PROC SQL

Is there an equivalent for the COALESCE function to find the first non-missing value in a column, rather than a row?

The COALESCE function allows for the first non-missing argument to be returned from a list of variables reading across a row: non_miss = coalesce(arg1,arg2,arg3) the only requirement being that the three variables arg1-arg3 are of the same type.

Sometimes data may be generated where individual rows contain a missing or non-missing value for the same variable, and it is desirable to consolidate these rows to retain only the non-missing values.

Submitting the code:

```
data mults ;
infile datalines dsd ;
input id char1 $ num1 char2 $ ;
datalines ;
1,Fred,.,Smith
1,,100,Smith
2,,200,Jones
2,Bert,.,
;
run ;
```

which generates a table:

1	Fred	.	Smith
1		100	Smith
2		200	Jones
2	Bert	.	

with a desired output of:

1	Fred	100	Smith
2	Bert	200	Jones

This can be achieved using a summary statistic in PROC SQL. The default action for summary statistics with a single argument in PROC SQL, is to calculate a value 'down' the column. Whilst it is usual to do this with numeric values, it is also possible with certain functions, to undertake the summarization with character variables.

The solution for this particular problem uses the MAX function in conjunction with the GROUP BY clause to create classification groups:

```
proc sql ;
create table flats as
select  id
       ,max(char1) as forename
       ,max(num1)  as value
       ,max(char2) as surname
from mults
group by id
;
quit ;
```

Unique solution ID: #1027
Author: Alan D Rudland
Last update: 2017-05-11 09:20