Abstract:

Although titled "Programming with SAS® Functions", this paper's main emphasis is understanding and utilizing SAS® Dates / Times. Based on questions, easy answers and examples, this paper will strive to explain INFORMATS, FORMATS, Date / Time calculations, conversions and SAS® Date / Time Functions.

What is a SAS® Date / Time variable?

It is a numeric variable which represents the amount of time which has passed since a specific time point. It has all of the qualities of a regular numeric variable.

DATES: The number of days since January 1, 1960.
TIMES: The number of seconds since midnight.
DATETIMES: The number of seconds since midnight, January 1, 1960.

Examples:
The SAS® Date value for April 2, 1990 is: 11049 (# of days)
The SAS® Time value for 10:30 A.M. is; 37800 (# of seconds)
The SAS® Datetime value for April 2, 1990:10:30 A.M. is: 954671400 (# of seconds)

What is the advantage to using SAS® Date / Time variables?

SAS® Date / Time variables allow you the capability to do calculations and comparisons between Dates / Times. They also offer you greater flexibility in the ways you can present (format) your data.

How do I let SAS® know my variable is a Date / Time?

By using INFORMATS (instructions to SAS® on how to read your data).

Examples:

Data Informat:
  Informat Datea MMDDYY6.;
  Input dataa;
  Cards;
122582;
Stored as: 8394

Data Informat2:
  Input Timea Time8.;
  Cards;
17:30:15;
Stored as: 63015

Data Informat3:
  Input Datatime $;
  Cards;
25JUN67:04:15;
Datetimb=Put(Datetima,Datetime13.);
Stored as: -7312700
(Remember to use delimiters and a 24 hour clock)

How do I print these variables out in a readable fashion?

By using FORMATS (instructions to SAS® on how to print your data):  

Examples: (note: the input values are SAS® Dates / Times)

Data Format1;
  Input Datea;
  Cards;
2736;
  Format Datea Ostar7.;
  Put Datea;
  Prints: 25JUN67

Data Format2;
  Timea=4793329;
  Put Timea Time8.;
  Prints: 13:31:28;

Data Format3;
  Input Datatime $;
  Cards;
892771199;
  Datetimb=Put(Datetima,Datetimo13.);
  Put Datetimb;
  Prints: 15APR88:23:59:59
How do I calculate Date / Time differences?

By using subtraction and addition. Remember that Dates, Times, and Datetimes are different and cannot be compared.

Calculating the time difference between two dates gives you the time difference in days.
Calculating the difference between two times or two datetimes gives you the difference in seconds.

Examples:

Dates:
30JUL87 - 01JAN87 = 210 (days)
(10072) - (9862)

Times:
18:45 - 08:00 = 38700 (sec)
(67500) - (28800)

Datetimes:
21NOV56:13:05 - 17NOV53:04:15
(-193175100) - (-98103300)
= 95071800 (seconds)

How can I convert these to more meaningful time units?

By using division and multiplication.

Date conversions:

| YearDif | = (Dateb-Datea)/365.25) |
| DayDiff | = (Dateb-Datea); |
| HourDif | = (Dateb-Datea)*24); |
| Mindiff | = (Dateb-Datea)*1440); |
| Secdiff | = (Dateb-Datea)*86400); |

Time and Datetime conversions:

| YearDif | = ((Timeb-Timea)/31557600); |
| DayDif | = ((Timeb-Timea)/864000); |
| HourDif | = ((Timeb-Timea)/86400); |
| Mindiff | = ((Timeb-Timea)/8640); |
| Secdiff | = (Timeb-Timea); |

How else can I present the decimal time differences?

The decimal time differences can be converted into more meaningful time units by using some calculations, functions and concatenations.

Convert decimal hours to HH:MM:SS

HourDif = (Timeb-Timea)/3600);
Hmms = COMRESS(INT(HourDif)) "=" |
COMRESS(INT(MOD(HourDif,1))"60)) "=" |
COMRESS(INT(MOD(MOD(HourDif,1))
"60,1)"60));

Example:

Timeb = INPUT (13:15:10,Time8.);
= 47710
Timea = INPUT (12:00:00,Time8.);
= 43200
HourDif = ((Timeb-Timea)/3600)
= 1.252778
Hmms = (see previous calculation);
= 1:15:10

Other Misc. Date / Time topics:

Date / Time Constants: An easy way to input SAS® Date / Time values:

Datea = '06FEB88'
Timea = '09:15'T;
Datetme = '19FEB66:11:15' DT;

Creating a character variable based on an associated format from raw data which is input into a SAS® Date:

Dateb = PUT(INPUT(Date, YYMMDD6.),Date7.);

Calculating age with the FLOOR function:

Age = FLOOR(Today-Bdate)/365.25);
If Age LT 0 then Age = Age+100;
My data isn't always formatted the way SAS® needs it to be in order to input it. How can I restructure it?

