Page 1 for Sales Rep. I

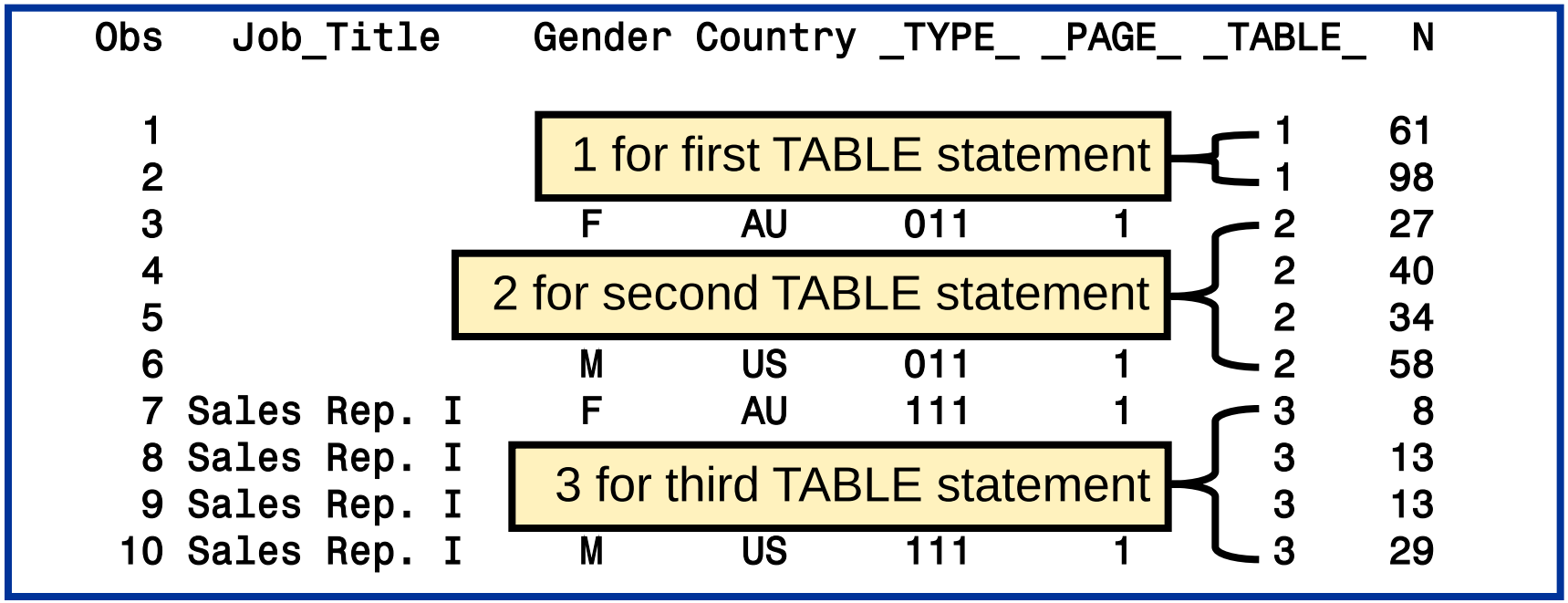
Page 2 for Sales Rep. II

Page 3 for Sales Rep. III

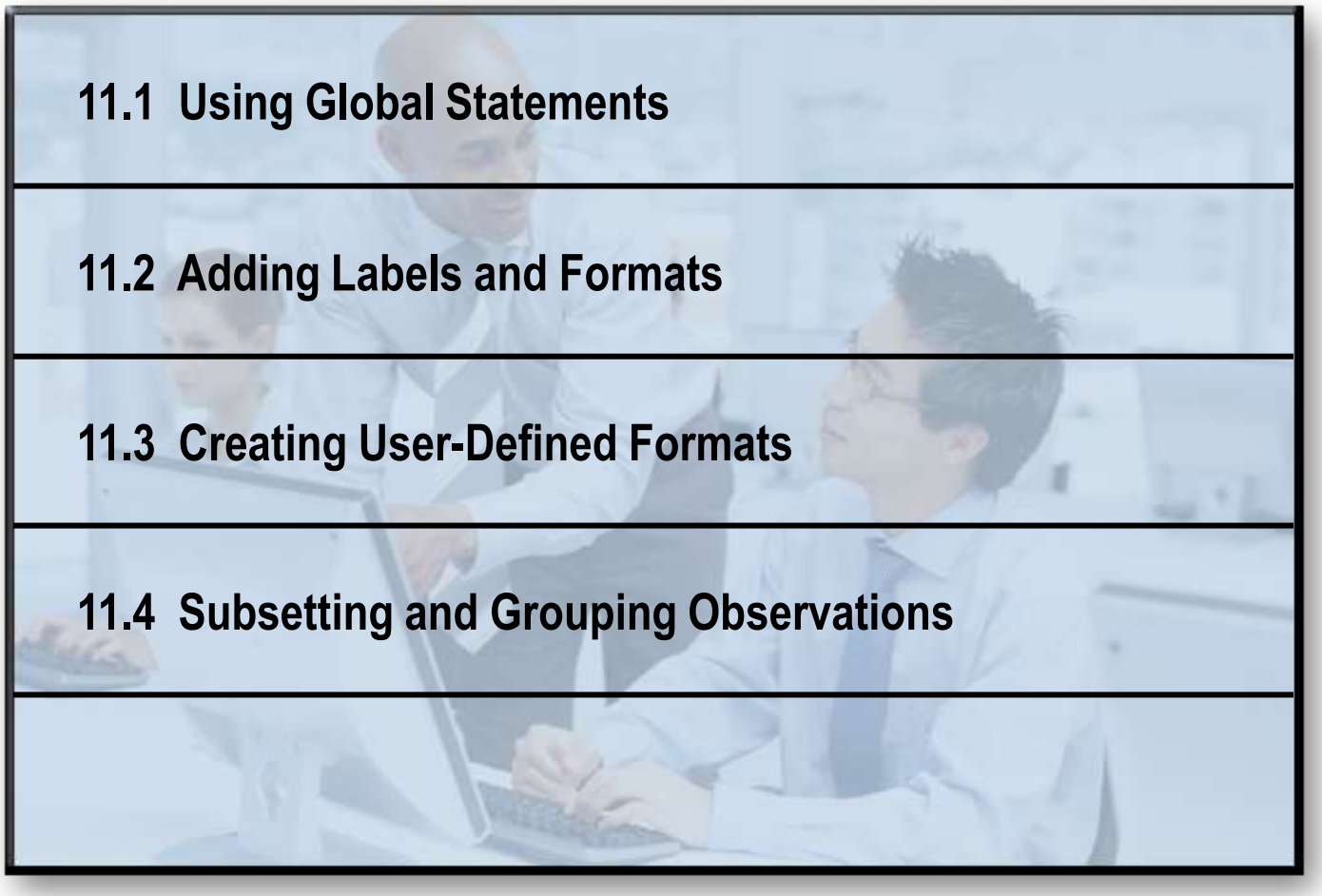
PROC Statement OUT= Option

TABLE is a numeric variable that shows the number of the TABLE statement that contains that observation.

