Basic Concepts #6: Introduction to Report Writing
Using By-line, PROC Report, PROC Means, PROC Freq

JC Wang
Outline

1. Use of Byline Information in Title Lines
   - By-Group Processing

2. PROC Report
   - Use of PROC Report
   - PROC REPORT Statement
   - Selected Statements Used in PROC Report

3. PROC Means
   - Comparing PROC Means and PROC Summary
   - Understanding the Use of Variables in PROC Means
   - Available Statistics in PROC Means
   - Creating SAS Data Set with PROC Means

4. PROC Freq
   - The Use of PROC Freq
   - PROC FREQ Statement
   - Statements Used in PROC Freq
By-Group Processing

- By-group processing in a procedure step, a BY line identifies each group in SAS output.
- By-group processing information can be inserted into titles.
- PROC PRINT, PROC MEANS/PROC SUMMARY and PROC STANDARD, each requires suppressing default BY lines by setting NOBYLINE system option.
- Set system option BYLINE to resume default by-group processing.
By-Group Processing

• By-group processing in a procedure step, a BY line identifies each group in SAS output.

• By-group processing information can be inserted into titles.

• PROC PRINT, PROC MEANS/PROC SUMMARY and PROC STANDARD, each requires suppressing default BY lines by setting NOBYLINE system option.

• Set system option BYLINE to resume default by-group processing.
By-Group Processing

By-group processing in a procedure step, a BY line identifies each group in SAS output.

By-group processing information can be inserted into titles.

PROC PRINT, PROC MEANS/PROC SUMMARY and PROC STANDARD, each requires suppressing default BY lines by setting NOBYLINE system option.

Set system option BYLINE to resume default by-group processing.
By-Group Processing

- By-group processing in a procedure step, a BY line identifies each group in SAS output.
- By-group processing information can be inserted into titles.
- PROC PRINT, PROC MEANS/PROC SUMMARY and PROC STANDARD, each requires suppressing default BY lines by setting NOBYLINE system option.
- Set system option BYLINE to resume default by-group processing.
By-Group Processing

By-group processing in a procedure step, a BY line identifies each group in SAS output.

By-group processing information can be inserted into titles.

PROC PRINT, PROC MEANS/PROC SUMMARY and PROC STANDARD, each requires suppressing default BY lines by setting NOBYLINE system option.

Set system option BYLINE to resume default by-group processing.
Inserting By-Line Information

**TITLE**n 'text-before #BY-spec<.suffix> text-after';

- **BY-spec** is one of
  - BYVALn or BYVAL(By-variable) places value of n-th or By-variable by-variable in title.
  - BYVARn or BYVAR(By-variable) places label/name of n-th or By-variable by-variable in title.
  - BYLINE places complete default by-line into title.

- **suffix** places text immediately following BY-group info. (Note: Period (.), following #BY-spec, works as a delimiter.)

See example report1.sas.
Inserting By-Line Information

**TITLE**n 'text-before #BY-spec<.suffix> text-after';

- **BY-spec** is one of:
  1. **BYVAL**n or **BYVAL**(By-variable) places value of n\textsuperscript{th} or **By-variable** by-variable in title.
  2. **BYVAR**n or **BYVAR**(By-variable) places label/name of n\textsuperscript{th} or **By-variable** by-variable in title.
  3. **BYLINE** places complete default by-line into title.

- **suffix** places text immediately following BY-group info. (Note: Period (.), following #BY-spec, works as a delimiter.)

See example report1.sas.
Inserting By-Line Information

\textbf{TITLEn} 'text-before \#BY-spec<.suffix> text-after';

- \textbf{BY-spec} is one of
  1. \texttt{BYVALn} or \texttt{BYVAL(By-variable)} places value of \textit{n}th or \textit{By-variable} by-variable in title.
  2. \texttt{BYVARn} or \texttt{BYVAR(By-variable)} places label/name of \textit{n}th or \textit{By-variable} by-variable in title.
  3. \texttt{BYLINE} places complete default by-line into title.

- \texttt{suffix} places text immediately following BY-group info. (Note: Period (.), following \#BY-spec, works as a delimiter.)

See example \texttt{report1.sas}.
Outline

1. Use of Byline Information in Title Lines
   - By-Group Processing

2. PROC Report
   - Use of PROC Report
   - PROC REPORT Statement
   - Selected Statements Used in PROC Report

3. PROC Means
   - Comparing PROC Means and PROC Summary
   - Understanding the Use of Variables in PROC Means
   - Available Statistics in PROC Means
   - Creating SAS Data Set with PROC Means

4. PROC Freq
   - The Use of PROC Freq
   - PROC FREQ Statement
   - Statements Used in PROC Freq
What PROC Report Does

A report writing tool combining features from

- DATA step
- PROC Print, PROC Means, and PROC Tabulate

which can be used to generate

1. listing reports
2. summary reports
What PROC Report Does

A report writing tool combining features from

- DATA step
- PROC Print, PROC Means, and PROC Tabulate

which can be used to generate

1. listing reports
2. summary reports
PROC Report Output

- Under windowing environment, output goes to Report window by default.
- Can use NOWINDOWS (or NOWD) option in PROC REPORT statement to redirect output to OUTPUT window.
PROC Report Output

