Introduction to the SAS® Output Delivery System (ODS)

Kirk Paul Lafler, sasNerd

Abstract

SAS® software is the "gold" standard for robust and reliable data access, manipulation, analytics, analysis, reporting, and data discovery. Microsoft Excel® is the most widely used software in the world. This paper introduces the basics of the Output Delivery System (ODS), ODS syntax; along with the application and construction of ODS code to share data, tables, statistics, images, reports, and results to create exciting and robust RTF, PDF, HTML, OUTPUT, and Microsoft Excel files and workbooks.

Keywords

SAS, ODS, ODS RTF, ODS PDF, ODS HTML, ODS Excel, Excel workbooks, styles, images

Introduction

Technology is moving at an astounding pace. With many new features and enhancements introduced in Version 9.4, users can finally sound the trumpets for what could be the most exciting time for output delivery since the introduction of the color monitor and printer. Never again will SAS users be confined to boring monospace output. Instead, output delivery has entered a new age taking full advantage of font characteristics, color, a variety of output layouts, and numerous other features. This paper and presentation provides users with many new and exciting features available with the ODS Excel destination. Sprinkled throughout will be numerous tips, tricks, and techniques that will, hopefully, make the learning process a little easier.

Attendees will learn numerous tips, tricks, and techniques in handling output including the advantages of using the ODS Excel destination; how ODS handles raw data; how ODS combines raw data with table definitions; how to open, close, and manage the ODS Excel destination; how the Results window stores links to ODS output; how selection and exclusion lists are used; and how to send and/or create Excel spreadsheets from SAS data sets and results using the ODS Excel destination.

Table Used in Examples

The data set used in all the examples in this paper is the SASHELP.CARS. The SASHELP.CARS data set consists of 428 observations and 15 variables, illustrated below.

	Make	Model	Type	Origin	DriveTrain	MSRP	Invoice	EngineSize	Cylinders	Horsepower	MPG_City	MPG_Highway	Weight	Wheelbase	Length
1	Acura	MDX	SUV	Asia	All	$36,945	$33,337	3.5	6	265	17	23	4451	106	189
2	Acura	RSX Type S 2dr	Sedan	Asia	Front	$23,820	$21,761	2	4	200	24	31	2778	101	172
3	Acura	TSX 4dr	Sedan	Asia	Front	$26,990	$24,647	2.4	4	200	22	29	3230	105	183
4	Acura	TL 4dr	Sedan	Asia	Front	$33,195	$30,299	3.2	6	270	20	28	3575	108	186
5	Acura	3.5 RL 4dr	Sedan	Asia	Front	$43,755	$39,014	3.5	6	225	18	24	3880	115	197
6	Acura	3.5 RL w/Navigation 4dr	Sedan	Asia	Front	$46,100	$41,100	3.5	6	225	18	24	3893	115	197
7	Acura	NSX coupe 2dr manual S	Sports	Asia	Rear	$89,765	$79,978	3.2	6	290	17	24	3153	100	174
8	Audi	A4 1.8T 4dr	Sedan	Europe	Front	$25,940	$23,508	1.8	4	170	22	31	3252	104	179
9	Audi	A41.8T convertible 2dr	Sedan	Europe	Front	$35,940	$32,506	1.8	4	170	23	30	3638	105	180
10	Audi	A4 3.0 4dr	Sedan	Europe	Front	$31,840	$28,846	3	6	220	20	28	3462	104	179
11	Audi	A4 3.0 Quattro 4dr manual	Sedan	Europe	All	$33,430	$30,366	3	6	220	17	26	3583	104	179
12	Audi	A4 3.0 Quattro 4dr auto	Sedan	Europe	All	$34,480	$31,388	3	6	220	18	25	3627	104	179
13	Audi	A6 3.0 4dr	Sedan	Europe	Front	$36,640	$33,129	3	6	220	20	27	3561	109	192
14	Audi	A6 3.0 Quattro 4dr	Sedan	Europe	All	$39,640	$35,992	3	6	220	18	25	3880	109	192
15	Audi	A4 3.0 convertible 2dr	Sedan	Europe	Front	$42,490	$38,325	3	6	220	20	27	3814	105	180
16	Audi	A4 3.0 Quattro convertible 2dr	Sedan	Europe	All	$44,240	$40,075	3	6	220	18	25	4013	105	180
17	Audi	A6 2.7 Turbo Quattro 4dr	Sedan	Europe	All	$42,840	$38,840	2.7	6	250	18	25	3836	109	192
18	Audi	A6 4.2 Quattro 4dr	Sedan	Europe	All	$49,690	$44,936	4.2	8	300	17	24	4024	109	193
19	Audi	A8 L Quattro 4dr	Sedan	Europe	All	$69,190	$64,740	4.2	8	330	17	24	4399	121	204
20	Audi	S4 Quattro 4dr	Sedan	Europe	All	$48,040	$43,556	4.2	8	340	14	20	3825	104	179
21	Audi	RS 6 4dr	Sports	Europe	Front	$84,600	$76,417	4.2	8	450	15	22	4024	109	191
22	Audi	TT 1.8 convertible 2dr (coupe)	Sports	Europe	Front	$35,940	$32,512	1.8	4	180	20	28	3131	95	159
23	Audi	TT 1.8 Quattro 2dr (convertible)	Sports	Europe	All	$37,390	$33,891	1.8	4	225	20	28	2921	96	159
24	Audi	TT 3.2 coupe 2dr (convertible)	Sports	Europe	All	$40,590	$36,739	3.2	6	250	21	29	3351	96	159
25	Audi	A6 3.0 Avant Quattro	Wago	Europe	All	$40,840	$37,060	3	6	220	18	25	4035	109	192
26	Audi	S4 Avant Quattro	Wago	Europe	All	$49,090	$44,446	4.2	8	340	15	21	3936	104	179
27	BMW	X3 3.0i	SUV	Europe	All	$37,000	$33,873	3	6	225	16	23	4023	110	180
28	BMW	X5 4.4i	SUV	Europe	All	$52,195	$47,720	4.4	8	325	16	22	4824	111	184

Output Delivery Basics

Years ago, ODS was released to the user community as a way to address the inherent weaknesses found with the creation of traditional reports and output results. ODS enables "quality" looking data, reports, and output results to be produced without having to import, or use copy and paste techniques, into word processors such as MS-Word. Although the SAS® Output Delivery System *(ODS)* continues to support the creation of traditional SAS Listing or monospace output (i.e., Listing), it provides users with many ways to access and format data, reports, and output results. ODS provides many new features providing users with greater flexibility including the Output, RTF, PDF, HTML, XML, PowerPoint, and Excel destinations. In Version 9.4 (M3), many new output formatting features and options are introduced for SAS users to take advantage of, including the ODS Excel destination. Users have a powerful and easy way to create and access formatted procedure and DATA step output.

Tip #1 – ODS and "Batch" Use

Many of the ODS features found in the interactive side of the SAS Display Manager System (DMS) can also be used in batch processing. ODS has been designed to make exciting new formatting options available to users. In a windowing environment, ODS can send output to the following destinations: the output window (DMS), the listing file, HTML, SAS dataset, rich text format (RTF), postscript file, external output file (non-SAS file), or output device. The only exception for batch processing is having output sent to the output window.

Tip #2 – What if I'm Still Not Using the latest Version of SAS Software

First introduced in Version 6.12, ODS offered users the capability to format output to destinations other than traditional line printers. Version 9.4 (M3) introduced the ODS Excel destination to deploy output to the web, the creation of SAS datasets and rich text format (RTF) files, and DATA step interaction. ODS was designed to address the inherent weaknesses found in traditional SAS output. It enables the creation of "quality" looking output without having to import it into word processors such as MS-Word. New output enhancements were introduced in Version 8 and then in Version 9, including the ability to create postscript files and output customizations. To take full advantage of the power offered in ODS, it is recommended that users upgrade to the latest Version as early as possible to take advantage of these features.

Tip #3 – ODS and System Resources

A very important efficiency consideration is to remember that ODS currently supports the following destinations: 1) Listing, 2) rich text format (RTF), 3) postscript, 4) HTML, 5) PowerPoint, 6) Excel and 7) Output. (Note: It also provides support with the creation of output and results in the DATA step.) Each ODS destination can be open or closed at the same time. For each open destination, ODS sends output object(s) to it. System resources are used when a destination is open. As a result, make sure any and all unwanted open destinations are closed to conserve on resources.

Tip #4 – Closing Destinations before and after use

The Listing destination is open by default at SAS invocation, while the other destinations are closed. If nothing is done to suppress output to the Listing destination, your SAS programs automatically produce Listing output, just as they always have in the SAS System. If you needed to suppress printed output from being sent to the Listing destination (or DMS Output window) before the execution of a procedure step, the following ODS statement would be issued:

```
ODS Listing Close;
  Proc univariate data=sashelp.Cars;
  Run;
ODS Listing;
```

By closing the Listing destination before the procedure code, the SAS System is actually suppressing output to that destination until it is reopened. The preceding example shows that at the end of the procedure step, the Listing destination is reopened by specifying ODS Listing; so output from subsequent steps can be sent to the Listing destination.

Tip #5 – Deleting Output from the Results Window

The Results window identifies procedure output that has been produced, providing users with an improved way to manage their output. It is customarily a good thing to remove unwanted output displayed in this window to conserve on system resources. The Results window is opened by specifying the command **ODSRESULTS** on the DMS command line or by selecting **View Results** from the pull-down menu.

To delete procedure output, use the following steps:

1. Select the procedure folder you want to remove.
2. Click the Delete button on the task bar.
3. Select "Yes" to confirm the deletion of the procedure output folder.

Tracing Output

Output producing procedures create one or more pieces or tables of information. In order to discriminate between the various pieces of information, it is advantageous to know the names assigned to each piece of information. The ability to display the names of individual pieces of information generated on output is referred to as *tracing*.

Tip #6 – Tracing Procedure Output

The trace record displays information about the data component, the table definition, and the output object. By specifying an ODS Trace ON ; statement, the SAS System turns the trace feature on and prints results to the SAS Log destination.

```
ODS Trace ON ;
  Proc univariate data=sashelp.Cars ;
  Run ;
ODS Trace Off ;
```

For example, the trace record displays the following output objects to the SAS Listing destination: 1) Moments, 2) BasicMeasures, 3) TestForLocation, 4) Quantiles, and 5) ExtremeObs. A sample trace record containing each output object's name, label, template, and path is displayed for the Univariate procedure. Note that for each output object, the name, label, template, and path is displayed.

```
Output Added:
-------------
Name:       Moments
Label:      Moments
Template:   base.univariate.Moments
Path:       Univariate.MSRP.Moments
-------------

Output Added:
-------------
Name:       BasicMeasures
Label:      Basic Measures of Location and Variability
Template:   base.univariate.Measures
Path:       Univariate.MSRP.BasicMeasures
-------------

Output Added:
-------------
Name:       TestsForLocation
Label:      Tests For Location
Template:   base.univariate.Location
Path:       Univariate.MSRP.TestsForLocation
-------------

Output Added:
```

```
-------------
Name:       Quantiles
Label:      Quantiles
Template:   base.univariate.Quantiles
Path:       Univariate.MSRP.Quantiles
-------------

Output Added:
-------------
Name:       ExtremeObs
Label:      Extreme Observations
Template:   base.univariate.ExtObs
Path:       Univariate.MSRP.ExtremeObs
-------------
```

Selecting Output with ODS

A selection or exclusion list exists for each open ODS destination. These lists determine which output objects to send to ODS destinations. To accomplish this, ODS verifies whether an output object is included in a destination's selection or exclusion list. If it does not appear in this list, then the output object is not sent to the ODS destination. If an output object is included in the list, ODS determines if the object is included in the overall list. If it does not appear in this list, then the output object is not sent to the ODS destination. If an output object is included in the overall list then ODS sends it to the selected destination.

