# STAT 425 and STAT 625 Statistical Software

## Lecture 5

## Creating Labels and Formats

Saida Zardi

# Creating Labels

You can enhance your PROC PRINT report by labeling columns with more descriptive text.

To label columns:

Use the LABEL statement to assign a descriptive label to a variable.

Use the LABEL option in the PROC PRINT statement to specify that the labels be displayed.

Labels can be up to 256 characters long and are enclosed in quotation marks.

Tip:The LABEL statement applies only to the PROC step in which it appears.

**Syntax, LABEL statement:**

LABEL variable1='label1'
variable2='label2'
... ;

Note: you can assign a single or multiple Label Statements:

LABEL variable1='label1';
LABEL variable2='label2';

```
Libname Lec5 'J:\CLASSES\STAT46';

DATA Lec5.Stocks;
   infile 'J:\CLASSES\STAT46\stockprices.txt';
   input @1 Stock $ 4.
         @5 PurDate mmddyy10.
         @15 PurPrice  comma8.
         @21 Number 4.;
   label Stock='Name of the Stock'
         PurDate='Purchase Date'
         PurPrice= 'Purchase Price'
         Number = 'Number of shares';
   format PurDate date9.
          PurPrice dollar8.2;

run;
title 'The Data Portion of Stocks.sas7bdat';
proc print data=Lec5.Stocks;
run;

title 'The Descriptor portion of Stocks.sas7bdat';
proc contents data=Lec5.Stocks varnum;
run;

title 'Summary Statistics';
proc means data=Lec5.Stocks;
run;
```

# Log window:

```
31            @5 PurDate mmddyy10.
32            @15 PurPrice  comma8.
33            @21 Number 4.;
34       label Stock='Name of the Stock'
35             PurDate='Purchase Date'
36             PurPrice= 'Purchase Price'
37             Number = 'Number of shares';
38       format PurDate date9.
39              PurPrice dollar8.2;
40
41    run;

NOTE: The infile 'J:\CLASSES\STAT46\stockprices.txt' is:
      Filename=J:\CLASSES\STAT46\stockprices.txt,
      RECFM=V,LRECL=32767,File Size (bytes)=208,
      Last Modified=13Sep2018:22:32:28,
      Create Time=13Sep2018:21:03:15

NOTE: 5 records were read from the infile 'J:\CLASSES\STAT46\stockprices.txt'.
      The minimum record length was 40.
      The maximum record length was 40.
NOTE: The data set LEC5.STOCKS has 5 observations and 4 variables.
NOTE: DATA statement used (Total process time):
      real time           0.11 seconds
      cpu time            0.04 seconds


42    title 'The Data Portion of Stocks.sas7bdat';
43    proc print data=Lec5.Stocks;
NOTE: Writing HTML Body file: sashtml1.htm
44    run;

NOTE: There were 5 observations read from the data set LEC5.STOCKS.
NOTE: PROCEDURE PRINT used (Total process time):
      real time           0.28 seconds
      cpu time            0.20 seconds


45
46    title 'The Descriptor portion of Stocks.sas7bdat';
47    proc contents data=Lec5.Stocks varnum;
48    run;

NOTE: PROCEDURE CONTENTS used (Total process time):
      real time           0.03 seconds
      cpu time            0.01 seconds
```

# Results :

**The Data Portion of Stocks.sas7bdat**

| Obs | Stock | PurDate | PurPrice | Number |
|---|---|---|---|---|
| 1 | IBM | 21MAY2006 | $80.01 | 100 |
| 2 | CSCO | 05APR2005 | $17.52 | 200 |
| 3 | MOT | 01MAR2004 | $14.75 | 500 |
| 4 | XMSR | 15APR2006 | $28.42 | 200 |
| 5 | BBY | 15FEB2005 | $45.21 | 100 |