- Under windowing environment, output goes to Report window by default.
- Can use NOWINDOWS (or NOWD) option in PROC REPORT statement to redirect output to OUTPUT window.
PROC Report Output

- Under windowing environment, output goes to Report window by default.
- Can use NOWINDOWS (or NOWD) option in PROC REPORT statement to redirect output to OUTPUT window.
PROC REPORT statement

PROC REPORT <option(s)>;

Selected options

- I/O: DATA=, OUT= (output data for further processing)
- LineSize/PageSize: LS=, PS=
- NOWD
- Surrounding box: BOX, FORMCHAR=
- Centering: CENTER | NOCENTER
- Column spacing: SPACING=
- Allowing missing value: MISSING
- Multi-panel report: PANELS=, PSPACE=
- Column-header appearance: HEADLINE, HEADSKIP, NOHEADER, SPLIT=
PROC REPORT statement

PROC REPORT <option(s)>;

Selected options

- I/O: DATA=, OUT= (output data for further processing)
- LineSize/PageSize: LS=, PS=
- NOWD
- Surrounding box: BOX, FORMCHAR=
- Centering: CENTER | NOCENTER
- Column spacing: SPACING=
- Allowing missing value: MISSING
- Multi-panel report: PANELS=, PSPACE=
- Column-header appearance: HEADLINE, HEADSKIP, NOHEADER, SPLIT=
Selected Statements Used in PROC Report

- COLUMN statement: describes columns and headers
- DEFINE statement: describes how to use or display a report item
- COMPUTE statement and ENDCOMP statement: contain programming statement(s) that builds report
- BREAK statement: produces summary at a break
- RBREAK statement: produces a default summary at the beginning/end of a report or at the beginning/end of each BY-group
- LINE statement: valid only in COMPUTE blocks, it provides a subset of PUT statement’s features to customize the report
- BY statement: BY-group processing
Selected Statements Used in PROC Report

- **COLUMN statement**: describes columns and headers
- **DEFINE statement**: describes how to use or display a report item
- **COMPUTE statement and ENDCOMP statement**: contain programming statement(s) that builds report
- **BREAK statement**: produces summary at a break
- **RBREAK statement**: produces a default summary at the beginning/end of a report or at the beginning/end of each BY-group
- **LINE statement**: valid only in COMPUTE blocks, it provides a subset of PUT statement’s features to customize the report
- **BY statement**: BY-group processing
Selected Statements Used in PROC Report

- COLUMN statement: describes columns and headers
- DEFINE statement: describes how to use or display a report item
- COMPUTE statement and ENDCOMP statement: contain programming statement(s) that builds report
- BREAK statement: produces summary at a break
- RBREAK statement: produces a default summary at the beginning/end of a report or at the beginning/end of each BY-group
- LINE statement: valid only in COMPUTE blocks, it provides a subset of PUT statement’s features to customize the report
- BY statement: BY-group processing
Selected Statements Used in PROC Report

- COLUMN statement: describes columns and headers
- DEFINE statement: describes how to use or display a report item
- COMPUTE statement and ENDCOMP statement: contain programming statement(s) that builds report
- BREAK statement: produces summary at a break
- RBREAK statement: produces a default summary at the beginning/end of a report or at the beginning/end of each BY-group
- LINE statement: valid only in COMPUTE blocks, it provides a subset of PUT statement’s features to customize the report
- BY statement: BY-group processing
Selected Statements Used in PROC Report

- COLUMN statement: describes columns and headers
- DEFINE statement: describes how to use or display a report item
- COMPUTE statement and ENDCOMP statement: contain programming statement(s) that builds report
- BREAK statement: produces summary at a break
- RBREAK statement: produces a default summary at the beginning/end of a report or at the beginning/end of each BY-group
- LINE statement: valid only in COMPUTE blocks, it provides a subset of PUT statement’s features to customize the report
- BY statement: BY-group processing
Selected Statements Used in PROC Report

- COLUMN statement: describes columns and headers
- DEFINE statement: describes how to use or display a report item
- COMPUTE statement and ENDCOMP statement: contain programming statement(s) that builds report
- BREAK statement: produces summary at a break
- RBREAK statement: produces a default summary at the beginning/end of a report or at the beginning/end of each BY-group
- LINE statement: valid only in COMPUTE blocks, it provides a subset of PUT statement’s features to customize the report
- BY statement: BY-group processing
Selected Statements Used in PROC Report

- COLUMN statement: describes columns and headers
- DEFINE statement: describes how to use or display a report item
- COMPUTE statement and ENDCOMP statement: contain programming statement(s) that builds report
- BREAK statement: produces summary at a break
- RBREAK statement: produces a default summary at the beginning/end of a report or at the beginning/end of each BY-group
- LINE statement: valid only in COMPUTE blocks, it provides a subset of PUT statement’s features to customize the report
- BY statement: BY-group processing
Selected Statements Used in PROC Report

