

```
Gastrointestinal disorders
  Nausea
  Gastrooesophageal reflux
    disease
  Vomiting
  Gastric ulcer
    perforation, obstructive

Neoplasms benign, malignant
and unspecified (incl cysts
and polyps)
  Adult T-cell lymphoma/
    leukaemia stage I

Vascular disorders
  Orthostatic hypotension
```

Indenting Lengthy Text Strings in PROC REPORT Output

Jeppe Rømer Juul (jxju@novonordisk.com)

Kavita Math (kvsm@novonordisk.com)

8/6/21

- **The Issue: Indentation using FLOW**
- **The Solution: The Indent Macro**
- **Macro parameters**
- **Macro call**
- **Macro algorithm**
- **Glimpse of the Macro**

The Issue: Indentation using FLOW

Adverse events by system organ class and preferred term - summary

	Drug A			Drug B		
	N	(%)	E	N	(%)	E
Gastrointestinal disorders	3	(30.0)	5	6	(60.0)	17
Gastrooesophageal reflux	3	(30.0)	5	5	(50.0)	6
disease						
Gastric ulcer	0	(0.0)	0	1	(10.0)	11
perforation, obstructive						
Neoplasms benign, malignant and unspecified (incl cysts and polyps)	0	(0.0)	0	1	(10.0)	11
Adult T-cell lymphoma/leukaemia stage I	0	(0.0)	0	1	(10.0)	11

The Issue: Indentation using FLOW

When lengthy text strings need to be indented in PROC REPORT outputs FLOW option allows the user to wrap the text to the next lines in a specified column.

However

- FLOW option does not carry the spaces used for indentation to the wrapped lines.
- Long words are not hyphenated when they are split across two lines
- Cannot control the line breaks if it's desired at a specific position in case if the text is having characters like / or -

Using Indent Macro

Adverse events by system organ class and preferred term - summary

	Drug A			Drug B		
	N	(%)	E	N	(%)	E
Gastrointestinal disorders	3	(30.0)	5	6	(60.0)	17
Gastrooesophageal reflux disease	3	(30.0)	5	5	(50.0)	6
Gastric ulcer perforation, obstructive	0	(0.0)	0	1	(10.0)	11
Neoplasms benign, malignant and unspecified (incl cysts and polyps)	0	(0.0)	0	1	(10.0)	11
Adult T-cell lymphoma/leukaemia stage I	0	(0.0)	0	1	(10.0)	11

Solution: The Indent Macro

- Inserts spaces and SPLIT characters in text strings based on the desired column width
- Controls where line break can occur by finding the last SPACE or other CHARACTER (/ -)
- Retains spaces used for indentation and additionally hyphenate words that break up with the remaining characters on the next line
- Flexible to handle different indentations for different rows of text
- Produces a well indented more desirable output for submission

Macro Parameters

The macro needs to be called within a data step prior to using PROC REPORT

Parameter	Description
INVAR	Name of input variable with text to indent
OUTVAR	Name of output variable with indented text
LINESIZE	Width of the column to be indented as in PROC REPORT
INDENT	Number of spaces to indent (can be specified as a number OR name of a numeric variable)
DELIMITER	The split character used in PROC REPORT. Default is

Macro call - Example 1

Example shows the macro call within a data step where the value of indent is specified as a number.

In this case all lines will have same indentation 1

```
data example_call;  
have = "This long and incomprehensible text shows/displays how it works";  
    %miscindent(invar      = have  
                ,outvar    = want  
                ,linesize = 17  
                ,indent    = 1);  
run;
```

Macro call - Example 2

Example shows the macro call where the value of indent is specified in a dataset
This allows different indentation for different rows.

The macro expects **INDENTVAR** to be present in data set before it's called

```
data example_call;  
have = "This long and incomprehensible text shows/displays it works";  
    %miscindent(invar      = have  
                ,outvar    = want  
                ,linesize = 17  
                ,indent     = indentvar) ;  
run;
```

Macro Algorithm

Starts with assigning `INVAR = REMANING_TEXT`

- Do loop starts while remaining text is not missing
 - Identifies position for splitting based on last space/character/hyphenation
 - Splitting based on last space/character (hyphen, backslash)
 - Splitting by hyphenation
- Do loop ends when remaining text is empty
- Final OUTVAR including spaces and split characters

Macro Algorithm

Let us investigate a simple example to understand the algorithm

`have = "This long and incomprehensible text shows/displays how it works"`

`want = " This long and| incomprehensib-| le text shows/| displays how
it| works| "`

