

Preserving Line Breaks When Exporting to Excel

Nelson Lee, Genentech, South San Francisco, CA

ABSTRACT

Do you have imported data with line breaks and want to export the data to Excel? What if you need to preserve the line breaks in the data export? This paper shares a quick tip on how to export SAS data with line breaks to Excel through PROC TEMPLATE and TAGSET attributes. This paper was written for audiences with beginner skills; the code was written using SAS® Version 9.2 on Windows operating system.

INTRODUCTION

It is common to enter line breaks (Alt+Enter) into Excel data to create multiple lines within a cell for easy reading or other formatting purposes. Exhibit 1 is an example where investigational subject data with comments are recorded. On row 2, the COMMENT column has multiple lines entered.

A number of published articles ^{1,2} shared different approaches on importing this type of data in SAS® from Excel. However, there are times that we need to preserve the data with line breaks when exporting to Excel.

	A	B	C	D
1	ID	GENDER	DOB	COMMENT
2	1001	M	1/1/1990	Subject is on combo treatment Check prior medical hx Transferred to site 305A
3	1002	F	3/15/1958	single arm subject adverse events reported in last 30 days
4	1003	M	6/30/1967	no prior medical history on record
5	1004	F	4/12/1963	
6	1005	F	12/3/1974	subject missing contact number

3 lines separated by line breaks

Exhibit 1: Source Data – Sheet1

	A	B	C
1	ID	TEST1	TEST2
2	1001	125	positive
3	1002	102	negative
4	1003	110	unknown
5	2201	99	positive
6	2202	105	negative

Exhibit 2: Source Data – Sheet2

AN EXAMPLE

Data in Exhibit 1 and Exhibit 2 come from two different sources but are in the same Excel file. There is a business need to combine (merge by ID) the 2 data sets into a single data set and export to Excel in a single spreadsheet. To complete this task, data in Exhibit 1 (sheet1) and Exhibit 2 (sheet2) are read in SAS using the LIBNAME statement. The following SAS code reads the data from the 2 spreadsheets (sheet1 and sheet2), manipulates the data and creates a single dataset, then exports the combined data to Excel.

Preserving Line Breaks When Exporting to Excel

```

/*read sheet1 COMMENT data*/
DATA in_one;
    set test.'sheet1$'n;
run;

/*read sheet2 TEST data*/
DATA in_two;
    set test.'sheet2$'n;
run;

PROC SORT data=in_one;
    by id;
run;

PROC SORT data=in_two;
    by id;
run;

/*merge sheet1 and sheet2 data by ID, data are sorted by ID*/
DATA one_two;
    merge in_one in_two;
    by id;
run;

ods listing close;
ods tagsets.excelxp file="Linebreakoutput.xls";

/* export to Excel */
ods tagsets.excelxp options(sheet_name="Combined"
Absolute_Column_Width="5,8,8,20,5,5" autofit_height = 'yes');

PROC PRINT data=one_two noobs; run;

ods tagsets.excelxp close;
ods listing;

```

Exhibit 3 is the exported data in Excel. Note that the data in the COMMENT column is exported successfully but the line breaks are not preserved. For example, the first data line (row 2), a single line is observed where the line break is replaced with a space.

	A	B	C	D	E	F
1	ID	GENDER	DOB	COMMENT	TEST1	TEST2
2	1001	M	01JAN1990	Subject is on combo treatment Check prior medical hx Transferred to site 305A	125	positive
3	1002	F	15MAR1958	single arm subject adverse events reported in last 30 days	102	negative
4	1003	M	30JUN1967	no prior medical history on record	110	unknown

Exhibit 3: Combined Data – Line Breaks Are Not Preserved

LINE BREAK HANDLING

Since the variable COMMENT has line breaks (refer to Exhibit 1), it needs special handling in order to preserve line breaks.

In the code below, the TRANWRD function replaces a line break (special embedded character 'OA'x) with the special character '^'. It can be any character as long as it does not mix up with any character in COMMENT. Without this character conversion, the line break is completely ignored when data is read in SAS and treated as a space (see Exhibit 3).