- COLUMN statement: describes columns and headers
- DEFINE statement: describes how to use or display a report item
- COMPUTE statement and ENDCOMP statement: contain programming statement(s) that builds report
- BREAK statement: produces summary at a break
- RBREAK statement: produces a default summary at the beginning/end of a report or at the beginning/end of each BY-group
- LINE statement: valid only in COMPUTE blocks, it provides a subset of PUT statement’s features to customize the report
- BY statement: BY-group processing
COLUMN Statement

COLUMN  column-specification(s);
where  column-specification is one of the following

- report-item: identifies an item that forms a column, where report-item is a data set variable, a computed variable, or a statistic
- report-item-1, report-item-2 <,..., report-item-n>: identifies stacked items
- (’header-1’ <... ’header-n’> report-item(s)): creates one/more headers spanning multiple items (columns)
- report-item=name: specifies an alias for a report item
COLUMN Statement

COLUMN column-specification(s);
where column-specification is one of the following

- report-item: identifies an item that forms a column, where report-item is a data set variable, a computed variable, or a statistic
- report-item-1, report-item-2 <..., report-item-n>: identifies stacked items
- ('header-1' <... 'header-n'> report-item(s)): creates one/more headers spanning multiple items (columns)
- report-item=name: specifies an alias for a report item
Available Statistics in PROC Report

- Descriptive Statistics: MIN, MAX, RANGE; MEAN, VAR, STDDEV|STD, STDERR, CV; SUM, SUMWGT, CSS, USS; N, NMISS; PCTN, PCTSUM.
- Quantiles: P1, P5, P10, Q1|P25, MEDIAN|P50, Q3|P75, P90, P95, and P99; QRANGE.
- Test Statistics: T and PROBT.
Available Statistics in PROC Report

- **Descriptive Statistics:** MIN, MAX, RANGE; MEAN, VAR, STDDEV|STD, STDERR, CV; SUM, SUMWGT, CSS, USS; N, NMISS; PCTN, PCTSUM.
- **Quantiles:** P1, P5, P10, Q1|P25, MEDIAN|P50, Q3|P75, P90, P95, and P99; QRANGE.
- **Test Statistics:** T and PROBT.
Available Statistics in PROC Report

- **Descriptive Statistics**: MIN, MAX, RANGE; MEAN, VAR, STDDEV|STD, STDERR, CV; SUM, SUMWGT, CSS, USS; N, NMISS; PCTN, PCTSUM.

- **Quantiles**: P1, P5, P10, Q1|P25, MEDIAN|P50, Q3|P75, P90, P95, and P99; QRANGE.

- **Test Statistics**: T and PROBT.
Available Statistics in PROC Report

- Descriptive Statistics: MIN, MAX, RANGE; MEAN, VAR, STDDEV|STD, STDERR, CV; SUM, SUMWGT, CSS, USS; N, NMISS; PCTN, PCTSUM.
- Quantiles: P1, P5, P10, Q1|P25, MEDIAN|P50, Q3|P75, P90, P95, and P99; QRANGE.
- Test Statistics: T and PROBT.
DEFINE Statement

DEFINE report-item / <option(s)>;

- use of items (specified in option): ANALYSIS (default for data set numeric variables), DISPLAY (default for data set character variables); ORDER; GROUP, ACROSS; COMPUTED.

- Selected item appearance options: SPACING=, WIDTH=; FORMAT=, ORDER=; MISSING; statistic.

- Selected report item options: DESCENDING, FLOW, NOPRINT, NOZERO, PAGE.

- Values/column heading placement: CENTER, LEFT, RIGHT, 'column-header'.
DEFINE Statement

DEFINE report-item / <option(s)>;

- use of items (specified in option): ANALYSIS (default for data set numeric variables), DISPLAY (default for data set character variables); ORDER; GROUP, ACROSS; COMPUTED.

- Selected item appearance options: SPACING=, WIDTH=; FORMAT=, ORDER=; MISSING; statistic.

- Selected report item options: DESCENDING, FLOW, NOPRINT, NOZERO, PAGE.

- Values/column heading placement: CENTER, LEFT, RIGHT, 'column-header'.
DEFINE Statement

DEFINE  report-item / <option(s)>;

- use of items (specified in option): ANALYSIS (default for data set numeric variables), DISPLAY (default for data set character variables); ORDER; GROUP, ACROSS; COMPUTED.

- Selected item appearance options: SPACING=, WIDTH=; FORMAT=, ORDER=; MISSING; statistic.

- Selected report item options: DESCENDING, FLOW, NOPRINT, NOZERO, PAGE.

- Values/column heading placement: CENTER, LEFT, RIGHT, 'column-header'.
DEFINE Statement

DEFINE report-item / <option(s)>;

- use of items (specified in option): ANALYSIS (default for data set numeric variables), DISPLAY (default for data set character variables); ORDER; GROUP, ACROSS; COMPUTED.

- Selected item appearance options: SPACING=, WIDTH=; FORMAT=, ORDER=; MISSING; statistic.

- Selected report item options: DESCENDING, FLOW, NOPRINT, NOZERO, PAGE.

