The New SAS ODS Excel Destination: A User Review and Demonstration

LeRoy Bessler PhD
Mequon, Wisconsin, USA – Le_Roy_Bessler@wi.rr.com

Copyright 2015. All rights reserved.

Abstract

The common destination for SAS-prepared results is often an Excel workbook. Everyone already has Excel and knows how to use it, to reformat or further explore their results however they wish. But SAS programmers want only one SAS tool to meet all of their highly formatted Excel reporting needs. Twice since June 2013, I have provided vigorous comparisons of five (Yes, 5) different ways of Excel Reporting from SAS: HTML, ExcelXP, MSOffice2K_x, TableEditor, and Dynamic Data Exchange (DDE). Deemed by me as inadequate compared to MSOffice2K_x, I omitted the MSOffice2K destination, but even it has its distinct advantages. DDE is clearly the strongest solution, and I still get requests for my DDE ToolKit from around the world, but it can be clumsy to work with if you have to run SAS on a remote server (especially via SAS Enterprise Guide), rather than on your own PC). The new SAS ODS Excel destination, which is available in pre-production experimental status in SAS 9.4 TS1M2, is intended to provide relief from the too many partial SAS solutions and to add capabilities not available in any of the partial solutions. It will not have the disadvantages of DDE. As of the date of this abstract, there is no formal documentation available. I show you what I have been able to do with the new ODS Excel technology, including production status features available in SAS 9.4 TS1M3, which is expected to be available in July 2015, and am providing herein a user-created substitute for currently unavailable vendor documentation. I’m really excited about this long desired solution to an important and popular need. The Excel files created by ODS Excel can be opened and viewed with Excel 2010+ or OpenOffice.org/LibreOffice 3.3+. The ODS Excel capability does not require Excel to be installed on the machine that creates ODS Excel output. An interesting feature in TS1M3 is the ability to use ODS Excel and the new PROC MSCHART to produce Excel Charts in a spreadsheet with SAS. Note that PROC MSCHART is pre-production experimental status in TS1M3. PROC MSCHART does not require Excel to be installed on the machine to produce Excel Charts in a spreadsheet with SAS. You can use ODS EXCEL and/or PROC MSCHART running SAS on MVS, Unix, Linux, or Windows.

Introduction

When I was still consulting in the winter of 2012-2013, I was working on an application where, besides graphs, I wanted to provide lots of Excel spreadsheets, linked backwards and forwards with the graphs, and to let summary spreadsheets have links to detail spreadsheets for each summary row. And I wanted a lot of control over spreadsheet features and format. My frustration was that there was no single SAS solution to address ALL of my spreadsheet function and feature needs. After this experience, I made two successive (published) forays into comparing
the capabilities of the various SAS-provided tools. In the Options Available Prior to the ODS Excel Destination section below, I summarize my conclusions.

My first in-depth adventure, ten years earlier, with using SAS to create highly formatted reports that can be opened with Excel was DDE (Dynamic Data Exchange). See Reference 1 (the latest update to my original 2003 paper), which mentions a DDE toolkit for which I still get requests frequently twelve years later. Unless you need to create a pivot table, for which the best resource is Reference 2, I suspect that DDE will remain a popular solution for some SAS users. Your SAS program runs as a client of an Excel session that your program starts to serve as its Excel server. It’s only clumsy to deal with when using Enterprise Guide and a remote SAS server. If something goes wrong, you end up with a hung SAS process and a hung Excel process that are Sanalogous tools can be created to deal with the hung Excel process.

The main content of this paper is the section ODS Excel – Demonstration By Examples. It is followed by lists in four sections: Options Demonstrated In This Paper; Options Not Demonstrated In This Paper; Printing Options (None Were Demonstrated In This Paper); and Options Not Retained From ExcelXP. Evidently, the developers regarded the capabilities of ODS tagset ExcelXP as a starting point for improvement. (I was never a fan of ExcelXP, which has no support for graphs.)

Before getting into the examples, there is a comparison of Options Available Prior to the ODS Excel Destination, as well as a section on Set-Up and Common Code for all of the Examples.

**NOTE:** Be sure to see Appendix A for documentation of ALL of the ODS Excel options available in TS1M3. Almost all are in release TS1M2 of SAS V9.4, but some require TS1M3. It does not have the convenient format of the SAS Tip Sheets that are available at SAS Global Forum conferences (which can be found online at the SAS web site), but it is the next best thing.

Acknowledgements

Essential to this project was assistance from Wayne Hester, Nancy Goodling, Chevell Parker, Scott Huntley, Dan O’Connor, and Amy Peters at SAS Institute.

Any errors or imperfections in this paper are my responsibility.
Options Available Prior to the ODS Excel Destination

Other users might have options interests other than those listed below, but it is my judgement that these are the ones most commonly desired.

<table>
<thead>
<tr>
<th>Feature</th>
<th>TableEditor*</th>
<th>ExcelXP</th>
<th>MSOffice2K_x</th>
<th>HTML</th>
<th>DDE**</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pivot Table</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>AutoFit Columns</td>
<td>By Default***</td>
<td>No</td>
<td>By Default***</td>
<td>By Default</td>
<td>Yes</td>
</tr>
<tr>
<td>Filters</td>
<td>Yes, but All Columns only</td>
<td>Yes</td>
<td>Yes, but All Columns only</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Freeze Panes</td>
<td>Row 1 only</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Side-By-Side Elements in WorkSheet</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Multi-Sheet WorkBook</td>
<td>Yes</td>
<td>Yes, Easy</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Graphs</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Titles</td>
<td>Not with other features</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

*Javascript must be enabled
**SAS must be running on Windows
***Headings of columns autofit by default can be overlaid by filter buttons

Note: The comparison above omits MSOffice2K. It was not evaluated by me because it has fewer capabilities than MSOffice2K_x.
Set-Up and Common Code for all of the Examples

%macro RunDayDateTime(RunDayDateTimeAsFileNameSuffix=NO);
%GLOBAL RunDayDateTime FileNameDTsuffix;
DATA _NULL_;
RunDate = DATE(); RunTime = TIME();
RunDayDateTimeText =
   TRIM(LEFT(PUT(RunDate,weekdatx37.))) || ' at ' ||
   TRIM(LEFT(PUT(RunTime,timeampm11.)));
CALL SYMPUT('RunDayDateTime',TRIM(LEFT(RunDayDateTimeText)));
RunDayDateTimeFileNameSuffix =
   TRIM(LEFT(PUT(RunDate,downame3.))) || '_' ||
   TRIM(LEFT(PUT(RunDate,date9.))) || '_' ||
   TRIM(LEFT(COMPRESS(PUT(RunTime,TOD8.),'.'))) || '_' ||
   SUBSTR(PUT(RunTime,TOD12.3),10,3); /* prevent duplicate timestamps when two successive macro invocations run during the same second */
CALL SYMPUT('FileNameDTsuffix',
   TRIM(LEFT(RunDayDateTimeFileNameSuffix)));
RUN;
%if %upcase(&RunDayDateTimeAsFileNameSuffix) EQ YES
 %then %let FileNameDTsuffix = %str(&FileNameDTsuffix);
%else %let FileNameDTsuffix = %str();
%mend RunDayDateTime;

%RunDayDateTime(RunDayDateTimeAsFileNameSuffix=YES);

/* Common Framing: */

%let SASenv = V94TS1M2; /* Modification Level 2 for V9.4 */
%let Path = D:\! ! ! ODS Excel Destination\results\;
%let CodePath = D:\! ! ! ODS Excel Destination\code\;
%let ODSstyle = HTMLblue; /* this is the shipped default */
%let OBScount = 10; /* fit results on 1 sheet without scroll */
%let N = 1; /* where Demo Step N will vary 1, 2, 3, . . . */
%let ZoomPct = P; /* P chosen to nearly fill Excel window */
ods noresults; /* not opening result in SAS session */
ods _all_ close;
< ODS Code Block goes here >
options obs=max;
/* Start of Common Framing for Early Adopter Release of SAS: */

%let SASenv = V94TS1M3; /* Modification Level 3 for V9.4 */

%let Path =
    /folders/myfolders/! ! ! ODS Excel Destination/results/;

%let CodePath =
    /folders/myfolders/! ! ! ODS Excel Destination/code/;

The Early Adopter Release is distributed only via SAS University Edition, which runs on a Linux Virtual Machine, and is accessed by SAS Studio.

In the examples that follow, the successful version of Demo 2 requires TS1M3;
Demo 23 and Demo 24 requires TS1M3; and Demo 9 was tested on TS1M3 to verify the difference in syntax for the Start_At option as compared with TS1M2.
ODS Excel - Demonstration By Examples

Each example on the following pages is accompanied by complete, or nearly complete, code. In some cases, inessential code was “abbreviated” in order to achieve the delivery goal of supplying all code on the same page as the Demonstration output.

**NOTE:** Available upon email request to the author are a zip file of all of the complete code and a zip file of all of the output Excel files.

