OpenDocument spreadsheet

All multiline directives such as #if, #else, #elseif, #foreach, and #macro must be used under the following conditions:

1. The beginning and ending statements must be declared within a single cell. The two figures below show two samples of invalid usage of the multiline statements between cells.

	A	B
1	#if ($e.elementType == "usecase")	
2	$e.name	
3	#end	
4		

Figure 1: A sample of an invalid multiline statement.

	A	B	C	D
1				
2		Use Case		
3		#foreach ($uc in $UseCase)	$name	#end
4				

Figure 2: Another sample of an invalid multiline statement.

> ✓ Since the body of the **#if** statement is contained in the cell A2, this cell will not be generated if the condition is not true (the element type is not "usecase"). Due to the constraints of spreadsheet document structure, the number of cells in a column must be equal to the number of cells in all columns, and the number of cells in a row must be equal to the number of cells in all rows.
>
> The codes in the figure above will break the structure of a spreadsheet document. The two figures below demonstrate two samples of valid usage of the multiline statements.

	A	B
1	#if ($e.elementType == "usecase")$e.name#end	
2		

Figure 3: A sample of a valid usage of multiline statements.

	A	B	C
1			
2		Use Case	
3		#foreach ($uc in $UseCase) $name #end	
4			

Figure 4: Another sample of a valid usage of multiline statements.

2. *VTL* Macro must be declared within a single cell. Do not insert the multi-cell recorded macros in a single cell, see figure below.

	A	B	C
1			
2		Macro	
3		#macro(insertCell $e)	
4		#if($e=="red")$e.name	
5		#else $e.name	
6		#end	
7		#end	
8			
9		Use Macro	
10		#foreach($e in $elements)	
11		#insertCell($e)	
12		#end	
13			

Figure 5: A sample of an invalid usage of a VTL macro.

The macro will copy all contents between **#macro** and **#end**. Cells and rows will be included in the macro as well. Once this record has been inserted, the macro content will break the document's structure.

Creating Data for Multiple Rows

The **#foreach** directive can only be used in a single cell record. To create data for multiple rows, use the **#forrow** directive instead, see figure below.

	A	B	C
1	#forrow($uc in $UseCase)	$uc.name	#endrow

Figure 6: The forrrow directive being used to create data for multiple rows.

As shown in figure below, the output will be:

	A	B	C
1		Use Case A	
2		Use Case B	
3		Use Case C	
4		Use Case D	
5			

Figure 7: The results of running the code from figure 6.

Creating Data for Multiple Columns

#forcol is used to create data for multiple columns, see figure below. This statement can be used with **#forrow** (see the sample from the "Other Document" template).

	A	B
1	#forcol($uc in $UseCase)$uc.name#endcol	
2		

Figure 8: The forcol directive being used to create data for multiple columns.

From the usage of #forcol shown in figure below, the output will be:

	A	B	C	D	E
1	Use Case A	Use Case B	Use Case C	Use Case D	
2					

Figure 9: The results of running the code from figure 8.