Previous Page | Next Page

Displaying Dates

Understanding How SAS Displays Values

To understand how to display the departure dates, you need to understand how SAS displays values in general. SAS displays all data values with a set of directions called a **format**. By default, SAS uses a standard numeric format with no commas, letters, or other special notation to display the values of numeric variables. Creating SAS Date Values from Calendar Dates shows that printing SAS date values with the standard numeric format produces numbers that are difficult to recognize. To display these numbers as calendar dates, you need to specify a SAS date format for the variable.

SAS date formats are available for the most common ways of writing calendar dates. The DATE9. format represents dates in the form **ddMMMyyyy**. If you want the month, day, and year to be spelled out, then use the WORDDATE18. format. The WEEKDATE29. format includes the day of the week. There are also formats available for number representations such as the format MMDDYY8., which displays the calendar date in the form **mm/dd/yy**, or the format MMDDYY10., which displays the calendar date in the form **mm/dd/yyyy**. Like informat names, each format name ends with a period and contains a width specification that tells SAS how many columns to use when displaying the date value.

Formatting a Date Value

You tell SAS which format to use by specifying the variable and the format name in a FORMAT statement. The following FORMAT statement assigns the MMDDYY10. format to the variable DepartureDate:

```
format DepartureDate mmddyy10.;
```

In this example, the FORMAT statement contains the following items:

- the name of the variable (DepartureDate)
- the name of the format to be used (MMDDYY10.)

The following PRINT procedures format the variable DepartureDate in both the two-digit year calendar format and the four-digit year calendar format:

```
options pagesize=60 linesize=80 pageno=1 nodate;
proc print data=mylib.tourdates;
   title 'Departure Dates in Two-Digit Calendar Format';
   format DepartureDate mmddyy8.;
run;

proc print data=mylib.tourdates;
   title 'Departure Dates in Four-Digit Calendar Format';
   format DepartureDate mmddyy10.;
run;
```

The following output displays the results:

Displaying a Formatted Date Value

```
                Departure Dates in Two-Digit Calendar Format                1

                                 Departure
 Obs    Country                    Date       Nights

   1    Japan                    05/13/00         8
   2    Greece                   10/17/99        12
   3    New Zealand              02/03/01        16
   4    Brazil                   02/28/01         8
   5    Venezuela                11/10/00         9
   6    Italy                    04/25/01         8
   7    Russia                   06/03/97        14
   8    Switzerland              01/14/01         9
   9    Australia                10/24/98        12
  10    Ireland                  08/27/00         7

               Departure Dates in Four-Digit Calendar Format                2

                                  Departure
 Obs    Country                       Date     Nights

   1    Japan                    05/13/2000         8
   2    Greece                   10/17/1999        12
   3    New Zealand              02/03/2001        16
   4    Brazil                   02/28/2001         8
   5    Venezuela                11/10/2000         9
   6    Italy                    04/25/2001         8
   7    Russia                   06/03/1997        14
   8    Switzerland              01/14/2001         9
```

```
 9    Australia      10/24/1998      12
10    Ireland        08/27/2000       7
```

Placing a FORMAT statement in a PROC step associates the format with the variable only for that step. To associate a format with a variable permanently, use the FORMAT statement in a DATA step.

Assigning Permanent Date Formats to Variables

The next example creates a new permanent SAS data set and assigns the DATE9. format in the DATA step. Now all subsequent procedures and DATA steps that use the variable DepartureDate will use the DATE9. format by default. The PROC CONTENTS step displays the characteristics of the data set MYLIB.TOURDATE.

```
options yearcutoff=1920 pagesize=60 linesize=80 pageno=1 nodate;

data mylib.fmttourdate;
   set mylib.tourdates;
   format DepartureDate date9.;
run;

proc contents data=mylib.fmttourdate nodetails;
run;
```

The following output shows that the DATE9. format is permanently associated with DepartureDate:

Assigning a Format in a DATA Step

```
                              The SAS System                              1

                          The CONTENTS Procedure

Data Set Name: MYLIB.FMTTOURDATE                      Observations:         10
Member Type:   DATA                                   Variables:            3
Engine:        V8                                     Indexes:              0
Created:       14:15 Friday, November 19, 1999        Observation Length:   32
Last Modified: 14:15 Friday, November 19, 1999        Deleted Observations: 0
Protection:                                           Compressed:           NO
Data Set Type:                                        Sorted:               NO
Label:


                 -----Engine/Host Dependent Information-----

Data Set Page Size:         8192
Number of Data Set Pages:   1
First Data Page:            1
Max Obs per Page:           254
Obs in First Data Page:     10
Number of Data Set Repairs: 0
filename:                  /SAS_DATA_LIBRARY/fmttourdate.sas7bdat
Release Created:            8.0001M0
Host Created:               HP-UX
Inode Number:               1498874206
Access Permission:          rw-r--r--
Owner Name:                 user01
File Size (bytes):          16384


              -----Alphabetic List of Variables and Attributes-----

               #    Variable         Type    Len     Pos    Format
               ------------------------------------------------
               1    Country          Char     11      16
               2    DepartureDate    Num       8       0    DATE9.
               3    Nights           Num       8       8
```

Changing Formats Temporarily

If you are preparing a report that requires the date in a different format, then you can override the permanent format by using a FORMAT statement in a PROC step. For example, to display the value for DepartureDate in the data set MYLIB.TOURDATES in the form of **month-name dd, yyyy**, you can issue a FORMAT statement in a PROC PRINT step. The following program specifies the WORDDATE18. format for the variable DepartureDate:

```
options pagesize=60 linesize=80 pageno=1 nodate;
proc print data=mylib.tourdates;
   title 'Tour Departure Dates';
   format DepartureDate worddate18.;
run;
```

The following output displays the results:

Overriding a Previously Specified Format

Tour Departure Dates 1

Obs	Country	DepartureDate	Nights
1	Japan	May 13, 2000	8
2	Greece	October 17, 1999	12
3	New Zealand	February 3, 2001	16
4	Brazil	February 28, 2001	8
5	Venezuela	November 10, 2000	9
6	Italy	April 25, 2001	8
7	Russia	June 3, 1997	14
8	Switzerland	January 14, 2001	9
9	Australia	October 24, 1998	12
10	Ireland	August 27, 2000	7

The format DATE9. is still permanently assigned to DepartureDate. Calendar dates in the remaining examples are in the form **ddMMMyyyy** unless a FORMAT statement is included in the PROC PRINT step.

Previous Page | Next Page | Top of Page