- Values/column heading placement: CENTER, LEFT, RIGHT, 'column-header'.
DEFINE Statement

```
DEFINE report-item / <option(s)>;
```

- Use of items (specified in `option`): ANALYSIS (default for data set numeric variables), DISPLAY (default for data set character variables); ORDER; GROUP, ACROSS; COMPUTED.

- Selected item appearance options: SPACING=, WIDTH=; FORMAT=, ORDER=; MISSING; statistic.

- Selected report item options: DESCENDING, FLOW, NOPRINT, NOZERO, PAGE.

- Values/column heading placement: CENTER, LEFT, RIGHT, `column-header`.
Leading/Ending Characters in Headers

Use of one of the following characters as the leading character and as the ending character in a header: repeats itself so the header spans over the multiple items.

+ − * \ _ : . =

Use of the pair < > or > < for leading-ending character pair are likewise.

See example report1.sas.
Outline

1. Use of Byline Information in Title Lines
   - By-Group Processing

2. PROC Report
   - Use of PROC Report
   - PROC REPORT Statement
   - Selected Statements Used in PROC Report

3. PROC Means
   - Comparing PROC Means and PROC Summary
   - Understanding the Use of Variables in PROC Means
   - Available Statistics in PROC Means
   - Creating SAS Data Set with PROC Means

4. PROC Freq
   - The Use of PROC Freq
   - PROC FREQ Statement
   - Statements Used in PROC Freq
PROC Means Versus PROC Summary

- They both produce descriptive statistics over all observations or within groups.
- They are virtually the same except for default action.
- PROC Means places output in output window by default while PROC Summary produces output data set by default.
PROC Means Versus PROC Summary

- They both produce descriptive statistics over all observations or within groups.
- They are virtually the same except for default action.
- PROC Means places output in output window by default while PROC Summary produces output data set by default.
PROC Means Versus PROC Summary

- They both produce descriptive statistics over all observations or within groups.
- They are virtually the same except for default action.
- PROC Means places output in output window by default while PROC Summary produces output data set by default.
PROC Means Versus PROC Summary

- They both produce descriptive statistics over all observations or within groups.
- They are virtually the same except for default action.
- PROC Means places output in output window by default while PROC Summary produces output data set by default.
Getting Started with PROC Means

Two types of variables

- **Analysis variables**: numeric variables that you want analyses. Analysis variables are declared in the `VAR` statement.

- **Classification variables**: numeric or character variables that you want by-group analyses. Classification variables are declared with a `BY` statement, or with one or more `CLASS` statements.
Getting Started with PROC Means

Two types of variables

- **Analysis variables**: numeric variables that you want analyses. Analysis variables are declared in `VAR` statement.

- **Classification variables**: numeric or character variables that you want by-group analyses. Classification variables are declared with a `BY` statement, or with one or more `CLASS` statements.
Getting Started with PROC Means

Two types of variables

- **Analysis variables**: numeric variables that you want analyses. Analysis variables are declared in `VAR` statement.

- **Classification variables**: numeric or character variables that you want by-group analyses. Classification variables are declared with a `BY` statement, or with one or more `CLASS` statements.
Using Classification Variables

- **BY <DESCENDING> variable ... <NOTSORTED>;**
  Observations need to be sorted or indexed according to the variables listed in BY statement unless NOTSORTED is requested.

- **CLASS variable(s) </options>;;**
  Observations are not required to be sorted or indexed. Unique combinations of variables’ levels are computed. *Options*, if specified, are applied to all variables in the list. Use multiple CLASS statements if you want different options for classification variables.

- **BY statement is preferred for large data set and classification variable with large number of levels.**

- Can use **BY statement in combination with CLASS statements.**
Using Classification Variables

- **BY** `<DESCENDING>` variable ... `<NOTSORTED>`;
  Observations need to be sorted or indexed according to the variables listed in BY statement unless `NOTSORTED` is requested.

- **CLASS** variable(s) `</options>`;
  Observations are not required to be sorted or indexed. Unique combinations of variables’ levels are computed. *Options*, if specified, are applied to all variables in the list. Use multiple **CLASS** statements if you want different options for classification variables.

- **BY** statement is preferred for large data set and classification variable with large number of levels.

- Can use **BY** statement in combination with **CLASS** statements.
Using Classification Variables

- **BY <DESCENDING> variable ... <NOTSORTED>;**
  Observations need to be sorted or indexed according to the variables listed in BY statement unless NOTSORTED is requested.

- **CLASS variable(s) <\options>;**
  Observations are not required to be sorted or indexed. Unique combinations of variables’ levels are computed. Options, if specified, are applied to all variables in the list. Use multiple CLASS statements if you want different options for classification variables.

- **BY statement is preferred for large data set and classification variable with large number of levels.**

- **Can use BY statement in combination with CLASS statements.**
Using Classification Variables

- **BY** `<DESCENDING>` variable ... `<NOTSORTED>`;

  Observations need to be sorted or indexed according to the variables listed in BY statement unless `NOTSORTED` is requested.