Tip #7 – Selecting Desired Pieces of Information

Once you know the individual names of each output object (from the trace), you can then select the desired object for reporting purposes. The syntax is:

```
ODS select output-component-name;
```

where *output-component-name* is the name of the desired output object. To select just the output object Moments from the Univariate procedure, the following syntax is specified:

```
ODS Select Moments ;
Proc univariate data=sashelp.Cars ;
Run ;
```

Export, Import and the LIBNAME Engine

Creating and accessing Excel files requires a SAS/ACCESS to PC Files license. SAS/ACCESS includes a component called PC Files Server. It accepts requests from SAS to convert to and from Excel (and other file types), allows newer 64-bit desktop machines to work with older 32-bit machines, and circumvents the bit architecture mismatch.

Using PROC EXPORT

PROC EXPORT reads data from a SAS data set and writes it to an external data source. A variety, but incomplete list, of external data sources include:

- ✓ MS-Access databases
- ✓ MS-Excel files
- ✓ SPSS files
- ✓ Stata files – rectangular files stored in memory (Reference: http://data.princeton.edu/stata/DataManagement.html)
- ✓ Delimited external files
- ✓ Lotus 1-2-3 spreadsheet files
- ✓ Paradox files

Tip #8 – Creating an Excel File Under Windows with PROC EXPORT

Creating an Excel file is easy with PROC EXPORT. To create an Excel spreadsheet under Windows from the SASHELP.CARS data set, specify the following statement and options:

```
PROC EXPORT DATA=sashelp.Cars
        OUTFILE='c:\Cars.xls'
            DBMS=EXCEL
            REPLACE;
    RUN;
```

Tip #9 – Creating an Excel File Under Unix with PROC EXPORT

To create an Excel file under Unix from the SASHELP.CARS data set, specify the following statement and options:

```
PROC EXPORT DATA=sashelp.Cars
        OUTFILE='c:\Cars.xls'
            DBMS=EXCELCS
            REPLACE;
    RUN;
```

Tip #10 – Creating a Subsetted Excel File with PROC EXPORT

Exporting a subset to an Excel file is accomplished using PROC EXPORT with a WHERE= data set option. For example, to export the "SUV" vehicles from the SASHELP.CARS data set to an Excel file, specify the following PROC EXPORT statement and WHERE= data set option:

```
PROC EXPORT DATA=SASHELP.CARS
        (WHERE=(type="SUV"))
        OUTFILE='c:\SUV_Vehicles.xls'
            DBMS=EXCEL;
    RUN;
```

Using PROC IMPORT

PROC IMPORT reads data from an Excel file to a SAS data set. A variety, but incomplete list, of a variety of external data sources include:

- ✓ MS-Excel files
- ✓ MS-Access databases
- ✓ Lotus 1-2-3 spreadsheet files

Tip #11 – Creating a SAS Data Set Under Windows with PROC IMPORT

To create a SAS data set from an Excel file under Windows from the SASHELP.CARS data set, specify the following PROC IMPORT statement and options:

```
PROC IMPORT OUT=sashelp.Cars
        DATAFILE='c:\Cars.xls'
            DBMS=EXCEL2002
            REPLACE ;
    RUN ;
PROC PRINT DATA=sashelp.Cars noobs ;
    RUN ;
```

Tip #12 – Creating a SAS Data Set Under Unix with PROC IMPORT

This example illustrates the syntax used to create a SAS data set from an Excel file under Unix.

```
PROC IMPORT OUT=sashelp.Cars
        DATAFILE='c:\Cars.xls'
            DBMS=EXCELCS
            REPLACE ;
RUN ;
PROC PRINT DATA=sashelp.Cars noobs ;
RUN ;
```

Using the LIBNAME Engine

The **LIBNAME** engine associates a SAS libref with a Microsoft Excel workbook or Microsoft Access database. The LIBNAME syntax to create a Windows Excel file is:

```
LIBNAME libref EXCEL <physical-file-name>
        <SAS/ACCESS-engine-connection-options>
        <SAS/ACCESS-libname-options>;
```

Where

libref *represents the user-defined alias to reference a SAS data set,*

EXCEL *is the SAS/ACCESS name for the interface to PC files under Windows,*

SAS/ACCESS-engine-connection-options *provides information to connect to your PC files,*

SAS/ACCESS-libname-options *defines how SAS is to interact with your data source.*

Tip #13 – Exporting a SAS Data Set to a Windows-based Excel File

A LIBNAME statement can be used to export a SAS data set to a Windows-based Excel file. For example, to export the data in the SASHELP.CARS data set to a Windows Excel file, specify the following LIBNAME statement and XLS options:

```
LIBNAME MYXLS EXCEL "c:/Cars.XLS" ;
  DATA MYXLS.Sheet1 ;
    SET sashelp.Cars ;
  RUN ;
LIBNAME MYXLS CLEAR ;
```

Tip #14 – Exporting a SAS Data Set to a Unix-based Excel File

A LIBNAME statement can be used to export a data set to a Unix-based Excel file. For example, to export the data in the SASHELP.Cars data set to an Excel file, specify the following LIBNAME statement and XLS options:

```
LIBNAME MYXLS PCFILES PATH="c:/Cars.XLS" ;
   DATA MYXLS.Sheet1 ;
     SET sashelp.Cars ;
   RUN ;
LIBNAME MYXLS CLEAR ;
```

Creating Unique Output with ODS

Output Delivery System (ODS) can be used to create a variety of output formats. ODS statements are classified as global statements and are processed immediately by the SAS System. ODS statement options control what format engine(s) are turned on and in effect during the step or until another ODS statement is specified. ODS has built-in format engines (e.g., Listing, Output, RTF, PDF, DATA Step, HTML, Excel, PowerPoint and XML). Specifying an ODS statement and destination at a particular point in a program is important, because output-producing PROC and DATA steps will respond by sending output to the open destination.
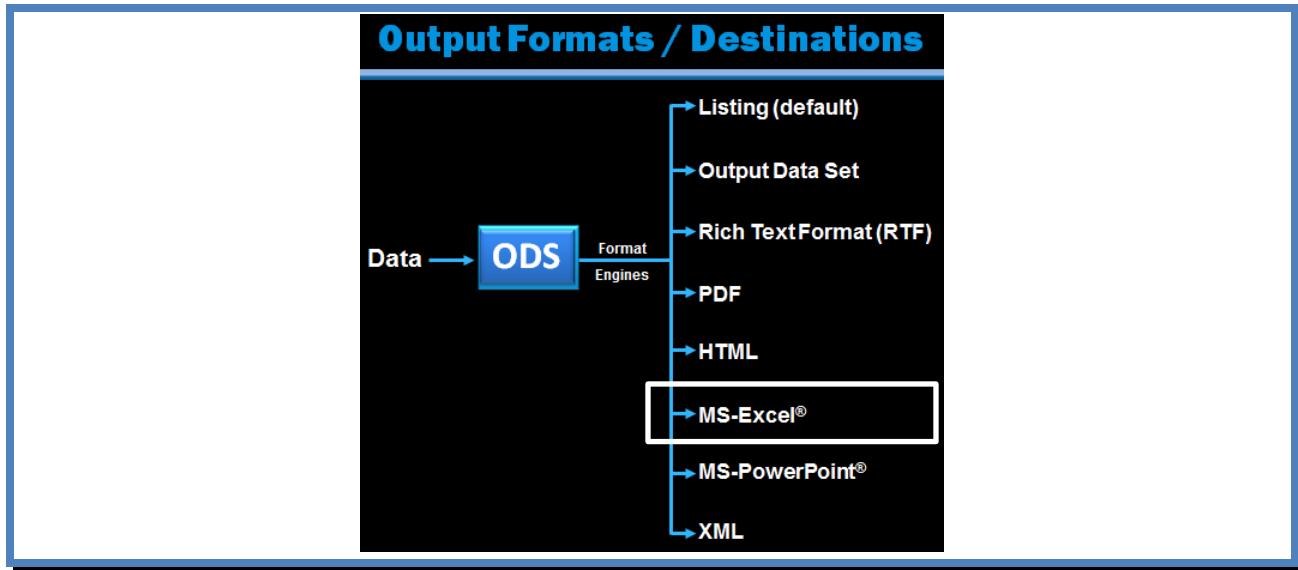
Creating SAS Output Data Sets

Occasionally, output results are needed in a SAS data set rather than in printed form such as the Listing destination. Re-directing SAS procedure output to a data set is relatively simple with ODS. The syntax is:

```
ODS Output output-table-name = user-defined-table-name;
    < SAS Code >
```

where **output-table-name** is the name of the desired output table (component) containing the information you want written to a data set, such as **Moments** in the UNIVARIATE procedure. **User-defined-table-name** is the name you supply for the newly created data set.

Tip #15 – SAS Output Data Sets

A data set can be defined as either a temporary or permanent data set. Once an object is selected, specify the object in the ODS OUTPUT statement. For example, the Moments from the Univariate procedure is selected and output to a SAS data set in the following code.

```
ODS Output Moments = Cars_Moments ;
  Proc univariate data=sashelp.Cars ;
  Run ;
```

When the OUTPUT destination is no longer needed, it can be closed with the following ODS statement:

```
ODS OUTPUT CLOSE;
```

Tip #16 – Creating Rich Text Format (RTF) with ODS

Rich text format (RTF) is text consisting of formatting attributes codes, such as boldface, italics, underline, etc. It is principally used to encapsulate text and formatting attributes during copy-and-paste operations. Because word-processing programs use RTF rather than ASCII when handling data, the need to reformat is a thing of the past. The syntax to create RTF output is:

```
ODS RTF FILE = 'user-specified-file-name';
```

where user-specified-file-name references a complete and fully-qualified output location for the creation and storage of the RTF file, data, and codes. For example, the following code creates an RTF file using the Univariate procedure output. (**Note:** The RTF extension is required).

```
ODS Select Moments ;
ODS RTF FILE='RTF-univariate-output.rtf' ;
  Proc univariate data=sashelp.Cars ;
    Title1 'Delivering RTF Output' ;
  Run ;
ODS RTF Close ;
```

Two RTF output objects are produced and displayed below:

Delivering RTF Output

The UNIVARIATE Procedure
Variable:
MSRP

Moments			
N	428	Sum Weights	428
Mean	32774.8551	Sum Observations	14027638
Std Deviation	19431.7167	Variance	377591613
Skewness	2.79809927	Kurtosis	13.8792055
Uncorrected SS	6.20985E11	Corrected SS	1.61232E11
Coeff Variation	59.2884899	Std Error Mean	939.267478

Delivering RTF Output

The UNIVARIATE Procedure
Variable:
Invoice

Moments			
N	428	Sum Weights	428
Mean	30014.7009	Sum Observations	12846292
Std Deviation	17642.1178	Variance	311244319
Skewness	2.8347404	Kurtosis	13.9461638
Uncorrected SS	5.18479E11	Corrected SS	1.32901E11
Coeff Variation	58.778256	Std Error Mean	852.763949

Tip #17 – Creating and Naming Postscript Files

ODS enables the creation of a postscript file from SAS output to help ensure that formatting and content is preserved while maintaining complete printer independence. The next example shows the creation of a postscript file from Print procedure output.

```
ODS Printer Postscript ;
  Proc print data=sashelp.Cars noobs n ;
    Title1 'Example Creating a Postscript File' ;
```

```
   Run ;
ODS Printer Close ;
```

Since the postscript file was created without assigning a filename, ODS assigns the name: **SASPRT.PS.** Frequently, a user-defined filename is desired rather than the default name. In these cases, the ODS statement and **File=** option can be used to assign a name to the postscript file.

```
ODS Printer Postscript File='Cars' ;
  Proc print data=sashelp.Cars noobs n ;
    Title1 'Example Creating a Postscript File' ;
  Run ;
ODS Printer Close ;
```

Tip #18 – The Custom Reporting Interface – ODS and the DATA Step

To provide greater capabilities when working with custom output created in the DATA step, ODS is integrated into the DATA step. Two new options are necessary to take advantage of ODS: 1) the ODS option in the FILE statement and 2) the _ODS_ option in the PUT statement. These two options are used to direct the results of a DATA step to ODS. For example, the following code directs output from a DATA step through the RTF format engine to create a RTF file.

```
ODS RTF FILE='ods-DATA-step.rtf' ;
  Title1 'Great Cars' ;
  Data NULL ;
    Set sashelp.Cars(where=(type="SUV")) ;
    File print ods ;
    Put _ods_ ;
  Run ;
ODS RTF Close ;
```

The RTF file created in the previous DATA step code is illustrated below.

Great Cars

Make	Model	Type	Origin	DriveTrain	MSRP	Invoice	Engine Size (L)	Cylinders	Horsepower	MPG (City)	MPG (Highway)	Weight (LBS)	Wheelbase (IN)	Length (IN)
Acura	MDX	SUV	Asia	All	$36,945	$33,337	3.5	6	265	17	23	4451	106	189
BMW	X3 3.0i	SUV	Europe	All	$37,000	$33,873	3	6	225	16	23	4023	110	180
BMW	X5 4.4i	SUV	Europe	All	$52,195	$47,720	4.4	8	325	16	22	4824	111	184
Buick	Rainier	SUV	USA	All	$37,895	$34,357	4.2	6	275	15	21	4600	113	193
Buick	Rendezvous CX	SUV	USA	Front	$26,545	$24,085	3.4	6	185	19	26	4024	112	187
Cadillac	Escalade	SUV	USA	Front	$52,795	$48,377	5.3	8	295	14	18	5367	116	199
Cadillac	SRX V8	SUV	USA	Front	$46,995	$43,523	4.6	8	320	16	21	4302	116	195
Chevrolet	Suburban 1500 LT	SUV	USA	Front	$42,735	$37,422	5.3	8	295	14	18	4947	130	219
Chevrolet	Tahoe LT	SUV	USA	All	$41,465	$36,287	5.3	8	295	14	18	5050	116	197
Chevrolet	TrailBlazer LT	SUV	USA	Front	$30,295	$27,479	4.2	6	275	16	21	4425	113	192
Chevrolet	Tracker	SUV	USA	Front	$20,255	$19,108	2.5	6	165	19	22	2866	98	163
Dodge	Durango SLT	SUV	USA	All	$32,235	$29,472	4.7	8	230	15	21	4987	119	201
Ford	Excursion 6.8 XLT	SUV	USA	All	$41,475	$36,494	6.8	10	310	10	13	7190	137	227
Ford	Expedition 4.6 XLT	SUV	USA	Front	$34,560	$30,468	4.6	8	232	15	19	5000	119	206
Ford	Explorer XLT V6	SUV	USA	All	$29,670	$26,983	4	6	210	15	20	4463	114	190
Ford	Escape XLS	SUV	USA	All	$22,515	$20,907	3	6	201	18	23	3346	103	173
GMC	Envoy XUV SLE	SUV	USA	Front	$31,890	$28,922	4.2	6	275	15	19	4945	129	208
GMC	Yukon 1500 SLE	SUV	USA	Front	$35,725	$31,361	4.8	8	285	16	19	5042	116	199
GMC	Yukon XL 2500 SLT	SUV	USA	All	$46,265	$40,534	6	8	325	13	17	6133	130	219
Honda	Pilot LX	SUV	Asia	All	$27,560	$24,843	3.5	6	240	17	22	4387	106	188
Honda	CR-V LX	SUV	Asia	All	$19,860	$18,419	2.4	4	160	21	25	3258	103	179
Honda	Element LX	SUV	Asia	All	$18,690	$17,334	2.4	4	160	21	24	3468	101	167
Hummer	H2	SUV	USA	All	$49,995	$45,815	6	8	316	10	12	6400	123	190
Hyundai	Santa Fe GLS	SUV	Asia	Front	$21,589	$20,201	2.7	6	173	20	26	3549	103	177

Tip #19 – Available "Custom" SAS Styles

Although many default "custom" styles are available, users have many more style choices to choose from – 57 style choices to be exact. To view the names of the available template styles, simply submit a LIST Styles statement in PROC TEMPLATE, as shown.

```
PROC TEMPLATE ;
   LIST STYLES ;
RUN ;
```

Output and SAS Styles

Listing of: SASHELP.TMPLMST
Path Filter is: Styles
Sort by: PATH/ASCENDING

Obs	Path	Type
1	Styles	Dir
2	Styles.Analysis	Style
3	Styles.BarrettsBlue	Style
4	Styles.BlockPrint	Style
5	Styles.DTree	Style
6	Styles.Daisy	Style
7	Styles.Default	Style
8	Styles.Dove	Style
10	Styles.Excel	Style
12	Styles.Festival	Style
13	Styles.FestivalPrinter	Style
14	Styles.Gantt	Style
15	Styles.GrayscalePrinter	Style
16	Styles.HTMLBlue	Style
17	Styles.Harvest	Style
18	Styles.HighContrast	Style
19	Styles.HighContrastLarge	Style
20	Styles.Journal	Style

21	Styles.Journal1a	Style
22	Styles.Journal2	Style
23	Styles.Journal2a	Style
24	Styles.Journal3	Style
25	Styles.Journal3a	Style
26	Styles.Listing	Style
27	Styles.Meadow	Style
28	Styles.MeadowPrinter	Style
29	Styles.Minimal	Style
30	Styles.MonochromePrinter	Style
31	Styles.Monospace	Style
32	Styles.Moonflower	Style
33	Styles.Netdraw	Style
34	Styles.NoFontDefault	Style
35	Styles.Normal	Style
36	Styles.NormalPrinter	Style
37	Styles.Ocean	Style
38	Styles.Pearl	Style
39	Styles.PearlJ	Style
40	Styles.Plateau	Style

41	Styles.PowerPointDark	Style
42	Styles.PowerPointLight	Style
43	Styles.Printer	Style
44	Styles.Raven	Style
45	Styles.Rtf	Style
46	Styles.Sapphire	Style
47	Styles.SasDocPrinter	Style
48	Styles.SasWeb	Style
49	Styles.Seaside	Style
50	Styles.SeasidePrinter	Style
51	Styles.Snow	Style
52	Styles.StatDoc	Style
53	Styles.Statistical	Style
54	Styles.Word	Style
55	Styles.vaDark	Style
56	Styles.vaHighContrast	Style
57	Styles.vaLight	Style

Tip #20 – Using the "Custom" BarrettsBlue SAS Style

This example specifies an ODS HTML statement and a STYLE=BarrettsBlue option with PROC PRINT to send the subset of data from the SASHELP.CARS data set to Excel.

```
ODS HTML FILE='PRINT-to-Excel.XLS'
         STYLE=BarrettsBlue ;
  proc print data=sashelp.Cars(where=(type="SUV")) noobs ;
    title ;
  run ;
ODS HTML CLOSE ;
```

Results

	Make	Model	Type	Origin	DriveTrai	MSRP	Invoice	EngineSize	Cylinders	Horsepower	MPG_City	MPG_Highway	Weight	Wheelbas	Length
1	Make	Model	Type	Origin	DriveTrai	MSRP	Invoice	EngineSize	Cylinders	Horsepower	MPG_City	MPG_Highway	Weight	Wheelbas	Length
2	Acura	MDX	SUV	Asia	All	$36,945	$33,337	3.5	6	265	17	23	4451	106	189
3	BMW	X3 3.0i	SUV	Europe	All	$37,000	$33,873	3	6	225	16	23	4023	110	180
4	BMW	X5 4.4i	SUV	Europe	All	$52,195	$47,720	4.4	8	325	16	22	4824	111	184
5	Buick	Rainier	SUV	USA	All	$37,895	$34,357	4.2	6	275	15	21	4600	113	193
6	Buick	Rendezvous CX	SUV	USA	Front	$26,545	$24,085	3.4	6	185	19	26	4024	112	187
7	Cadillac	Escalade	SUV	USA	Front	$52,795	$48,377	5.3	8	295	14	18	5367	116	199
8	Cadillac	SRX V8	SUV	USA	Front	$46,995	$43,523	4.6	8	320	16	21	4302	116	195
9	Chevrolet	Suburban 1500 LT	SUV	USA	Front	$42,735	$37,422	5.3	8	295	14	18	4947	130	219
10	Chevrolet	Tahoe LT	SUV	USA	All	$41,465	$36,287	5.3	8	295	14	18	5050	116	197
11	Chevrolet	TrailBlazer LT	SUV	USA	Front	$30,295	$27,479	4.2	6	275	16	21	4425	113	192
12	Chevrolet	Tracker	SUV	USA	Front	$20,255	$19,108	2.5	6	165	19	22	2866	98	163
13	Dodge	Durango SLT	SUV	USA	All	$32,235	$29,472	4.7	8	230	15	21	4987	119	201
14	Ford	Excursion 6.8 XLT	SUV	USA	All	$41,475	$36,494	6.8	10	310	10	13	7190	137	227
15	Ford	Expedition 4.6 XLT	SUV	USA	Front	$34,560	$30,468	4.6	8	232	15	19	5000	119	206
16	Ford	Explorer XLT V6	SUV	USA	All	$29,670	$26,983	4	6	210	15	20	4463	114	190
17	Ford	Escape XLS	SUV	USA	All	$22,515	$20,907	3	6	201	18	23	3346	103	173
18	GMC	Envoy XUV SLE	SUV	USA	Front	$31,890	$28,922	4.2	6	275	15	19	4945	129	208
19	GMC	Yukon 1500 SLE	SUV	USA	Front	$35,725	$31,361	4.8	8	285	16	19	5042	116	199
20	GMC	Yukon XL 2500 SLT	SUV	USA	All	$46,265	$40,534	6	8	325	13	17	6133	130	219
21	Honda	Pilot LX	SUV	Asia	All	$27,560	$24,843	3.5	6	240	17	22	4387	106	188
22	Honda	CR-V LX	SUV	Asia	All	$19,860	$18,419	2.4	4	160	21	25	3258	103	179
23	Honda	Element LX	SUV	Asia	All	$18,690	$17,334	2.4	4	160	21	24	3468	101	167
24	Hummer	H2	SUV	USA	All	$49,995	$45,815	6	8	316	10	12	6400	123	190
25	Hyundai	Santa Fe GLS	SUV	Asia	Front	$21,589	$20,201	2.7	6	173	20	26	3549	103	177
26	Isuzu	Ascender S	SUV	Asia	All	$31,849	$29,977	4.2	6	275	15	20	4967	129	208
27	Isuzu	Rodeo S	SUV	Asia	Front	$20,449	$19,261	3.2	6	193	17	21	3836	106	178
28	Jeep	Grand Cherokee Laredo	SUV	USA	Front	$27,905	$25,686	4	6	195	16	21	3790	106	181
29	Jeep	Liberty Sport	SUV	USA	All	$20,130	$18,973	2.4	4	150	20	24	3826	104	174
30	Jeep	Wrangler Sahara convertible 2dr	SUV	USA	All	$25,520	$23,275	4	6	190	16	19	3575	93	150
31	Kia	Sorento LX	SUV	Asia	Front	$19,635	$18,630	3.5	6	192	16	19	4112	107	180
32	Land Rover	Range Rover HSE	SUV	Europe	All	$72,250	$65,807	4.4	8	282	12	16	5379	113	195

Tip #21 – Using the "Custom" Excel SAS Style

This example specifies an ODS HTML statement and a STYLE=Excel option with PROC PRINT to send the contents of the SASHELP.CARS data set to Excel.

```
ODS HTML FILE='PRINT-to-Excel.XLS'
        STYLE=Excel ;
  proc print data=sashelp.Cars(where=(type="SUV")) noobs ;
    title ;
  run ;
ODS HTML CLOSE ;
```

Results

	Make	Model	Type	Origin	DriveTrain	MSRP	Invoice	EngineSize	Cylinders	Horsepower	MPG_City	MPG_Highway	Weight	Wheelbas	Length
1	Make	Model	Type	Origin	DriveTrain	MSRP	Invoice	EngineSize	Cylinders	Horsepower	MPG_City	MPG_Highway	Weight	Wheelbas	Length
2	Acura	MDX	SUV	Asia	All	######	######	3.5	6	265	17	23	4451	106	189
3	BMW	X3 3.0i	SUV	Europ	All	######	######	3	6	225	16	23	4023	110	180
4	BMW	X5 4.4i	SUV	Europ	All	######	######	4.4	8	325	16	22	4824	111	184
5	Buick	Rainier	SUV	USA	All	######	######	4.2	6	275	15	21	4600	113	193
6	Buick	Rendezvous CX	SUV	USA	Front	######	######	3.4	6	185	19	26	4024	112	187
7	Cadillac	Escalade	SUV	USA	Front	######	######	5.3	8	295	14	18	5367	116	199
8	Cadillac	SRX V8	SUV	USA	Front	######	######	4.6	8	320	16	21	4302	116	195
9	Chevrolet	Suburban 1500 LT	SUV	USA	Front	######	######	5.3	8	295	14	18	4947	130	219
10	Chevrolet	Tahoe LT	SUV	USA	All	######	######	5.3	8	295	14	18	5050	116	197
11	Chevrolet	TrailBlazer LT	SUV	USA	Front	######	######	4.2	6	275	16	21	4425	113	192
12	Chevrolet	Tracker	SUV	USA	Front	######	######	2.5	6	165	19	22	2866	98	163
13	Dodge	Durango SLT	SUV	USA	All	######	######	4.7	8	230	15	21	4987	119	201
14	Ford	Excursion 6.8 XLT	SUV	USA	All	######	######	6.8	10	310	10	13	7190	137	227
15	Ford	Expedition 4.6 XLT	SUV	USA	Front	######	######	4.6	8	232	15	19	5000	119	206
16	Ford	Explorer XLT V6	SUV	USA	All	######	######	4	6	210	15	20	4463	114	190
17	Ford	Escape XLS	SUV	USA	All	######	######	3	6	201	18	23	3346	103	173
18	GMC	Envoy XUV SLE	SUV	USA	Front	######	######	4.2	6	275	15	19	4945	129	208
19	GMC	Yukon 1500 SLE	SUV	USA	Front	######	######	4.8	8	285	16	19	5042	116	199
20	GMC	Yukon XL 2500 SLT	SUV	USA	All	######	######	6	8	325	13	17	6133	130	219
21	Honda	Pilot LX	SUV	Asia	All	######	######	3.5	6	240	17	22	4387	106	188
22	Honda	CR-V LX	SUV	Asia	All	######	######	2.4	4	160	21	25	3258	103	179
23	Honda	Element LX	SUV	Asia	All	######	######	2.4	4	160	21	24	3468	101	167
24	Hummer	H2	SUV	USA	All	######	######	6	8	316	10	12	6400	123	190
25	Hyundai	Santa Fe GLS	SUV	Asia	Front	######	######	2.7	6	173	20	26	3549	103	177
26	Isuzu	Ascender S	SUV	Asia	All	######	######	4.2	6	275	15	20	4967	129	208
27	Isuzu	Rodeo S	SUV	Asia	Front	######	######	3.2	6	193	17	21	3836	106	178
28	Jeep	Grand Cherokee Laredo	SUV	USA	Front	######	######	4	6	195	16	21	3790	106	181
29	Jeep	Liberty Sport	SUV	USA	All	######	######	2.4	4	150	20	24	3826	104	174
30	Jeep	Wrangler Sahara convertible	SUV	USA	All	######	######	4	6	190	16	19	3575	93	150
31	Kia	Sorento LX	SUV	Asia	Front	######	######	3.5	6	192	16	19	4112	107	180
32	Land Rover	Range Rover HSE	SUV	Europ	All	######	######	4.4	8	282	12	16	5379	113	195
33	Land Rover	Discovery SE	SUV	Europ	All	######	######	4.6	8	217	12	16	4576	100	185
34	Land Rover	Freelander SE	SUV	Europ	All	######	######	2.5	6	174	18	21	3577	101	175

Tip #22 – Using the "Custom" HighContrast SAS Style

In this example, a data subset is sent to Excel using an ODS HTML statement along with an XLS extension and a STYLE= HighContrast option, a KEEP= data set option and a WHERE statement in a PROC PRINT.

```
ODS HTML FILE='PRINT-subset-to-Excel.XLS'
       STYLE=HighContrast ;
  title ;
  proc print data=sashelp.Cars(keep=make type origin msrp invoice) ;
    Where type="SUV" ;
  run ;
ODS HTML CLOSE ;
```

Results

	A	B	C	D	E	F
1	Obs	Make	Type	Origin	MSRP	Invoice
2	1	Acura	SUV	Asia	$36,945	$33,337
3	27	BMW	SUV	Europe	$37,000	$33,873
4	28	BMW	SUV	Europe	$52,195	$47,720
5	47	Buick	SUV	USA	$37,895	$34,357
6	48	Buick	SUV	USA	$26,545	$24,085
7	56	Cadillac	SUV	USA	$52,795	$48,377
8	57	Cadillac	SUV	USA	$46,995	$43,523
9	64	Chevrolet	SUV	USA	$42,735	$37,422
10	65	Chevrolet	SUV	USA	$41,465	$36,287
11	66	Chevrolet	SUV	USA	$30,295	$27,479
12	67	Chevrolet	SUV	USA	$20,255	$19,108
13	106	Dodge	SUV	USA	$32,235	$29,472
14	119	Ford	SUV	USA	$41,475	$36,494
15	120	Ford	SUV	USA	$34,560	$30,468
16	121	Ford	SUV	USA	$29,670	$26,983
17	122	Ford	SUV	USA	$22,515	$20,907
18	142	GMC	SUV	USA	$31,890	$28,922
19	143	GMC	SUV	USA	$35,725	$31,361
20	144	GMC	SUV	USA	$46,265	$40,534
21	152	Honda	SUV	Asia	$27,560	$24,843
22	153	Honda	SUV	Asia	$19,860	$18,419
23	154	Honda	SUV	Asia	$18,690	$17,334
24	167	Hummer	SUV	USA	$49,995	$45,815
25	168	Hyundai	SUV	Asia	$21,589	$20,201
26	188	Isuzu	SUV	Asia	$31,849	$29,977
27	189	Isuzu	SUV	Asia	$20,449	$19,261
28	202	Jeep	SUV	USA	$27,905	$25,686
29	203	Jeep	SUV	USA	$20,130	$18,973
30	204	Jeep	SUV	USA	$25,520	$23,275
31	205	Kia	SUV	Asia	$19,635	$18,630
32	216	Land Rover	SUV	Europe	$72,250	$65,807
33	217	Land Rover	SUV	Europe	$39,250	$35,777
34	218	Land Rover	SUV	Europe	$25,995	$23,969

The ODS Excel Destination

The **ODS Excel** Destination became production in SAS 9.4 (M4) and serves as an interface between SAS and Excel. It offers the ability to:

- ✓ Send Results and Output directly from SAS to Excel;
- ✓ Create Excel workbooks;
- ✓ Produce Reports, Tables, Statistics and Graphs;
- ✓ Format Data into Excel Worksheet cells;
- ✓ Automate the creation of Production-level Workbooks.

Simplifying the process of sending output, reports, tables, statistics and graphs to Excel files, users are able to use ODS Excel to:

- ✓ Programmatically generate output and results;
- ✓ Control font sizes;
- ✓ Add special features to row and column headers;
- ✓ Adjust row and column sizes;
- ✓ Format data values;
- ✓ Align data to the left, center or right;
- ✓ Add hyperlinks for drill-down capability.