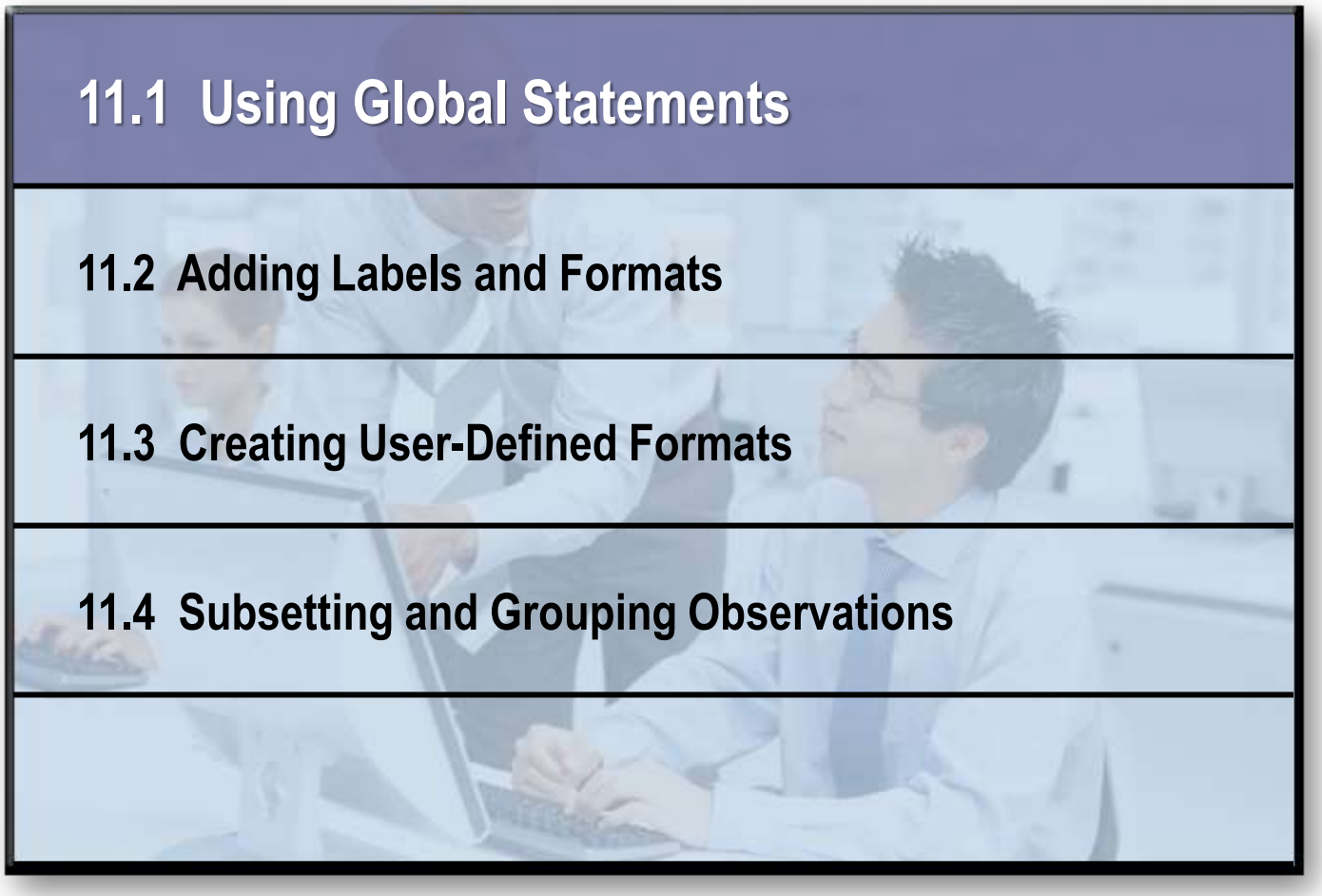
Partial PROC PRINT Output



Chapter 11: Enhancing Reports



Chapter 11: Enhancing Reports

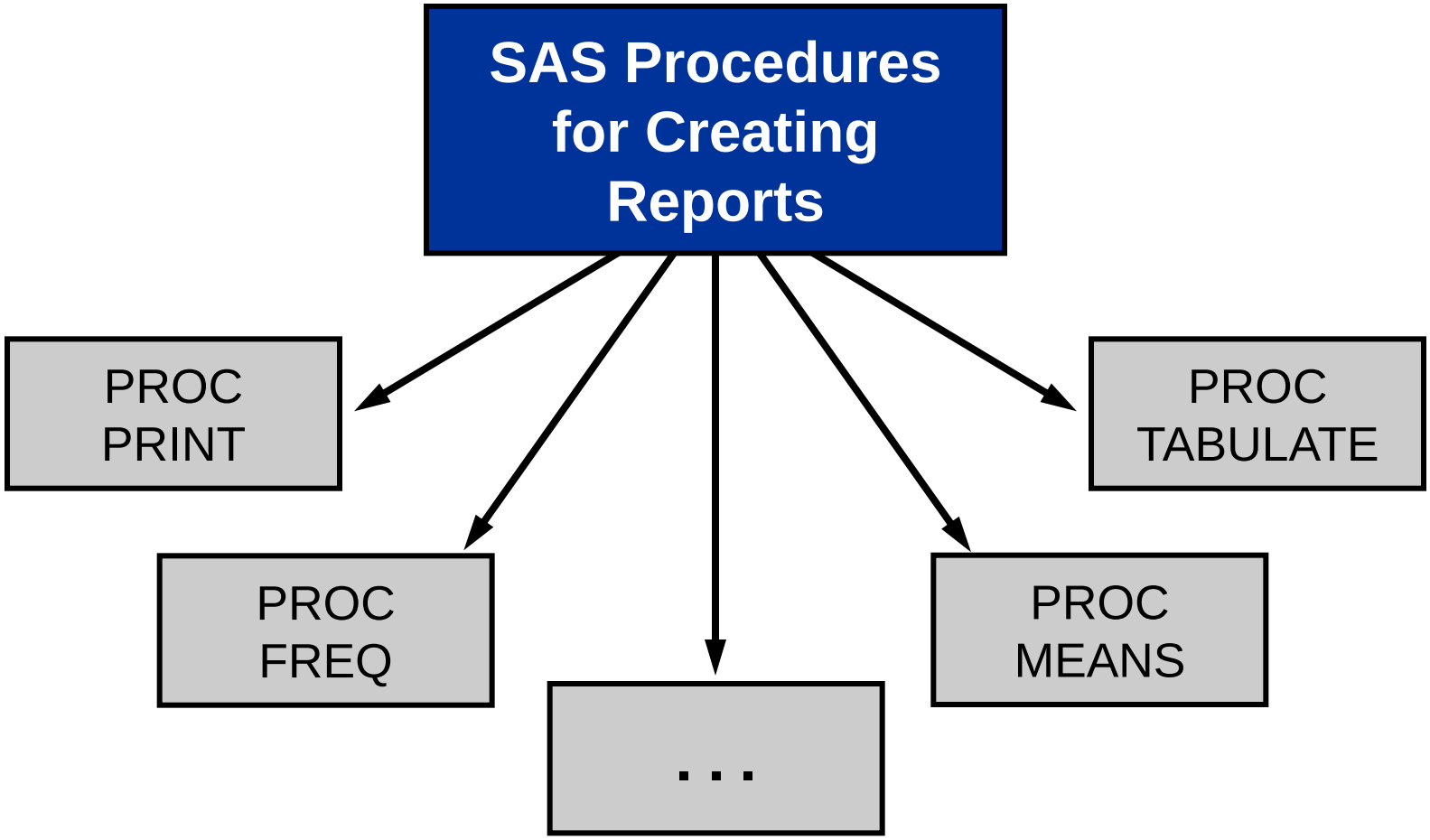


Objectives

- Identify SAS statements that are used with most reporting procedures.
- Enhance reports by using SAS system options.
- Enhance reports by adding titles and footnotes.
- Add dates and times to titles.

Creating Reports

A procedure step is a primary method for creating reports.



Example of a Basic Report

```
proc print data=orion.sales;
   var Employee_ID First_Name Last_Name Salary;
run;
```

Partial PROC PRINT Output

Obs	Employee_ID	First_ Name	Last_Name	Salary
1	120102	Tom	Zhou	108255
2	120103	Wilson	Dawes	87975
3	120121	Irenie	Elvish	26600
4	120122	Christina	Ngan	27475
5	120123	Kimiko	Hotstone	26190
6	120124	Lucian	Daymond	26480
7	120125	Fong	Hofmeister	32040
8	120126	Satyakam	Denny	26780
9	120127	Sharryn	Clarkson	28100
10	120128	Monica	Kletschkus	30890

p111d01

Example of an Enhanced Report

```
options nocenter;
ods html file='enhanced.html' style=sasweb;
proc print data=orion.sales label;
   var Employee_ID First_Name Last_Name Salary;
   title1 'Orion Sales Employees';
   title2 'Males Only';
   footnote 'Confidential';
   label Employee_ID='Sales ID'
         First_Name='First Name'
         Last_Name='Last Name'
         Salary='Annual Salary';
   format Salary dollar8.;