A companion slide presentation, just as is the case for this paper, is in the Conference Proceedings and Tools package, which is available only to conference attendees.

Any graphs in the paper are not intended as examples of good graphic design. They are provided only to serve as proof of concept. For my latest paper, as of 24 June 2015, on communication-effective graphic design, see Reference 4.

To conserve vertical space, the “captions” for each example are the comment that is immediately below each screen image of the spreadsheet output.

There are few comments in the code, but the ODS Excel option(s) and/or SAS statement(s) required to deliver the distinguishing feature of each example are highlighted with blue or red.
/* Simplest Example, Default Sheet Name, Title Line Wrap */

%RunDayDateTime(RunDayDateTimeAsFileNameSuffix=YES);
%let SASenv = V94TS1M2;
%let Path = D:\results\;
%let CodePath = D:\code\;
%let ZoomPct = 220; /* want to fill the slide */
%let ODSstyle = HTMLblue; /* this is the shipped default */
%let OBScount = 10;
%let N = 0; /* Demo Step */
footnote1 justify=left bold "Source Data: SASHELP.SHOES";
footnote2 justify=left bold "Code: &CodePath.Demo&N..sas";
footnote3 justify=left bold "Run on: &RunDayDateTime";
ods noresults;
ods _all_ close;
ods excel style=styles.&ODSstyle
   file="&Path.\&SASenv.\&SASenv.\&CodePath.\&CodePath.\&FilenameDTsuffix..xlsx"
   options(embedded_titles='yes' embedded_footnotes='yes'
            zoom="&ZoomPct");
title1 justify=left bold "&OBScount Obs - Shoe Sales - &SASenv
   ODS EXCEL - ODS Style &ODSstyle - &ZoomPct.% Zoom";
title2 justify=left bold color=red "Default Sheet Name (maximum
   is 28 characters)"
   options obs=&OBScount;
proc print data=sashelp.shoes noobs;
id region subsidiary product;
run;
ods excel close;
options obs=max;
/* Custom Sheet Name, But Unwanted Title Line Wrap */
%RunDayDateTime(RunDayDateTimeAsFileNameSuffix=YES);
%let SASenv = V94TS1M2;
%let Path = D:\ ODS Excel Destination\results\;
%let CodePath = D:\ ODS Excel Destination\code\;
%let ZoomPct = 220; /* want to fill the slide */
%let ODSstyle = HTMLblue; /* this is the shipped default */
%let OBScount = 10;
%let N = 1; /* Demo Step */
footnote1 justify=left bold "Source Data: SASHELP.SHOES";
footnote2 justify=left bold "Code: &CodePath.Demo&N..sas";
footnote3 justify=left bold "Run on: &RunDayDateTime";
ods noresults;
ods _all_ close;
ods excel style=styles.&ODSstyle
   file="&Path.&SASenv. Demo&N._&FileNameDTsuffix..xlsx"
   options(embedded_titles='yes' embedded_footnotes='yes'
   zoom="&ZoomPct" sheet_name='Custom Sheet Name');
title1 justify=left bold "&OBScount Obs - Shoe Sales - &SASenv ODS EXCEL - ODS Style &ODSstyle - &ZoomPct.% Zoom";
options obs=&OBScount;
proc print data=sashelp.shoes noobs;
id region subsidiary product;un;
ods excel close;
options obs=max;
/* title_footnote_nobreak='yes', but Title Line Wrap Persists */
%RunDayDateTime(RunDayDateTimeAsFileNameSuffix=YES);
%let SASenv = V94TS1M2;
%let Path = D:\ ODS Excel Destination\results\;
%let CodePath = D:\ ODS Excel Destination\code\;
%let ZoomPct = 220; /* want to fill the slide */
%let ODSstyle = HTMLblue; /* this is the shipped default */
%let OBScount = 10;
%let N = 2; /* Demo Step */
footnote1 justify=left bold "Source Data: SASHELP.SHOES";
footnote2 justify=left bold "Code: &CodePath.Demo&N..sas";
footnote3 justify=left bold "Run on: &RunDayDateTime";
ods noresults;
ods _all_ close;
ods excel style=styles.&ODSstyle
   file="&Path.&SASenv._Demo&N._&FileNameDTsuffix..xlsx"
   options(embedded_titles='yes' embedded_footnotes='yes'
   zoom="&ZoomPct" sheet_name='Title Wrap Persists'
   title_footnote_nobreak='yes');
title1 justify=left bold "&OBScount Obs - Shoe Sales - &SASenv
ODS EXCEL - ODS Style &ODSstyle - &ZoomPct.% Zoom";
ods excel close;
options obs=max;
/* allocate enough title space, but the row is too high */
%RunDayDateTime(RunDayDateTimeAsFileNameSuffix=YES);
%let SASenv = V94TS1M2;
%let Path = D:\! ! ! ODS Excel Destination\results\;
%let CodePath = D:\! ! ! ODS Excel Destination\code\;
%let ZoomPct = 220; /* want to fill the slide */
%let ODSstyle = HTMLblue; /* this is the shipped default */
%let OBScount = 10;
%let N = 3; /* Demo Step */
footnote1 justify=left bold "Source Data: SASHELP.SHOES";
footnote2 justify=left bold "Code: &CodePath.Demo&N..sas";
footnote3 justify=left bold "Run on: &RunDayDateTime";
ods noresults;
ods _all_ close;
ods excel style=styles.&ODSstyle
   file="&Path.&SASenv._Demo&N.__FileNameDTsuffix..xlsx"
   options(embedded_titles='yes' embedded_footnotes='yes'
      zoom="&ZoomPct" sheet_name='No Title Wrap, Row Too High'
      title_footnote_width='8');
title1 justify=left bold "&OBScount Obs - Shoe Sales - &SASenv
   ODS EXCEL - ODS Style &ODSstyle - &ZoomPct.% Zoom";
options obs=&OBScount;
proc print data=sashelp.shoes noobs;
id region subsidiary product;
run;
ods excel close;
options obs=max;
/* title row height set to that of other rows */
%RunDayDateTime(RunDayDateTimeAsFileNameSuffix=YES);
%let SASenv = V94TS1M2;
%let Path = D:\ODS Excel Destination\results\;
%let CodePath = D:\ODS Excel Destination\code\;
%let ZoomPct = 220; /* want to fill the slide */
%let ODSstyle = HTMLblue; /* this is the shipped default */
%let OBScount = 10;
%let N = 4; /* Demo Step */
footnote1 justify=left bold "Source Data: SASHELP.SHOES";
footnote2 justify=left bold "Code: &CodePath.Demo&N..sas";
footnote3 justify=left bold "Run on: &RunDayDateTime";
ods noresults;
ods _all_ close;
ods Excel style=styles.&ODSstyle
    file="&Path.&SASenv._Demo&N._&FileNameDTsuffix..xlsx"
    options(embedded_titles='yes' embedded_footnotes='yes'
             zoom="&ZoomPct" sheet_name='Set Row Height To Necessary'
             title_footnote_width='8' absolute_row_height='14');
title1 justify=left bold "&OBScount Obs - Shoe Sales - &SASenv ODS EXCEL - ODS Style &ODSstyle - &ZoomPct.% Zoom";
ods obs=&OBScount;
proc print data=sashelp.shoes noobs;
id region subsidiary product;
run;
ods Excel close;
options obs=max;
/* make title2 a hyperlink, it could link to another workbook */
%RunDayDateTime(RunDayDateTimeAsFileNameSuffix=YES);
%let SASenv = V94TS1M2;
%let Path = D:\ODS Excel Destination\results\;
%let CodePath = D:\ODS Excel Destination\code\;
%let ZoomPct = 220; /* want to fill the slide */
%let ODSstyle = HTMLblue; /* this is the shipped default */
%let OBScount = 10;
%let N = 5; /* Demo Step */
footnote1 justify=left bold "Source Data: SASHELP.SHOES";
footnote2 justify=left bold "Code: &CodePath.Demo&N..sas";
footnote3 justify=left bold "Run on: &RunDayDateTime";
ods noresults;
ods _all_ close;
ods excel style=styles.&ODSstyle
    file="&Path.&SASenv._Demo&N._&FileNameDTsuffix..xlsx"
    options(embedded_titles='yes' embedded_footnotes='yes'
        zoom="&ZoomPct" sheet_name='Title2 Is Hyperlink'
        title footnote_width='8' absolute_row_height='14');
title1 justify=left bold "&OBScount Obs - Shoe Sales - &SASenv"
ods EXCEL - ODS Style &ODSstyle - &ZoomPct.% Zoom"
title2 justify=left bold color=blue underlin=1
    link='http://www.wiilsu.org'
    'Go to Wisconsin Illinois SAS Users Home Page';
ods excel close; options obs=max;

/* Source Data: SASHELP.SHOES */
/* Code: D:\ODS Excel Destination\code\Demo5.sas */
/* Run on: Wednesday, 10 June 2015 at 1:14:17 AM */
Clicking on the hyperlink in Title2 of the spreadsheet takes you to one of my favorite web sites. It could instead take you a relevant graph or other document, which could be linked back to the spreadsheet.

