

# SetUp for Sample Programs from the Slide Presentation *Excel Reports from SAS with HTML, ExcelXP, MSOffice2K\_x, TableEditor, or DDE, Which Is Best?*

This document prepared on 9 October 2013

LeRoy Bessler PhD

Bessler Consulting and Research

Le\_Roy\_Bessler@wi.rr.com

Development and Test Context

SAS Display Manager Environment

Output Examples in Slides for 4 Nov 2013 were created with **SAS Version 9.4 (TS1M0)**

This is a discussion of my personal (and recommended) SetUp.

**NOTE:** For brevity (in order to fit on the slides) some code in the slides does not explicitly adhere to all of these practices.

## ODS Code Blocks

Regardless of what you have done  
using Tools > Options > Preferences > Results  
as discussed below,  
always package your ODS step this way:

```
ods listing close;  
ods YourDestination . . . ;  
< Your DATA steps, if any, and Your PROC steps >  
ods YourDestination close;  
ods listing;
```

Leaving the ODS LISTING destination open before your code execution step will create extra output in your Display Manager Results Viewer. If you happen to have submitted statement "ODS NORESULTS;", then that will not happen, but, if you are creating graphic output, then an extra graphic image file will be created for the LISTING destination, and, what is worse, if you have assigned a specific filename to your graphic image file, rather than your taking the default filename, then the actual filename will be different, because your filename will go to the image that was created for the LISTING destination (and placed wherever that happens to go).

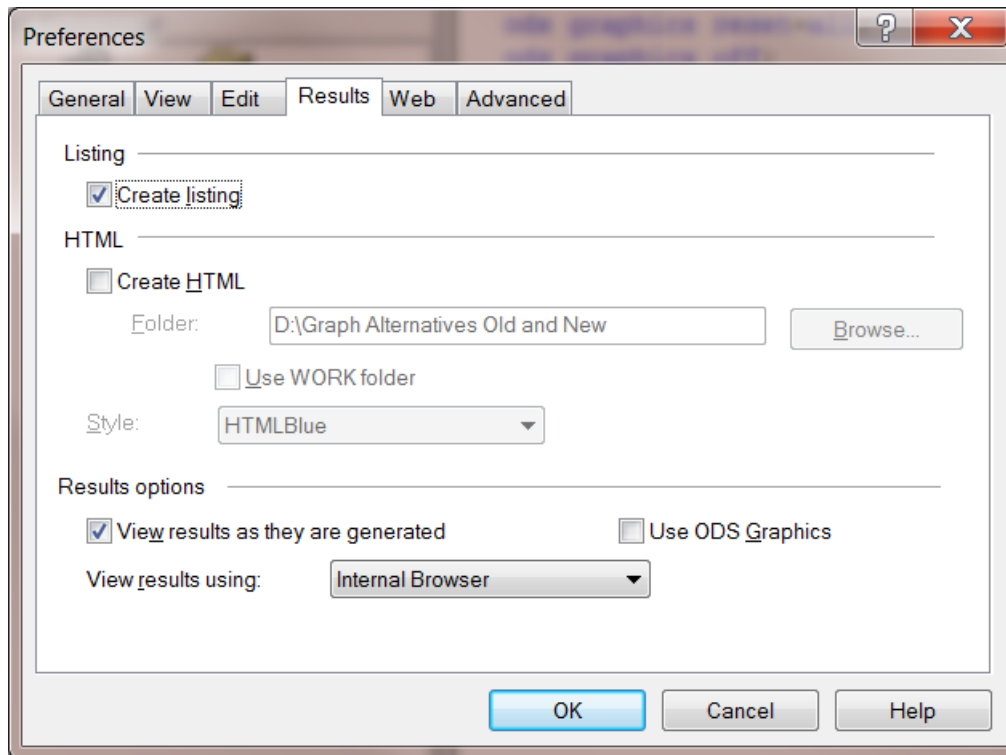
It is always best to close the LISTING destination before your non-LISTING destination code begins.