   where Gender='M';
   by Country;
run;
ods html close;
```


p111d01

Example of an Enhanced Report

Partial PROC PRINT Output

Orion Sales Employees
Males Only

Country=AU

Obs	Sales ID	First Name	Last Name	Annual Salary
1	120102	Tom	Zhou	$108,255
2	120103	Wilson	Dawes	$87,975
6	120124	Lucian	Daymond	$26,480
7	120125	Fong	Hofmeister	$32,040
8	120126	Satyakam	Denny	$26,780
11	120129	Alvin	Roebuck	$30,070
12	120130	Kevin	Lyon	$26,955
13	120131	Marinus	Surawski	$26,910
16	120134	Sian	Shannan	$28,015
17	120135	Alexei	Platts	$32,490
18	120136	Atul	Leyden	$26,605

Statements That Enhance Reports

Many statements are used with most reporting procedures to enhance the report.

```
options nocenter;
ods html file='enhanced.html' style=sasweb;
proc print data=orion.sales label;
   var Employee_ID First_Name Last_Name Salary;
   title1 'Orion Sales Employees';
   title2 'Males Only';
   footnote 'Confidential';
   label Employee_ID='Sales ID'
         First_Name='First Name'
         Last_Name='Last Name'
         Salary='Annual Salary';
   format Salary dollar8.;
   where Gender='M';
   by Country;
run;
ods html close;
```

Global Statements

The following are global statements that enhance reports:

- OPTIONS
- TITLE
- FOOTNOTE
- ODS

Global statements are specified anywhere in your SAS program and they remain in effect until canceled, changed, or your SAS session ends.