**NOTE:** I have stated elsewhere that a graph and a spreadsheet of its input data as an interlinked pair is always an excellent delivery method (see Reference 5)—as an advocate for and creator of communication-effective graphs, I have long said:  
**Image + Precise Numbers = quick, easy inference + reliable inference**
/* freeze headers and row headers */
%RunDayDateTime(RunDayDateTimeAsFileNameSuffix=YES);
%let SASenv = V94TS1M2;
%let Path = D:\ODS Excel Destination\results\;
%let CodePath = D:\ODS Excel Destination\code\;
%let ZoomPct = 220; /* want to fill the slide */
%let ODSstyle = HTMLblue; /* this is the shipped default */
%let OBScount = 10;
%let N = 6; /* Demo Step */
footnote1 justify=left bold "Source Data: SASHELP.SHOES'';
footnote2 justify=left bold "Code: &CodePath..sas'';
footnote3 justify=left bold "Run on: &RunDayDateTime'';
ods noresults;
ods _all_ close;
ods excel style=styles.&ODSstyle
file="&Path..xlsx"
options(embedded_titles='yes' embedded_footnotes='yes'
zoom="&ZoomPct" sheet_name='Freeze Headers + RowHeaders'
/* & in sheet_name would get converted to */
title_foote note width='8' absolute_row_height='14'
frozen_headers='4' frozen_rowheaders='3');
title1 justify=left bold "&OBScount Obs - Shoe Sales - &SASenv
ODS EXCEL - ODS Style &ODSstyle - &ZoomPct.% Zoom'';
title2 justify=left bold color=blue underlin=1
link='http://www.wiilsu.org' 'Go to ...';
options obs=&OBScount;
proc print data=sashelp.shoes noobs;
   id region subsidiary product;
ods excel close; options obs=max;
/* turn on some autofilters */

%RunDayDateTime(RunDayDateTimeAsFileNameSuffix=YES);
%let SASenv = V94TS1M2;
%let Path = D:\! ! ! ODS Excel Destination\results\;
%let CodePath = D:\! ! ! ODS Excel Destination\code\;
%let ZoomPct = 220; /* want to fill the slide */
%let ODSstyle = HTMLblue; /* this is the shipped default */
%let OBScount = 10;
%let N = 7; /* Demo Step */
footnote1 justify=left bold "Source Data: SASHELP.SHOES";
footnote2 justify=left bold "Code: &CodePath.Demo&N..sas";
footnote3 justify=left bold "Run on: &RunDayDateTime";
ods noresults;
ods _all_ close;
ods excel style=styles.&ODSstyle
   file="&Path.&SASenv._Demo&N._&FileNameDTsuffix..xlsx"
   options(/embedded_titles='yes' embedded_footnotes='yes'
     zoom="&ZoomPct" autofilter='1-4'
     sheet_name='AutoFilters Columns A-D'
     title_footnote_width='8' absolute_row_height='14'
     frozen_headers='4' frozen_rowheaders='3');
title1 justify=left bold "&OBScount Obs - Shoe Sales - &SASenv
ODS EXCEL - ODS Style &ODSstyle - &ZoomPct.% Zoom";
title2 justify=left bold color=blue underlin=1
   link='http://www.wiilsu.org' 'Go to ...;
options obs=&OBScount;
proc print data=sashelp.shoes noobs;
id region subsidiary product;run;
ods excel close; options obs=max;
/* s in Chars is dropped */
/* verify that the maximum sheet name length is 28 characters */
%RunDayDateTime(RunDayDateTimeAsFileNameSuffix=YES);
%let SASenv = V94TS1M2;
%let Path = D:\! ! ! ODS Excel Destination\results\;
%let CodePath = D:\! ! ! ODS Excel Destination\code\;
%let ZoomPct = 220; /* want to fill the slide */
%let ODSstyle = HTMLblue; /* this is the shipped default */
%let OBScount = 10;
%let N = 8; /* Demo Step */
footnote1 justify=left bold "Source Data: SASHELP.SHOES";
footnote2 justify=left bold "Code: &CodePath.Demo&N..sas";
footnote3 justify=left bold "Run on: &RunDayDateTime";
ods noresults; ods _all_ close;
ods excel style=styles.&ODSstyle
file="&Path.&SASenv._Demo&N..&FileNameDTsuffix..xlsx"
options(embedded_titles='yes' embedded_footnotes='yes'
zoom="&ZoomPct" autofilter='1-4'
title_footnote_width='8' absolute_row_height='14'
frozen_headers='4' frozen_rowheaders='3'
sheet_name='SheetName Max Length 28 Chars');
title1 justify=left bold "&OBScount Obs - Shoe Sales - &SASenv
ODS EXCEL - ODS Style &ODSstyle - &ZoomPct.% Zoom";
title2 justify=left bold color=blue underlin=1
link='http://www.wiilsu.org' 'Go to ...';
options obs=&OBScount;
proc print data=sashelp.shoes noobs;
  id region subsidiary product;
run;
ods excel close; options obs=max;
/* Start at Column C Row 2. If TS1M3 Start at Column 3 Row 2. */
%RunDayDateTime(RunDayDateTimeAsFileNameSuffix=YES);
%let SASenv = V94TS1M2;
%let Path = D:\! ! ! ODS Excel Destination\results\;
%let CodePath = D:\! ! ! ODS Excel Destination\code\;
%let ZoomPct = 220; /* want to fill the slide */
%let ODSstyle = HTMLblue; /* this is the shipped default */
%let OBScount = 10;
%let N = 9; /* Demo Step */
footnote1 justify=left bold "Source Data: SASHELP.SHOES";
footnote2 justify=left bold "Code: &CodePath.Demo&N..sas";
footnote3 justify=left bold "Run on: &RunDayDateTime";
ods noresults; ods _all_ close;
ods excel style=styles.&ODSstyle
   file="&Path.&SASenv._Demo&N._&FileNameDTsuffix..xlsx"
   options(embedded_titles='yes' embedded_footnotes='yes'
      zoom="&ZoomPct"
      title footnote width='8' absolute_row_height='14'
      frozen_headers='5' frozen_rowheaders='5'
      autofilter='1-4' start at='C2'
      sheet name='Start Content at ColCRow2');
title1 justify=left bold "&OBScount Obs - Shoe Sales - &SASenv ODS EXCEL - ODS Style &ODSstyle - &ZoomPct.% Zoom";
title2 justify=left bold color=blue underlin=1 link='http://www.wiilsu.org' 'Go to ...';
opts obs=&OBScount;
proc print data=sashelp.shoes noobs;
id region subsidiary product;
sales inventory returns;
ods excel close; options obs=max;
/* Start At=(3,2) for Col3 Row2 on TS1M3, unlike TS1M2 */
%RunDayDateTime(RunDayDateTimeAsFileNameSuffix=YES);
%let SASenv = V94TS1M3;
%let Path = /folders/myfolders/!
%let CodePath = /folders/myfolders/!
%let ZoomPct = 180; want to fill the slide */
%let ODSstyle = HTMLblue; /* EXCEL is the TS1M3 default */
%let OBScount = 10;
%let N = 9; /* Demo Step */
footnote1 justify=left bold "Source Data: SASHELP.SHOES";
footnote2 justify=left bold "Code: &CodePath.Demo&N..sas";
footnote3 justify=left bold "Run on: &RunDayDateTime";
ods noresults; ods _all_ close;
ods excel style=styles.&ODSstyle
file="&Path.&SASenv._Demo&N._&FileNameDTsuffix..xlsx"
options(embedded_titles='yes' embedded_footnotes='yes'
    title_footnote_width='8' absolute_row_height='14'
    frozen_headers='5' frozen_rowheaders='5'
    start_at='3,2' autofilter='1-4' zoom="&ZoomPct"
    sheet_name='Start Content at Col3Row2');
title1 justify=left bold "&OBScount Obs - Shoe Sales ...";
title2 justify=left bold color=blue underlin=1
    link='http://www.wiilsu.org' 'Go to ...';
opts obs=&OBScount;
proc print data=sashelp.shoes noobs;
id region subsidiary product; run;
ods excel close; options obs=max;
/* hide rows 13-14 & cols 6-7, narrow title space to cols 1-5 */