## The Descriptor portion of Stocks.sas7bdat

### The CONTENTS Procedure

| | | | |
|---|---|---|---|
| Data Set Name | LEC5.STOCKS | Observations | 5 |
| Member Type | DATA | Variables | 4 |
| Engine | V9 | Indexes | 0 |
| Created | 09/23/2018 11:36:02 | Observation Length | 32 |
| Last Modified | 09/23/2018 11:36:02 | Deleted Observations | 0 |
| Protection | | Compressed | NO |
| Data Set Type | | Sorted | NO |
| Label | | | |
| Data Representation | WINDOWS_64 | | |
| Encoding | wlatin1 Western (Windows) | | |

| Engine/Host Dependent Information | |
|---|---|
| Data Set Page Size | 65536 |
| Number of Data Set Pages | 1 |
| First Data Page | 1 |
| Max Obs per Page | 2039 |
| Obs in First Data Page | 5 |
| Number of Data Set Repairs | 0 |
| ExtendObsCounter | YES |
| Filename | J:\CLASSES\STAT46\stocks.sas7bdat |
| Release Created | 9.0401M4 |
| Host Created | X64_SR12R2 |
| Owner Name | AMERICAN\zardi |
| File Size | 128KB |
| File Size (bytes) | 131072 |

| Variables in Creation Order | | | | | |
|---|---|---|---|---|---|
| # | Variable | Type | Len | Format | Label |
| 1 | Stock | Char | 4 | | Name of the Stock |
| 2 | PurDate | Num | 8 | DATE9. | Purchase Date |
| 3 | PurPrice | Num | 8 | DOLLAR8.2 | Purchase Price |
| 4 | Number | Num | 8 | | Number of shares |

program

```
49
50    title 'Summary Statistics';
51    proc means data=Lec5.Stocks;
52    run;

NOTE: There were 5 observations read from the data set LEC5.STOCKS.
NOTE: PROCEDURE MEANS used (Total process time):
      real time                0.03 seconds
      cpu time                 0.03 seconds
```

Results

## Summary Statistics

### The MEANS Procedure

| Variable | Label | N | Mean | Std Dev | Minimum | Maximum |
|---|---|---|---|---|---|---|
| PurDate | Purchase Date | 5 | 16598.40 | 335.1243053 | 16131.00 | 16942.00 |
| PurPrice | Purchase Price | 5 | 37.1820000 | 26.7640051 | 14.7500000 | 80.0100000 |
| Number | Number of shares | 5 | 220.0000000 | 164.3167673 | 100.0000000 | 500.0000000 |

**Note:**

When you use a LABEL or FORMAT statement within a **PROC step**, the label or format applies only to the output from that step.

However, in PROC steps, you can also use permanently assigned labels or formats. **Permanent labels** and formats can be assigned **in the DATA step**. These labels and formats are saved with the data set, and they can be reused by procedures that reference the data set.

# Using Formats to enhance Output

In your SAS reports, formats control how the data values are displayed. To make data values more understandable when they are displayed in your procedure output, you can use the FORMAT statement, which associates formats with variables.

Formats affect only how the data values appear in output, not the actual data values as they are stored in the SAS data set.

- You can use a separate FORMAT statement for each variable, or you can format several variables (using either the same format or different formats) in a single FORMAT statement.

- Syntax, FORMAT statement:FORMAT variable(s) format-name;
- variable(s) is the name of one or more variables whose values are to be written according to a particular pattern
- format-name specifies a SAS format or a user-defined format that is used to write out the values.
- Tip:The FORMAT statement applies only to the PROC step in which it appears.

# Using Proc Format to create User-Defined Formats

```
proc Format;

  value   PriceLevel   10-30='Low Price'
                       31-50='Medium Price'
                       51-90='High Price';

  value   Number  0-100='Low Number of Shares'
                  101-300='Medium Number of Shares'
                  301-600='High Number of Shares';
run;
```

# Adding a Format Statement in Proc Print

```
title 'Data Set with Formatted Values';
proc print data=Lec5.Stocks;
 var Stock PurDate PurPrice Number;
 Format PurPrice  PriceLevel.
        Number    Number.;
run;
```