The OPTIONS Statement

The *OPTIONS statement* changes the value of one or more SAS system options.

General form of the OPTIONS statement:

OPTIONS *option(s)*;

- Some SAS system options change the appearance of a report.
- The OPTIONS statement is **not** usually included in a PROC or DATA step.

SAS System Options for Reporting

Selected SAS System Options:

DATE (default)	displays the date and time that the SAS session began at the top of each page of SAS output.
NODATE	does not display the date and time that the SAS session began at the top of each page of SAS output.
NUMBER (default)	prints page numbers on the first line of each page of SAS output.
NONUMBER	does not print page numbers on the first line of each page of SAS output.
PAGENO=*n*	defines a beginning page number (*n*) for the next page of SAS output.

continued...

SAS System Options for Reporting

Selected SAS System Options:

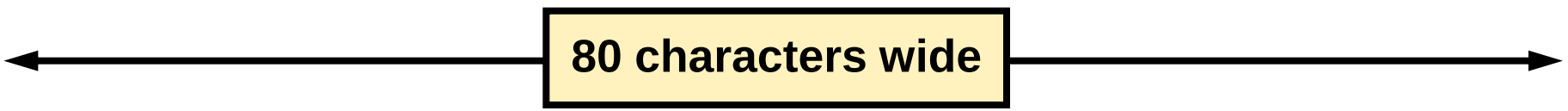
CENTER (default)	centers SAS output.
NOCENTER	left-aligns SAS output.
ORIENTATION = portrait (default)	Controls paper rotation to be taller than wide
ORIENTATION = landscape	Controls paper rotation to be wider than tall
PAGESIZE=*n** PS=*n*	defines the number of lines (*n*) that can be printed per page of SAS output.
LINESIZE=*width** LS=*width*	defines the line size (*width*) for the SAS log and SAS output.

*Affects listing output only

SAS System Options for Reporting

```
options ls=80 date number;

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

```
                                                09:11 Monday, January 14, 2008   35

                                The MEANS Procedure

                                Analysis Variable : Salary

        N            Mean         Std Dev         Minimum         Maximum
-------------------------------------------------------------------------------
      165        31160.12        20082.67        22710.00        243190.00
-------------------------------------------------------------------------------
```



p111d02

SAS System Options for Reporting

```
options nodate pageno=1;

proc freq data=orion.sales;
   tables Country;
run;
```

1

The FREQ Procedure

Country	Frequency	Percent	Cumulative Frequency	Cumulative Percent
AU	63	38.18	63	38.18
US	102	61.82	165	100.00

80 characters wide

p111d02


Poll
Quiz

Setup for the Poll

- Retrieve and submit program **p111a01**.
- Review the results including the date, time, and page number in the top-right corner of each page of output.
- Add the DTRESET system option to the OPTIONS statement.
- Submit the program and review the results.

DTRESET	updates date and time at the top of each page of SAS output.
NODTRESET (Default)	does not update date and time at the top of each page of SAS output.

11.01 Poll

Did the date and/or time change?

- Yes
- No

11.01 Poll – Correct Answer

Did the date and/or time change?

- ❍ Yes
- ❍ No

The DTRESET option uses the current date and time versus the SAS invocation date and time.

```
options date number pageno=1 ls=100 dtreset;
```

The TITLE Statement

The *TITLE statement* specifies title lines for SAS output.

General form of the TITLE statement:

TITLE*n* '*text* '**;**

- Titles appear at the top of the page.
- The default title is **The SAS System**.
- The value of ***n*** can be from 1 to 10.
- An unnumbered **TITLE** is equivalent to **TITLE1**.
- Titles remain in effect until they are changed, canceled, or you end your SAS session.

The FOOTNOTE Statement

The *FOOTNOTE statement* specifies footnote lines for SAS output.

General form of the FOOTNOTE statement:

FOOTNOTE*n* '*text* '**;**

- Footnotes appear at the bottom of the page.
- No footnote is printed unless one is specified.
- The value of ***n*** can be from 1 to 10.
- An unnumbered **FOOTNOTE** is equivalent to FOOTNOTE1.
- Footnotes remain in effect until they are changed, canceled, or you end your SAS session.

The TITLE and FOOTNOTE Statements

```
footnote1 'By Human Resource Department';
footnote3 'Confidential';

proc means data=orion.sales;
   var Salary;
   title 'Orion Star Sales Employees';
run;
```

The TITLE and FOOTNOTE Statements

Orion Star Sales Employees

The MEANS Procedure

Analysis Variable : Salary

N	Mean	Std Dev	Minimum	Maximum
165	31160.12	20082.67	22710.00	243190.00

By Human Resource Department

Confidential

Changing Titles and Footnotes

TITLE*n* or **FOOTNOTE*n***

- replaces a previous title or footnote with the same number
- cancels all titles or footnotes with higher(larger) numbers.

Canceling All Titles and Footnotes

- The null TITLE statement cancels all titles.

```
title;
```

- The null FOOTNOTE statement cancels all footnotes.

```
footnote;
```

Changing and Canceling Titles and Footnotes

PROC PRINT Code	Resultant Title(s)

`proc print data=orion.sales;`
`  title1 'The First Line';`
`  title2 'The Second Line';`
`run;`	

`proc print data=orion.sales;`
`  title2 'The Next Line';`
`run;`	

`proc print data=orion.sales;`
`  title 'The Top Line';`
`run;`	

`proc print data=orion.sales;`
`  title3 'The Third Line';`
`run;`	

`proc print data=orion.sales;`
`  title;`
`run;`	

...

Changing and Canceling Titles and Footnotes

PROC PRINT Code	Resultant Title(s)

`proc print data=orion.sales;`
`  title1 'The First Line';`
`  title2 'The Second Line';`
`run;`	

`proc print data=orion.sales;`
`  title2 'The Next Line';`
`run;`	

`proc print data=orion.sales;`
`  title 'The Top Line';`
`run;`	

`proc print data=orion.sales;`
`  title3 'The Third Line';`
`run;`	

`proc print data=orion.sales;`
`  title;`
`run;`	

...

Changing and Canceling Titles and Footnotes

PROC PRINT Code	Resultant Title(s)

`proc print data=orion.sales;`
`  title1 'The First Line';`
`  title2 'The Second Line';`
`run;`
	The First Line
	The Second Line

`proc print data=orion.sales;`
`  title2 'The Next Line';`
`run;`	

`proc print data=orion.sales;`
`  title 'The Top Line';`
`run;`	

`proc print data=orion.sales;`
`  title3 'The Third Line';`
`run;`	

`proc print data=orion.sales;`
`  title;`
`run;`	

...

Changing and Canceling Titles and Footnotes

PROC PRINT Code	Resultant Title(s)

`proc print data=orion.sales;`
`  title1 'The First Line';`
`  title2 'The Second Line';`
`run;`
	The First Line
	The Second Line

`proc print data=orion.sales;`
`  title2 'The Next Line';`
`run;`	

`proc print data=orion.sales;`
`  title 'The Top Line';`
`run;`	

`proc print data=orion.sales;`
`  title3 'The Third Line';`
`run;`	

`proc print data=orion.sales;`
`  title;`
`run;`	

...

Changing and Canceling Titles and Footnotes

PROC PRINT Code	Resultant Title(s)

`proc print data=orion.sales;`
`  title1 'The First Line';`
`  title2 'The Second Line';`
`run;`
	The First Line
	The Second Line

`proc print data=orion.sales;`
`  title2 'The Next Line';`
`run;`
	The First Line
	The Next Line

`proc print data=orion.sales;`
`  title 'The Top Line';`
`run;`	

`proc print data=orion.sales;`
`  title3 'The Third Line';`
`run;`	

`proc print data=orion.sales;`
`  title;`
`run;`	

...

Changing and Canceling Titles and Footnotes

PROC PRINT Code	Resultant Title(s)

`proc print data=orion.sales;`
`  title1 'The First Line';`
`  title2 'The Second Line';`
`run;`
	The First Line
	The Second Line

`proc print data=orion.sales;`
`  title2 'The Next Line';`
`run;`
	The First Line
	The Next Line

`proc print data=orion.sales;`
`  title 'The Top Line';`
`run;`	

`proc print data=orion.sales;`
`  title3 'The Third Line';`
`run;`	

`proc print data=orion.sales;`
`  title;`
`run;`	

...

