

Paper #TF13



Rediscovering the DATA _NULL_ for Creating a Report, And Putting That Text File Into RTF Format In a Single Datastep

David Franklin
TheProgrammersCabin.com

Introduction

- DATA_NULL_ is one of the most flexible methods for generating a text output file
- as a programmer you are able to control where a character is put on the page and where page breaks begin
- if is possible wrap text over multiple lines and place them where you want them to be
- The DATA_NULL_ technique presented here produces a text output file that can be read in a text editor, and therefore by its very nature does not allow for proportional fonts like Times Roman

Aims

- Rediscovered the DATA _NULL_ as it is used for creating a text output file
- presents a few macros, tricks and techniques that will make your report shine
 - a macro for wrapping text so it can go onto multiple lines
 - doing “Page x of y” processing
 - centering or right aligning text
 - page breaking where you want it
 - carrying over group values from one page to another
- transforming the text file created into an RTF file by a macro in a single datastep

The PUT Statement is Key

- `<n*>"char-str"` -- writes out the given quoted character string *n* times (if *n* is missing, then output once)
- `@(n|num-var|(expn))` -- moves the column pointer to the given column
- `+(n|num-var|(expn))` -- moves the column pointer *n* columns from the current position
- `#(n|num-var|(expn))` – move the column pointer to row *n*
- `/` -- moves the line pointer to the first column on the next line
- `_PAGE_` -- forces a page break where pointer goes to the first column of the first line of the next page
- `var<:format>` -- value from variable being output, with optional format
- `@ | @@` -- Pointer to be held for the next PUT statement

The Anatomy Of The Data Step To Produce The Report, Part 1

The DATA step has four main parts:

- Initialization
- Body
- Header
- Footer

The Anatomy Of The Data Step To Produce The Report, Part 2

```
data _null_;  
  FILE "<filename>" N=PAGESIZE PRINT LINESLEFT=11  
    HEADER=hdr NOTITLES NOFOOTNOTES;  
  SET <dataset> end=eof;  
  put <@position> <variable> <format>;  
  if eof then link ftr;  
  return;  
  hdr: put <@position> "<text>";  
    return;  
  ftr: put <#line> <@position> "<text>";  
    return;  
run;
```

Our First Example, Part 1

```
data _airpassengers;  
  infile cards;  
  input year $ number comma10.;  
cards;  
2011 23,073,700  
2010 39,253,999  
2009 37,338,942  
2008 37,234,592  
;  
run;
```

Our First Example, Part 2

```
options ls=64 ps=20;
data _null_;
  call symput('ls',getoption('ls'));
  call symput('titl1',"Passenger Traffic at San "
              "Francisco International Airport");
  call symput('titl2',"2008-20111");
data _null_;
  file "c:\temp\AirPassengerNumbers.txt"
      N=PAGESIZE PRINT LINESLEFT=11 HEADER=hdr
      NOTITLES NOFOOTNOTES;
  set _airpassengers end=eof;
  put @20 year @35 number comma10.;
  if eof then link ftr;
  return;
```

<continued next slide>

Our First Example, Part 3

```
hdr: put @((&ls-length("&titl1"))/2) "&titl1";
      put @((&ls-length("&titl2"))/2) "&titl2";
      put;
      put @35 "Passenger";
      put @20 "Year" @36 "Numbers";
      put @20 4*"-" @35 10*"-" ;
      return;
ftr: put #18 @1 &ls*"-" ;
      put @1 "1 January thru July numbers for 2011";
      put @1 "Reference: Airport Council "
           "International, December 8, 2011";
      return;
run;
```

Our First Example, Part 4

Passenger Traffic at San Francisco International Airport
2008-2011¹

Year	Passenger Numbers
2011	23,073,700
2010	39,253,999
2009	37,338,942
2008	37,234,592

