

# Macros

## SAS ignores comments, right?

You'd like to think so, and mostly this is true - but not always!

When it comes to using comments, the two main options are comment blocks `/* ... */` and comment statements `* ... ;` and it is often a case of personal choice which to use.

There is a school of thought which suggests that the comment statement should be used where possible, as this allows for the comment block style to be used to block-off a series of statements, including comment statements:

```
*** Listing Report *** ;
proc print data = sashelp.class ;
run ;

/*** Do not run this code block
*** Summary Report *** ;
proc means data = sashelp.class ;
    var age ;
run ;
***/
```

So far, so good. Anything coded as a comment is essentially 'ignored', and has no impact on processing.

Problems can arise however when the Macro Processor is involved: Macro Language Statements are processed by the Macro Processor, while all other syntax is passed to the DATA Step Compiler.

The following code appears innocuous, containing Macro Language Statements (`%IF... %THEN... %ELSE...`) and DATA Step syntax comments:

```
%macro humpday ;
%let dow = %sysfunc(weekday(%sysfunc(today()))) ;
*** Do some Logic on Wednesday ***;
%if &dow = 4 %then
%do ;
    %put Happy Humpday! ;
%end ;
*** Earlier ***;
%else %if &dow < 4 %then
%do ;
    %put No Luck! Keep Working... ;
%end ;
*** Later ***;
```

# Macros

```
%else
%do ;
    %put Almost the weekend! ;
%end ;
%mend humpday ;

%humpday
```

Submitting this code however generates the following in the LOG:

```
15      %macro humpday ;
16      %let dow = %sysfunc(weekday(%sysfunc(today()))) ;
17      *** Do some Logic on Wednesday ***;
18      %if &dow = 4 %then
19      %do ;
20          %put Happy Humpday! ;
21      %end ;
22      *** Earlier ***;
23      %else %if &dow < 4 %then
ERROR: There is no matching %IF statement for the %ELSE. A dummy macro
will be compiled.
24      %do ;
25          %put No Luck! Keep Working... ;
26      %end ;
27      *** Later ***;
28      %else
ERROR: There is no matching %IF statement for the %ELSE. A dummy macro
will be compiled.
29      %do ;
30          %put Almost the weekend! ;
31      %end ;
32      %mend humpday ;
33
34      %humpday
—
180
WARNING: Apparent invocation of macro HUMPDAY not resolved.
ERROR 180-322: Statement is not valid or it is used out of proper order.
```

although there appears to be no syntax errors within the code.

The issue lies with the comment statements embedded within the Conditional Macro Language

# Macros

statements, which are passed to the DATA Step Compiler, where they are compiled and executed. This focus shift from the Macro Processor appears to cause the Macro Processor to 're-start' when control is passed back, giving an apparent syntax error on the succeeding %ELSE statements.

Replacing the comment statements with the comment block syntax, or with macro comment syntax %\* ... ; resolves the problem:

```
%macro humpday ;
%let dow = %sysfunc(weekday(%sysfunc(today()))) ;
*** Do some Logic on Wednesday ***;
%if &dow = 4 %then
%do ;
    %put Happy Humpday! ;
%end ;
/*** Earlier ***/
%else %if &dow < 4 %then
%do ;
    %put No Luck! Keep Working... ;
%end ;
%*** Later ***;
%else
%do ;
    %put Almost the weekend! ;
%end ;
%mend humpday ;

%humpday
```

Unique solution ID: #1040

Author: Alan D Rudland

Last update: 2019-01-11 10:03