



Call Execute: Let Your Program Run Your Macro



Artur Usov, OCS Consulting, Netherlands



Table of contents

- ▶ Introduction
- ▶ Syntax
- ▶ Scenarios
- ▶ Limitations
- ▶ Questions



Introduction

- ▶ Allows to generate and execute **any** SAS code within a data step
- ▶ Values of variables initiated in the PDV inside the data step can be part of the executable code
- ▶ Efficient way of re-executing macros where parameters are taken from a dataset
- ▶ Data driven programs

Syntax



- ▶ Simple Syntax : call execute('SAS CODE');
- ▶ **SAS CODE:**
 - ▶ Any SAS code to be executed
 - ▶ Is executed for every observation from the dataset
 - ▶ Can contain values of the variables from the dataset
- ▶ Example:

```
data _null_;  
    call execute('%put Hello world;');  
run;
```

↓

%put Hello world;

↓

Hello world



Scenario 1: variable values are code

- Use variable value from the data step as part of executable code:

```
data _null_;  
  set sashelp.class;  
  call execute('data work.' || strip(name) || '  
              set sashelp.class;  
              where name="' || strip(name) || '"';  
              run;');  
  
run;
```

First observation with name Alfred :

```
data work.Alfred;  
  set sashelp.class;  
  where name="Alfred";  
  
run;
```



Scenario 2: run macros

- ▶ Data step within Data Step – avoid, better use macros
- ▶ Pass macro parameters from dataset

```
%macro datasets(name= );  
  
    data work.<u>&name</u>;  
        set sashelp.class;  
        where name="<u>&name</u>";  
    run;  
  
%mend datasets;  
  
data _null_;  
    set sashelp.class;  
    call execute('%datasets(name='||strip(<u>name</u>)||')');  
run;
```

First observation with name Alfred:

```
%<b>datasets</b>(name=<u>Alfred</u>);
```

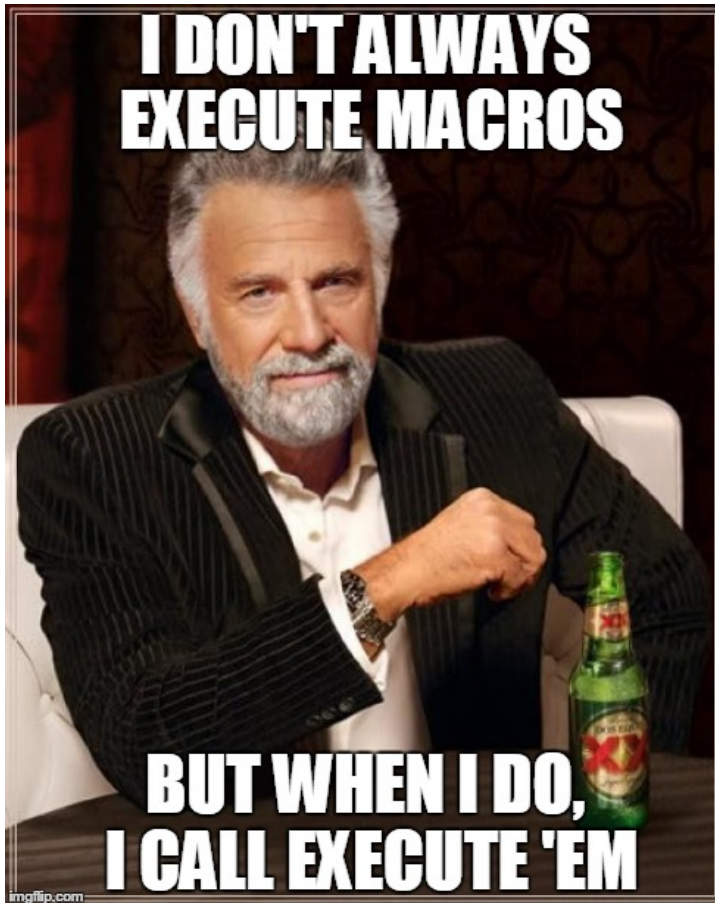


Limitations

- ▶ Call Execute will create macro variables created in a macro with Call Symput or INTO function within PROC SQL only at the end of macro execution
- ▶ Use %NRSTR with Call Execute

```
data _null_;  
  set sashelp.class;  
  call execute('%nrstr(%limit(name='||strip(name)||'))');  
  call execute('%limit(name='||strip(name)||'))';  
run;
```

Questions



- ▶ Thank you for your attention