Tip #23 – Sending Results to an Excel File

In this example, the PROC MEANS Statistics is sent to Excel by specifying an ODS Excel statement along with an XLSX extension.

```
ods excel file='e:/PROCMEANS.xlsx' ;
  proc means data=sashelp.cars ;
    class type ;
  run ;
ods excel close ;
```

Results

Tip #24 – Sending Tabular and Graphical Results to an Excel File

In this example, a PROC FREQ HBAR Histogram is sent to Excel by specifying an ODS Excel statement along with an XLSX extension.

```
ods Excel file="e:/Freq-HBAR-Histogram.xlsx" ;
  PROC FREQ DATA=sashelp.cars
           ORDER=data ;
ods exclude twowayfreqs ;
    TABLES Type * Origin /
               PLOTS=freqplot(orient=horizontal) ;
  RUN ;
ods Excel close ;
```
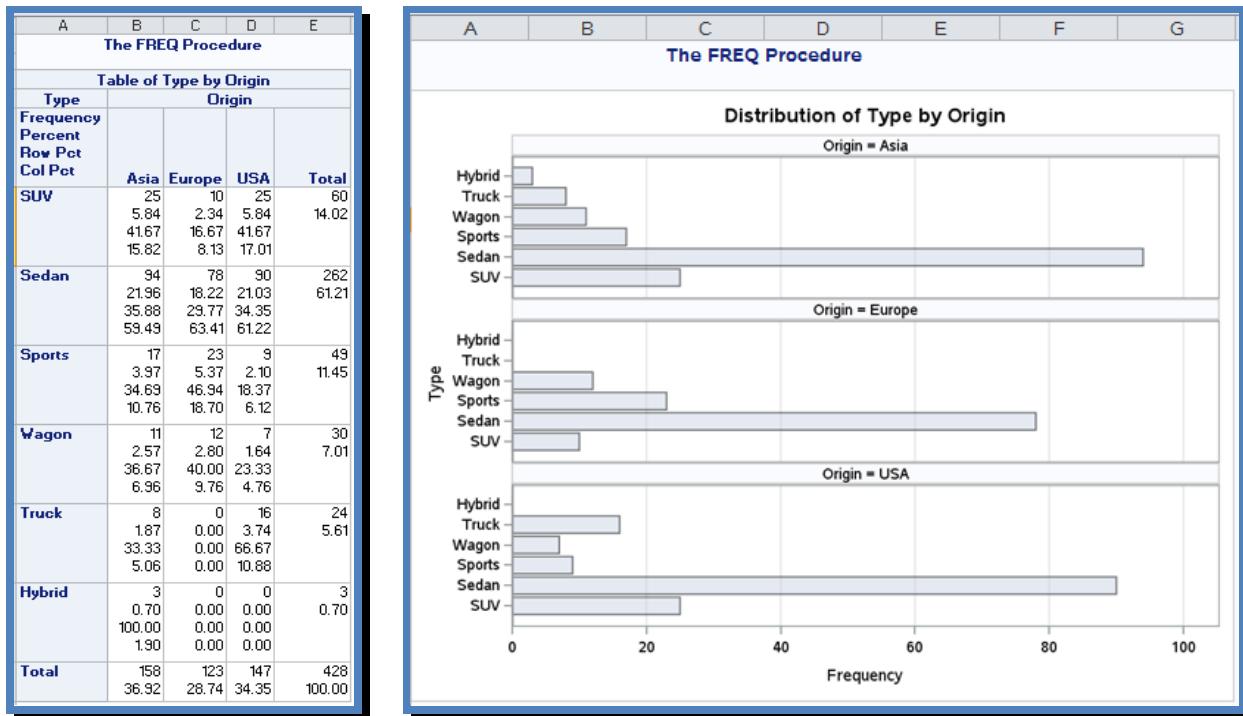
Results

Tip #25 – Sending Report Results to an Excel File with ODS Excel

In this example, PROC REPORT sends output to Excel by specifying an ODS Excel statement.

```
ods Excel file="PROC-REPORT.xlsx" ;
  PROC REPORT DATA=SASHELP.CARS
             (KEEP=Type Make Model Origin MSRP)  NOWINDOWS ;
    WHERE MSRP < 30000 AND Type = 'Sports' ;
    COLUMNS Type Origin Make Model MSRP ;
    DEFINE Type    / ORDER  WIDTH=8 ;
    DEFINE Origin / ORDER  WIDTH=6 ;
    DEFINE Make    / ORDER  WIDTH=13 CENTER ;
    DEFINE Model  / DISPLAY WIDTH=40 ;
    DEFINE MSRP    / DISPLAY 'MSRP' WIDTH=8 ;
  RUN ;
ods Excel close ;
```

Results

A	B	C	D	E
Type	**Origin**	**Make**	**Model**	**MSRP**
Sports	Asia	Hyundai	Tiburon GT V6 2dr	$18,739
		Mazda	MX-5 Miata convertible 2dr	$22,388
			MX-5 Miata LS convertible 2dr	$25,193
			RX-8 4dr automatic	$25,700
			RX-8 4dr manual	$27,200
		Mitsubishi	Eclipse GTS 2dr	$25,092
			Eclipse Spyder GT convertible 2dr	$26,992
			Lancer Evolution 4dr	$29,562
		Nissan	350Z coupe 2dr	$26,910
		Subaru	Impreza WRX 4dr	$25,045
		Toyota	Celica GT-S 2dr	$22,570
			MR2 Spyder convertible 2dr	$25,130
	USA	Ford	Mustang 2dr (convertible)	$18,345
			Mustang GT Premium convertible 2dr	$29,380

Tip #26 – Sending Statistical and Graphical Results to an Excel File

In this example, PROC UNIVARIATE sends output to Excel by specifying an ODS Excel statement.

```
OPTIONS ORIENTATION=PORTRAIT ;
ods Excel file="/folders/myfolders/Univariate-VBAR-Histogram.xlsx" ;
  TITLE "Histogram for Chart Variable" ;
  PROC UNIVARIATE DATA=sashelp.cars
                 (KEEP=type origin make cylinders MSRP)
                  NOPRINT ;
    VAR cylinders ;
    HISTOGRAM cylinders / NORMAL;
  RUN ;
ods Excel close ;
```

Results

A	B	C
The UNIVARIATE Procedure		
Fitted Normal Distribution for Cylinders		
Quantiles for Normal Distribution		
	Quantile	
Percent	**Observed**	**Estimated**
1.0	4.00000	2.18203
5.0	4.00000	3.24410
10.0	4.00000	3.81029
25.0	4.00000	4.75636
50.0	6.00000	5.80751
75.0	6.00000	6.85867
90.0	8.00000	7.80474
95.0	8.00000	8.37092
99.0	10.00000	9.43299
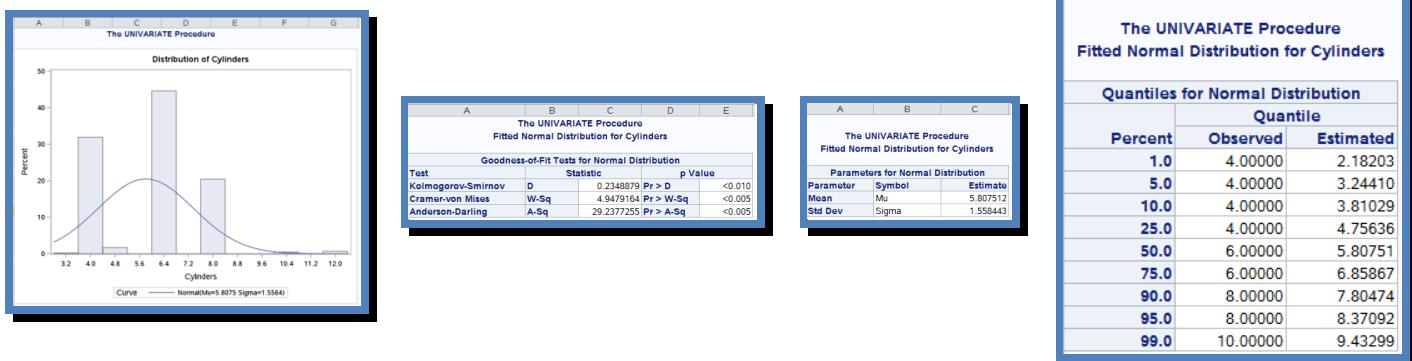
Tip #27 – Creating an Excel File with the %DS2CSV Macro

Users are able to create an Excel file using the %ds2csv macro. To create an Excel file, specify the CSV extension, runmode=b (batch), and the filename in the csvfile= in the macro. A list of available parameters used with the %DS2CSV macro is illustrated in the following table.

Parameter	Description
DATA=	Name of SAS data set to export.
CSVFILE=	CSV file to be created.
RUNMODE=	Specify "B" for batch or interactive SAS programs.
OPENMODE=	Specify "Replace" or "Append".
COLHEAD=	Specify "Y" or "N" to add column headers in CSV file.
FORMAT=	Specify "Y" or "N" to keep formats.
LABELS=	Specify "Y" or "N" to keep labels on column headings.
WHERE=	Specify expression to subset data in resulting CSV file.

In this example, the %DS2CSV macro reads the SASHELP.CARS data set and creates a CSV file called, DS2CSV.

```
%ds2csv (data=sashelp.cars
        ,runmode=b
        ,csvfile=c:\mwsug\lafler\DS2CSV.csv);
```

Log:

NOTE: CSV file successfully generated for SASHELP.CARS.

