How can I convert a datetime variable to a date variable in SAS?

Authored by

stats writer

June 26, 2024

To convert a datetime variable to a date variable in SAS, the following steps can be followed:

1. First, identify the datetime variable that needs to be converted. This variable should be in SAS datetime format, which includes both date and time information.

2. Create a new variable to store the converted date values. This variable should be in SAS date format, which only includes date information.

3. Use the SAS function INTNX to convert the datetime variable to a date variable. This function takes three arguments: the interval, the number of intervals, and the datetime variable.

4. Set the interval to 'DAY' and the number of intervals to 0. This will extract only the date portion from the datetime variable.

5. Assign the result of the INTNX function to the new date variable.

6. If needed, format the new date variable to display the date in the desired format.

By following these steps, the datetime variable can be converted to a date variable in SAS.

Convert Datetime to Date in SAS

The easiest way to convert a datetime to a date in SAS is to use the DATEPART function.

This function uses the following basic syntax:

date = put(datepart(some_datetime), mmddyy10.);

The argument mmddyy10. specifies that the date should be formatted like 10/15/2022.

The following example shows how to use this syntax in practice.

Example: Convert Datetime to Date in SAS

Suppose we have the following dataset in SAS that contains one column of datetimes:

```
/*create dataset*/
data original_data;
format some_datetime datetime23.;
input some_datetime :datetime23.;
datalines;
14OCT2022:0:0:0
09NOV2022:0:0:0
14NOV2022:0:0:0
15NOV2022:0:0:0
22DEC2022:0:0:0
24DEC2022:0:0:0
04JAN2023:0:0:0
;
run;

/*view dataset*/
proc printdata=original_data;
```

Obs	some_datetime
1	14OCT2022:00:00:00
2	09NOV2022:00:00:00
3	14NOV2022:00:00:00
4	15NOV2022:00:00:00
5	22DEC2022:00:00:00
6	24DEC2022:00:00:00
7	04JAN2023:00:00:00

The following code shows how to use the DATEPART function to create a new dataset in which the values in the datetime column are formatted as dates with various formats:

```
/*create new dataset with datetime formatted as date*/
data new_data;
set original_data;
date_mmddyyyy = put(datepart(some_datetime), mmddyy10.);
date_yyyymmdd = put(datepart(some_datetime), yymmdd10.);
date_date9 = put(datepart(some_datetime), date9.);
date_default = datepart(some_datetime);
run;/*view new dataset*/
proc printdata=new_data;
```

Obs	some_datetime	date_mmddyyyy	date_yyyymmdd	date_date9	date_default
1	14OCT2022:00:00:00	10/14/2022	2022-10-14	14OCT2022	22932
2	09NOV2022:00:00:00	11/09/2022	2022-11-09	09NOV2022	22958
3	14NOV2022:00:00:00	11/14/2022	2022-11-14	14NOV2022	22963
4	15NOV2022:00:00:00	11/15/2022	2022-11-15	15NOV2022	22964
5	22DEC2022:00:00:00	12/22/2022	2022-12-22	22DEC2022	23001
6	24DEC2022:00:00:00	12/24/2022	2022-12-24	24DEC2022	23003
7	04JAN2023:00:00:00	01/04/2023	2023-01-04	04JAN2023	23014

Notice that the four new columns display the date from the original datetime column in various formats.

By default, the DATEPART function converts a datetime to the number of days since January 1, 1960.

Thus, the new column called date_default displays the number of days since January 1, 1960 for each datetime.

Note: You can find the complete documentation for the SAS DATEPART function .