%RunDayDateTime(RunDayDateTimeAsFileNameSuffix=YES);
%let SASenv = V94TS1M2;
%let Path = D:\! ! ! ODS Excel Destination\results\;
%let CodePath = D:\! ! ! ODS Excel Destination\code\;
%let ZoomPct = 220; /* want to fill the slide */
%let ODSstyle = HTMLblue; /* this is the shipped default */
%let OBScount = 10;
%let N = 10; /* Demo Step */
footnote1 justify=left bold "Source Data: SASHELP.SHOES";
footnote2 justify=left bold "Code: &CodePath.Demo&N..sas";
footnote3 justify=left bold "Run on: &RunDayDateTime";
ods noresults; ods _all_ close;
ods excel style=styles.&ODSstyle
   file="&Path.&SASenv._Demo&N._&FileNameDTsuffix..xlsx"
   options(embedded_titles='yes' embedded_footnotes='yes'
      zoom="&ZoomPct" autofilter='1-4' title_footnote_width='5'
      /* adjust to absence of columns 6 & 7 */
      frozen_headers='5' frozen_rowheaders='3'
      sheet_name='hide rows 13-14 and cols 6-7'
      hidden_rows='13-14' hidden_columns='6-7');
title1 justify=left bold "&OBScount Obs - Shoe Sales - &SASenv
   ODS EXCEL - ODS Style &ODSstyle - &ZoomPct.% Zoom";
title2 justify=left bold color=blue underlin=1
   link='http://www.wiilsu.org' 'Go to ...';
opts obs=&OBScount;
proc print data=sashelp.shoes noobs;
   id region subsidiary product; run;
ods excel close; options obs=max;
/* open workbook on Sex=F sheet | after click on Sex=M sheet */
/* above are in workbook with sheet_interval='bygroup' */
%RunDayDateTime(RunDayDateTimeAsFileNameSuffix=YES);

%let SASenv = V94TS1M2;
%let Path = D:\11 ODS Excel Destination\results\;
%let CodePath = D:\11 ODS Excel Destination\code\;
%let ZoomPct = 220; /* want to fill the slide */
%let ODSstyle = HTMLblue; /* this is the shipped default */
%let N = 11; /* Demo Step */
footnote1 justify=left bold "Source Data: SASHELP.CLASS";
footnote2 justify=left bold "Code: &CodePath.Demo&N..sas";
footnote3 justify=left bold "Run on: &RunDayDateTime";
ods noresults;
ods _all_ close;
ods excel style=styles.&ODSstyle
  file="&Path.&SASenv_ Demo&N_&FileNameDTsuffix..xlsx"
  options(embedded_titles='yes' embedded_footnotes='yes'
    title_footnote_width='8' absolute_row_height='14'
    sheet_interval='bygroup' suppress_bylines='yes'
    sheet_label='Sex' zoom="&ZoomPct")
;
title1 justify=left bold "Students - &SASenv ODS EXCEL - ODS Style &ODSstyle - &ZoomPct.% Zoom";
title2 justify=left bold color=blue underlin=1
  link='http://www.wiilsu.org 'Go to ...';
proc sort data=sashelp.class out=work.ToPrint; by sex; run;
proc print data=work.ToPrint noobs; by sex; id name; run;
ods excel close;
options obs=max;

<table>
<thead>
<tr>
<th></th>
<th>Name</th>
<th>Age</th>
<th>Height</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Alice</td>
<td>13</td>
<td>56.5</td>
<td>84.0</td>
</tr>
<tr>
<td>2</td>
<td>Barbara</td>
<td>14</td>
<td>65.3</td>
<td>98.0</td>
</tr>
<tr>
<td>3</td>
<td>Carol</td>
<td>14</td>
<td>62.8</td>
<td>102.5</td>
</tr>
<tr>
<td>4</td>
<td>Jane</td>
<td>15</td>
<td>59.8</td>
<td>84.5</td>
</tr>
<tr>
<td>5</td>
<td>Janet</td>
<td>15</td>
<td>62.5</td>
<td>112.5</td>
</tr>
<tr>
<td>6</td>
<td>Joyce</td>
<td>11</td>
<td>51.3</td>
<td>50.5</td>
</tr>
<tr>
<td>7</td>
<td>Judy</td>
<td>14</td>
<td>64.3</td>
<td>90.0</td>
</tr>
<tr>
<td>8</td>
<td>Louise</td>
<td>14</td>
<td>63.7</td>
<td>70.0</td>
</tr>
<tr>
<td>9</td>
<td>Mary</td>
<td>13</td>
<td>66.5</td>
<td>112.0</td>
</tr>
</tbody>
</table>

Source Data: SASHELP.CLASS
Code: D:\11 ODS Excel Destination\code\Demo11.sas
Run on: Wednesday, 10 June 2015 at 4:22:50 PM
Demonstration of the Effect of Using the Tab_Color Option

<table>
<thead>
<tr>
<th>Name</th>
<th>Sex</th>
<th>Age</th>
<th>Height</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alice</td>
<td>F</td>
<td>13</td>
<td>56.5</td>
<td>84.0</td>
</tr>
<tr>
<td>Barbara</td>
<td>F</td>
<td>13</td>
<td>65.3</td>
<td>98.0</td>
</tr>
<tr>
<td>Carol</td>
<td>F</td>
<td>14</td>
<td>62.8</td>
<td>102.5</td>
</tr>
<tr>
<td>Jane</td>
<td>F</td>
<td>12</td>
<td>59.8</td>
<td>84.5</td>
</tr>
<tr>
<td>Janet</td>
<td>F</td>
<td>15</td>
<td>62.5</td>
<td>112.5</td>
</tr>
<tr>
<td>Joyce</td>
<td>F</td>
<td>11</td>
<td>91.3</td>
<td>50.5</td>
</tr>
<tr>
<td>Judy</td>
<td>F</td>
<td>14</td>
<td>94.3</td>
<td>90.0</td>
</tr>
<tr>
<td>Louise</td>
<td>F</td>
<td>12</td>
<td>56.3</td>
<td>77.0</td>
</tr>
<tr>
<td>Mary</td>
<td>F</td>
<td>15</td>
<td>66.5</td>
<td>112.0</td>
</tr>
</tbody>
</table>

Source Data: SASHELP.CLASS
Code: D:\!!1\ODS Excel Destination\code\Demo12.sas
Run on: Wednesday, 10 June 2015 at 5:34:40 PM

Tab Colors at Initial Open (faint colors at bottom of tabs):

After Clicking on the Boys Tab:

After Clicking back to the Girls Tab:
Code for Using the Tab_Color Option

**NOTE:** Using full-strength colors, unless they are light colors, for the tabs is likely to make the sheet name difficult to read.

```sas
%RunDayDateTime(RunDayDateTimeAsFileNameSuffix=YES);
%let SASenv = V94TS1M2;
%let Path = D:\results\;
%let CodePath = D:\code\;
%let ZoomPct = 220; /* want to fill the slide */
%let ODSstyle = HTMLblue; /* this is the shipped default */
%let N = 12; /* Demo Step */

footnote1 justify=left bold "Source Data: SASHELP.CLASS";
footnote2 justify=left bold "Code: &CodePath.Demo&N..sas";
footnote3 justify=left bold "Run on: &RunDayDateTime";
title1 justify=left bold color=red "WorkBook with color-coded WorkSheet tabs";
title2 justify=left bold color=red "At WorkBook Open & After, the Active Sheet tab has only a tinge of color at its bottom";
title3 justify=left bold color=red "After either Sheet tab has been pressed, the InActive sheet tab is fully colored";

ods noresults; ods _all_ close;

ods excel style=styles.&ODSstyle
    file="&Path.&SASenv._Demo&N._&FileNameDTsuffix..xlsx"
    options(embedded_titles='yes' embedded_footnotes='yes'
              title_footnote_width='10' absolute_row_height='14'
              sheet_interval='proc' zoom="&ZoomPct");

ods excel options(sheet_name='Girls' tab_color='LightRed');
title4 justify=left bold "Female Students - &SASenv ODS EXCEL - ODS Style &ODSstyle - &ZoomPct.% Zoom";
proc print data=sashelp.class noobs;
    where sex eq 'F';
    id name;
run;

ods excel options(sheet_name='Boys' tab_color='LightBlue');
title4 justify=left bold "Male Students - &SASenv ODS EXCEL - ODS Style &ODSstyle - &ZoomPct.% Zoom";
proc print data=sashelp.class noobs;
    where sex eq 'M';
    id name;
run;

ods excel close; options obs=max;
```
/* Empty worksheet has Girls and Boys companion worksheets */
/* For more than one empty sheet, your sheet_name= is ignored */
/* They will be named Sheet1, Sheet2, etc. */