- **CLASS** `variable(s)` `</ options>`;

  Observations are not required to be sorted or indexed. Unique combinations of variables’ levels are computed. *Options*, if specified, are applied to all variables in the list. Use multiple **CLASS** statements if you want different options for classification variables.

- **BY** statement is preferred for large data set and classification variable with large number of levels.

- Can use **BY** statement in combination with **CLASS** statements.
Available Statistics in PROC Means

PROC MEANS <option(s)> <statistic(s)>;

- Descriptive statistics: N, NMISS; MIN, MAX, RANGE; MEAN, VAR, STDDEV|STD, SKEWNESS|SKEW, KURTOSIS|KURT, CV; STDERR, CLM, LCLM, UCLM; SUM, SUMWGT, CSS, USS.
- Quantiles: P1, P5, P10, Q1|P25, MEDIAN|P50, Q3|P75, P90, P95, and P99; QRANGE.
- Test Statistics: T and PROBT.
- Default set of statistics: N, MEAN, STD, MIN, and MAX.
Available Statistics in PROC MEANS

PROC MEANS <option(s)> <statistic(s)>;

- Descriptive statistics: N, NMISS; MIN, MAX, RANGE; MEAN, VAR, STDDEV|STD, SKEWNESS|SKEW, KURTOSIS|KURT, CV; STDERR, CLM, LCLM, UCLM; SUM, SUMWGT, CSS, USS.
- Quantiles: P1, P5, P10, Q1|P25, MEDIAN|P50, Q3|P75, P90, P95, and P99; QRANGE.
- Test Statistics: T and PROBT.
- Default set of statistics: N, MEAN, STD, MIN, and MAX.
Available Statistics in PROC Means

PROC MEANS <option(s)> <statistic(s)>;

- Descriptive statistics: N, NMISS; MIN, MAX, RANGE; MEAN, VAR, STDDEV|STD, SKEWNESS|SKEW, KURTOSIS|KURT, CV; STDERR, CLM, LCLM, UCLM; SUM, SUMWGT, CSS, USS.
- Quantiles: P1, P5, P10, Q1|P25, MEDIAN|P50, Q3|P75, P90, P95, and P99; QRANGE.
- Test Statistics: T and PROBT.
- Default set of statistics: N, MEAN, STD, MIN, and MAX.
Available Statistics in PROC Means

PROC MEANS <option(s)> <statistic(s)>;

- Descriptive statistics: N, NMISS; MIN, MAX, RANGE; MEAN, VAR, STDDEV|STD, SKEWNESS|SKEW, KURTOSIS|KURT, CV; STDERR, CLM, LCLM, UCLM; SUM, SUMWGT, CSS, USS.
- Quantiles: P1, P5, P10, Q1|P25, MEDIAN|P50, Q3|P75, P90, P95, and P99; Q RANGE.
- Test Statistics: T and PROBT.
- Default set of statistics: N, MEAN, STD, MIN, and MAX.
Available Statistics in PROC Means

PROC MEANS <option(s)> <statistic(s)>;

- Descriptive statistics: N, NMISS; MIN, MAX, RANGE; MEAN, VAR, STDDEV|STD, SKEWNESS|SKEW, KURTOSIS|KURT, CV; STDERR, CLM, LCLM, UCLM; SUM, SUMWGT, CSS, USS.
- Quantiles: P1, P5, P10, Q1|P25, MEDIAN|P50, Q3|P75, P90, P95, and P99; QRANGE.
- Test Statistics: T and PROBT.
- Default set of statistics: N, MEAN, STD, MIN, and MAX.
Creating SAS Data Set with PROC Means

- Use **OUTPUT** statements to create SAS data sets.
- Within same **OUTPUT** statement, need to specify what statistics needed.
- Can request different statistics for different variables listed in **VAR** statement.
- In an **OUTPUT** statement, it’s good practice to name output data set variables or use **AUTONAME** option.
Creating SAS Data Set with PROC Means

- **Use** `OUTPUT` **statements** to create SAS data sets.
  - Within same `OUTPUT` statement, need to specify what statistics needed.
  - Can request different statistics for different variables listed in `VAR` statement.
  - In an `OUTPUT` statement, it’s good practice to name output data set variables or use `AUTONAME` option.
Creating SAS Data Set with PROC Means

- Use **OUTPUT** statements to create SAS data sets.
- Within same **OUTPUT** statement, need to specify what statistics needed.
- Can request different statistics for different variables listed in **VAR** statement.
- In an **OUTPUT** statement, it’s good practice to name output data set variables or use **AUTONAME** option.
Creating SAS Data Set with PROC Means

- Use **OUTPUT** statements to create SAS data sets.
- Within same **OUTPUT** statement, need to specify what statistics needed.
- Can request different statistics for different variables listed in **VAR** statement.
- In an **OUTPUT** statement, it’s good practice to name output data set variables or use **AUTONAME** option.
Creating SAS Data Set with PROC Means

- Use `OUTPUT` statements to create SAS data sets.
- Within same `OUTPUT` statement, need to specify what statistics needed.
- Can request different statistics for different variables listed in `VAR` statement.
- In an `OUTPUT` statement, it’s good practice to name output data set variables or use `AUTONAME` option.
_TYPE_ & _FREQ_ Automatic Variables in Output Data Set

- **_TYPE_** contains information about class variables:
  - _TYPE_ is numeric by default if 32 or less classification variables are specified, and character otherwise.
  - If CHARTYPE option is specified in PROC MEANS statement then _TYPE_ is character.
  - The character value of _TYPE_ is the binary representation of the numeric values. For instance, if three classification variables are specified in this order: A then B then C, then '011' (value of 3) is the subgroup represented by B*C; '101' (value of 5) is for A*C; '100' (value of 4) is for A, etc.