¹ January thru July numbers for 2011

Reference: Airport Council International, December 8, 2011

Wrapping Lines of Text, Part 1

- Have a 200 character text with only 40 characters of space

```
%MACRO WRAP (  
  TXTVAR /*var or constant to be wrapped*/  
  ,SLPTTXT /*chars to split text*/  
  ,TXTLEN /*width of column*/  
  ,TXTINDENT /*indent n chars for second  
             and later lines*/  
  ,MAXLINES /*max number of lines to  
            allow*/  
);
```

Wrapping Lines of Text, Part 2

```
LENGTH _TMPTXT $32767 _&TXTVAR.1- _&TXTVAR.&MAXLINES
    $&TXTLEN;
INFORMAT _&TXTVAR.1- _&TXTVAR.&MAXLINES $char&TXTLEN..;
FORMAT _&TXTVAR.1- _&TXTVAR.&MAXLINES $char&TXTLEN..;
ARRAY _&TXTVAR.{&MAXLINES} $&TXTLEN;
_TMPTXT=&TXTVAR;
ENDPOS=0;
k=1;
```

Wrapping Lines of Text, Part 3

```
DO UNTIL (MISSING (STRIP (_TMPTXT)));
  IF K>1 THEN _TXTINDENT=input("&TXTINDENT",best.);
  ELSE _TXTINDENT=0;
  ENDPOS=(input(&TXTLEN.,best.)-_TXTINDENT) -
    INDEXC(REVERSE(SUBSTR(_TMPTXT,1,(input(&TXTLEN.,best.)-
      _TXTINDENT))),&SLPTTXT);
  IF (_TXTINDENT-1)>=0 then
    _&TXTVAR.{k}=repeat(' ',_TXTINDENT-1)||
      LEFT(SUBSTR(_TMPTXT,1,ENDPOS));
  ELSE _&TXTVAR.{k}=LEFT(SUBSTR(_TMPTXT,1,ENDPOS));
  _TMPTXT=strip(SUBSTR(_TMPTXT,ENDPOS+1));
  _&TXTVAR.0=k; k+1; END;
drop k _tmptxt endpos _TXTINDENT;
%MEND;
```

Wrapping Lines of Text, Part 4

- Example: Splitting variable AETERM text using space as a delimiter, each line a maximum of 30 characters, indent second and later lines 2 characters, allowing for a maximum of 10 variables to hold the text

```
%wrap(AETERM, " ", 30, 2, 10);
```

- Macro creates internal variables:
 - AETERM0 (number of lines for the text in question)
 - AETERM1-AETERM10 (variables to store split text)

Wrapping Lines of Text, Part 5

```
%wrap(AETERM," ",30,2,10);
%wrap(AEOUT," ",15,0,10);
_maxlines=max(1,_aeterm0,_aeout0);
if (linesleft-_maxlines)<5 then do;
    link ftr; /*Link to print footer text*/
    put _page_;
end;
array aza {*} aeterm1-aeterm10;
array azb {*} aeout1-aeout10;
put @1 subject @10 aza{1} $char30.
    @45 azb{1} $char15.;
do i=2 to _maxlines;
    put @10 aza{i} $char30. @45 azb{i} $char15.;
end;
```

Aligning Text, Part 1

```
%macro aligntxt(var,len,align,lenB);  
  %if (&lenB eq ) %then %let lenB=&len;;  
  informat &var $char&lenB.; format &var $char&lenB.;  
  if lengthn(&var)>&len then  
    put 'WAR' "NING: Len text in var &var > &len, " &var=;  
  else if ^missing(&var) then do;  
    %if %upcase(&align)=L %then %do;  
      &var=strip(&var); %end;  
    %else %if %upcase(&align)=C %then %do;  
      if &len=length(&var) then &var=strip(&var);  
      else &var=repeat(' ',  
        floor(((&len-length(&var))/2)-1))||strip(&var); %end;  
    %else %if %upcase(&align)=R %then %do;  
      if &len=length(&var) then &var=strip(&var);  
      else  
        &var=repeat(' ',&len-length(&var)-1)||strip(&var);  
    %end;  
  end;  
%mend aligntxt;
```


Aligning Text, Part 2

- Example: Right aligning a character variable AESTDTC 10 characters but allowing for 12 in the dataset

```
%aligntxt(AESTDTC,10,R,11);
```

