

Manipulating SAS Date Values with Functions

SAS Date and Time Values

SAS includes a variety of functions that enable you to work with SAS date values. SAS stores a date value as the number of days from January 1, 1960, to a given date. For example:

Jan. 1, 1959	Jan. 1, 1960	Jan. 1, 1961
← -365	0	366 →

A SAS time value is stored as the number of seconds since midnight. For example:

(12:00 am) midnight	12:15 pm	17:00 (or 5:00 pm)
0	44100	61200 →

Consequently, a SAS datetime value is stored as the number of seconds between midnight on January 1, 1960, and a given date and time. For example:

July 4, 1776 11:30:23	Jan. 1, 1960 midnight	July 4, 1994 16:10:45
← -5790400177	0	1088957445 →

SAS stores date values as numbers so that you can easily sort the values or perform arithmetic computations. You can use SAS date values as you use any other numeric values.



SAS Date Functions

SAS stores dates, times, and datetimes as numeric values. You can use several functions to create these values.

You use other functions to extract months, quarters, days, and years from SAS date values.

We will see several SAS date functions, showing how they are used to both create and extract date values.



MDY Function

The MDY function creates a SAS date value from numeric values that represent the month, day, and year.

Having the date in three variables makes it difficult to perform calculations. You can convert these numeric values to useful SAS date values by applying the MDY function.

General form, MDY function:

MDY(month,day,year)

where

month can be a variable that represents the month, or a number from 1-12

day can be a variable that represents the day, or a number from 1-31

year can be a variable that represents the year, or a number that has 2 or 4 digits (4 digits recommended).



MDY Function

Example:

`mdy (Bmonth,Bday,Byear)`

Then place this function in an assignment statement to create a new variable to contain the SAS date values, such as

`Birth_date = MDY (Bmonth, Bday, Byear);`

Remember, to display SAS date values in a more readable form, you can associate a SAS format with the values.



MDY Function

The MDY function can also add the same SAS date to every observation. This may be useful if you want to compare a fixed beginning date with differing end dates. Just use numbers of the fixed date instead of data set variables when providing values to the MDY function.

Example:

```
Date_start = mdy(6,17,2002);
```