**Data Set with Formatted Values**

| Obs | Stock | PurDate | PurPrice | Number |
|---|---|---|---|---|
| 1 | IBM | 21MAY2006 | High Price | Low Number of Shares |
| 2 | CSCO | 05APR2005 | Low Price | Medium Number of Shares |
| 3 | MOT | 01MAR2004 | Low Price | High Number of Shares |
| 4 | XMSR | 15APR2006 | Low Price | Medium Number of Shares |
| 5 | BBY | 15FEB2005 | Medium Price | Low Number of Shares |

# Regrouping Values Using Formats

You can use format to group various values together. In this example we grouped the values into two categories only: Low and High. We did this by creating a new format:

```
proc format;
  value MedHigh 0-100 ='Low'
                101-300,301-600 ='High';
run;
```

# Applying the New Format with Proc Freq

```
title 'Number of Shares using the Format MedHigh';
proc print data=Lec5.Stocks;
 var Stock PurDate PurPrice Number;
 Format PurPrice  PriceLevel.
        Number    MedHigh.;

title ' Number of shares frequencies using MedHigh';
proc freq data=Lec5.Stocks;
  table Number;
  format Number MedHigh.;
run;
```

**Number of shares frequencies using MedHigh**

| Obs | Stock | PurDate | PurPrice | Number |
|---|---|---|---|---|
| 1 | IBM | 21MAY2006 | High Price | Low |
| 2 | CSCO | 05APR2005 | Low Price | High |
| 3 | MOT | 01MAR2004 | Low Price | High |
| 4 | XMSR | 15APR2006 | Low Price | High |
| 5 | BBY | 15FEB2005 | Medium Price | Low |

**Number of shares frequencies using MedHigh**

The FREQ Procedure

| Number of shares | | | | |
|---|---|---|---|---|
| Number | Frequency | Percent | Cumulative Frequency | Cumulative Percent |
| Low | 2 | 40.00 | 2 | 40.00 |
| High | 3 | 60.00 | 5 | 100.00 |

# More on format Ranges

When you define a Format, you can specify individual values or ranges to the left of the equal sign in your VALUE statement.

```
proc format;
  value NumberN low-<150 = 'Low'
                150-<250 = 'Medium'
                250-<500 = 'High'
                500 - high = 'Very High';
 run;
 title 'Number of Shares using the Format NumberN.';
proc print data=Lec5.Stocks;
  var Stock PurDate PurPrice Number;
  Format Number    NumberN.;
  run;
```

High and Low are predefined formats in SAS.

**Number of Shares using the Format NumberN.**

| Obs | Stock | PurDate | PurPrice | Number |
|---|---|---|---|---|
| 1 | IBM | 21MAY2006 | $80.01 | Low |
| 2 | CSCO | 05APR2005 | $17.52 | Medium |
| 3 | MOT | 01MAR2004 | $14.75 | Very High |
| 4 | XMSR | 15APR2006 | $28.42 | Medium |
| 5 | BBY | 15FEB2005 | $45.21 | Low |

# More on format Ranges

Note how the sign < is used:

**Program 1:**
to exclude the last value, then put the < sign before that value

```
proc format;
  value NumberN low-<150 = 'Less than 150'
                150-<250 = '150 to less than 250'
                250-<500 = '250 to less than 500'
                500 - high = '500+';
run;
```

**Program 2:**
to exclude the first value, then put the < sign after the sign"-"

```
proc format;
  value NumberN low-150 = 'Less than or equal to 150'
                150<-250 = 'greater than 150 to 250'
                250-<500 = 'Greater than 250 to 500'
                500<- high = 'Greater than 500';
run;
```

# Storing the Formats in a Format Library

- Create a library reference (*libref*) to indicate where you want to store your SAS formats. It could be the same library where you store your data sets.
- Use the option LIBRARY=*libref* when you run PROC FORMAT. You only have to run this procedure once.