Preserving Line Breaks When Exporting to Excel

```

/*read sheet1 COMMENT data with line breaks*/
DATA in_one;
  set test.'sheet1$'n;
  COMMENT=tranwrd(COMMENT, '0A'x, "^"); /*replace line breaks*/
run;

```

EXPORT TO EXCEL

Use PROC TEMPLATE and modify TAGSET attributes to preserve line breaks in data export. Without this modification, the line breaks are not preserved during export and would have the character “^” exported instead as shown in Exhibit 4.

	A	B	C	D	E	F
1	ID	GENDER	DOB	COMMENT	TEST1	TEST2
2	1001	M	01JAN1990	Subject is on combo treatment Check prior medical hx Transferred to site 305A	125	positive
3	1002	F	15MAR1958	single arm subject adverse events reported in last 30 days	102	negative

Each line break is replaced by the special character ^

Exhibit 4: Combined Data – Line Breaks Converted to a Special Character

The following code (PROC TEMPLATE) is added before exporting the merged SAS dataset one_two in the previous section. This will allow the converted line break character “^” to export to Excel and preserve the line break attribute as shown in Exhibit 5.

```

PROC TEMPLATE;
  edit tagsets.excelbase;
  mapsub = %nrstr("</>/&quot;/&#10;");
  map = %nrstr("<>\"^");
end;
run;

```

Unicode
 is the line break character

“^” matches the line break replacement character in data_in_one step

	A	B	C	D	E	F
1	ID	GENDER	DOB	COMMENT	TEST1	TEST2
2	1001	M	01JAN1990	Subject is on combo treatment Check prior medical hx Transferred to site 305A	125	positive
3	1002	F	15MAR1958	single arm subject adverse events reported in last 30 days	102	negative
4	1003	M	30JUN1967	no prior medical history on record	110	unknown
5	1004	F	12APR1963	.	.	.
6	1005	F	03DEC1974	subject missing contact number	.	.
7	2201	.	.	.	99	positive
8	2202	.	.	.	105	negative

Line break preserved

Exhibit 5: Desired Output – Line Breaks Are Preserved

CONCLUSIONS

The purpose of this paper is to show that with a simple special character conversion and modification to tagset attributes, you can easily export SAS data to Excel while preserving (or creating) the line break attributes.

ACKNOWLEDGEMENTS

A big thank you to SAS technical support for assisting me with addressing the line break export question and identifying a solution.

REFERENCE

1. Reading an Excel Spreadsheet with Cells Containing Line Endings, Larry Hoyle, Institute for Policy & Social Research, University of Kansas, SAS Global Forum 2013 Quick Tips, paper 315-2013
2. Line Feeds –How to detect them and get rid of them? Kalyani Kothapalli, MDS Pharma Services, Lincoln, NE, Pharmasug 2008, paper CC06

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Nelson Lee, CCDM
Genentech, Inc.
1 DNA Way, MS #88
South San Francisco, CA 94080
(650) 225-2121
lee.nelson@gene.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. © indicates USA registration.
Other brand and product names are trademarks of their respective companies.

APPENDIX

The following code generates the data shown in Exhibit 5.

```
LIBNAME test "C:\MYSAS\Data\Clinicallab.xls";
/*read sheet1 COMMENT data with line breaks*/
DATA in_one;
    set test.'sheet1$'n;
    comment=tranwrd(comment, '0A'x, "^"); /*replace line breaks*/
run;

/*read sheet2 TEST data*/
DATA in_two;
    set test.'sheet2$'n;
run;

PROC SORT data=in_one;
    by id;
run;

PROC SORT data=in_two;
    by id;
run;

/*merge sheet1 and sheet2 data by ID, data are sorted by ID*/
DATA one_two;
    merge in_one in_two;
    by id;
run;

/* Add the ODS PATH statement so that this does not get overwritten in the
SASUSER */
ods path work.templat(update) sashelp.tmplmst(read);

PROC TEMPLATE;
    edit tagsets.excelbase;
        mapsub = %nrstr("&lt;/&gt;/&amp;/&quot;/&#10;");
        map = %nrstr("<>"^");
    end;
run;

ods listing close;
ods tagsets.excelxp file="Linebreakoutput.xls";

/* export to Excel */
ods tagsets.excelxp options(sheet_name="Combined"
Absolute_Column_Width="5,8,8,20,5,5" autofit_height = 'yes');

PROC PRINT data=one_two noobs; run;

ods tagsets.excelxp close;
ods listing;
```