Working with Dates in SAS® Software

Christine A. Smiley, Kestnbaum, a KnowledgeBase Marketing Company, Chicago, IL

ABSTRACT

Using dates in the SAS System can be very simple if you handle date fields properly. This paper will illustrate how to read in date fields from raw data. The various informats and formats that apply to dates will be explained. Code is provided that will convert an existing variable into a SAS date value. Techniques for printing, manipulating, and referring to particular dates will be discussed.

WHAT IS A CALENDAR DATE?

In raw data files or printed reports, dates may appear in a variety of formats. These are usually calendar dates, which are a readable representation of a particular date. All of the following calendar date values represent the date April 13, 1999:

4/13/99	041399	4131999
19990413	991304	04-13-99
13APR1999	13 APR 99	130499
13APR99	13-apr-1999	

WHAT IS A SAS DATE?

SAS represents a date internally as the number of days between January 1, 1960 and the specified date. Dates before 1960 are represented with negative numbers. A SAS date value is a numeric variable.

SAS Date Value	Calendar Date Value
0	January 1, 1960
30	January 31, 1960
366	January 1, 1961
-365	January 1, 1959
14347	April 13, 1999

HOW DO YOU READ IN A CALENDAR DATE TO CREATE A SAS DATE?

Suppose you receive a raw data file with a field containing calendar dates. It is best to read in this field as a SAS date value rather than a character or standard numeric value. This will almost always prevent much trouble. To tell the SAS System that the field is a date and should be converted to a SAS date value, use an informat. The available informats are listed in the next section. They vary depending on the form and width of the date. The width, represented by w, is determined by how many spaces the raw date uses. The width includes any dashes, slashes, or spaces within the date. For example, dates in a form such as 13-APR-1999 would use the date11. informat.

SAS DATE INFORMATS

```
DATEw.     form: ddmmmyy or ddmmmyyyy
13APR1999    13 APR 99    13-APR-1999


DDMMYYw.     form: ddmmyy or ddmmyyyy
130499 13/04/99 13 04 99 13-04-1999


MMDDYYw.     form: mmddyy or mmddyyyy
041399       04/13/99      04-13-99
4 13 99      4131999       04131999


YYMMDDw.    form: yymmdd or yyyymmdd
990413      99 04 13      99-04-13
99/04/13    19990413      1999-04-13
```

HOW DO YOU USE AN INFORMAT TO CREATE A SAS DATE?

If the raw fields are aligned in columns, use formatted input and specify the informat.

```
   input @1 mydate yymmdd8.;
```

With list input, use a separate INFORMAT statement or a modifier.

```
   informat mydate yymmdd8.;
         input mydate;


   input mydate : yymmdd8.;
```

HOW DO YOU PRINT A SAS DATE WITH A SPECIFIC FORMAT?

If a SAS date value is printed without a format it will appear as the number of days since 1960. If a format is applied to a variable, then it will appear with that format in PROC PRINT and many other procedures. When a date field is read in, it is a good idea to associate a permanent format with the variable. The formats that can be used with SAS date values parallel the informats listed above. To permanently associate a format with a variable, use a FORMAT statement in the DATA step.

```
 format mydate yymmdd8.;
```

EXAMPLE: READ DATES WITH FORMATTED INPUT

```
  data keydates;
   input @1 A date11.  @13 B ddmmyy6.
         @20 C mmddyy10.  @31 D yymmdd8.;
   format A B C D mmddyy10.;
cards;
13/APR/1999 130499 04-13-1999 99 04 13
01/JAN/1960 010160 01-01-1960 60 01 01
31/JAN/1960 310160 01-31-1960 60 01 31
run;
```

```
  proc print data=keydates;
```

```
OBS        A          B          C          D
1    04/13/1999  04/13/1999 04/13/1999 04/13/1999
2    01/01/1960  01/01/1960 01/01/1960 01/01/1960
3    01/31/1960  01/31/1960 01/31/1960 01/31/1960
```

EXAMPLE: READ DATES USING LIST INPUT

```
   data keydate2;
      informat A date11. C mmddyy10.;
      input A C;
      format A C mmddyy10.;
cards;
13/APR/1999 4-13-1999
1/JAN/1960 1-1-1960
31/JAN/1960 1-31-1960
run;
```

```
  proc print data=keydate2;
```

```
  OBS         A          C
   1   04/13/1999  04/13/1999
   2   01/01/1960  01/01/1960
   3   01/31/1960  01/31/1960
```

HOW DO YOU CHANGE A DATA SET VARIABLE TO A SAS DATE VALUE?