%RunDayDateTime(RunDayDateTimeAsFileNameSuffix=YES);
%let SASenv = V94TS1M2;
%let Path = D:\ODS Excel Destination\results\;
%let CodePath = D:\ODS Excel Destination\code\;
%let ZoomPct = 220; /* want to fill the slide */
%let ODSstyle = HTMLblue; /* this is the shipped default */
%let N = 13; /* Demo Step */
footnote1 justify=left bold "Source Data: SASHELP.CLASS";
footnote2 justify=left bold "Code: &CodePath.Demo&N..sas";
footnote3 justify=left bold "Run on: &RunDayDateTime";
title1 ...; title2 ...
ods noresults; ods _all_ close;
ods excel style=styles.&ODSstyle file="&Path.&SASenv._Demo&N._&FileNameDTsuffix..xlsx"
options(Embedded_titles='yes' embedded_footnotes='yes'
  title_footnote_width='10' absolute_row_height='14'
  sheet_interval='proc' zoom="&ZoomPct");
ods excel options(sheet_name='Girls');
title3 justify=left bold "Female Students - &SASenv ODS EXCEL -
  ODS Style &ODSstyle - &ZoomPct.% Zoom";
proc print data=sashelp.class noobs;where sex eq 'F'; run;
ods excel options(blank_sheet='Custom Named Empty Sheet');
ods excel close; options obs=max;
/* Click on Girls or Boys. Index most useful for many sheets. */
%RunDayDateTime(RunDayDateTimeAsFileNameSuffix=YES);
%let SASenv = V94TS1M2;
%let Path = D:\!! ! ! ODS Excel Destination\results\;
%let CodePath = D:\!! ! ! ODS Excel Destination\code\;
%let ZoomPct = 220; /* want to fill the slide */
%let ODSstyle = HTMLblue; /* this is the shipped default */
%let N = 14; /* Demo Step */
footnote1 justify=left bold "Source Data: SASHELP.CLASS";
footnote2 justify=left bold "Code: &CodePath.Demo&N..sas";
footnote3 justify=left bold "Run on: &RunDayDateTime";
title1 justify=left bold color=red "WorkBook with an Index";
ods noresults;
ods _all_ close;
ods excel style=styles.&ODSstyle
   file="&Path.&SASenv._Demo&N._&FileNameDTsuffix..xlsx"
   options(Embedded_titles='yes' Embedded_footnotes='yes'
      Title_footnote_width='10' Absolute_row_height='14'
      Zoom="&ZoomPct" Sheet_interval='proc' Index='yes');
ods excel options(sheet_name='Girls');
title2 justify=left bold "Female Students - &SASenv ODS EXCEL -
   ODS Style &ODSstyle - &ZoomPct.% Zoom";
proc print data=sashelp.class noobs; where sex eq 'F'; run;
ods excel options(sheet_name='Boys');
title2 justify=left bold "Male Students - &SASenv ODS EXCEL -
   ODS Style &ODSstyle - &ZoomPct.% Zoom";
proc print data=sashelp.class noobs; where sex eq 'M'; run;
ods excel close;
opts obs=max;
/* Table of Contents = TWO UNinformative links per worksheet */
%RunDayDateTime(RunDayDateTimeAsFileNameSuffix=YES);
%let SASenv = V94TS1M2;
%let Path = D:\O ! ! ! ODS Excel Destination\results\;
%let CodePath = D:\O ! ! ! ODS Excel Destination\code\;
%let ZoomPct = 220; /* want to fill the slide */
%let ZoomPct = 180; want to fill the slide */
%let ODSstyle = HTMLblue; /* this is the shipped default */
%let N = 15; /* Demo Step */
footnote1 justify=left bold "Source Data: SASHELP.CLASS";
footnote2 justify=left bold "Code: &CodePath.Demo&N..sas";
footnote3 justify=left bold "Run on: &RunDayDateTime";
title1 justify=left bold color=red "WorkBook with Table of
Contents (All Entries are Hyperlinks to Respective Sheets)";
title2 justify=left bold color=red "Entries are less informative
than those in an Index, and needless two per sheet.";
ods noresults; ods _all_ close;
ods excel style=styles.&ODSstyle
   file="&Path.&SASenv._Demo&N._&FileNameDTsuffix..xlsx"
   options(embedded_titles='yes' embedded_footnotes='yes'
           title_footnote_width='10' absolute_row_height='14'
           sheet_interval='proc' contents='yes' zoom="&ZoomPct")
ods excel options(sheet_name='Girls');
title3 justify=left bold "Female Students ...";
proc print data=sashelp.class noobs; where sex eq 'F'; run;
ods excel options(sheet_name='Boys');
title3 justify=left bold "Male Students ...";
proc print data=sashelp.class noobs; where sex eq 'M'; run;
ods excel close; options obs=max;
/* Listing and Plot on Separate WorkSheets */
%RunDayDateTime(RunDayDateTimeAsFileNameSuffix=YES);
%let SASenv = V94TS1M2;
%let Path = D:\! ! ! ODS Excel Destination\results\;
%let CodePath = D:\! ! ! ODS Excel Destination\code\;
%let ZoomPct = 170; /* want to fill the slide */
%let ZoomPct = 180; want to fill the slide */
%let ODSstyle = HTMLblue; /* this is the shipped default */
%let N = 16; /* Demo Step */
footnote1 justify=left bold "Source Data: SASHELP.CLASS";
footnote2 justify=left bold "Code: &CodePath.Demo&N..sas";
footnote3 justify=left bold "Run on: &RunDayDateTime";
ods noresults; ods _all_ close;
ods excel style=styles.&ODSstyle
    file="&Path.&SASenv._Demo&N..&FileNameDTsuffix..xlsx"
    options(embedded_titles='yes' embedded_footnotes='yes'
        title_footnote_width='10' absolute_row_height='14'
        sheet_interval='proc' zoom="&ZoomPct")
ods excel options(sheet_name='Class Listing');
title1 justify=left bold "List of Students - &SASenv ODS EXCEL -
    ODS Style &ODSstyle - &ZoomPct.% Zoom";
proc print data=sashelp.class noobs; run;
ods excel options(sheet_name='Plot Weight vs Height')
gtitle gfootnote; /* so that text is in the image */
goptions xpixels=1000 ypixels=750;
title1 justify=left bold "Student Weight vs Height - &SASenv ODS
    EXCEL - ODS Style &ODSstyle - &ZoomPct.% Zoom";
proc gplot data=sashelp.class; plot weight*height; run; quit;
ods excel close; options obs=max;
/* Listing & Plot on One WorkSheet. In omitted Demo18, ODS Excel Start At option could NOT move Plot to right of Listing */

