SAS 9 ODS Report Writing Interface Tip Sheet

Table Example

```
data _null_;
  set sashelp.class(obs=5) end=done;
  /* do header only at beginning */
  if _N_ = 1 then do;
    /* create odsout object */
    declare odsout o();

    /* start table */
    o.table_start();
      o.head_start();
        o.row_start();
          o.format_cell(
            data: "Name, Gender,
                   Age & BMI",
            column_span: 4);
        o.row_end();
      o.head_end();
      o.body_start();
  end;
  bmi = (weight/(height*height))*703;
  o.row_start();
    o.format_cell(data: name);
    if sex = "F" then
      o.format_cell(data: "Female");
    else
      o.format_cell(data: "Male");
    o.format_cell(data: age);
    o.format_cell(data: bmi,
              format: "8.2");
  o.row_end();

  if done then do;
    o.body_end();
    o.table_end();
  end;
run;
```

Name, Gender, Age & BMI			
Alfred	Male	14	16.61
Alice	Female	13	18.50
Barbara	Female	13	16.16
Carol	Female	14	18.27
Henry	Male	14	17.87

Layout Example

```
data _null_;
  set sashelp.cars(obs=80) end=done;
  by make;

if _N_ = 1 then do;
dcl odsout o();
/* start absolute layout */
o.layout_absolute(
  style_attr: "background=lightgray");
  o.region(x: "0.5in",
    style_attr: "background=gray");
    o.format_text(data: "Region 1",
      style_attr: "background=gray");
  o.region(width: "2in", height: "1in",
    y:"1in", style_attr:"background=blue");
    o.format_text(data:"Region 2", just:"c",
      style_attr:"background=blue
                  color=white");
o.layout_end();

o.page();


/*start gridded layout */
o.layout_gridded(columns: 3);
end;

if first.make then do;
  o.region(style_attr: "background=ligr");
  o.format_text(data: make,
    style_elem: "SystemTitle");
  o.format_text(data: model);
end;

if done then do;
  o.layout_end();
end;
run;
```
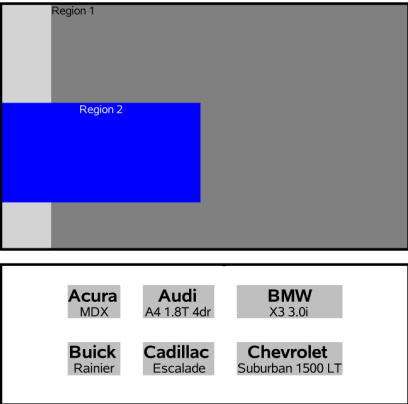

This tip sheet places frequently used information in one place, on one sheet of paper, so you don't have to search through the online documentation. It also gives you something to take home, type in, and try.

The ODS Report Writing Interface allows you to create highly customized reports from within the DATA step. As it is fully integrated with the ODS system, it lets you combine the powerful programming features from the DATA step with ODS features giving you flexibility and control for every piece of output in your report. The ODS Report Writing Interface is currently supported for the HTML and PDF output destinations.

For complete information, refer to the Base SAS® documentation at
support.sas.com/base

SAS | THE POWER TO KNOW®

SAS 9 ODS Report Writing Interface Tip Sheet

Common Arguments

These arguments are common to all methods.

style_elem : '*style-element*-name'
specifies the style element to apply. The value can also be a character variable.

style_attr : '*style-attribute(s)*'
specifies style overrides to apply. The value can also be a character variable.

Table Methods

Methods that come in _start / _end pairs, must both be invoked for the item to be well formed.

obj.**table_start(*arguments*)**
obj.**table_end( )**
start or end a table, respectively.

obj.**head_start(*arguments*)**
obj.**head_end( )**
start or end a table header section, respectively.

obj.**body_start(*arguments*)**
obj.**body_end( )**
start or end a table body section, respectively.

obj.**foot_start(*arguments*)**
obj.**foot_end( )**
start or end a table footer section, respectively.

obj.**row_start(*arguments*)**
obj.**row_end( )**
start or end a row, respectively.

    type : 'H' | 'B' | 'F'
    specifies the type (header, body or footer) of the table row. Default is body. This is used as an alternative to the head, body, and foot methods.

Table Methods
(continued)

obj.**format_cell(*arguments*)**
    or
obj.**cell_start(*arguments*)**
obj.**cell_end( )**
create a table cell.

    data : '*string*' | *number* | *variable-name*
    specifies the value to display in the table cell. This is a required argument for this method.

    format : '*sas-format-name*'
    specifies the format used to display the cell value.

    just : 'L' | 'C' | 'R'
    specifies the horizontal alignment of the cell.

    vjust : 'T' | 'M' | 'B'
    specifies the vertical alignment of the cell.

    inhibit : 'T' | 'B' | 'L' | 'R' | 'X'
    specifies the cell borders that should not be displayed. *(Printer destinations only)*

    height : '*dimension*'
    specifies the vertical height of the cell.

    width : '*dimension*'
    specifies the horizontal width of the cell.

    column_span : *integer*
    specifies the number of columns that the cell will occupy. The default value is 1.

    row_span : *integer*
    specifies the number of rows that the cell will occupy. The default value is 1.

    inline_attr : '*style attribute(s)*'
    overrides specific style attributes for the preceding **data** argument only.

    inline_elem : '*style-element-name*'
    changes the style element for the preceding **data** argument only.

Text Methods

obj.**format_text(*arguments*)**
obj.**note(*arguments*)**
create text output or note, respectively.

    data : '*string*' | *number* | *variable-name*
    specifies the value to display. This is a required argument for this method.

    format : '*sas-format-name*'
    specifies the format used to display the text.

    just : 'L' | 'C' | 'R'
    specifies the horizontal alignment of the text.

    vjust : 'T' | 'M' | 'B'
    specifies the vertical alignment of the text.

    inline_attr : '*style attribute(s)*'
    overrides specific style attributes for the preceding **data** argument only.

    inline_elem : '*style-element-name*'
    changes the style element for the preceding **data** argument only.

Page Methods

obj.**page( )**
force a page break.

obj.**title(*arguments*)**
obj.**footnote(*arguments*)**
add a title or footnote to the system.

    text : '*string*' | *variable-name*
    specifies the text value of the title / footnote. This is a required argument for this method.

    start : *integer*
    specifies the title / footnote index to start with. Valid numeric range is 1–10.

Absolute Layout Methods
(Printer destinations only)

obj.**layout_absolute(*arguments*)**
obj.**layout_end( )**
start and end an absolute layout, respectively.

    x : '*dimension*'
    y : '*dimension*'
    specifies the x and y starting point.

    height : '*dimension*'
    width : '*dimension*'
    specifies the height and width.

obj.**region(*arguments*)**
start a new region within the layout. Arguments are same as for **layout_absolute( ).**

Gridded Layout Methods

obj.**layout_gridded(*arguments*)**
obj.**layout_end( )**
start and end a gridded layout, respectively.

    columns : *integer*
    rows : *integer*
    specifies the number of columns and rows.

    height : '*dimension*'
    width : '*dimension*'
    specifies the height and width.

obj.**region(*arguments*)**
start a new region within the gridded layout.

    column_span : *integer*
    specifies the number of columns that the region will occupy. The default value is 1.

    row_span : *integer*
    specifies the number of rows that the region will occupy. The default value is 1.