

Data Science Stack Exchange is a question and answer site for Data science professionals, Machine Learning specialists, and those interested in learning more about the field. Join them; it only takes a minute:

Sign up

Here's how it works:

Anybody can ask a question

Anybody can answer

The best answers are voted up and rise to the top

Why do we need to use sysfunc when we call a SAS function inside a SAS macro

I saw this piece of code in my project:

```
%let num = test;
%let x=%sysfunc(trim(&num));
```

Why could not I write:

```
%let x= %trim(&num);
```

Why did I need to use sysfunc?

Under what circumstances can I call a function inside a macro without using sysfunc?

sas

asked Nov 13 '15 at 21:41

 Victor
142 6

2 Answers

Without the sysfunc(), the expression will not be evaluated. You will not be assigning the value of the expression trim(&num) to the macrovariable, but rather the whole expression.

If you want to store the result of an expression, you need to execute that function with sysfunc()

<http://support.sas.com/documentation/cdl/en/mcrolref/61885/HTML/default/viewer.htm#z3514sysfunc.htm>

answered Nov 13 '15 at 23:03

 jamesmf
1,750 3 13

Thanks. Why we need the % sign in front of sysfunc but not in front of trim? – Victor Nov 14 '15 at 4:00

The argument to sysfunc is the precise code that sysfunc should execute as though it were not in a macro - so it interprets trim correctly – jamesmf Nov 14 '15 at 15:12

The trim() function is part of the SAS Language. Without the %sysfunc() macro function, trim() can only be used within a data step or in a macro definition that gets called inside of a data step. You can't have it out in open code.

if %trim() existed, which it does not, it would work just fine the way you used it. But, there is no function named %trim() that is defined in the SAS macro language.

Doing what you want without %sysfunc() , would have to go something like:

```
data _NULL_;
  call symput("x", trim("&num."));
run;
```

or if you want x to be a data set variable:

```
%let expr= trim("&num.");

DATA ds;
  x = &expr.;
RUN;
```

answered Apr 1 at 11:56

CommanderP