%RunDayDateTime(RunDayDateTimeAsFileNameSuffix=YES);
%let SASenv = V94TS1M2;
%let Path = D:\! ! ! ODS Excel Destination\results\;
%let CodePath = D:\! ! ! ODS Excel Destination\code\;
%let ZoomPct = 100; /* want to fill the slide */
%let ODSstyle = HTMLblue; /* this is the shipped default */
%let N = 17; /* Demo Step */
ods noresults; ods _all_ close;
ods excel style=styles.&ODSstyle
   file="&Path.&SASenv._Demo&N._&FileNameDTsuffix..xlsx"
   options(embedded_titles='yes' embedded_footnotes='yes'
         title_footnote_width='10' absolute_row_height='14'
         zoom="&ZoomPct" sheet_interval='none'
         sheet_name='Listing and Wgt vs Hgt Plot');
title1 justify=left bold "List of Students - &SASenv ODS EXCEL - ODS Style &ODSstyle - &ZoomPct.% Zoom";
footnote1;
proc print data=sashelp.class noobs; run;
ods excel gtitle gfootnote; /* so that text is in the image */
gooprtions xpixels=800 ypixels=600;
title1 justify=left bold "Student Weight vs Height";
footnote1 justify=left bold "Source Data: SASHHELP.CLASS";
footnote2 justify=left bold "Code: &CodePath.Demo&N_.sas";
footnote3 justify=left bold "Run on: &RunDayDateTime";
proc gplot data=sashelp.class;
symbol value=dot; plot weight*height; run; quit;
ods excel close; options obs=max;
/* Five Outputs from PROC UNIVARIATE on Separate WorkSheets */
%RunDayDateTime(RunDayDateTimeAsFileNameSuffix=YES);
%let SASenv = V94TS1M2;
%let Path = D:\! ODS Excel Destination\results\;
%let CodePath = D:\! ODS Excel Destination\code\;
%let ZoomPct = 210; /* want to fill the slide */
%let ODSstyle = HTMLblue; /* this is the shipped default */
%let N = 19; /* Demo Step */
footnote1 justify=left bold "Source Data: SASHELP.CLASS";
footnote2 justify=left bold "Code: &CodePath.Demo&N.sas";
footnote3 justify=left bold "Run on: &RunDayDateTime";
ods noresults; ods _all_ close;
ods noproctitle;
ods excel style=styles.&ODSstyle
    file="&Path.&SASenv.Demo&N._&FileNameDTsuffix..xlsx"
    options(embedded_titles='yes' embedded_footnotes='yes'
        title_footnote_width='11' absolute_row_height='14'
        sheet_interval='output' zoom="&ZoomPct");
/* Presumably use of sheet_name= is impossible in this context,
   or would deliver unintended consequences. I did not test it.
   Your findings? */
title1 justify=left bold "Univariate Statistics for Student Weight - &SASenv ODS EXCEL - ODS Style &ODSstyle - &ZoomPct.% Zoom"
proc univariate data=sashelp.class;
var weight;
run;
ods excel close; options obs=max;
/* Five Outputs from PROC UNIVARIATE on Same WorkSheet */
%RunDayDateTime(RunDayDateTimeAsFileNameSuffix=YES);
%let SASenv = V94TS1M2;
%let Path = D:\IVI SODS Excel Destination\results\;
%let CodePath = D:\IVI SODS Excel Destination\code\;
%let ZoomPct = 85; /* shrink to fit display area */
%let ODSstyle = HTMLblue; /* this is the shipped default */
%let N = 20; /* Demo Step */
footnote1 justify=left bold "Source Data: SASHELP.CLASS";
footnote2 justify=left bold "Code: &CodePath.Demo&N..sas";
footnote3 justify=left bold "Run on: &RunDayDateTime";
ods noresults;
ods _all_ close;
ods noproctitle;
ods excel style=styles.&ODSstyle
   file="&Path.&SASenv._Demo&N._&FileNameDTsuffix..xlsx"
   options(embedded_titles='yes' embedded_footnotes='yes'
         title_footnote_width='7' absolute_row_height='14'
         zoom="%ZoomPct" sheet_interval='none'
         sheet_name='All Outputs On Same Sheet');
title1 justify=left bold "Univariate Statistics for Student Weight - &SASenv ODS EXCEL - ODS Style &ODSstyle - &ZoomPct."
proc univariate data=sashelp.class;
var weight;
run;
ods excel close;
options obs=max;
/* Using TAGATTR to apply static Microsoft numeric formats */
%RunDayDateTime(RunDayDateTimeAsFileNameSuffix=YES);
%let SASenv = V94TS1M2;
%let Path = D:\ ODS Excel Destination\results\;
%let CodePath = D:\ ODS Excel Destination\code\;
%let ZoomPct = 300; /* want to fill the slide */
%let ODSstyle = HTMLblue; /* this is the shipped default */
%let N = 21; /* Demo Step */
footnote1 justify=left bold "Source Data: SASHELP.PRDSALE";
footnote2 ..."; footnote3 ...";
data work.ToSummary;
set sashelp.prdsale; Difference = predict - actual; run;
proc summary data=work.ToSummary nway;
class country; var predict actual Difference;
output out=ToPrint sum=; run;
ods noresults; ods _all_ close;
ods excel style=styles.&ODSstyle
   file="&Path.&SASenv._Demo&N._&FileNameDTsuffix..xlsx"
   options(embedded_titles='yes' embedded_footnotes='yes'
       zoom="&ZoomPct" sheet_name='Using Tag Attributes');
title1 justify=left bold "Actual Vs Predicted ...";
proc print data=work.ToPrint noobs label; id country;
var predict actual /
   style={tagattr='format:$#,##0_';[Red]($#,##0\')};
var difference /
   style={tagattr='format:$#,##0_';[Red]($#,##0\')};
sum predict actual Difference /
   style={tagattr='format:$#,##0_';[Red]($#,##0\')};
run; ods excel close; options obs=max;
/* Using TAGATTR to apply a dynamic Microsoft numeric format */
/* See next page for result of changing Germany Actual Sales */
%RunDayDateTime(RunDayDateTimeAsFileNameSuffix=YES);
%let SASenv = V94TS1M2;
%let Path = D:\ ! ! ! ODS Excel Destination\results\;
%let CodePath = D:\ ! ! ! ODS Excel Destination\code\;
%let ZoomPct = 270; /* want to fill the slide */
%let ODSstyle = HTMLblue; /* this is the shipped default */
%let N = 22; /* Demo Step */
footnote1 justify=left bold "Source Data: SASHELP.PRDSALE"
footnote2 ...; footnote3 ...
data work.ToSummary;
set sashelp.prdsale; Difference = predict - actual; run;
proc summary data=work.ToSummary nway;
class country; var predict actual Difference;
output out=ToPrint sum=; run;
ods noresults; ods _all_ close;
ods excel style=styles.&ODSstyle
   file="&Path.&SASenv._Demo&N._&FileNameDTsuffix..xlsx"
   options(embedded_titles='yes' embedded_footnotes='yes'
      sheet_name='Tag Attributes and Formula' zoom="&ZoomPct");
title1 justify=left bold "Actual Vs Predicted Sales ...";
title2 justify=left bold color=red "Any Manual Changes to Actual and/or Predicted Will Be Reflected in a Revised Difference";
proc print data=work.ToPrint noobs label; id country;
var predict actual Difference;
var Difference / style={tagattr='format:$#,##0_');[Red]((#,##0)')};
run; ods excel close; options obs=max;
Actual Vs Predicted Sales By Country - V94TS1M2
ODS EXCEL - ODS Style HTMLblue - 270% Zoom

Any Manual Changes to Actual and/or Predicted Will
Be Reflected in a Revised Difference

<table>
<thead>
<tr>
<th>Country</th>
<th>Predicted Sales</th>
<th>Actual Sales</th>
<th>Difference</th>
</tr>
</thead>
<tbody>
<tr>
<td>CANADA</td>
<td>$233,019</td>
<td>$246,990</td>
<td>$13,971</td>
</tr>
<tr>
<td>GERMANY</td>
<td>$231,554</td>
<td>$200,000</td>
<td>($31,554)</td>
</tr>
<tr>
<td>U.S.A.</td>
<td>$241,722</td>
<td>$237,349</td>
<td>($4,373)</td>
</tr>
</tbody>
</table>

Source Data: SASHELP.PRDLSALE
Code: D:!! !! ODS Excel Destination:codeIDemo22.sas
Run on: Saturday, 13 June 2015 at 2:23:07 PM

/* Using TAGATTR to apply a dynamic Microsoft numeric format */
/* New difference value after changing Germany Actual Sales */
Microsoft Excel Charts Directly From SAS

An interesting new capability in TS1M3 is the ability to use ODS Excel and the new PROC MSCHART to create Excel charts directly from SAS.

The most frequent graphic destination for data prepared with SAS has probably been Excel charts. Though SAS/GRAPH and ODS Graphics have always been far superior from a features and functions standpoint, many users are inclined to rely on a tool which might be easier for them to learn and to use. Not everyone is as fussy as I am in what I want to achieve graphically. In TS1M3, this capability is still in experimental status, but what you can do is shown on the next two pages. PROC MSCHART requires ODS Excel, and is not a part of SAS/GRAPH or ODS Graphics. It does not require Excel to be installed on the machine where SAS is running, which could be MVS, Unix, Linux, or Windows.

The first Excel chart is modeled on a SAS Global Forum 2015 examples handout, which did not provide all of the code needed, and which did not get into the broader story that is shown on the second Excel chart. Until I learned that the data needed to be pre-sorted by date (which was not revealed in the handout), I was baffled by the chart being drawn backwards.

Note that both charts are automatically web-enabled with “data tips”. No coding is required to request that.

The table of chart input data is always created. If the chart is positioned at top left, it hides the table, IF the table is short enough. When the chart is positioned at top right, the table is always apparent. The developer is contemplating providing an option to suppress the table, but the user will have the option to retain the table.

I like the table. As an advocate for and creator of communication-effective graphs, I have long said: Image + Precise Numbers = quick, easy inference + reliable inference
It’s more convenient to have the chart and table side-by-side than to use hyperlinks to toggle between a chart and a table. Guessing the y-value (or x-value) of a plot, or the value of a bar end, by estimating where it lines up with tick mark values on an axis is an unreliable substitute for a table. Data tips are nice, but they disappear when you move the mouse. If you want to compare different points or bars with precision, you need to transcribe the information from pop-up boxes.
/* PROC MSCHART with chart position="topleft" */
/* >>> If data is not pre-sorted, chart is drawn backwards! */
%RunDayDateTime(RunDayDateTimeAsFileNameSuffix=YES);
%let SASenv = V94TS1M3;
%let Path = /folders/myfolders/! ! ! ODS Excel Destination/results/;
%let CodePath = /folders/myfolders/! ! ! ODS Excel Destination/code/;
%let ZoomPct = 143;
%let N = 23; /* Demo Step */
proc sort out=work.stocks data=sashelp.stocks
   (where=((stock="IBM") and ('01Jan00'd LE date LE '31dec00'd));
   by date; run;
ods noresults; ods _all_ close;
ods excel file="&Path.&SASenv._Demo&N._&FileNameDTsuffix..xlsx"
   options(zoom="&ZoomPct" sheet_name='Microsoft Chart');
title1 "IBM Stock - 1 Jan 2000 to 31 Dec 2000 - &SASenv - ODS EXCEL - PROC MSCHART - &ZoomPct% Zoom - position='topleft' hides table of input";
proc mschart data=work.stocks category=date
   width=8in height=4in position="topleft";
   chartattrs borderattrs =(type=solid solid_color=black)
   backfillattrs=(type=solid solid_color=yellow) nowall;
   line close; vcolumn volume / secondary;
   categoryaxis type=date;
   primaryaxis title; secondaryaxis title;
   format date monyy5.;
   label close = 'Close'; label volume = 'Volume(millions)';
run;
ods excel close;
/* PROC MSCHART with chart position="topright" */
/* >>> If data is not pre-sorted, chart is drawn backwards! */
%
%RunDayDateTime(RunDayDateTimeAsFileNameSuffix=YES);
%let SASenv = V94TS1M3;
%let Path = /folders/myfolders/!
%let CodePath = /folders/myfolders/!
%let ZoomPct = 143;
%let N = 24; /* Demo Step */
proc sort out=work.stocks data=sashelp.stocks
   (where=((stock="IBM") and ('01Jan00'd LE date LE '31dec00'd)));
by date; run;
ods noresults; ods _all_ close;
odsexcel file="&Path.&SASenv._Demo&N._&FileNameDTsuffix..xlsx"
options(zoom="&ZoomPct" sheet_name='Microsoft Table and Chart');
title1 "IBM Stock - 1 Jan 2000 to 31 Dec 2000 - &SASenv - ODS EXCEL - PROC MSCHART - &ZoomPct% Zoom - position='topright' shows table of input";
proc mschart data=work.stocks category=date
   width=8in height=4in position="topright";
chartattrs borderattrs  =(type=solid solid_color=black)
   backfillattrs=(type=solid solid_color=yellow) nowall;
line close; vcolumn volume / secondary;
categoryaxis type=date;
primaryaxis title; secondaryaxis title;
format date monyy5.;
label close  = 'Close'; label volume = 'Volume(millions)';
run;
odsexcel close;
Alternative to an Index