Changing and Canceling Titles and Footnotes

PROC PRINT Code	Resultant Title(s)

`proc print data=orion.sales;`
`  title1 'The First Line';`
`  title2 'The Second Line';`
`run;`
	The First Line
	The Second Line

`proc print data=orion.sales;`
`  title2 'The Next Line';`
`run;`
	The First Line
	The Next Line

`proc print data=orion.sales;`
`  title 'The Top Line';`
`run;`
	The Top Line

`proc print data=orion.sales;`
`  title3 'The Third Line';`
`run;`	

`proc print data=orion.sales;`
`  title;`
`run;`	

...

Changing and Canceling Titles and Footnotes

PROC PRINT Code	Resultant Title(s)

`proc print data=orion.sales;`
`  title1 'The First Line';`
`  title2 'The Second Line';`
`run;`
	The First Line
	The Second Line

`proc print data=orion.sales;`
`  title2 'The Next Line';`
`run;`
	The First Line
	The Next Line

`proc print data=orion.sales;`
`  title 'The Top Line';`
`run;`
	The Top Line

`proc print data=orion.sales;`
`  title3 'The Third Line';`
`run;`	

`proc print data=orion.sales;`
`  title;`
`run;`	

...

Changing and Canceling Titles and Footnotes

PROC PRINT Code	Resultant Title(s)

`proc print data=orion.sales;`
`  title1 'The First Line';`
`  title2 'The Second Line';`
`run;`
	The First Line
	The Second Line

`proc print data=orion.sales;`
`  title2 'The Next Line';`
`run;`
	The First Line
	The Next Line

`proc print data=orion.sales;`
`  title 'The Top Line';`
`run;`
	The Top Line

`proc print data=orion.sales;`
`  title3 'The Third Line';`
`run;`
	The Top Line
	
	The Third Line

`proc print data=orion.sales;`
`  title;`
`run;`	

...

Changing and Canceling Titles and Footnotes

PROC PRINT Code	Resultant Title(s)

`proc print data=orion.sales;`
`  title1 'The First Line';`
`  title2 'The Second Line';`
`run;`
	The First Line
	The Second Line

`proc print data=orion.sales;`
`  title2 'The Next Line';`
`run;`
	The First Line
	The Next Line

`proc print data=orion.sales;`
`  title 'The Top Line';`
`run;`
	The Top Line

`proc print data=orion.sales;`
`  title3 'The Third Line';`
`run;`
	The Top Line
	
	The Third Line

`proc print data=orion.sales;`
`  title;`
`run;`	

...

Changing and Canceling Titles and Footnotes

PROC PRINT Code	Resultant Title(s)

`proc print data=orion.sales;`
`  title1 'The First Line';`
`  title2 'The Second Line';`
`run;`
	The First Line
	The Second Line

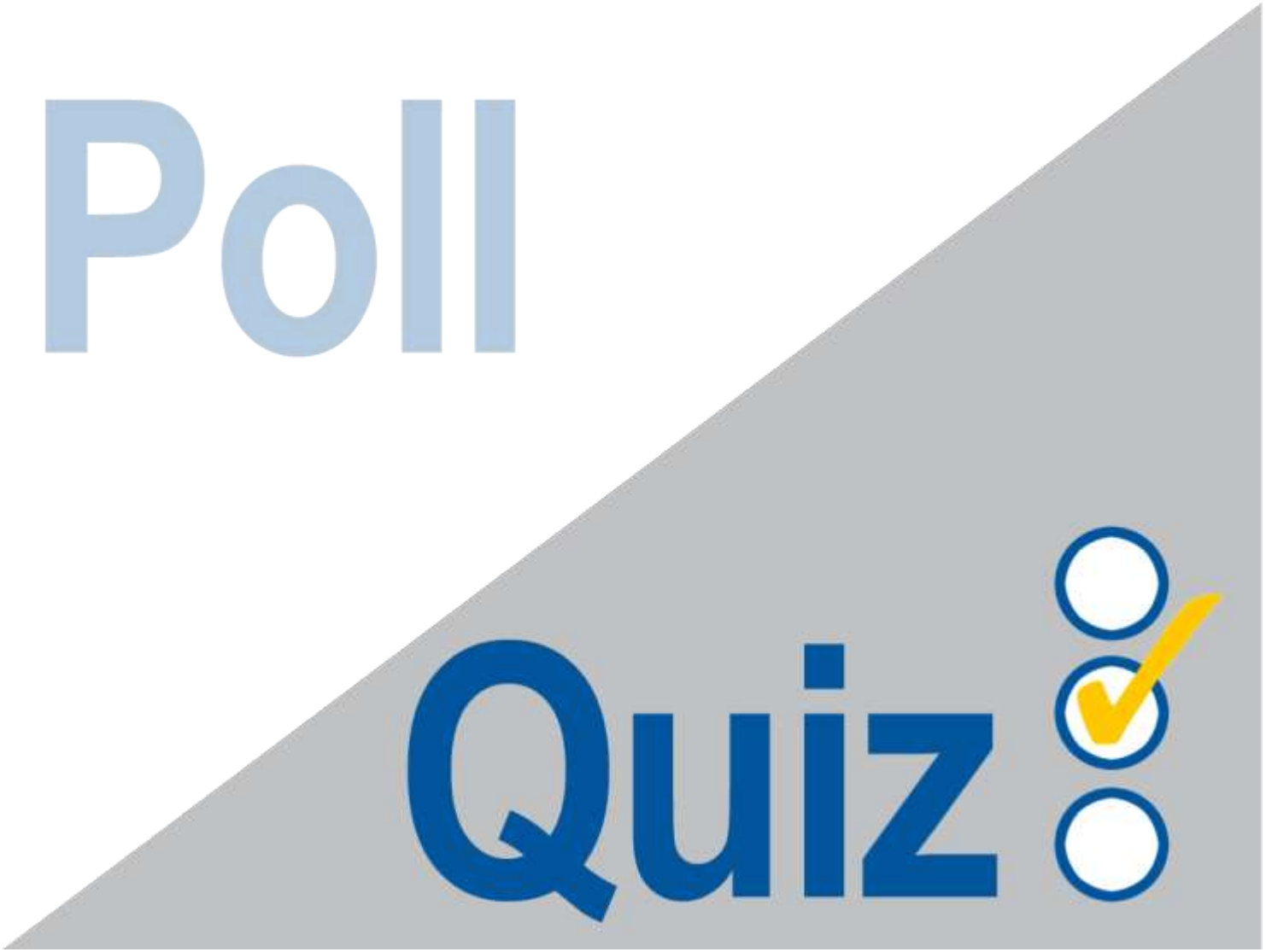
`proc print data=orion.sales;`
`  title2 'The Next Line';`
`run;`
	The First Line
	The Next Line

`proc print data=orion.sales;`
`  title 'The Top Line';`
`run;`
	The Top Line

`proc print data=orion.sales;`
`  title3 'The Third Line';`
`run;`
	The Top Line
	
	The Third Line

`proc print data=orion.sales;`
`  title;`
`run;`	

Poll



11.02 Quiz

Which footnote(s) appears in the second procedure output?

a.
```
Non Sales Employees
```

b.
```
Orion Star
Non Sales Employees
```

c.
```
Non Sales Employees
Confidential
```

d.
```
Orion Star
Non Sales Employees
Confidential
```