- Macro code is in the paper

Decimal Alignment of Numbers

```
%macro alndat(vr /*Character field that contains number*/  
              ,dp /*Position of Decimal Point in Field*/);  
  attrib _vr length=$15 format=$char15. _rn format=$15.;  
  drop _rn;  
  if index(&vr, '.')>0 then do;  
    _rn=strip(scan(&vr,1, '.'));  
    _vr=repeat(' ',&dp-length(strip(_rn))-2)||strip(&vr);  
  end;  
  else  
    &vr=repeat(' ',&dp-length(strip(&vr))-2)||strip(&vr);  
%mend alndat;
```

- Usage

```
%alndat(WBC,5);
```

- Macro not in paper but available on Powerpoint and as download

Page x of y, Part 1

- In header section of DATA _NULL_ put 'Page x of y', usually top right of each page
- Read output file created and count number of 'Page x of y' occur
- Modify output file replacing 'Page x of y' with "Page <n> of <number of pages>"
- SETPAGE macro in paper has full code

Page x of y, Part 2

```
%macro SetPage(outrpt);  
  data _null_;  
  retain _pagenum 0;  
  infile "&outrpt" length=len end=eof lrecl=200;  
  input _txt $varying200. len;  
  if index(_txt,'Page x of y') then _pagenum+1;  
  if eof then  
    call symput('pgnum',compress(put(_pagenum,8.))));
```

Page x of y, Part 3

```
data _null_;
  attrib _txt length=$200 format=$char200.
  _txt2 length=$200 format=$char200.;
  infile "&outrpt" length=len SHAREBUFFERS;
  file "&outrpt" lrecl=200;
  input _txt $varying200. len;
  if index(_txt,'Page x of y') then do;
    _pagenum+1;
    _txt2='Page '||compress(put(_pagenum,4.))||" of &pgnum";
    _txt2l=length(strip(_txt2));
    _txt2=repeat(' ',17-_txt2l-1)||strip(_txt2);
    _txt=tranwrd(_txt,'Page xxxx of yyyy',_txt2); end; run;
%mend SetPage;
```

Our Next Example, Part 1

```
data _ae0;
  attrib PTNO length=$4 AETERM length=$100
  AESER length=$4 AESTDT length=$10 AEENDT length=$10
  AEACN length=$25 AEREL length=$25 AETXGR length=$25
  AEOUT length=$25;
  infile cards dlm="~";
  input PTNO $ AETERM $ AESER $ AESTDT $ AEENDT $ AEACN $
        AEREL $ AETXGR $ AEOUT $;
cards;
0001~HEARTBURN~NO~2009-02-21~2009-12-09~NO ACTION~NOT
RELATED~MODERATE~RESOLVED
0001~UPSET STOMACH~NO~2010-04-20~2011-04-09~NO
ACTION~POSSIBLE~MILD~RESOLVED
```

Our Next Example, Part 2

- Calculate max. width needed for a variable

```
data _null_;  
  array len {9} len1-len9;  
  retain len1-len9; *9 variables in dataset;  
  set _ae0 end=eof;  
  array ary {*} _character_;  
  do i=1 to dim(ary);  
    len(i)=max(len(i),lengthn(ary{i})); end;  
  if eof then do;  
    do i=1 to dim(ary);  
      vname=vname(ary(i)); *Get variable name in array;  
      put @1 vname= len(i)= ; end; end;  
run;
```

Our Next Example, Part 3

```
vname=PTNO len1=4  
vname=AETERM len2=34  
vname=AESER len3=3  
vname=AESTDT len4=10  
vname=AEENDT len5=10  
vname=AEACN len6=24  
vname=AEREL len7=11  
vname=AETXGR len8=8  
vname=AEOUT len9=22
```


Our Next Example, Part 4

Yum, Yum and Yum, Inc.
Protocol: Sweet Sauce II

Page x of y

Listing 8
Adverse Events

Subject Number	Adverse Event	Start Date	Stop Date	Serious?	Action Taken	Related to Study Drug?	Toxicity	Outcome
ccc	cccccccccccccccccccccccc	ddmmmyyyy	ddmmmyyyy	cc	ccccccccccc	cccccccccc	cccccccccccc	cccccccccccccccccccc