From time to time I encounter an Excel workbook that has so many tabs that it’s difficult to find the sheet I’m looking for. In fact, an index sheet is a solution. But then, once I’m in a worksheet far enough to the right, the index tab is no longer within view, and I need to remember to find the full scroll left arrow. I don’t like complicated workbooks. I think that a workbook should have all tabs in view at the same time, and their labels should all be readable. I like the simpler solution shown below. It’s not an example substitute for the extreme case that I am objecting to because there are only ten plus one spreadsheets, but I have used it, in Real Life, where the number of spreadsheets was much, much larger. The code is on the following pages.

After clicking on UnitedStates (note the link back to the summary workbook):
%RunDayDateTime(RunDayDateTimeAsFileNameSuffix=YES);

%let SASenv = V94TS1M2;
%let Path = D:\!! ! ! ODS Excel Destination\results\;
%let CodePath = D:\!! ! ! ODS Excel Destination\code\;
%let ZoomPct = 250; /* want to fill the slide */
%let ODSstyle = HTMLblue; /* this is the shipped default */
%let N = 25; /* Demo Step */

data work.ShoesWithRevisedRgnNames;
set sashelp.shoes;
Region = compress(Region,' ');
Region = translate(Region,'_','/');
run;

proc summary data=work.ShoesWithRevisedRgnNames;
class Region Product;
var sales;
output out=SalesSummary(drop=_freq_ where=(_type_ IN (2,3)))
  sum=;
run;

data work.SalesByRegion(drop=Product _type_)
  work.SalesByProductWithInRegion;
set SalesSummary;
if _type_ EQ 2
then output work.SalesByRegion;
else output work.SalesByProductWithInRegion;
run;

proc sort data=work.SalesByRegion(keep=Region)
  out=work.Regions nodupkey;
by Region;
run;

data ToFormat(drop=Region RegionFileName);
length fmtname $ 7 type $ 1 Start $ 25 Label $ 256;
retain fmtname 'LnkXLSX' type 'C';
set work.Regions;
Start = left(Region);
Label = "&Path.|| trim(left(Region)) ||
          "&_SASenv._Demo&N._&FileNameDTsuffix..xlsx";
run;

proc format lib=work cntlin=ToFormat;
run;
quit;
data _null_;  
set Regions end=LastOne;  
call symput('Region'||trim(left(_N_)),trim(left(Region)));  
if LastOne;  
call symput('RegionCount',_N_);  
run;  
footnote1 justify=left bold "Source Data: SASHELP.SHOES";  
footnote2 justify=left bold "Code: &CodePath.Demo&N..sas";  
footnote3 justify=left bold "Run on: &RunDayDateTime";  
ods noresults;  
ods _all_ close;  
%macro RegionSalesDetail;  
%do i = 1 %to &RegionCount %by 1;  
ods excel style=styles.&ODSstyle  
file="&Path.&Region&i..&SASenv..Demo&N..&FileNameDTsuffix..xlsx"  
options(embedded_titles='yes' embedded_footnotes='yes'  
zoom="&ZoomPct"  
sheet_name='Sales This Region By Product'  
title_footnote_width='8' absolute_row_height='14');  
title1 justify=left bold "&&Region&i Shoe Sales - &SASenv ODS EXCEL - ODS Style &ODSstyle - &ZoomPct.% Zoom";  
title2 justify=left bold color=blue underlin=1  
link="&Path.&SASenv..Demo&N..&FileNameDTsuffix..xlsx"  
'Click here to get back to By Region Sales Summary';  
proc print data=work.SalesByProductWithInRegion noobs;  
where Region EQ "&&Region&i";  
id Region;  
var Product Sales;  
run;  
ods excel close;  
%end;  
%mend RegionSalesDetail;  
%RegionSalesDetail;  
%RegionSalesDetail;  
ods excel style=styles.&ODSstyle  
file="&Path.&SASenv..Demo&N..&FileNameDTsuffix..xlsx"  
options(embedded_titles='yes' embedded_footnotes='yes'  
zoom="&ZoomPct"  
sheet_name='Sales By Region'
NOTE: Among my disappointments with the old ODS tagset solutions was the fact that a tagset I needed for a certain feature did not support the use of hyperlinks in the worksheet table cells.

The disadvantage of this solution is that it creates multiple files. If they can be used in situ, then there is no inconvenience. But if you want to share results via email, you need to send the whole collection, either as eleven file attachments, or as an attached zipfile (probably the better choice).
Revisiting the ODS Excel Index

Well, I decided to see what the result would look like, for THIS CASE of only ten detail worksheets plus their corresponding summary worksheet. It’s OK. All eleven worksheet tabs are in view, as you can verify if you use your Acrobat Reader magnifier. See results below.

Index Sheet:

After Clicking on Sales By Region tab:

After Clicking on United States tab:

Code is on the following pages.
data work.ShoesWithRevisedRgnNames;
set sashelp.shoes;
Region = compress(Region,' ');
Region = translate(Region,'_','/');
run;

proc summary data=work.ShoesWithRevisedRgnNames;
class Region Product;
var sales;
output out=SalesSummary(drop=_freq_ where=(_type_ IN (2,3)))
sum=;
run;

data work.SalesByRegion(drop=Product)
  work.SalesByProductWithInRegion;
set SalesSummary;
drop _type_;  
if _type_ EQ 2
then output work.SalesByRegion;
else output work.SalesByProductWithInRegion;
run;

proc sort data=work.SalesByRegion(keep=Region) out=work.Regions
  nodupkey;
by Region;
run;

data _null_; 
set Regions end=LastOne;
call symput('Region'||trim(left(_N_)),trim(left(Region)));
if LastOne;
call symput('RegionCount',_N_);
run;

footnote1 justify=left bold "Source Data: SASHELP.SHOES";
footnote2 justify=left bold "Code: &CodePath.Demo&N..sas";
footnote3 justify=left bold "Run on: &RunDayDateTime";
ods noresults; ods _all_ close;
%macro RegionSalesDetail;
  %do i = 1 %to &RegionCount %by 1;

  ods excel options(sheet_name="&&Region&i");
title1 justify=left bold "&&Region&i Shoe Sales - &SASenv ODS EXCEL - ODS Style &ODSstyle - &ZoomPct.% Zoom";
proc print data=work.SalesByProductWithInRegion noobs;
  where Region EQ "&&Region&i";
  id Region;
  var Product Sales;
run;

%end;
%mend RegionSalesDetail;

ods excel style=styles.&ODSstyle
  file="&Path.&SASenv._Demo&N._&FileNameDTsuffix..xlsx"
  options(embedded_titles='yes' embedded_footnotes='yes'
    zoom="&ZoomPct"
    sheet_interval='proc' 'index='yes'
    title_footnote_width='8' absolute_row_height='14');
ods excel options(sheet_name='Sales By Region');
title1 justify=left bold "Shoe Sales By Region - &SASenv ODS EXCEL - ODS Style &ODSstyle - &ZoomPct.% Zoom";
proc print data=work.SalesByRegion noobs;
  id Region;
  var Sales;
run;

%RegionSalesDetail;

ods excel close;

options obs=max;
Options Demonstrated In This Paper

autofilter
colors (Table of Contents)
embedded_titles, embedded footnotes
frozen_headers, frozen_rowheaders
hidden_columns, hidden_rows
index (better choice than Table of Contents)
sheet_interval
sheet_label
sheet_name
start_at /* assignable values are different in TS1M2 vs TS1M3 */
suppress_byline
tab_color
title_footnote_nobreak /* TS1M3 */
title_footnote_height, title_footnote_width
zoom, blank_sheet
PROC MSCHART requires ODS Excel (TS1M3)
use of tagattr in style= on var statements
creating a hyperlink in a worksheet subtitle