- **_FREQ_** contains the number of observations that a given output level represents.
_TYPE_ & _FREQ_ Automatic Variables
in Output Data Set

- **_TYPE_** contains information about class variables
  1. _TYPE_ is numeric by default if 32 or less classification variables are specified, and character otherwise.
  2. If CHARTYPE option is specified in PROC MEANS statement then _TYPE_ is character.
  3. The character value of _TYPE_ is the binary representation of the numeric values. For instances, if three classification variables are specified in this order: A then B then C, then ‘011’ (value of 3) is the subgroup represented by B*C; ’101’ (value of 5) is for A*C; ’100’ (value of 4) is for A; etc.

- **_FREQ_** contains the number of observations that a given output level represents.
**_TYPE_ & _FREQ_ Automatic Variables**

in Output Data Set

- **_TYPE_** contains information about class variables
  1. **_TYPE_** is numeric by default if 32 or less classification variables are specified, and character otherwise.
  2. If CHARTYPE option is specified in PROC MEANS statement then **_TYPE_** is character.
  3. The character value of **_TYPE_** is the binary representation of the numeric values. For instances, if three classification variables are specified in this order: A then B then C, then ’011’ (value of 3) is the subgroup represented by B*C; ’101’ (value of 5) is for A*C; ’100’ (value of 4) is for A; etc.

- **_FREQ_** contains the number of observations that a given output level represents.
_TYPE_ & _FREQ_ Automatic Variables

in Output Data Set

- _TYPE_ contains information about class variables
  1. _TYPE_ is numeric by default if 32 or less classification variables are specified, and character otherwise.
  2. If CHARTYPE option is specified in PROC MEANS statement then _TYPE_ is character.
  3. The character value of _TYPE_ is the binary representation of the numeric values. For instances, if three classification variables are specified in this order: A then B then C, then ‘011’ (value of 3) is the subgroup represented by B*C; ‘101’ (value of 5) is for A*C; ‘100’ (value of 4) is for A; etc.

- _FREQ_ contains the number of observations that a given output level represents.
_TYPE_ & _FREQ_ Automatic Variables
in Output Data Set

(TYPE_ contains information about class variables
1 _TYPE_ is numeric by default if 32 or less classification variables are specified, and character otherwise.
2 if CHARTYPE option is specified in PROC MEANS statement then _TYPE_ is character.
3 the character value of _TYPE_ is the binary representation of the numeric values. For instances, if three classification variables are specified in this order: A then B then C, then ’011’ (value of 3) is the subgroup represented by B*C; ’101’ (value of 5) is for A*C; ’100’ (value of 4) is for A; etc.

_FREQ_ contains the number of observations that a given output level represents.
_TYPE_ & _FREQ_ Automatic Variables
in Output Data Set

- **_TYPE_** contains information about class variables
  1. **_TYPE_** is numeric by default if 32 or less classification variables are specified, and character otherwise.
  2. If CHARTYPE option is specified in PROC MEANS statement then **_TYPE_** is character.
  3. The character value of **_TYPE_** is the binary representation of the numeric values. For instances, if three classification variables are specified in this order: A then B then C, then '011' (value of 3) is the subgroup represented by B*C; '101' (value of 5) is for A*C; '100' (value of 4) is for A; etc.

- **_FREQ_** contains the number of observations that a given output level represents.
WAYS and TYPES Statements

- Both statements are used to select a subsets of class combinations.

  - TYPES request(s);
    For instances, TYPES (A B) *C;,
    TYPES (A B) *(C D);,
    TYPES ();.

  - WAYS list;
    For instances, WAYS 0 TO 2;,
    WAYS 1,3;.

- Output data are arranged in ascending order of _TYPE_ unless otherwise DESCENDINGTYPES option is specified in PROC MEANS statement.
WAYS and TYPES Statements

- Both statements are used to select subsets of class combinations.
  - TYPES request(s);
    For instances, TYPES (A B) *C;,
    TYPES (A B) *(C D);,
    TYPES ();.
  - WAYS list;
    For instances, WAYS 0 TO 2;,
    WAYS 1,3;.
  - Output data are arranged in ascending order of _TYPE_ unless otherwise DESCENDTYPES option is specified in PROC MEANS statement.
WAYS and TYPES Statements

- Both statements are used to select a subsets of class combinations.
  
  - **TYPES** request(s);
  
    For instances, TYPES (A B) *C;,
    TYPES (A B) *(C D);,
    TYPES ();.