**Dates:**
SAS® is very flexible in the different ways Dates can be input. SAS® will even accept partial dates by using the MONYY format and will assume the day to be the first day of the month.

You may also reformat your data to allow for your own assumptions. For example, partial dates (including year only) may be converted to full dates using your own assumptions:

**Example:**
(where the raw data is in YYMMDD format)
\[
\text{Cdate} = \text{COMPRESS(Datea)};
\]
\[
\text{If LENGTH(Cdate) = 4 then Cdate} = \text{Cdate || '15'};
\]
\[
\text{Else if LENGTH(Cdate) = 2 then Cdate} = \text{Cdate || '01'};
\]
\[
\text{Dateb} = \text{INPUT(Cdate, YYMMDD6.)};
\]

**Times:**
SAS® is not as flexible when entering Time variables. Delimiters such as colons or blank spaces must be used between the hours, minutes, and seconds.

You may have to add colons or blank spaces to your data before inputting it.

**Example:**
(where the time is in HHMM format)
\[
\text{Ctime} = \text{COMPRESS(Timea)};
\]
\[
\text{Do while (length(Ctime) LT 4);}
\]
\[
\text{Ctime} = '0' || \text{Ctime};
\]
\[
\text{End;}
\]
\[
\text{Time} = \text{SUBSTR(Ctime,1,2)} || ':' || \text{SUBSTR(Ctime,3,2)};
\]
\[
\text{Fintime} = \text{PUT(\text{INPUT(\text{Time},\text{TIME5.})},\text{TIME5.});}
\]

Times continued:
SAS® dates must also be in 24 hour clock times. You may need to convert your A.M. / P.M. times to 24 hour clock times. (Note: You must have a variable to flag the P.M. values):

**Example:**
Dataa = '17JUL47D';
\[
\text{Month} = \text{MONTH(Dataa)};
\]
\[
\text{Day} = 17;
\]
\[
\text{Dataa} = \text{MDY(Month,Day,56)};
\]
\[
\text{Returns: -1130}
\]

Similar functions return Time values (HMS) and Datetime values (OAMHMS).

**Datetimes:**
SAS® is quite specific about the input format for Datetime values. The date portion must be in the DDMYYMM format followed by a special character or blank, and the time portion must follow the same restrictions as SAS® Time variables.

Often your Datetime variable will be calculated using SAS® Date and Time variables. This can be done quite easily by simply converting the SAS® Date value to number of seconds and then adding the SAS® Time value to it.

**Example:**
Dataa = '21NOV69D';
\[
\text{Times} = '12:15'T;
\]
\[
\text{Datetime} = \text{DATE(MDY(Month,Day,56), Time});
\]
\[
\text{Format Datetime DATETIME13;};
\]
\[
\text{Stored as: 943272900}
\]
\[
\text{Prints as: 21NOV69:12:15}
\]

**Doesn't SAS® have a whole bunch of Date / Time functions?**
Yes. Several are shown here. See the SAS® Basics manual for more.

**MDY -** returns a SAS® Date value from month, day and year arguments:
\[
\text{Month} = 11;
\]
\[
\text{Day} = 21;
\]
\[
\text{Dataa} = \text{MDY(Month,Day,56)};
\]
\[
\text{Returns: -1130}
\]

**MONTH -** Returns the numeric value of the month from a SAS® date variable.
\[
\text{Dataa} = '17JUL47D';
\]
\[
\text{Month} = \text{MONTH(Dataa)};
\]
\[
\text{Returns: 7}
\]

Similar functions return day values (DAY) and year values (YEAR).
DATE - Returns today's date as a SAS® Date.

    Tday = DATE();
    Returns: 11049
    May also be written as: Tday = TODAY();

Similar functions return the current Time value (TIME) and current Datetime value (DATETIME).

DATEPART - Returns the Date portion of a SAS® Datetime value.

    Datetime = '02APR90:10:30'DT;
    Datea = DATEPART(Datetime);
    Returns: 11049

Similar functions return the Time portion (TIMEPART) of a SAS® Datetime value.

HOUR - Returns the hour from a SAS® Datetime or Time value.

    Hour = HOUR(Datetime);
    Returns: 10

Similar functions return the minute (MINUTE) or second (SECOND) time values from SAS® Datetime or Time values.