Options Not Demonstrated In This Paper

absolute_column_width
absolute_row_height
embed_titles_once
embed_footnotes_once
formulas
msg_level
row_heights

Printing Options (None Were Demonstrated In This Paper)

center_horizontal
center_vertical
column_repeat
blackandwhite
dpi
draftquality
fittopage
gridlines
orientation
page_order_across
pages_fitheight
pages_fitwidth
print_area /* TS1M3 */
print_footer
print_footer_margin
print_header
print_header_margin
rowbreaks_count /* TS1M3 */
rowbreaks_interval /* TS1M3 */
rowcolheadings
row_repeat
scale

Options Not Retained From ExcelXP

ascii_dots autofilter_table
autofit_height
auto_subtotals
contents_workbook
convert_percentages
currency_format
currency_symbol
decimal_separator
default_column_width
merge_titles_footnotes
minimize_style
missing_align
numeric_test_format
pagebreaks
row_height_fudge
skip_space
thousands_separator
width_fudge
width_points
wraptext
configuration_name
configuration_file
Conclusion

NOTE: Be sure to see Appendix A for documentation of ALL of the ODS Excel options available in TS1M3. Almost all are in release TS1M2 of SAS V9.4, but some require TS1M3.

ODS Excel is a definite improvement on all the SAS non-DDE tools to create highly formatted reports that can be opened in Excel. For me, its only significant limitations at this point are no support for Pivot Tables and no support for creating a horizontal panel of tables, graphs, or mixed tables and graphs across a worksheet. The latter unmet capability is most useful if you want to deliver a graph with its supporting input detail next to it on the same worksheet. As an advocate for and creator of communication-effective graphs, I have long said:

**Image + Precise Numbers = quick, easy inference + reliable inference**

References


Author Information

Your questions, comments, suggestions, and alternate solutions are always welcome.

LeRoy Bessler PhD
Mequon, Wisconsin, USA
Le_Roy_Bessler@wi.rr.com

Dr. LeRoy Bessler is Senior Data Scientist in Business Analytics at DentaQuest. He has presented at software user conferences in the US, Canada, and Europe, on effective visual communication (using graphs, tables, web pages, maps, or color), highly formatted Excel reporting from SAS, custom-developed tools to assist SAS server administrators, users, and managers, and Software-Intelligent Application Development methods to maximize Reliability, Reusability, Maintainability, Extendibility, and Flexibility. His SAS experience includes application development and supporting users, servers, software, and data.

SAS, SAS/GRAPH, and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration. Other brand and product names are trademarks of their respective companies.
Appendix A. ODS Excel Documentation

ODS Excel Options Received from SAS Developer Wayne Hester
Reformatted and With Added Comments by LeRoy Bessler
(Any errors, typos, or confusion are the responsibility of LeRB.)

Basic Syntax:

```sas
ods noresults; /* avoid opening result in SAS session */
ods _all_ close;
ods excel file="AnyPath\AnyFileName.xlsx"
    options(name1='value1' name2='value2' ... );
< SAS code goes here >
ods _all_ close;
```

Comments By LeRB:

The information below is documentation.

For each option is listed:
the **type** of value to be assigned;
if **type='word'**, the list of valid values is given, as **value=**;
the **default** for that option; and
the description of the option as **summary=**.

The assigned value to any option must be quoted.

Since, in the ExcelXP Tip Sheet of March 2015 (document no. 1130598_0315),
which is NOT documentation for ODS Excel,
Boolean choices (for **type='bool'**) are usually ‘yes’ | ‘no’ | ‘on’ | ‘off’,
but sometimes ‘yes’ | ‘no’ | ‘true’ | ‘false’,
I conclude that the safest rule of thumb is to always use either ‘yes’ or ‘no’.

**type='number_bool'** means that the valid choices are either ‘yes’ | ‘no’ or a ‘number’

**type='number_list'** means that the valid choice is a comma-separated list of numbers

**type='number_list_range'** means that the valid choices are either a number, or a
comma-separated list of numbers, or a numeric range of the form numberJ-numberK

Common Options

```sas
absolute_column_width={type='number_list',default=nil,summary='List of widths to use for columns instead of measured widths'}

absolute_row_height={type='number_list',default=nil,summary='List of heights to use for rows instead of measured heights'}
```
autofilter={type='range',default='none',summary='Turn all filtering for data sections of tables in the worksheet'}

blank_sheet={type='string',default=nil,summary='Create a blank worksheet with the given name'}

contents={type='bool',default='no',summary='Create a worksheet that contains the table of contents'} /* LeRB: more verbose, less informative than index */

embedded_titles={type='bool',default='no',summary='Embed titles in the worksheet'}

embedded_footnotes={type='bool',default='no',summary='Embed footnotes in the worksheet'}

embed_titles_once={type='bool',default='no',summary='Embed titles only at the top of the worksheet'}

embed_footnotes_once={type='bool',default='no',summary='Embed footnotes only at the bottom of the worksheet'}

formulas={type='bool',default='yes',summary='Data values that begin with a "=" will become formulas'}

frozen_headers={type='number_bool',default='no',summary='Prevent headers from scrolling with scrollbar'}

frozen_rowheaders={type='number_bool',default='no',summary='Prevent row headers from scrolling with scrollbar'}

hidden_columns={type='number_list_range',default=nil,summary='Range or list of columns to hide'}

hidden_rows={type='number_list_range',default=nil,summary='Range or list of rows to hide'}

index={type='bool',default='no',summary='Create a worksheet that contains an index of all worksheets'}

msg_level={type='string',default='no',summary='Suppress messages from Excel tagset'}

row_heights={type='number_list',default=nil,summary='Positional array of row heights values to use as over rides'}
sheet_interval={type='word',values={'output','table','page','bygroup','bygroups','proc','none'},default='output',summary='Specifies how often a new worksheet is created'}

sheet_label={type='string',default=nil,summary='Specifies a prefix for the worksheet name'}

sheet_name={type='string',default=nil,summary='Specifies the name for the next worksheet'}

start_at={type='string',default='1,1',summary='Specify the starting position for the report'} /* column-row assignment protocol differs in TS1M2 vs TS1M3. In TS1M2, Column 3 Row 2 is assigned with Start_At=(C2). In TS1M3, Column 3 Row 2 is assigned with Start_At=(3,2). */

suppress_bylines={type='bool',default='no',summary='Suppress bylines in the worksheet'}

tab_color={type='string',default=nil,summary='Specifies the color for the next worksheet tab'}

title_footnote_nobreak={type='bool',default='no',summary='Do not allow titles and footnotes to wrap across lines'} /* TS1M3 */

title_footnote_width={type='number',default='0',summary='The number of columns titles and footnotes span'},

zoom={type='number',default='100',summary='Indicates the initial zoom level on the worksheet'}

Printing Options

center_horizontal={type='bool',default='no',summary='Centers worksheet horizontally when printing'}

center_vertical={type='bool',default='no',summary='Centers worksheet vertically when printing'}

column_repeat={type='string',default=nil,summary='Columns to repeat across page when printing'}

blackandwhite={type='bool',default='no',summary='Prints the worksheet in black and white'}

dpi={type='number',default='300',summary='Print resolution'}
draftquality={type='bool',default='no',summary='Indicates that
draft quality should be used for printing'}

fittopage={type='bool',default='no',summary='Fit worksheet to
the page when printing'}

gridlines={type='bool',default='no',summary='Enables grid lines
when printing'}

orientation={type='word',values={'portrait','landscape'},default =
'portrait',summary='Print orientation for the worksheet'}

page_order_across={type='bool',default='no',summary='Set the
page print order to across and then down'}

pages_fitheight={type='number',summary='The number of pages down
to fit the worksheet when printing'}

pages_fitwidth={type='number',summary='The number of pages
across to fit the worksheet when printing'}

print_area={type='text_list',default=nil,summary='Description of
the printed area, as start column, start row, end column, end row'} /* TS1M3 */

print_footer={type='string',summary='The footer text to be used
when printing'}

print_footer_margin={type='number',summary='The footer margin as
set in the page setup dialog'}

print_header={type='string',summary='The header text to be used
when printing'}

print_header_margin={type='number',summary='The header margin as
set in the page setup dialog'}

rowbreaks_count={type='number',default=nil,summary='Inserts a
print page break every number of data rows'} /* TS1M3 */

rowbreaks_interval={type='word',values={'output','proc'},default
='none',summary='Inserts a print page break between each
interval'} /* TS1M3 */

rowcolheadings={type='bool',default='no',summary='Specifies
whether or not row and column headings should be printed'}
row_repeat={type='string',default=nil,summary='Rows to repeat across page when printing'}

scale={type='number',default='100',summary='Indicates the scale level for printing'}