- **WAYS** list;
  
    For instances, WAYS 0 TO 2;,
    WAYS 1,3;.

- Output data are arranged in ascending order of _TYPE_ unless otherwise DESCENDTYPES option is specified in PROC MEANS statement.
WAYS and TYPES Statements

- Both statements are used to select a subsets of class combinations.

- **TYPES request(s);**
  - For instances, TYPES (A B) *C;,
  - TYPES (A B) *(C D);,
  - TYPES ();.

- **WAYS list;**
  - For instances, WAYS 0 TO 2;;
  - WAYS 1,3;;

- Output data are arranged in ascending order of _TYPE_ unless otherwise DESCENDTYPES option is specified in PROC MEANS statement.
WAYS and TYPES Statements

- Both statements are used to select a subsets of class combinations.

- TYPES request(s);
  For instances, TYPES (A B) *C;,
  TYPES (A B) *(C D);,
  TYPES ();.

- WAYS list;
  For instances, WAYS 0 TO 2;,
  WAYS 1,3;.

- Output data are arranged in ascending order of _TYPE_ unless otherwise DESCENDTYPES option is specified in PROC MEANS statement.
Outline

1. Use of Byline Information in Title Lines
   - By-Group Processing

2. PROC Report
   - Use of PROC Report
   - PROC REPORT Statement
   - Selected Statements Used in PROC Report

3. PROC Means
   - Comparing PROC Means and PROC Summary
   - Understanding the Use of Variables in PROC Means
   - Available Statistics in PROC Means
   - Creating SAS Data Set with PROC Means

4. PROC Freq
   - The Use of PROC Freq
   - PROC FREQ Statement
   - Statements Used in PROC Freq
What PROC Freq Does

- Produce one-way frequency tables, by default, of all variables. Statistics can be computed to test for equal proportions, specified proportions, or the binomial proportion.

- Can produce multi-way frequency and crosstabulation (contingency) tables. Statistics can be computed for the examination of the relationships between two classification variables adjusting for any stratification variables.
What PROC Freq Does

- Produce one-way frequency tables, by default, of all variables. Statistics can be computed to test for equal proportions, specified proportions, or the binomial proportion.

- Can produce multi-way frequency and crosstabulation (contingency) tables. Statistics can be computed for the examination of the relationships between two classification variables adjusting for any stratification variables.
What PROC Freq Does

- Produce one-way frequency tables, by default, of all variables. Statistics can be computed to test for equal proportions, specified proportions, or the binomial proportion.

- Can produce multi-way frequency and crosstabulation (contingency) tables. Statistics can be computed for the examination of the relationships between two classification variables adjusting for any stratification variables.
PROC FREQ Statement

PROC FREQ \(<options>\);

Options include

- DATA=: specifies the input data set.
- COMPRESS: begins next one-way table on current page.
- FORMCHAR=: specifies outline and cell divider characters in crosstabulation table.
- NLEVELS: displays the number of levels for all TABLES variables.
- NOPRINT: suppresses all displayed output.
- ORDER=: specifies the order for listing variable values.
- PAGE: displays one table per page.
PROC FREQ Statement

PROC FREQ <options>;

Options include

- \texttt{DATA=} : specifies the input data set.
- \texttt{COMPRESS} : begins next one-way table on current page.
- \texttt{FORMCHAR=} : specifies outline and cell divider characters in crosstabulation table.
- \texttt{NLEVELS} : displays the number of levels for all TABLES variables.
- \texttt{NOPRINT} : suppresses all displayed output.
- \texttt{ORDER=} : specifies the order for listing variable values.
- \texttt{PAGE} : displays one table per page.
PROC FREQ Statement

PROC FREQ <options>;
Options include

- DATA=: specifies the input data set.
- COMPRESS: begins next one-way table on current page.
- FORMCHAR=: specifies outline and cell divider characters in crosstabulation table.
- NLEVELS: displays the number of levels for all TABLES variables.
- NOPRINT: suppresses all displayed output.
- ORDER=: specifies the order for listing variable values.
- PAGE: displays one table per page.
PROC FREQ Statement

PROC FREQ <options>;

Options include

- DATA=: specifies the input data set.
- COMPRESS: begins next one-way table on current page.
- FORMCHAR=: specifies outline and cell divider characters in crosstabulation table.
- NLEVELS: displays the number of levels for all TABLES variables.
- NOPRINT: suppresses all displayed output.
- ORDER=: specifies the order for listing variable values.
- PAGE: displays one table per page.
PROC FREQ Statement

PROC FREQ <options>;
Options include

- DATA=: specifies the input data set.
- COMPRESS: begins next one-way table on current page.
- FORMCHAR=: specifies outline and cell divider characters in crosstabulation table.
- NLEVELS: displays the number of levels for all TABLES variables.
- NOPRINT: suppresses all displayed output.
- ORDER=: specifies the order for listing variable values.
- PAGE: displays one table per page.
PROC FREQ Statement