Program Name: AELIST.SAS Date/Time of Creation: ddmmmyyyy hh:mm



Our Next Example, Part 5

```
%let ls=134; %let ps=58;
options ls=&ls ps=&ps; *Set our output pagesize;
data _null_;
  retain np 0; *New page flag;
  file "c:\temp\aelist.txt" N=PAGESIZE print linesleft=11
    header=hdr notitles;
  set _ae0 end=eof;
  by ptno; *Have a blank line between each PTNO value;
  length _aestdtn _aeendtn 8;
  format _aestdtn _aeendtn date9.;
  _aestdtn=input(aestdt, yymmdd10.);
  _aeendtn=input(aeendt, yymmdd10.);
```

Our Next Example, Part 6

```
%WRAP(aeterm," ",26,2,3);
%WRAP(aeacn," ",12,0,3);
maxlines=sum(0,_aeterm0,_aeacn0);
if (ll-maxlines)<5 then do;
  link ftr; put _page_; end;
if first.ptno or np=1 then do;
  put; put @2 ptno $4. @@; np=0; end;
put @9 _aeterm1 $char26. @37 _aestdtn @48 _aeendtn @61
  aeser @69 _aeacn1 @83 aere1 @96 aetxgr @113 aeout;
if ^missing(_aeterm2) or ^missing(_aeacn2) then
  put @9 _aeterm2 $char26. @69 _aeacn2;
if ^missing(_aeterm3) or ^missing(_aeacn3) then
  put @9 _aeterm3 $char26. @69 _aeacn3;
if eof then link ftr; return;
```

Our Next Example, Part 7

```
hdr: put @1 'Yum, Yum and Yum, Inc.`
      @(&ls-length("Page x of y")+1) "Page x of y";
put @1 'Protocol: Sweet Sauce II';
put @((&ls-length("Listing 8"))/2) "Listing 8";
put @((&ls-length("Adverse Events"))/2)
      "Adverse Events";
put; put @1 &ls*'-' ;
put @1 'Subject' @39 'Start' @83 'Related to';
put @1 'Number' @9 'Adverse Event' @39 'Date`
      @48 'Stop Date' @59 'Serious?' @69 'Action Taken`
      @83 'Study Drug?' @96 'Toxicity` @113 'Outcome';
put @1 &ls*'-' ;
return;
```

Our Next Example, Part 8

```
ftr: put #(&ps-1) @1 &ls*'-' ;  
      put "Program Name: AELIST.SAS "  
          "Date/Time of Creation: &sysdate9. &systeme." ;  
      np=1 ;  
      return ;  
run ;
```

