How can the INTNX function be used in SAS, and what are some examples of its applications?

Authored by

stats writer

June 25, 2024

The INTNX function in SAS is a powerful tool used for manipulating and calculating dates. It allows users to increment or decrement a date by a specified interval, such as days, months, or years. This function is particularly useful for creating time series data or performing time-based calculations.

One example of using the INTNX function is for creating a series of dates, such as for a monthly sales report. By specifying an interval of one month, the function can generate a sequence of dates for each month in a given time period.

Another application of the INTNX function is for calculating the difference between two dates, such as the age of a person or the length of a project. This can be done by specifying a larger interval, such as years, and subtracting the result from the original date.

Overall, the INTNX function provides flexibility and efficiency in managing and manipulating dates in SAS, making it a valuable tool for data analysis and reporting.

Use INTNX Function in SAS (With Examples)

You can use the INTNX function in SAS to increment a date by a specific interval such as a day, week, month, etc.

This function uses the following basic syntax:

INTNX(interval, start_date, increment)

where:

interval: The interval to add to date (day, week, month, year, etc.)start_date: Variable that contains start datesincrement: The number of intervals to add

To subtract an interval, supply a negative number to the increment argument.

The following examples show some common ways to use the INTNX function in practice with the following dataset in SAS:

```
/*create dataset*/
data original_data;
format date date9.;
input date :date9. sales;
datalines;
01JAN2022 50
01FEB2022 34
14MAR2022 26
01MAY2022 22
24AUG2022 27
28OCT2022 48
14NOV2022 97
04DEC2022 88
;
run;

/*view dataset*/
```

```
proc printdata=original_data;
```

Obs	date	sales
1	01JAN2022	50
2	01FEB2022	34
3	14MAR2022	26
4	01MAY2022	22
5	24AUG2022	27
6	28OCT2022	48
7	14NOV2022	97
8	04DEC2022	88

Example 1: Use INTNX to Add Days to Date

We can use the INTNX function to create a new column called plus5days that adds five days to each date in the date column:

```
/*create new dataset with column that adds 5 days to date*/
data new_data;
set original_data;
plus5days=intnx('day', date, 5);
format plus5days date9.;
run;

/*view dataset*/
```

proc printdata=new_data;

Obs	date	sales	plus5days
1	01JAN2022	50	06JAN2022
2	01FEB2022	34	06FEB2022
3	14MAR2022	26	19MAR2022
4	01MAY2022	22	06MAY2022
5	24AUG2022	27	29AUG2022
6	28OCT2022	48	02NOV2022
7	14NOV2022	97	19NOV2022
8	04DEC2022	88	09DEC2022

Notice that the new column called plus5days contains the values in the date column with fives days added to them.

Example 2: Use INTNX to Subtract Days from Date

You can also subtract days by simply using a negative value in the INTNX function.

For example, we can use the following code to subtract five days from each value in the date column:

/*create new dataset with column that subtracts 5 days from date*/
data new_data;

```
set original_data;
minus5days=intnx('day', date, -5);
format minusdays date9.;
run;

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

Obs	date	sales	minus5days
1	01JAN2022	50	27DEC2021
2	01FEB2022	34	27JAN2022
3	14MAR2022	26	09MAR2022
4	01MAY2022	22	26APR2022
5	24AUG2022	27	19AUG2022
6	28OCT2022	48	23OCT2022
7	14NOV2022	97	09NOV2022
8	04DEC2022	88	29NOV2022

Notice that the new column called minus5days contains the values in the date column with fives days subtracted from them.

Example 3: Use INTNX to Find First Day of Month

We can use the INTNX function to create a new column called firstmonth that contains the first day of the month for each date in the date column:

```
/*create new dataset with column that contains first day of the month*/
data new_data;
set original_data;
firstmonth=intnx('month', date, 0);
formatfirstmonth date9.;
run;

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

Obs	date	sales	firstmonth
1	01JAN2022	50	01JAN2022
2	01FEB2022	34	01FEB2022
3	14MAR2022	26	01MAR2022
4	01MAY2022	22	01MAY2022
5	24AUG2022	27	01AUG2022
6	28OCT2022	48	01OCT2022
7	14NOV2022	97	01NOV2022
8	04DEC2022	88	01DEC2022

Notice that the new column called firstmonth contains the first day of the month for each date in the date column.

Example 4: Use INTNX to Find First Day of Year

We can also use the INTNX function to create a new column called firstyear that contains the first day of the year for each date in the date column:

```
/*create new dataset with column that contains first day of the year*/
data new_data;
set original_data;
firstyear=intnx('year', date, 0);
formatfirstyear date9.;
run;

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

Obs	date	sales	firstyear
1	01JAN2022	50	01JAN2022
2	01FEB2022	34	01JAN2022
3	14MAR2022	26	01JAN2022
4	01MAY2022	22	01JAN2022
5	24AUG2022	27	01JAN2022
6	28OCT2022	48	01JAN2022
7	14NOV2022	97	01JAN2022
8	04DEC2022	88	01JAN2022

Notice that the new column called firstyear contains the

first day of the year for each date in the date column.

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

The following tutorials explain how to perform other common tasks in SAS: