

PROC SQL

How can I create a macro variable containing a quoted list from a dataset?

Using PROC SQL and the INTO clause along with the SEPARATED BY option and a compound separator creating quoted lists is easy.

Using the PROC SQL code:

```
proc sql noprint ;
  select name into :namelist separated by '"', "'
  from sashelp.class
  ;
quit ;
```

will generate a partially quoted list with text looking like:

```
Alfred", "Alice", "Barbara", "Carol
```

Use a %LET statement to concatenate an opening quote, the resolved text and a closing quote:

```
%let namelist = "&namelist" ;
```

The quotation mark before the ampersand is the opening quote at the beginning of the first item, and the one before the semi-colon is the closing quote after the last item. The macro variable now contains the text:

```
"Alfred", "Alice", "Barbara", "Carol"
```

It is also possible to create a list of items with single quotes, but additional MACRO Quoting functions are required to mask the single quotes, which would otherwise prevent the macro variable reference from resolving:

```
*** Use Single Quotes *** ;
```

```
proc sql noprint ;
  select name into :namelist separated by "' '"
  from sashelp.class
```

PROC SQL

```
;
quit ;
```

```
%let sq = %bquote(%sysfunc(byte(39))) ;
%let namelist = %unquote(&sq%bquote(&namelist)&sq) ;
%put &namelist ;
```

```
*** Sample Use *** ;
```

```
data names ;
  set sashelp.classfit ;
  where name in (&namelist) ;
run ;
```

Unique solution ID: #1003

Author: Alan D Rudland

Last update: 2017-02-23 14:57