It is essential to close YourDestination, whatever it is, after your DATA steps, if any, and your PROC steps. If you do not do so, an empty file will be created, despite the fact that your SAS log will contain a message like (in the case of the HTML destination):

**NOTE:** Writing HTML Body file: YourFileName.html

## Display Manager Environment Results Settings

When I go to Tools > Options > Preferences > Results  
My Results Panel Settings look like this:



I have no idea what is the purpose of the underscore below l, H, B, U, w, G, and r. It is NOT a clickable hyperlink.

“Create listing” is a good default.

If your code has not turned on some other output destination, this prevents your getting a message in the SAS log to notify you that no output destinations are active, in which case, if your code has created some displayable output, you cannot see it. However, see remarks above about the “ODS LISTING CLOSE;” statement.

When either creating HTML output with your own code, or creating Excel output with your own code, I recommend UnChecking “Create HTML”. This prevents any possible interference between something that SAS might be doing on its own, which MIGHT interfere with what you want to do.

**NOTE:** There is an unexpected (at least unexpected by me) effect of having identified a default folder for HTML output. In my SetUp above, you can see that it is

D:\Graph Alternatives Old and New

If I were to Check “Create HTML”, then whenever my code submission created displayable report output, it would be routed to that default folder. However, the default

folder comes into play even when “Create HTML” is UnChecked if I use an ODS statement of this form:

```
ODS HTML file="FileName.html" ...;
```

SAS routes this file to the default folder, which is unsurprising and might be OK.

The best syntax, which ALWAYS works (and also works well for non-HTML destinations), is:

```
ODS HTML path="C:\Folder" body="FileName.xls" ...;
```

and, if graphs are included in the output,

```
ODS HTML path=" C:\Folder" (url=none) body="FileName.xls" ...;
```

Let me close this section by repeating:

When either creating HTML output with your own code, or creating Excel output with your own code, I recommend UnChecking “Create HTML”. This prevents any possible interference between something that SAS might be doing on its own, which MIGHT interfere with what you want to do.

### Graphics-Related Setting and Statements

I UnCheck “Use ODS Graphics”.

I like to keep control. Sometimes I use SAS/GRAPH (the old graphics alternative), and other times I use ODS Graphics (the new graphics alternative). If I need ODS Graphics, I am likely to want to customize the result, in which case I need to include an explicit ODS GRAPHICS statement in my code, which, even if the “ON” parameter is omitted from the statement, will turn on ODS Graphics. Furthermore, if I submit an SG (Statistical Graphics) procedure, then ODS Graphics is, in effect, turned on automatically, even if prior code submitted explicitly turned ODS Graphics OFF. In that situation, all of the ODS Graphics customization parameters will be at their defaults.

```
options reset=all;
```

The above statement prevents any previously set GOPTIONS values from code submitted earlier during the session to affect the current code to be submitted.

Even if you are not creating a graph in the current code, this is good to do and is, at worst, harmless.

```
ods graphics / reset=all;
```

The above statement prevents any previously set ODS GRAPHICS values from code previously submitted during the session to affect the current code to be submitted. Even if you are not creating a graph in the current code, this is good to do and is, at worst, harmless.

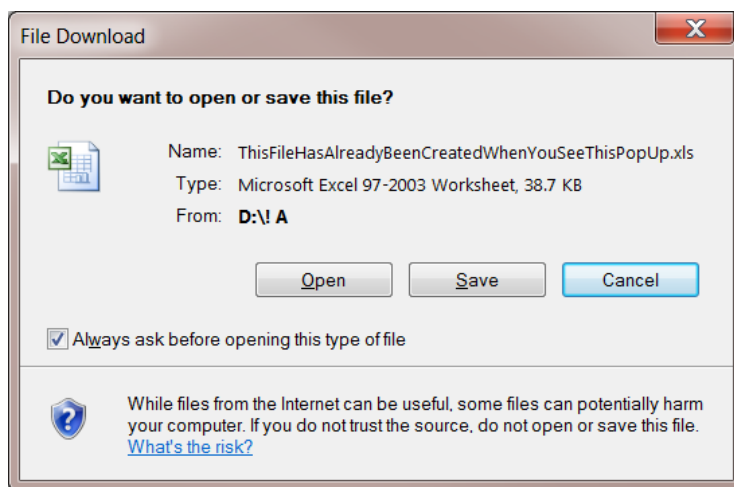
```
ods graphics off;
```

In case ODS Graphics is by default turned on, or was turned on for code previously submitted the session, the above statement turns off the feature. If you are not creating a graph in the current code, this is good to do and is, at worst, harmless.