If you have a date field which was not read in as a SAS date value, but now you need to work with it as a date value, then it must be converted. To do this, use the INPUT and PUT functions in a DATA step. The method is different depending on whether the variable is character or numeric. See the next two sections.

HOW DO YOU CHANGE A CHARACTER VARIABLE TO A SAS DATE VALUE?

Use an INPUT function with an appropriate format. The formats that can be used mirror the informats listed previously. Suppose that you have two character variables as shown in the next PROC PRINT. To

change these to SAS date values, create new variables using the INPUT function.

```
OBS               A          C
1     13/APR/1999  14-13-99
2     31/JAN/1960  01-31-60
```

```
  A2 = input (A, date11.);
  C2 = input (C, mmddyy8.);
```

HOW DO YOU CHANGE A NUMERIC VARIABLE TO A SAS DATE VALUE?

To change a numeric variable to a SAS date value, use both the PUT and INPUT functions. The PUT function converts the value from numeric to character. The INPUT function will then convert the character variable to the numeric SAS date.

```
OBS         B               D
1     130499          19990413
2     310160          19600131
```

```
  B2 = input
     (  put (B, Z6.) , ddmmyy6.);
  D2 = input
     ( put (D, Z8.) , yymmdd8.);
```

HOW DO YOU WORK WITH SAS DATE VALUES?

When a variable is a SAS date value, you can easily apply operations such as addition and subtraction. To find the number of days between two dates, simply subtract the two SAS date variables.

```
  daysbtwn = date1 - date2;
```

Comparison operators can also be used.

```
  if date1 < date2 then do;
```

HOW DO YOU REFER TO A PARTICULAR DATE?

To create a SAS date constant, write the date enclosed in single or double quotes, followed by a D. Here are some examples:

```
  age = '13-APR-1999'd - birthday;
  where date1 lt "1jan90"D;
```

YEAR 2000 ISSUES

The YEARCUTOFF= option determines the century to which date values will default when only the last two digits of the year are given.

CONCLUSION

When dates are treated as SAS date values, it can be very straightforward to work with them. SAS software is quite flexible as to the types of dates it will read and display. It is almost always best to read in raw date fields with the proper informat to convert to a SAS date value. If the date was not handled properly to begin with, you can perform a conversion later. Once a variable is in SAS date format, there are many functions available that allow you to work with dates quite easily.

REFERENCES

Beam, David (1997), "Handling the Year 2000 and Other Timely Issues," *Proceedings of the Twenty-Second Annual SAS® Users Group International Conference*, 22, 303-310.

Karp, Andrew H. (1998), "Working with SAS® Date and Time Functions," *Proceedings of the Twenty-Third Annual SAS® Users Group International Conference*, 23, 348-354.

SAS Institute Inc. (1990), *SAS® Language: Reference, Version 6, First Edition,* Cary, NC: SAS Institute Inc.

ACKNOWLEDGMENTS

SAS is a registered trademark or trademark of SAS Institute Inc. in the USA and other countries. ® indicates registration.

Brand and product names are registered trademarks or trademarks of their respective companies.

Thank you to the following people for their suggestions and contributions to this paper: Heather Cole, Stephanie Papaioannou, Chris Rellinger, and Ken Smiley.

CONTACT INFORMATION

Please contact the author with any questions or comments:
Christine A. Smiley
Kestnbaum, a KnowledgeBase
    Marketing Company
55 W. Wacker Dr., Suite 1210
Chicago, IL 60601-1612
Phone: 312-782-1360 x232
Fax: 312-782-1362
E-mail: cas@kestnbaum.com