Our Next Example, Part 9

Yum, Yum and Yum, Inc.
Protocol: Sweet Sauce II

Page xxxx of yyyy

Listing 8
Adverse Events

Subject Number	Adverse Event	Start Date	Stop Date	Serious?	Action Taken	Related to Study Drug?	Toxicity	Outcome
0001	HEARTBURN	21FEB2009	09DEC2009	NO	NO ACTION	NOT RELATED	MODERATE	RESOLVED
	NAUSEA	26APR2009	17MAY2009	NO	NO ACTION	NOT RELATED	MILD	RESOLVED
	NAUSEA	30OCT2009	28AUG2010	NO	NO ACTION	NOT RELATED	MILD	RESOLVED
	BLACK AND BLOODY STOOLS	12APR2010	15APR2010	YES	TEMPORARILY DISCONTINUED	POSSIBLE	SEVERE	RESOLVED
	UPSET STOMACH	20APR2010	09APR2011	NO	NO ACTION	POSSIBLE	MILD	RESOLVED
0002	UPSET STOMACH	22JUN2008	18APR2009	NO	NO ACTION	NOT RELATED	MODERATE	RESOLVED
	FLATULENCE	10JUN2009	10JUN2010	NO	NO ACTION	PROBABLE	MILD	ONGOING
	BELCHING	23FEB2010	24FEB2010	NO	NO ACTION	NOT RELATED	MILD	RESOLVED
	HEARTBURN	25MAY2010	09OCT2010	NO	NO ACTION	NOT RELATED	MILD	RESOLVED
0003	NAUSEA	19DEC2008	18JUL2009	NO	NO ACTION	NOT RELATED	MILD	RESOLVED
	SWELLING OF THE MOUTH	23MAY2009	16MAR2010	YES	TEMPORARILY DISCONTINUED	NOT RELATED	SEVERE	RESOLVED
0004	ITCHING	26NOV2009	10DEC2009	NO	NO ACTION	NOT RELATED	MILD	ONGOING
0005	FATIGUE	26JUN2008	25JAN2009	NO	NO ACTION	PROBABLE	MODERATE	RESOLVED
	HEARTBURN	09APR2009	28APR2009	NO	NO ACTION	NOT RELATED	MILD	RESOLVED
	RASH	22APR2009	06OCT2009	NO	NO ACTION	NOT RELATED	MILD	RESOLVED

Our Next Example, Part 10

- `%SetPage(%str(c:\temp\aelist.txt));`

Yum, Yum and Yum, Inc.
Protocol: Sweet Sauce II

Page 1 of 2

Listing 8
Adverse Events

Subject Number	Adverse Event	Start Date	Stop Date	Serious?	Action Taken	Related to Study Drug?	Toxicity	Outcome
0001	HEARTBURN	21FEB2009	09DEC2009	NO	NO ACTION	NOT RELATED	MODERATE	RESOLVED
	NAUSEA	26APR2009	17MAY2009	NO	NO ACTION	NOT RELATED	MILD	RESOLVED

Adding a line of text under a Record, Part 2

```
%WRAP (aeacn, " ", 12, 0, 3);
%WRAP (comment, " ", 112, 0, 2);
maxlines=sum(0, _aeterm0, _aeacn0);
if ^missing(comment) then
    maxlines=sum(maxlines, _comment0);
if (ll-maxlines)<5 then do;
    <other lines of code>;
if ^missing(_aeterm3) or ^missing(_aeacn3) then
    put @9 _aeterm3 $char26. @69 _aeacn3;
if ^missing(comment) then do;
    put @14 'Comment:' @23 _comment1 $char112.;
    if ^missing(_comment2) then put @23 _comment2 $char112.;
end;
```

Adding a line of text under a Record, Part 3

0002	UPSET STOMACH	22JUN2008	18APR2009	NO	NO ACTION	NOT RELATED	MODERATE	RESOLVED
	Comment: SUBJECT ATE 6KG OF CHOCOLATE AFTER BREAKUP WITH PARTNER							
	FLATULENCE	10JUN2009	10JUN2010	NO	NO ACTION	PROBABLE	MILD	ONGOING
	BELCHING	23FEB2010	24FEB2010	NO	NO ACTION	NOT RELATED	MILD	RESOLVED
	HEARTBURN	25MAY2010	09OCT2010	NO	NO ACTION	NOT RELATED	MILD	RESOLVED
0003	NAUSEA	19DEC2008	18JUL2009	NO	NO ACTION	NOT RELATED	MILD	RESOLVED
	SWELLING OF THE MOUTH	23MAY2009	16MAR2010	YES	TEMPORARILY DISCONTINUED	NOT RELATED	SEVERE	RESOLVED
0004	ITCHING	26NOV2009	10DEC2009	NO	NO ACTION	NOT RELATED	MILD	ONGOING
0005	FATIGUE	26JUN2008	25JAN2009	NO	NO ACTION	PROBABLE	MODERATE	RESOLVED
	HEARTBURN	09APR2009	28APR2009	NO	NO ACTION	NOT RELATED	MILD	RESOLVED
	RASH	22APR2009	06OCT2009	NO	NO ACTION	NOT RELATED	MILD	RESOLVED
	HIVES	24AUG2009	09DEC2009	NO	NO ACTION	NOT RELATED	MILD	ONGOING
	HEARTBURN	12DEC2009	06MAY2010	NO	NO ACTION	NOT RELATED	MILD	RESOLVED
	DROWSINESS	28DEC2009	08OCT2010	NO	NO ACTION	NOT RELATED	MILD	RESOLVED
	DROWSINESS	04JAN2010	01DEC2010	NO	NO ACTION	NOT RELATED	MILD	ONGOING
	NAUSEA	07APR2011	20NOV2011	NO	NO ACTION	NOT RELATED	MILD	RESOLVED
0006	HEARTBURN	06MAR2009	17JUL2009	NO	NO ACTION	NOT RELATED	MILD	RESOLVED
	ITCHING	26APR2010	13JUL2010	NO	NO ACTION	NOT RELATED	MILD	ONGOING
	BIRD BITE	29MAY2010	01JUL2010	NO	NO ACTION	NOT RELATED	MILD	RESOLVED
	Comment: BIRD WAS A PET OF THE SUBJECT. BIRD WELL CARED FOR AND IT IS BELIEVED THAT SHOTS FOR BIRD WERE UP TO DATE SO NO EXAMINATION NEEDED BY A VET TO SEE IF BIRD WAS ILL.							
0007	DIZZINESS	15NOV2008	29SEP2009	NO	NO ACTION	NOT RELATED	MILD	RESOLVED

