

SAS Date/Time Basics: What's Wrong When There Is No "ERROR"

Diahn L. Allen

ABSTRACT

This document will talk about how using dates in SAS® can produce false results but not give you an error, and how to correct that. The point of this paper is to show that even routine coding can result in data inconsistencies. This paper emphasizes the need to thoroughly check your code, your log and your results.

Keywords: Date Time Error

INTRODUCTION

Since SAS deals with data, quite a bit of programming is the ETL process. That is extracting the data, transforming it, and loading it into some repository. To that end, my task was simple enough. Read in some raw data, and transform the necessary raw fields to our standard fields. It's something I've done literally over 1,000 times. After some simple data transformations and running of frequencies to see the results, something seemed wrong, and it was rather unusual. The attached frequency indicates what happened. If you look at this frequency, even without too much knowledge of the data you can see that the variable ELIG_TERM_DATE looks odd as you scroll down. Some of the values look like SAS dates(numerics) and some look like actual dates:

18718	86	0.39	15671	70.24
18748	117	0.52	15788	70.76
18779	83	0.37	15871	71.14
19145	11	0.05	16432	73.65
19237	2	0.01	16434	73.66
22097	3	0.01	16437	73.67
19680905	2	0.01	16439	73.68
19740331	2	0.01	16441	73.69
19761231	2	0.01	16443	73.7
19770227	1	0	16444	73.7
19770531	4	0.02	16448	73.72

Huh? I've seen where the dates are missing (.) or where I received an error, but not a combination of types. Why? My original code was:

```
if i_EMPSTATUS='2' then ELIG_TERM_DATE=RETIRE_DATE ;  
  
else ELIG_TERM_DATE=PLANENDDATE;
```

What I intended to do was use different date variables (RETIRE_DATE and PLANENDDATE) to assign the values for the variable elig_term_date based on the value of the variable empstatus. For example, if empstatus='2' then use the first date (RETIRE_DATE) to assign the value of elig_term_date. In all other cases use the second date, which is the variable (PLANENDDATE).

This seemed simple enough, however, upon further examination, I found the problem was one of the variables I was using (RETIRE_DATE) was a character variable and the other, PLANENDDATE is a SAS numeric variable, a SAS date value. While this won't, and didn't result in any compile or syntax errors, it could and did produce problems later. It seemed to me that I should've received a log note telling me that my variable had been changed to CHAR; I didn't. Even if you don't get any errors, this is not an example of good coding technique. It's not good programming practice, in general, to have one variable with visibly inconsistent data types although the actual data type is listed as numeric in the contents. One of the simple errors it will produce is an error when you try to format it. As we all know, SAS dates are a special type of numeric variable. You cannot use SAS date formats to format a character value. Another problem that could arise is if and when you use SAS date functions: SAS date functions won't work with character values. One of the issues that arose was that I could not use a SAS date format. Even though in the contents ELIG_TERM_DATE is listed as numeric, SAS also recognized that some of its values were CHAR...

What alarmed me is that when I tried to format my variables in a frequency I got an error. My 2 statements were simple enough, but the variable RETIRE_DATE is char as seen in a PROC CONTENTS of the dataset. PLANENDDATE is numeric also seen in the PROC CONTENTS. So, my results are inconsistent. Here, I use two simple procedures I use to verify my results are the contents and frequency procedures. Careful examination of both showed that although the data type was listed as numeric the result was clearly not a SAS date.

This was an anomaly for me because I didn't have a note in the log showing that my data type had been changed. But it had not changed. I first told assigned ELIG_TERM_DATE to a character date named RETIRE_DATE then I assigned it to a numeric date named PLANENDDATE. Thus, there was no note in the log stating that the variable had been changed from NUMERIC to CHAR. The variable was numeric. There was a simple correction to my issue. I added the input function to my statement using the CHAR variable. That would make that numeric and would give me consistent results. One of my final slides will illustrate this.

Final code:

```
if i_EMPSTATUS='2' then ELIG_TERM_DATE=input(RETIRE_DATE, yymmdd8.) ;  
  
else ELIG_TERM_DATE=PLANENDDATE;
```

New results:

01/28/2005	2011-06-30	20050128	1	0.01	5317	41.36	2
01/31/2005	2009-07-01	20050131	2	0.02	5319	41.37	2
01/31/2005	2010-04-01	20050131	4	0.03	5323	41.40	2
01/31/2005	2010-07-01	20050131	2	0.02	5325	41.42	2
01/31/2005	2011-06-30	20050131	2	0.02	5327	41.44	2
02/28/2005	2011-06-30	20050228	4	0.03	5341	41.54	2

.
. .
.

CONCLUSION

- When using date variables in assignment statements and transformations be sure you are aware if you are using SAS dates or dates that are char variables.
- Be sure to use all tools to check for correctness.
- Use input/put functions as necessary to transform data to proper data type.

The message here is to know your data. Know the data types, the expected results and check your log and lst files. If you are doing data transformations, in particular with SAS dates, pay attention to the data types of the variables you are using. Use SAS' tools such as frequencies to look at your old variables crossed with your new variables to see if your results are what you expected. As seen here, because there is no error in the log does NOT mean your results are correct. I've come across a lot of people who check the log for errors, and maybe give a quick glance to the lst file especially in cases of production or standard processing. This is an example of how very rudimentary coding can go awry if you don't know your data and do NOT pay attention to your results—both expected and unexpected.

ACKNOWLEDGMENTS

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Diahn Allen

Email: diahnallen@gmail.com