PROC FREQ <options>;
Options include

- DATA=: specifies the input data set.
- COMPRESS: begins next one-way table on current page.
- FORMCHAR=: specifies outline and cell divider characters in crosstabulation table.
- NLEVELS: displays the number of levels for all TABLES variables.
- NOPRINT: suppresses all displayed output.
- ORDER=: specifies the order for listing variable values.
- PAGE: displays one table per page.
PROC FREQ Statement

PROC FREQ <options>;

Options include

- DATA=: specifies the input data set.
- COMPRESS: begins next one-way table on current page.
- FORMCHAR=: specifies outline and cell divider characters in crosstabulation table.
- NLEVELS: displays the number of levels for all TABLES variables.
- NOPRINT: suppresses all displayed output.
- ORDER=: specifies the order for listing variable values.
- PAGE: displays one table per page.
PROC FREQ Statement

PROC FREQ <options>;
Options include

- DATA=: specifies the input data set.
- COMPRESS: begins next one-way table on current page.
- FORMCHAR=: specifies outline and cell divider characters in crosstabulation table.
- NLEVELS: displays the number of levels for all TABLES variables.
- NOPRINT: suppresses all displayed output.
- ORDER=: specifies the order for listing variable values.
- PAGE: displays one table per page.
Statements Used in PROC Freq

- **BY** statement: By-group processing.
- **TABLES** statement: requests tables and statistics.
- **TEST** statement: requests asymptotic tests for the specified measures of association and measures of agreement.
- **EXACT** statement: requests exact tests or confidence limits for the specified statistics.
- **OUTPUT** statement: creates a SAS data set containing statistics computed by PROC Freq.
- **WEIGHT** statement: specifies a numeric variable with a value that represents the frequency of the observation.
Statements Used in PROC Freq

- **BY** statement: By-group processing.
- **TABLES** statement: requests tables and statistics.
- **TEST** statement: requests asymptotic tests for the specified measures of association and measures of agreement.
- **EXACT** statement: requests exact tests or confidence limits for the specified statistics.
- **OUTPUT** statement: creates a SAS data set containing statistics computed by PROC Freq.
- **WEIGHT** statement: specifies a numeric variable with a value that represents the frequency of the observation.
TABLES Statement

`TABLES requests < / options >;`

where requests are in the form

- variable by itself requests one-way table
- `variable-1*variable-2*... variable-n` requests n-way table; the last two variables in the request constitute row and column while others define strata.
- Can use parentheses for grouping. For instance, `TABLES A*(B C);`, and `TABLES (A B)*(C D);`.
- Can use variable-list shortcuts. For instance, `TABLES A * (B-D) * (E1-E3);`.
TABLES Statement

TABLES requests < / options >;
where requests are in the form

- variable by itself requests one-way table

- variable-1*variable-2*... variable-n requests n-way table; the last two variables in the request constitute row and column while others define strata.

- Can use parentheses for grouping. For instances, TABLES A*(B C);, and
TABLES (A B)*(C D);.

- Can use variable-list shortcuts. For instance, TABLES A * (B-D) * (E1-E3);.
TABLES Statement

TABLES requests < / options >;
where requests are in the form

- variable by itself requests one-way table
- variable-1*variable-2*... variable-n requests n-way table; the last two variables in the request constitute row and column while others define strata.

- Can use parentheses for grouping. For instances, TABLES A*(B C);, and
  TABLES (A B)*(C D);
- Can use variable-list shortcuts. For instance, TABLES A *(B-D) * (E1-E3);.
TABLES Statement

TABLES requests < / options >;
where requests are in the form

- variable by itself requests one-way table
- variable-1*variable-2*... variable-n requests n-way table; the last two variables in the request constitute row and column while others define strata.
- Can use parentheses for grouping. For instances, TABLES A*(B C);, and
  TABLES (A B)*(C D);.
- Can use variable-list shortcuts. For instance, TABLES A * (B-D) * (E1-E3);.
TABLES Statement

TABLES requests < / options >;
where requests are in the form

- variable by itself requests one-way table
- variable-1*variable-2*... variable-n requests n-way table; the last two variables in the request constitute row and column while others define strata.

Can use parentheses for grouping. For instances, TABLES A * (B C);, and
TABLES (A B) * (C D);.

Can use variable-list shortcuts. For instance, TABLES A * (B-D) * (E1-E3);.
OUTPUT Statement

OUTPUT <OUT= SAS-data-set> options ;

- Only one OUTPUT statement is allowed.
- A TABLES statement is required to go with the OUTPUT statement.
- If multiple TABLES statements are specified, the contents of the OUTPUT data set correspond to the last TABLES statement.
- If multiple tables are requested in a TABLES statement, the contents of the OUTPUT data set correspond to the last table request.
OUTPUT Statement

OUTPUT <OUT= SAS-data-set> options ;

- Only one OUTPUT statement is allowed.
- A TABLES statement is required to go with the OUTPUT statement.
- If multiple TABLES statements are specified, the contents of the OUTPUT data set correspond to the last TABLES statement.
- If multiple tables are requested in a TABLES statement, the contents of the OUTPUT data set correspond to the last table request.