### Control of Code Execution and Aftermath

```
ods noresults;
```

When your code creates an Excel file, the above statement prevents the pop-up message that asks you to Open or Save or Cancel. The file HAS already been created. You do not need to explicitly Save it. Here is what that pop-up message looks like, which requires you to respond:



However, to create certain kinds of Excel output with the ODS TableEditor tagset, you must precede your code with:

```
ods results;
```

If omitted, your code will run without error or warning, but the expected output will not appear. Such outputs are the result of a two-step process, where the first step is running your code, and the second step is your responding to prompts. If you have specified “ods noresults;”, then they will not appear. In that case, you will be unable to respond to them (the FIRST prompt is the essential one), and your Excel output will not be created.

### General Tip

Some SAS procedures by default include a subtitle to identify themselves as the source of a tabular report, such as “UNIVARIATE Procedure”. You can prevent that with this statement:

```
ods noproctitle;
```

before the PROC step code. If creating output strictly for personal use, leaving in this information is probably OK, and might even be useful documentation.

## Important Tip for Relocatability of Excel Files with Imbedded SAS Graphs

If your web page output or your Excel file output contains any graph(s) created with SAS/GRAPH or ODS Graphics, there is (are) a separate image file(s) created. If you want to ensure relocatability, i.e., that the HTML file and the image file(s) can be moved to another location, or attached to an email (either as separate files, or as a zip file), you must use this code structure:

```
ods html path="YourOutputFolder" (url=none)
        body="YourFileName.ext";
```

with optional parameters such as STYLE=, and where ext could be xls or html. Do not use a second folder, designated with GPATH= for the SAS-sourced graphs. With the code above, they will go to the folder designated with PATH=. But (url=none) is also essential for relocatability. Without it, the HTML file's internal pointer to image file will include the full path, rather than indicating, by default, that the image file is in the same folder as the HTML file.

The use of path= and (url=none) described above also applies to any ODS tagset used to create an Excel file (or HTML file) with imbedded SAS graph(s).

## If You Are Using SAS/GRAPH

IF you are specifying the name of the image file(s) to be created, AND IF you are possibly going to rerun the code during the same SAS session, then, before the first invocation of a SAS/GRAPH procedure, you should include this code:

```
proc catalog cat=work.gseg kill; run; quit;
/* Delete all content so image files can be overwritten */
/* if rerunning the code during the same session.          */
/* The first run of this PROC step during a session
   will cause these messages:
   ERROR: Catalog WORK.GSEG does not exist.
   WARNING: Command CATALOG not processed
           because of errors noted above. */

/* The above PROC step is needed only if you are using
SAS/GRAPH AND you are using NAME= to specify the image filename
AND you might rerun the code during the same SAS session.
E.g., in PROC GPLOT, if you code:
    PLOT y*x / name='MyGPLOT' ...;
and if you rerun the code in the same session without clearing
work.gseg, you will get this message in the SAS log:
    NOTE: Graph's name, MYGPLOT, changed to MYGPLOT1.
           MYGPLOT is already used or not a valid SAS name. */
```

If you wish to avoid the first run ERROR and WARNING messages described above, you could make the PROC step conditional on it being NOT the first chance to possibly run the code during the session. To make it conditional, replace the PROC step with this code:

```
%macro CatClean;

%global FirstRun;
%if %length(&FirstRun) EQ 0
%then %do;
    %let FirstRun = N;
    %put First Run Has Occurred;
%end;
%else %do;
proc catalog cat=work.gseg kill; run; quit;
%end;

%mend CatClean;

%CatClean;
```