Results

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	Make	Model	Type	Origin	DriveTrai	MSRP	Invoice	Engine Siz	Cylinders	Horsepov	MPG (City	MPG (High	Weight (L	Wheelbas	Length (IN
2	Acura	MDX	SUV	Asia	All	$36,945	$33,337	3.5	6	265	17	23	4451	106	189
3	Acura	RSX Type S	Sedan	Asia	Front	$23,820	$21,761	2	4	200	24	31	2778	101	172
4	Acura	TSX 4dr	Sedan	Asia	Front	$26,990	$24,647	2.4	4	200	22	29	3230	105	183
5	Acura	TL 4dr	Sedan	Asia	Front	$33,195	$30,299	3.2	6	270	20	28	3575	108	186
6	Acura	3.5 RL 4dr	Sedan	Asia	Front	$43,755	$39,014	3.5	6	225	18	24	3880	115	197
7	Acura	3.5 RL w/N	Sedan	Asia	Front	$46,100	$41,100	3.5	6	225	18	24	3893	115	197
8	Acura	NSX coupe	Sports	Asia	Rear	$89,765	$79,978	3.2	6	290	17	24	3153	100	174
9	Audi	A4 1.8T 4c	Sedan	Europe	Front	$25,940	$23,508	1.8	4	170	22	31	3252	104	179
10	Audi	A41.8T co	Sedan	Europe	Front	$35,940	$32,506	1.8	4	170	23	30	3638	105	180
11	Audi	A4 3.0 4dr	Sedan	Europe	Front	$31,840	$28,846	3	6	220	20	28	3462	104	179
12	Audi	A4 3.0 Qu	Sedan	Europe	All	$33,430	$30,366	3	6	220	17	26	3583	104	179
13	Audi	A4 3.0 Qu	Sedan	Europe	All	$34,480	$31,388	3	6	220	18	25	3627	104	179
14	Audi	A6 3.0 4dr	Sedan	Europe	Front	$36,640	$33,129	3	6	220	20	27	3561	109	192
15	Audi	A6 3.0 Qu	Sedan	Europe	All	$39,640	$35,992	3	6	220	18	25	3880	109	192
16	Audi	A4 3.0 con	Sedan	Europe	Front	$42,490	$38,325	3	6	220	20	27	3814	105	180
17	Audi	A4 3.0 Qu	Sedan	Europe	All	$44,240	$40,075	3	6	220	18	25	4013	105	180
18	Audi	A6 2.7 Tur	Sedan	Europe	All	$42,840	$38,840	2.7	6	250	18	25	3836	109	192
19	Audi	A6 4.2 Qu	Sedan	Europe	All	$49,690	$44,936	4.2	8	300	17	24	4024	109	193
20	Audi	A8 L Quat	Sedan	Europe	All	$69,190	$64,740	4.2	8	330	17	24	4399	121	204
21	Audi	S4 Quattr	Sedan	Europe	All	$48,040	$43,556	4.2	8	340	14	20	3825	104	179
22	Audi	RS 6 4dr	Sports	Europe	Front	$84,600	$76,417	4.2	8	450	15	22	4024	109	191
23	Audi	TT 1.8 con	Sports	Europe	Front	$35,940	$32,512	1.8	4	180	20	28	3131	95	159
24	Audi	TT 1.8 Qua	Sports	Europe	All	$37,390	$33,891	1.8	4	225	20	28	2921	96	159
25	Audi	TT 3.2 cou	Sports	Europe	All	$40,590	$36,739	3.2	6	250	21	29	3351	96	159
26	Audi	A6 3.0 Ava	Wagon	Europe	All	$40,840	$37,060	3	6	220	18	25	4035	109	192
27	Audi	S4 Avant C	Wagon	Europe	All	$49,090	$44,446	4.2	8	340	15	21	3936	104	179
28	BMW	X3 3.0i	SUV	Europe	All	$37,000	$33,873	3	6	225	16	23	4023	110	180
29	BMW	X5 4.4i	SUV	Europe	All	$52,195	$47,720	4.4	8	325	16	22	4824	111	184
30	BMW	325i 4dr	Sedan	Europe	Rear	$28,495	$26,155	2.5	6	184	20	29	3219	107	176
31	BMW	325Ci 2dr	Sedan	Europe	Rear	$30,795	$28,245	2.5	6	184	20	29	3197	107	177
32	BMW	325Ci con	Sedan	Europe	Rear	$37,995	$34,800	2.5	6	184	19	27	3560	107	177
33	BMW	325xi 4dr	Sedan	Europe	All	$30,245	$27,745	2.5	6	184	19	27	3461	107	176

Tip #28 – Sending Statistical Results with a Background Image to an Excel File

In this example, PROC UNIVARIATE sends output to Excel by specifying an ODS Excel statement.

```
proc template ;
  define style styles.background ;
    parent=styles.excel ;
    class body/background=_undef_
           backgroundimage="e:/Analytics.jpg" ;
  end ;
run ;
ods excel file="ODS-Excel-MEANS-Background.xlsx"
        options(embedded_titles="yes"
        start_at="5,5")
        style=styles.background ;
  proc means data=sashelp.cars ;
    class type ;
  run ;
ods excel close ;
```

Results
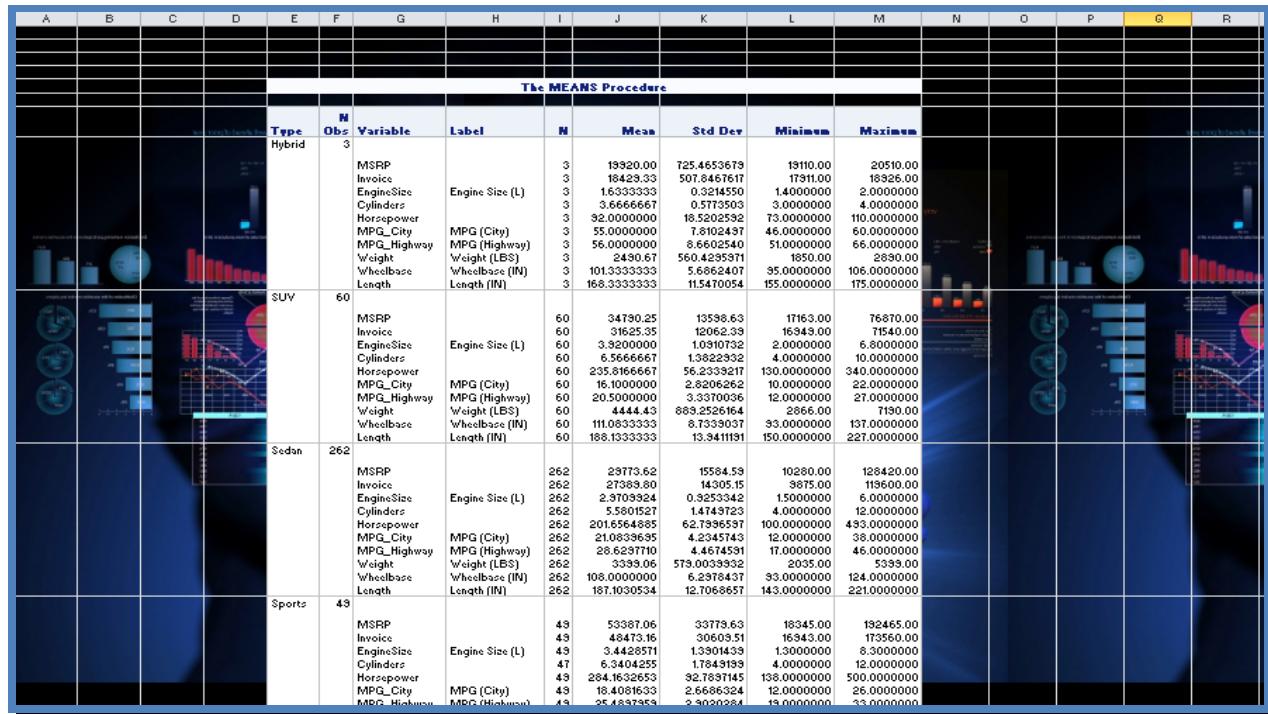
Output Delivery Goes Web

The Web offers incredible potential that impacts all corners of society. With its increasing popularity as a communications medium, Web publishers have arguably established the Web as the greatest medium ever created. Businesses, government agencies, professional associations, schools, libraries, research agencies, and a potpourri of society's true believers have endorsed the Web as an efficient means of conveying their messages to the world.

The Web is not a static environment, but a dynamic medium capable of distributing content anywhere and at any time. The 24/7 model permits information to be refreshed and updated continuously as new material becomes available. A primary objective of Web publishers everywhere is to engage visitors with timely and interesting content that brings them back for more.

Tip #29 – Pagesize / Linesize Settings

The Options PS= and LS= have no effect when used with the HTML destination (opposed to most other output-producing steps that generate output to a print destination). If the PS= and/or LS= options are used with the HTML destination, they are simply ignored. The SAS System creates a type of "streaming" or continuous output and adds elevator bars (horizontal and/or vertical) for easy navigation.

The SAS System does provide a way for users to paginate through output displayed in a body file. The HTML destination provides a way to designate an optional description of each page of the body file. The PAGE= file (when specified) recognizes each new page of output produced by ODS. What ODS does is create a section called *Table of Pages* containing links to the body file for easy navigation through output.

Tip #30 – Deploying Output to the Web

With the popularity of the Internet, you may find it useful to deploy selected pieces of output on your web site. ODS makes deploying procedure output to the web a simple process. Syntactically correct HTML code is automatically produced and made ready for deployment using one of the Internet browser software products (e.g., Internet Explorer, Netscape Navigator, etc.). As a result, the SAS System and the HTML destination create a type of "streaming" or continuous output by adding elevator bars (horizontal and/or vertical) for easy navigation.

Tip #31 – Creating HTML Destination Files with ODS

Four types of files can be created with the ODS HTML destination: 1) body, 2) contents, 3) page, and 4) frame. Each file is described below.

The **Body** file contains the results from the procedure embedded in ODS-generated HTML code. Horizontal and vertical scroll bars are automatically placed on the generated page, if necessary.

The **Contents** file consists of a link to each HTML table within the body file. It uses an anchor tag to link to each table. By using your browser software, you can view the contents file directly or as part of the frame file.

The **Page** file consists of a link to each page of ODS created output. By using your browser, you can view the page file directly or as part of the frame file.

The **Frame** file displays the body file and the contents file, the page file, or both. The next example shows the creation of Web-ready Univariate procedure output using the HTML format engine with the body=, contents=, page=, and frame= options.

```
ODS HTML body='ods-body.htm'
      contents='ods-contents.htm'
        page='ods-page.htm'
       frame='ods-frame.htm'
        path="e:/"
        (url=none) ;
  proc univariate data=sashelp.Cars ;
    Title1 'Creating HTML Output with ODS' ;
  Run ;
ODS HTML Close ;
```

A snippet of the HTML output appears on the next page:

Table of Contents

Creating HTML Output with ODS

The UNIVARIATE Procedure
Variable: MSRP

Moments			
N	428	Sum Weights	428
Mean	32774.8551	Sum Observations	14027638
Std Deviation	19431.7167	Variance	377591613
Skewness	2.79809927	Kurtosis	13.8792055
Uncorrected SS	6.20985E11	Corrected SS	1.61232E11
Coeff Variation	59.2884899	Std Error Mean	939.267478

Basic Statistical Measures			
Location		Variability	
Mean	32774.86	Std Deviation	19432
Median	27635.00	Variance	377591613
Mode	13270.00	Range	182185
		Interquartile Range	18886

Note: The mode displayed is the smallest of 18 modes with a count of 2.