Revisiting TXT2RTF Macro, Part 1

- At PharmaSUG 2010, a macro was presented that took a text file and put it into a RTF file
- One small issue with the ‘_’ character – it is there in the file but is not visible (screen issue)

```
%macro txt2rtf(_txtfn= /*Text file to convert to RTF*/  
,_rtffn= /*RTF file that is to be output to*/  
,_pgwh=15840 /*Page width in twips*/  
,_pght=12240 /*Page height in twips*/  
,_mgns=1440 /*Margin width in twips*/  
,_lspe=%str(\landscape) /*Landscape, need to add this*/  
,_fnsz=8 /*Font point size - integer*/  
,_cndns=Y /*Condensed Y|N*/ );
```

Revisiting TXT2RTF Macro, Part 2

```
data _null_;
infile "&_txtfn" end=eof;
length _txt $200; input; file "&_rtffn";
if _n_=1 then do;
  put '{\rtf1\ansi\ansicpg1252\deff0\deflang1033' /
  '{\fonttbl{\f0\fmodern\fprql\fcharset0 Courier New;}}' /
  '{\colortbl \red0\green0\blue0;}' /
  "\paperw&_pgwh.\paperh&_pght.\margl&_mgns.\margr&_mgns." /
  "\margt&_mgns.\margb&_mgns.&_lspe." /
  "\viewkind4\uc1\pard\ql\fi0\li0\ri0\sb0\sa0" @@;
  %if %upcase(&_cndns)=Y %then put "\s1-%eval(&_fnsz*20)";
  %else put;;
  put "\cf0\f0\fs%eval(&_fnsz*2) " _infile_ end;
else do; if substr(_infile_,1,1)=byte(12) then do;
  _txt='\page '||substr(_infile_,2); put _txt; end;
else if ^eof then put '\par ' _infile_;
else if eof then put '\par ' _infile_ '}}'; end; run;
```

Revisiting TXT2RTF Macro, Part 3

- LS and PS settings with `_CNDNS` option

Paper Size	Font Size	PAGESIZE <code>_cndns=Y</code>	PAGESIZE <code>_cndns=N</code>	LINESIZE
LETTER	8	58	51	134
LETTER	9	52	45	119
LETTER	10	46	41	104
A4	8	56	49	145
A4	9	50	44	129
A4	10	45	39	116

Contact Information



Name: David Franklin

Enterprise: TheProgrammersCabin.com

City, State ZIP: Litchfield, NH 03052

Work Phone: 603-275-6809

E-mail: dfranklin@TheProgrammersCabin.com

Web:

<http://www.TheProgrammersCabin.com>
[#pharmasug](#)