INDENT macro prior to PROC REPORT	FLOW with PROC REPORT
<code>This long and incomprehensib- le text shows/ displays how it works</code>	<code>This long and inc omprehensible tex t shows displays how it works</code>

First iteration of DO loop

REMANING_TEXT = INVAR



REMANING_TEXT = “**This long and in**comprehensible text shows/displays how it works“

Variable/Parameter	Value
LINESIZE	17
INDENT	1
LAST_SPACE_INDEX	14
OUTVAR	" This long and"
REMANING_TEXT	"incomprehensible text shows/displays how it works"

Next iteration of DO loop



REMANING_TEXT= “ **incomprehensib**le text shows/displays how it works“

Variable/Parameter	Value
LINESIZE	17
INDENT	2
LAST_SPACE_INDEX	0
OUTVAR	" This long and incomprehensib-
REMANING_TEXT	"le text shows/displays how it works"

Next iteration of do loop



REMANING_TEXT= “le text shows/displays how it works”

Variable/Parameter	Value
LINESIZE	17
INDENT	2
LAST_SPACE_INDEX	14
OUTVAR	" This long and incomprehensib- le text shows/"
REMANING_TEXT	"displays how it works"

Next iteration of do loop



REMANING_TEXT= “**displays how it** works“

Variable/Parameter	Value
LINESIZE	17
INDENT	2
LAST_SPACE_INDEX	16
OUTVAR	" This long and incomprehensib- le text shows/ displays how it"
REMANING_TEXT	"works"

Next iteration of do loop

REMANING_TEXT= "works"

Variable/Parameter	Value
LINESIZE	17
INDENT	2
LAST_SPACE_INDEX	17
OUTVAR	" This long and incomprehensib- le text shows/ displays how it works"
REMANING_TEXT	" "

Exits DO loop

REMANING_TEXT= " "

The delimiter from the start of the OUTVAR is inserted at the end

Variable/Parameter	Value
OUTVAR	" This long and incomprehensib- le text shows/ displays how it works "

If displayed using proc report, this will be formatted as:

```
This long and  
incomprehensib-  
le text shows/  
displays how it  
works
```

```

1 macro miscindent(invar =, outvar =, linesize =, indent =, delimiter = | );
2   length remaining_text &outvar. $2000;
3   remaining_text = &invar.;
4
5   * Add lines in outvar as long as there is remaining text that has not been copied from invar;
6   do while( lengthn(remaining_text) > 0);
7
8       * Indent by 1 more space after a had a line break, if the line is indented to start with;
9       new_indent = &indent.+(&indent.>0)*(&remaining_text ne &invar.);
10
11      * Find the last space (or other character), where the text will get a line break;
12      last_space_index = max(find(remaining_text, ' ', -(&linesize.-new_indent+1))
13                             ,find(remaining_text, '-', -(&linesize.-new_indent))
14                             ,find(remaining_text, '/', -(&linesize.-new_indent)));
15
16      * Transfer text to outvar and let remaining text take one more loop;
17      if last_space_index > 1 then do;
18          &outvar. = trim(&outvar.)
19              ||ifc(new_indent = 0,"&delimiter.", "&delimiter."||repeat(' ', max(0,new_indent-1)))
20              ||substr(remaining_text,1,last_space_index);
21          remaining_text = left(substr(remaining_text,last_space_index+1));
22      end;
23      else do;
24          &outvar. = trim(&outvar.)
25              ||ifc(new_indent = 0,"&delimiter.", "&delimiter."||repeat(' ', max(0,new_indent-1)))
26              ||substr(remaining_text,1,&linesize.-new_indent-1)
27              ||'-';
28          remaining_text = left(substr(remaining_text,&linesize.-new_indent));
29      end;
30  end;
31
32  * Remove the delimiter from the start of the outvar text, and add one at the end;
33  &outvar. = trim(substr(&outvar.,2+lengthn("&delimiter."))||"&delimiter.");
34  drop remaining_text last_space_index new_indent;
35 %mend;

```

Thank You

Acknowledgements

Durga Ramana, Girish J Rao, Jothi Krishan, Jesper Zeth, Kasper Klitgaard Berthelsen, Lars Erichsen, Lea Nørgreen Gustafsson, Michael Hedegaard Thomsen, Mikkel Willemoes, Rasmus Stenholt, Roshan Stephen, Srinivasa Dass, Sonali Das ,Vikram BR