Tests for Location: Mu0=0				
Test		Statistic	p Value	
Student's t	t	34.89406	Pr > |t|	<.0001
Sign	M	214	Pr >= |M|	<.0001
Signed Rank	S	45903	Pr >= |S|	<.0001

Quantiles (Definition 5)	
Level	Quantile
100% Max	192465.0
99%	94820.0
95%	73195.0
90%	52795.0
75% Q3	39215.0
50% Median	27635.0

Tip #32 – Creating PDF Output

To share output electronically, SAS provides users with the Adobe proprietary portable document facility called PDF. The objective of PDF is to enable the printing of output exactly as it is seen. The significance of PDF output is that it is a great format for Web deployment since it is completely independent of any printer destination. To create PDF output from the UNIVARIATE procedure, the ODS PDF option can be specified as follows.

```
ODS PDF FILE='ods-univariate.pdf' ;
  proc univariate data=sashelp.Cars ;
    title1 'Creating PDF Output with ODS' ;
  run ;
ODS PDF Close ;
```

Tip #33 – Combining Output Results

With the streaming capabilities of HTML output, results can be combined so they appear on the same screen (or page). Rather than having output controlled by one or more page breaks, HTML automatically displays output without page boundaries. The following example code illustrates combined output from the PRINT and MEANS procedures.

```
ods html body='ods-body-combined.html'
     contents='ods-contents-combined.html'
       page='ods-page-combined.html'
      frame='ods-frame-combined.html'
      path='c:\sas app'
      (url=none) ;
  proc print data=sashelp.Cars noobs n ;
    title1 'Classic Cars Listing' ;
    where type in ('SUV') ;
  run;
  proc means data=sashelp.Cars ;
    title1 'Summary of Classic Cars' ;
    class type ;
  run ;
ods html close ;
```

Output from Combining Output

Table of Contents	Classic Cars Listing													
	Make	Model	Type	Origin	DriveTrain	MSRP	Invoice	EngineSize	Cylinders	Horsepower	MPG_City	MPG_Highway	Weight	Wheelbase	Length
	Acura	MDX	SUV	Asia	All	$36,945	$33,337	3.5	6	265	17	23	4451	106	189
	BMW	X3 3.0i	SUV	Europe	All	$37,000	$33,873	3.0	6	225	16	23	4023	110	180
	BMW	X5 4.4i	SUV	Europe	All	$52,195	$47,720	4.4	8	325	16	22	4824	111	184
	Buick	Rainier	SUV	USA	All	$37,895	$34,357	4.2	6	275	15	21	4600	113	193
	Buick	Rendezvous CX	SUV	USA	Front	$26,545	$24,085	3.4	6	185	19	26	4024	112	187
	Cadillac	Escalade	SUV	USA	Front	$52,795	$48,377	5.3	8	295	14	18	5367	116	199
	Cadillac	SRX V8	SUV	USA	Front	$46,995	$43,523	4.6	8	320	16	21	4302	116	195
	Chevrolet	Suburban 1500 LT	SUV	USA	Front	$42,735	$37,422	5.3	8	295	14	18	4947	130	219
	Chevrolet	Tahoe LT	SUV	USA	All	$41,465	$36,287	5.3	8	295	14	18	5050	116	197
	Chevrolet	TrailBlazer LT	SUV	USA	Front	$30,295	$27,479	4.2	6	275	16	21	4425	113	192
	Chevrolet	Tracker	SUV	USA	Front	$20,255	$19,108	2.5	6	165	19	22	2866	98	163
	Dodge	Durango SLT	SUV	USA	All	$32,235	$29,472	4.7	8	230	15	21	4987	119	201
	Ford	Excursion 6.8 XLT	SUV	USA	All	$41,475	$36,494	6.8	10	310	10	13	7190	137	227
	Ford	Expedition 4.6 XLT	SUV	USA	Front	$34,560	$30,468	4.6	8	232	15	19	5000	119	206
	Ford	Explorer XLT V6	SUV	USA	All	$29,670	$26,983	4.0	6	210	15	20	4463	114	190
	Ford	Escape XLS	SUV	USA	All	$22,515	$20,907	3.0	6	201	18	23	3346	103	173
	GMC	Envoy XUV SLE	SUV	USA	Front	$31,890	$28,922	4.2	6	275	15	19	4945	129	208
	GMC	Yukon 1500 SLE	SUV	USA	Front	$35,725	$31,361	4.8	8	285	16	19	5042	116	199
	GMC	Yukon XL 2500 SLT	SUV	USA	All	$46,265	$40,534	6.0	8	325	13	17	6133	130	219
	Honda	Pilot LX	SUV	Asia	All	$27,560	$24,843	3.5	6	240	17	22	4387	106	188
	Honda	CR-V LX	SUV	Asia	All	$19,860	$18,419	2.4	4	160	21	25	3258	103	179
	Honda	Element LX	SUV	Asia	All	$18,690	$17,334	2.4	4	160	21	24	3468	101	167
	Hummer	H2	SUV	USA	All	$49,995	$45,815	6.0	8	316	10	12	6400	123	190
	Hyundai	Santa Fe GLS	SUV	Asia	Front	$21,589	$20,201	2.7	6	173	20	26	3549	103	177
	Isuzu	Ascender S	SUV	Asia	All	$31,849	$29,977	4.2	6	275	15	20	4967	129	208
	Isuzu	Rodeo S	SUV	Asia	Front	$20,449	$19,261	3.2	6	193	17	21	3836	106	178
	Jeep	Grand Cherokee Laredo	SUV	USA	Front	$27,905	$25,686	4.0	6	195	16	21	3790	106	181
	Jeep	Liberty Sport	SUV	USA	All	$20,130	$18,973	2.4	4	150	20	24	3826	104	174
	Jeep	Wrangler Sahara convertible 2dr	SUV	USA	All	$25,520	$23,275	4.0	6	190	16	19	3575	93	150
	Kia	Sorento LX	SUV	Asia	Front	$19,635	$18,630	3.5	6	192	16	19	4112	107	180
	Land Rover	Range Rover HSE	SUV	Europe	All	$72,250	$65,807	4.4	8	282	12	16	5379	113	195
	Land Rover	Discovery SE	SUV	Europe	All	$39,250	$35,777	4.6	8	217	12	16	4576	100	185

Building Interactive Drill-Down Applications for SAS and Microsoft Users

As a general rule, the best type of user interface design for transaction-based applications is a drill-down user interface opposed to a character-based one. It's referred to as drill-down because a user drills down through the data, layer by layer, until the desired information is found.

The key to building successful drill-down applications requires systems analysts and system designers to understand what users are trying to achieve with the data. These individuals must recognize the tasks users engage in while trying to access the desired information. These tasks are then translated into a series of selection criteria that users should be able to select from.

Tip #34 – Building a Drill-down Application

To simplify the process of building a graphical drill-down application in the SAS System, six easy steps are presented.

1. Create a data set containing the location of the HTML link variable.

2. Create HTML path with BODY and optional files.

3. Create graph using HTML= option and link variable with PROC TEMPLATE.

4. Produce 2-panel dashboard with PROC SGRENDER.

5. Create detail list drill-down results for Pie Chart and Bar Chart.

6. Use Web browser to navigate through resulting application.

Example Drill-down Application

In the following 2-panel drill-down application, a pie chart is used to display summary information about SUV vehicles in the left panel, and a vertical bar chart is used to display information about the total Price of SUV vehicles by origin (e.g., Asia, Europe and USA). To display detailed information on a movie category, a user would only need to click on the desired piece of the pie chart or bar chart that they had interest in, as shown in the illustration below. Control would then be passed, via hyperlinks, to the underlying detail output that is created.
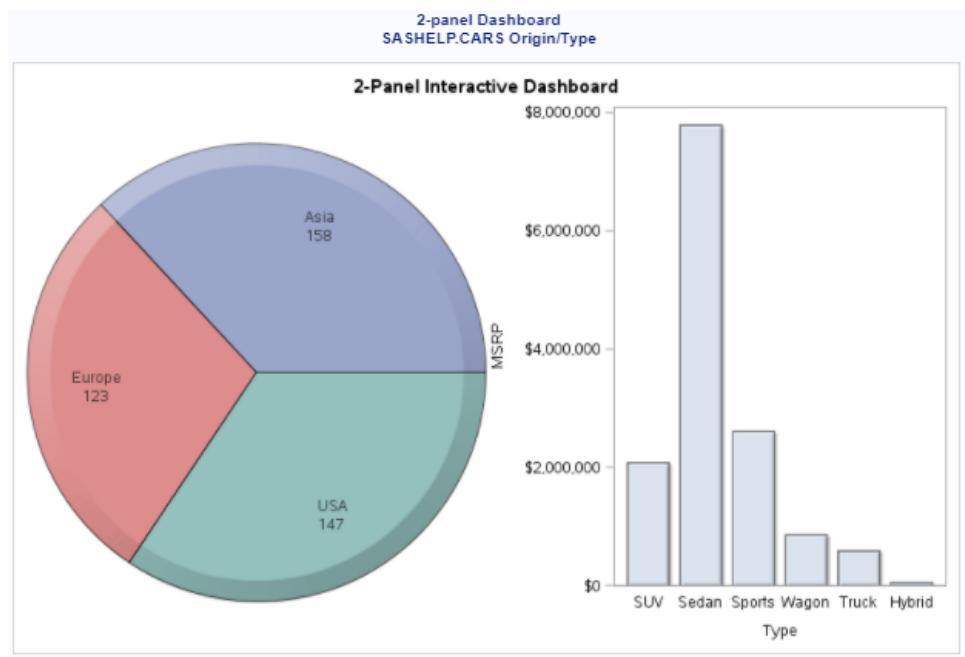
Example Code

