Conditional Statements

Traffic Lighting

cellstyle expr-1 as style-element(style-attributes),
    expr-2 as style-element(style-attributes),
    ...
    expr-n as style-element(style-attributes);

Execution stops.

WHERE expression.

expression that is evaluated for each item/
paragraph value in the output.

The result of the expression is true, the
given style element and style
attributes are applied. Once a matching
expression is found, execution stops.

Translating Values

translate expr-1a into expr-1b,
    expr-2a into expr-2b,
    ...
    expr-na into expr-nb;

The TRANSLATE-INTO statement is valid
at the top-level of both PROC ODSLIST and
PROC ODSTEXT as well as within list and
item blocks. Each expression is a WHERE
expression that is evaluated for each item/
paragraph. If the result of the expression
is true, the given style element and style
attributes are applied. Once a matching
expression is found, execution stops.

Example

/* static text */
proc odstext;
    p 'Class Information' /
        style={fontweight=bold};
    p 'The following is height, weight,
        and age information from the class
        data set.';
    run;

/* data dependent text / list */
proc odstext data=sashelp.class(obs=2);
    p 'Student name is ' || name || ' . '
        style={fontweight=bold};
    list;
        item 'Weight - ' || put(weight,3.);
        item 'Height - ' || put(height,2.);
    end;
    run;

/* item block with nested list */
proc odstext data=sashelp.class(obs=2);
    item / style={liststyletype=none};
    p name || ' is ' || "'s age" /
        style={fontweight=bold};
    list;
        cellstyle age > 13 as
            {fontstyle=italic};
        translate age = 13 into
            'Thirteen';
        item age / format=2.;
    end;
    run;

Class Information
The following is height, weight, and age information from the class data set.

Student name is Alice.
  - Weight: 113
  - Height: 60

Student name is Alfred.
  - Weight: 61
  - Height: 57

Alfred's age = 74
Alice's age = Thirteen

For complete information, refer to the
Base SAS® 9.4 documentation at
http://support.sas.com/v9doc

ODS List and Text Block Tip Sheet

This tip sheet presents the most common
statements and options used in creating lists
and text blocks.

PROC ODSLIST and PROC ODSTEXT allow you
to create bulleted lists and formatted blocks of
text in your reports. These lists and text blocks
can be static content (i.e., strings) or based on
data from a data set.

This tip sheet places frequently used
information in one place, on one sheet of
paper, so you don't have to search through the
online documentation. It also gives you
something to take home, type in, and try.
PROC ODSLIST

Basic Usage

.proc odslst data=data-set;
... statements ...
run;

Procedure Options

contents="string"
specifies the string to be displayed in the
table of contents for the output object.

data="data-set"
specifies the data set to use for variable
references in items and paragraphs. Data=
is not needed if there are no variable
references.

name=template-name
specifies the name of the template to store.
This option enables you to store list
templates for reuse with the DATA step.

pagebreak=yes | no | on | off
specifies whether or not the procedure
should generate page breaks. The default is
"no".

store=template-store
specifies the template store to save the
template in if name= is also used.

print
specifies that the output object should be
printed. This is for use when name= is
specified on static lists (i.e., lists with no
variable references or data set).

Common ODSLIST Statements

item expression </ / option(s) >;
specifies the content of the item.

item / </ option(s) >;
... item block ...
end;
specifies a block of content for an item.
This is used for using multiple paragraphs
and/or creating nested lists.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>format=</td>
<td>SAS data format</td>
</tr>
<tr>
<td>style=</td>
<td>Style override</td>
</tr>
<tr>
<td>value=</td>
<td>Numeric value of bullet</td>
</tr>
</tbody>
</table>

Item Block Statements

list </ / option(s) >;
... list block ...
end;
specifies the content of the nested list. The
list block statements are the same as for
PROC ODSLIST (see above).

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>start=</td>
<td>Starting value for bullet numbering</td>
</tr>
<tr>
<td>style=</td>
<td>Style override</td>
</tr>
</tbody>
</table>

p expression </ / option(s) >;
specifies the content of the paragraph.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>format=</td>
<td>SAS data format</td>
</tr>
<tr>
<td>style=</td>
<td>Style override</td>
</tr>
</tbody>
</table>

PROC ODSTEXT

Basic Usage

.proc ods txt data=data-set;
... statements ...
run;

Procedure Options

contents="string"
specifies the string to be displayed in the
table of contents for the output object.

data="data-set"
specifies the data set to use for variable
references in items and paragraphs. Data=
is not needed if there are no variable
references.

name=template-name
specifies the name of the template to store.
This option enables you to store list
templates for reuse with the DATA step.

pagebreak=yes | no | on | off
specifies whether or not the procedure
should generate page breaks. The default is
"no".

store=template-store
specifies the template store to save the
template in if name= is also used.

print
specifies that the output object should be
printed. This is for use when name= is
specified on static lists (i.e., lists with no
variable references or data set).

Common ODSTEXT Statements

list </ / option(s) >;
... list block ...
end;
specifies the content of the nested list. The
statements are the same as for PROC
ODSLIST.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>start=</td>
<td>Starting value for bullet numbering</td>
</tr>
<tr>
<td>style=</td>
<td>Style override</td>
</tr>
</tbody>
</table>

p where-expression </ / option(s) >;
specifies the content of the paragraph.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>format=</td>
<td>SAS data format</td>
</tr>
<tr>
<td>style=</td>
<td>Style override</td>
</tr>
</tbody>
</table>

List Block Statements

item expression </ / option(s) >;
specifies the content of the item.

item / </ option(s) >;
... item block ...
end;
specifies a block of content for an item.
This is used for using multiple paragraphs
and/or creating nested lists.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>format=</td>
<td>SAS data format</td>
</tr>
<tr>
<td>style=</td>
<td>Style override</td>
</tr>
<tr>
<td>value=</td>
<td>Numeric value of bullet</td>
</tr>
</tbody>
</table>