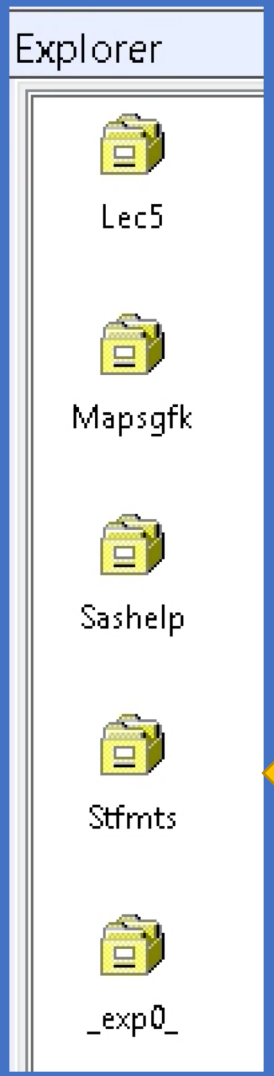
```
libname Stfmts 'J:\CLASSES\STAT46';

proc format library=Stfmts;
 value NumberN low-<150 = 'Less than 150'
               150-<250 = '150 to less than 250'
               250-<500 = '250 to less than 500'
               500 - high = '500+';
run;
```

```
1755  libname Stfmts 'J:\CLASSES\STAT46';
NOTE: Libref STFMTS refers to the same physical library as LEC5.
NOTE: Libref STFMTS was successfully assigned as follows:
      Engine:        V9
      Physical Name: J:\CLASSES\STAT46
1756
1757  proc format library=Stfmts;
1758   value NumberN low-<150 = 'Less than 150'
1759                 150-<250 = '150 to less than 250'
1760                 250-<500 = '250 to less than 500'
1761                 500 - high = '500+';
NOTE: Format NUMBERN has been written to STFMTS.FORMATS.
1762  run;

NOTE: PROCEDURE FORMAT used (Total process time):
      real time           0.04 seconds
      cpu time            0.00 seconds


1763
1764  /*proc format;
1765   value NumberN low-150 = 'Less than or equal to 150'
1766                 150<-250 = 'greater than 150 to 250'
1767                 250-<500 = 'Greater than 250 to 500'
1768                 500<- high = 'Greater than 500';
1769  run;
1770
1771
1772
1773  title 'Number of Shares using the Format NumberN.';
1774  proc print data=Lec5.Stocks;
1775   var Stock PurDate PurPrice Number;
1776   Format Number    NumberN.;
1777   run;
```




Format library created

# Permanent Data Set Attributes

- If you add your LABEL and FORMAT in the data step, the labels and formats become permanently associated with their respective variables.
- This makes a very convenient way to document the data set.

```
Libname Lec5 'J:\CLASSES\STAT46';
libname Stfmts 'J:\CLASSES\STAT46';
option fmtsearch=(Stfmts);

DATA Lec5.Stocks;
  infile 'J:\CLASSES\STAT46\stockprices.txt';
  input @1 Stock $ 4.
        @5 PurDate mmddyy10.
        @15 PurPrice  comma8.
        @21 Number 4.;
  label Stock='Name of the Stock'
        PurDate='Purchase Date'
        PurPrice= 'Purchase Price'
        Number = 'Number of shares';
  format PurDate date9.
         PurPrice dollar8.2
         PurPrice PriceLevel
         Number NumberN.;
run;
```

You could also use PROC CONTENTS or the SAS Explorer to list the labels and formats used for each variable.

```
title 'Data Set Stocks';
proc contents data=Lec5.Stocks varnum;
run;
```

**Data Set Stocks**

The CONTENTS Procedure

| | | | |
|---|---|---|---|
| Data Set Name | LEC5.STOCKS | Observations | 5 |
| Member Type | DATA | Variables | 4 |
| Engine | V9 | Indexes | 0 |
| Created | 09/24/2018 16:46:03 | Observation Length | 32 |
| Last Modified | 09/24/2018 16:46:03 | Deleted Observations | 0 |
| Protection | | Compressed | NO |
| Data Set Type | | Sorted | NO |
| Label | | | |
| Data Representation | WINDOWS_64 | | |
| Encoding | wlatin1 Western (Windows) | | |

| Engine/Host Dependent Information | |
|---|---|
| Data Set Page Size | 65536 |
| Number of Data Set Pages | 1 |
| First Data Page | 1 |
| Max Obs per Page | 2039 |
| Obs in First Data Page | 5 |
| Number of Data Set Repairs | 0 |
| ExtendObsCounter | YES |
| Filename | J:\CLASSES\STAT46\stocks.sas7bdat |
| Release Created | 9.0401M4 |
| Host Created | X64_SR12R2 |
| Owner Name | AMERICAN\zardi |
| File Size | 128KB |
| File Size (bytes) | 131072 |