```
footnote1 'Orion Star';
proc print data=orion.sales;
   footnote2 'Sales Employees';
   footnote3 'Confidential';
run;
proc print data=orion.nonsales;
   footnote2 'Non Sales Employees';
run;
```

11.02 Quiz – Correct Answer

Which footnote(s) appears in the second procedure output?

a.
```
Non Sales Employees
```

b.
```
Orion Star
Non Sales Employees
```

c.
```
Non Sales Employees
Confidential
```

d.
```
Orion Star
Non Sales Employees
Confidential
```

```
footnote1 'Orion Star';
proc print data=orion.sales;
   footnote2 'Sales Employees';
   footnote3 'Confidential';
run;
proc print data=orion.nonsales;
   footnote2 'Non Sales Employees';
run;
```

Titles with Dates and Times

The automatic macro variables &SYSDATE9 and &SYSTIME can be used to add the SAS invocation date and time to titles and footnotes.

```
title1 'Orion Star Employee Listing';
title2 "Created on &sysdate9 at &systime";
```

Double quotation marks must be used when you reference a macro variable.

Example Title Output:

```
                    Orion Star Employee Listing
                    Created on 11MAR2008 at 15:53
```

Titles with Dates and Times

The %LET statement can be used with %SYSFUNC and the TODAY function or the TIME function to create a macro variable with the current date or time.

%LET *macro-variable* **= %SYSFUNC(today(),** *date-format***);**

%LET *macro-variable* **= %SYSFUNC(time(),** *time-format***);**

- %LET is a macro statement that creates a macro variable and assigns it a value without leading or trailing blanks.
- %SYSFUNC is a macro function that executes SAS functions outside of a step.

Titles with Dates and Times

```
%let currentdate=%sysfunc(today(),worddate.);
%let currenttime=%sysfunc(time(),timeampm.);

proc freq data=orion.sales;
   tables Gender Country;
   title1 'Orion Star Employee Listing';
   title2 "Created &currentdate";
   title3 "at &currenttime";
run;
```

Example Title Output:

```
                    Orion Star Employee Listing
                      Created March 11, 2008
                           at 4:09:43 PM
```