The SAS coding steps used in building the 2-panel drill-down application under the Windows® platform is presented below.

```
*****************************************************************************;
***** Program Name: 2-panel Dashboard.SAS                              ***;
***** Purpose.....: Create and display a 2-panel interactive dashboard with two   ***;
*****              categorical variables using a DATA step, PROC TEMPLATE, Pie,   ***;
*****              Bar Chart, and PROC SGRENDER with the ODS HTML destination.    ***;
*****                                                                  ***;
***** Author......: Kirk Paul Lafler, Joshua Horstman and Roger Muller    ***;
***** Date Written: 03/28/2016                                         ***;
***** SAS Version.: SAS 9.4                                            ***;
***** Input Files.: SASHELP.CARS                                       ***;
*****                                                                  ***;
***** Output Files: HTML Output (17 Files)                             ***;
*****               2-panel-Dashboard.htm (Qty 1)                      ***;
*****               2-panel-Drilldown-PieChart.htm (Qty 3)             ***;
*****               2-panel-Drilldown-BarChart.htm (Qty 6)             ***;
***** Subroutines.: None                                               ***;
***** User-defined Formats: None                                       ***;
***** Macro Variables: &DSN and &MYRESULTS                             ***;
***** Includes....: None                                               ***;
***** Modification History:                                           ***;
*****  06/20/2017  KPL  Added Macro logic.                             ***;
*****  03/28/2016  KPL  Added Header information.                      ***;
*****************************************************************************;
%let dsn      = sashelp.Cars ;
%let myresults = /folders/myfolders/ ;
libname myresult "&myresults" ;
libname mydata '/folders/myfolders/';

*STEP 1a - Create User-defined Origin Format ;
PROC FORMAT ;
  VALUE $ORIGFMT
    'Asia'    = '2-panel-Drilldown-PieChart-Asia-Vehicles.htm'
    'Europe'  = '2-panel-Drilldown-PieChart-Europe-Vehicles.htm'
    'USA'     = '2-panel-Drilldown-PieChart-USA-Vehicles.htm' ;
RUN ;

*STEP 1b - Create User-defined Type Format ;
PROC FORMAT ;
  VALUE $TYPEFMT
    'Hybrid'  = '2-panel-Drilldown-BarChart-Hybrid-Vehicles.htm'
    'SUV'     = '2-panel-Drilldown-BarChart-SUV-Vehicles.htm'
    'Sedan'   = '2-panel-Drilldown-BarChart-Sedan-Vehicles.htm'
    'Sports'  = '2-panel-Drilldown-BarChart-Sports-Vehicles.htm'
    'Truck'   = '2-panel-Drilldown-BarChart-Truck-Vehicles.htm'
    'Wagon'   = '2-panel-Drilldown-BarChart-Wagon-Vehicles.htm' ;
RUN ;

* STEP 2 - Assign HTML Link to URL (Hyperlink) Variable ;
data work.Cars_with_Hyperlinks ;
  set &dsn ;
  URLOriginLink =
      cats("2-panel-Drilldown-Piechart-",origin,"-Vehicles.htm") ;
  URLTypeLink  =
```

```
          cats("2-panel-Drilldown-BarChart-",type,"-Vehicles.htm") ;
run ;
/**********************************************************************/
/* Step 3 - Create PieChart / BarChart Template Overlay with PROC TEMPLATE */
/**********************************************************************/
proc template ;
  define statgraph TwoPanelDashboard ;
    begingraph ;
    entrytitle "2-Panel Interactive Dashboard" ;
      layout gridded / columns=1 ;
        layout lattice / columns=2 ;
          cell ;
            layout region ;
              piechart category=Origin / url=URLOriginLink
                                         dataskin=crisp
                                         datatransparency=0.3 ;
            endlayout ;
          endcell ;
          cell ;
            layout overlay / width=250px  xaxisopts=(display=(label tickvalues line))
;
              barchart x=Type y=MSRP / url=URLTypeLink
                                       dataskin=crisp
                                       datatransparency=0.3
                                       orient=vertical
                                       barwidth=0.8 ;
            endlayout ;
          endcell ;
        endlayout ;
      endlayout ;
    endgraph ;
  end ;
run ;
quit ;


/********************************************************/
/* Step 4 - Produce Graphical Output from PieChart BarChart */
/*          Template using PROC SGRENDER                */
/********************************************************/
ods html file='2-panel-Dashboard.htm'
         path='/folders/myfolders/' (url=none) ;
ods graphics / reset imagemap=on width=8in height=5in
               imagename='2-panel-Dashboard' ;
title1 "2-panel Dashboard" ;
title2 "SASHELP.CARS Origin/Type" ;
proc sgrender data=work.Cars_with_Hyperlinks
              template=TwoPanelDashboard ;
run ;
quit ;
title ;
ods html close ;
ods graphics off ;


/**********************************************************************/
/* Step 5a - Create Pie Chart Drill-down Output for Each Origin Group with */
/*           PROC PRINT                                               */
```

```
/***************************************************************************/
proc sql noprint ;
  select distinct origin into :originlist separated by '~'
    from sashelp.Cars
      order by origin ;
  select count(distinct origin) into
              :numorigin separated by ' '
    from sashelp.Cars ;
quit ;

%macro create_drilldowns_by_origin ;
   %do i = 1 %to &numorigin ;
       ODS html
           body="2-panel-Drilldown-PieChart-%SCAN(&originlist,&i,~)-Vehicles.htm"
           path="/folders/myfolders/" (url=none) ;
       PROC PRINT DATA=sashelp.Cars NOOBS N ;
         TITLE "Detail Listing of %scan(&originlist,&i,~) Vehicles" ;
         WHERE origin = "%SCAN(&originlist,&i,~)" ;
         var make model type origin MSRP Invoice ;
       RUN ;
       ODS html close ;
   %end ;
%mend create_drilldowns_by_origin ;
%create_drilldowns_by_origin ;


/****************************************************************************/
/* Step 5b - Create Bar Chart Drill-down Output for Each Type Group with    */
/*           PROC PRINT                                                      */
/****************************************************************************/
proc sql noprint ;
  select distinct type into :typelist separated by '~'
    from sashelp.Cars
      order by type ;
  select count(distinct type) into
              :numtype separated by ' '
    from sashelp.Cars ;
quit ;

%macro create_drilldowns_by_type ;
   %do i = 1 %to &numtype ;
       ODS html
           body="2-panel-Drilldown-BarChart-%SCAN(&typelist,&i,~)-Vehicles.htm"
           path="/folders/myfolders/" (url=none) ;
       PROC PRINT DATA=sashelp.Cars NOOBS N ;
         TITLE "Detail Listing of %scan(&typelist,&i,~) Vehicles" ;
         WHERE type = "%SCAN(&typelist,&i,~)" ;
         var make model type origin MSRP Invoice ;
       RUN ;
       ODS html close ;
   %end ;
%mend create_drilldowns_by_type ;
%create_drilldowns_by_type ;
```

Conclusion

The time has come to turn tired-looking output listings into great looking information. The SAS® Output Delivery System **(ODS)** provides new and exciting ways to improve the way output is generated. Gone are the days when the only available formatting choice for output was basic output listings consisting of monospace fonts printed on a white background.

ODS' built-in format engines addresses the inherent weaknesses found in traditional SAS output by enabling "quality" output to be produced without the need of importing it into word processors. The latest version boasts many new format destinations including RTF, PDF, HTML, SAS data sets, and EXCEL while supporting traditional Listing output as well. Users have a powerful and easy way to create, access and share formatted procedure and DATA step output and results to Microsoft Excel.

References

Davis, Michael (2000), *"You Could Look It Up: An Introduction to SASHELP Dictionary Views,"* Proceedings of the North East SAS Users Group (NESUG) 2000 Conference, Bassett Consulting Services, North Haven, CT, USA.

Hamilton, Jack (1998), "*Some Utility Applications of the Dictionary Tables in PROC SQL*," Proceedings of the 1998 Western Users of SAS Software (WUSS) Conference, 85-90.

Lafler, Kirk Paul (2017), *"Hands-on Introduction to the SAS® ODS Excel® Destination,"* Mid West SAS Users Group (MWSUG) Conference, Software Intelligence Corporation, Spring Valley, CA, USA.

Lafler, Kirk Paul; Josh Horstman; Roger Muller (2017), "Building High-Impact Dashboards Using SAS® Base Software," Proceedings of the 2017 Pharmaceutical SAS Users Group (PharmaSUG) Conference.

Lafler, Kirk Paul (2016), *"Valuable Things You Can Do with SAS DICTIONARY Tables and SASHELP Views,"* Wisconsin Illinois SAS Users (WIILSU) Conference, Software Intelligence Corporation, Spring Valley, CA, USA.

Lafler, Kirk Paul (2012), *"Exploring DICTIONARY Tables and SASHELP Views,"* Kansas City SAS Users Group (KCSUG) Meeting, Software Intelligence Corporation, Spring Valley, CA, USA.

Lafler, Kirk Paul (2012), *"Exploring DICTIONARY Tables and SASHELP Views,"* South Central SAS Users Group (SCSUG) Conference, Software Intelligence Corporation, Spring Valley, CA, USA.

Lafler, Kirk Paul (2010), *"DATA Step and PROC SQL Programming Techniques,"* Ohio SAS Users Group (OSUG) 2010 One-Day Conference, Software Intelligence Corporation, Spring Valley, CA, USA.

Lafler, Kirk Paul (2009), *"Exploring DICTIONARY Tables and SASHELP Views,"* South Central SAS Users Group (SCSUG) Conference, Software Intelligence Corporation, Spring Valley, CA, USA.

Lafler, Kirk Paul (2009), *"Exploring DICTIONARY Tables and SASHELP Views,"* Western Users of SAS Software (WUSS) Conference, Software Intelligence Corporation, Spring Valley, CA, USA.

Lafler, Kirk Paul (2009), *"Exploring DICTIONARY Tables and SASHELP Views,"* PharmaSUG SAS Users Group Conference, Software Intelligence Corporation, Spring Valley, CA, USA.

Lafler, Kirk Paul (2008), *"Kirk's Top Ten Best PROC SQL Tips and Techniques,"* Wisconsin Illinois SAS Users Conference (June 26th, 2008), Software Intelligence Corporation, Spring Valley, CA, USA.

Lafler, Kirk Paul (2008), *"Undocumented and Hard-to-find PROC SQL Features,"* Greater Atlanta SAS Users Group (GASUG) Meeting (June 11th, 2008), Software Intelligence Corporation, Spring Valley, CA, USA.

Lafler, Kirk Paul (2008), *"Undocumented and Hard-to-find PROC SQL Features,"* PharmaSUG SAS Users Group Conference (June 1st - 4th, 2008), Software Intelligence Corporation, Spring Valley, CA, USA.

Lafler, Kirk Paul (2008), *"Undocumented and Hard-to-find PROC SQL Features,"* Michigan SAS Users Group (MSUG) Meeting (May 29th, 2008), Software Intelligence Corporation, Spring Valley, CA, USA.

Lafler, Kirk Paul (2008), *"Undocumented and Hard-to-find PROC SQL Features,"* Vancouver SAS Users Group Meeting (April 23rd, 2008), Software Intelligence Corporation, Spring Valley, CA, USA.

Lafler, Kirk Paul (2008), *"Undocumented and Hard-to-find PROC SQL Features,"* PhilaSUG 2008 User Group Meeting (March 13th, 2008), Software Intelligence Corporation, Spring Valley, CA, USA.

Lafler, Kirk Paul (2006), "*Exploring Dictionary Tables with PROC SQL*," SAS Press Webinar Series – June 27, 2006.

Lafler, Kirk Paul (2005), "*Exploring Dictionary Tables and SASHELP Views*," Proceedings of the Thirteenth Annual Western Users of SAS Software Conference.

Lafler, Kirk Paul (2004). *PROC SQL: Beyond the Basics Using SAS*, SAS Institute Inc., Cary, NC, USA.

Trademark Citations

SAS 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.

About The Author

Kirk Paul Lafler is a consultant, developer, programmer, educator, and data scientist; and teaches SAS Programming and Data Management in the Statistics Department at San Diego State University. Kirk also provides project-based consulting and programming services to client organizations in a variety of industries including healthcare, life sciences, and business; and teaches "virtual" and "live" SAS, SQL, Python, Database Management Systems (DBMS) technologies (e.g., Oracle, SQL-Server, Teradata, MySQL, MongoDB, PostgreSQL, AWS), Excel, R, cloud-based technologies as well as other software and tools. Currently, Kirk serves as the Western Users of SAS Software (WUSS) Executive Committee (EC) Open-Source Advocate and Coordinator and is actively involved with several proprietary and open-source software, DBMS, machine learning, cloud-computing user groups and conference committees. Kirk is the author of several books including the popular PROC SQL: Beyond the Basics Using SAS, Third Edition (SAS Press. 2019), along with other technical books and publications. He is also an Invited speaker, educator, keynote, and leader; and is the recipient of 28 "Best" contributed paper, hands-on workshop (HOW), and poster awards.

Comments and suggestions are encouraged and can be sent to:

Kirk Paul Lafler, sasNerd
Consultant, Developer, Programmer, Data Scientist, Educator, and Author
Specializing in SAS® / Python / SQL / Database Management Systems / Excel / R / AWS / Cloud-based Technologies
E-mail: KirkLafler@cs.com
LinkedIn: https://www.linkedin.com/in/KirkPaulLafler/