**Variables in Creation Order**

| # | Variable | Type | Len | Format | Label |
|---|---|---|---|---|---|
| 1 | Stock | Char | 4 | | Name of the Stock |
| 2 | PurDate | Num | 8 | DATE9. | Purchase Date |
| 3 | PurPrice | Num | 8 | DOLLAR8.2 | Purchase Price |
| 4 | Number | Num | 8 | NUMBERN. | Number of shares |

# Accessing a permanent SAS data set with a User-Defined Formats

- To use a permanent SAS data set that has user defined formats you need to specify the option: ***FMTSEARCH=system option*** as in the program below:

```
Libname Lec5 'J:\CLASSES\STAT46';
libname Stfmts 'J:\CLASSES\STAT46';
option fmtsearch=(Stfmts);


title 'Using User-Defined-Formats';
proc freq data=Lec5.Stocks;
 tables Number;
 run;
```

**Using User-Defined-Formats**

The FREQ Procedure

| Number of shares | | | | |
|---|---|---|---|---|
| Number | Frequency | Percent | Cumulative Frequency | Cumulative Percent |
| Less than 150 | 2 | 40.00 | 2 | 40.00 |
| 150 to less than 250 | 2 | 40.00 | 4 | 80.00 |
| 500+ | 1 | 20.00 | 5 | 100.00 |

Once you submit the FMTSRCH=option, you can use your own formats as if they were built-in SA

Note: if you give a copy of a SAS dataset to another user, be sure to give a copy of the format library to them as well.

# Displaying the Format Definitions

A useful PROC FORMAT option is FMTLIB. This option creates a listing for each format in the specified library with the ranges and labels.

In this example, we display the definition of all the formats in Stfmts Library, here's the code:

```
title 'Format Definitions in the Stfmts Library';
proc format library=Stfmts fmtlib;
run;
```

**Format Definitions in the Stfmts Library**

FORMAT NAME: NUMBERN LENGTH: 20 NUMBER OF VALUES: 4
MIN LENGTH: 1 MAX LENGTH: 40 DEFAULT LENGTH: 20 FUZZ: STD

| START | END | LABEL (VER. V7\|V8 24SEP2018:17:17:16) |
|---|---|---|
| LOW | 150 | 150<Less than 150 |
| 150 | 250 | 250<150 to less than 250 |
| 250 | 500 | 500<250 to less than 500 |
| 500 | HIGH | 500+ |

FORMAT NAME: PRICELEVEL LENGTH: 12
MIN LENGTH: 1 MAX LENGTH: 40 DEFAULT LENGTH: 12 FUZZ: STD

| START | END | LABEL (VER. 9.4 24SEP2018:17:17:16) |
|---|---|---|
| 10 | 30 | Low Price |
| 31 | 50 | Medium Price |
| 51 | 90 | High Price |

- Select Statement:

```
libname Stfmts 'J:\CLASSES\STAT46';
option fmtsearch=(Stfmts);
proc format library=Stfmts;
     select NumberN;
run;
```

- Note that the Libname is part of the program.
- There's also an Exclude statement that enables you to name the formats you don't want to see displayed:

```
libname Stfmts 'J:\CLASSES\STAT46';
option fmtsearch=(Stfmts);
proc format library=Stfmts;
     exclude NumberN;
run;
```

- So we will have displayed **PriceLevel** as in the output:

FORMAT NAME: PRICELEVEL LENGTH: 12
MIN LENGTH: 1 MAX LENGTH: 40 DEFAULT LENGTH: 12 FUZZ: STD

| START | END | LABEL (VER. 9.4 24SEP2018:17:17:16) |
|---|---|---|
| 10 | 30 | Low Price |
| 31 | 50 | Medium Price |
| 51 | 90 | High Price |