Excel in SAS®

Iowa SAS Users Group
13 May 2024

Charu Shankar
SAS Education

Bio

With a background in computer systems management. SAS Instructor Charu Shankar engages with logic, visuals, and analogies to spark critical thinking.

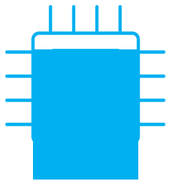
A SAS Instructor since 2007, she curates and delivers unique content via the SAS YouTube channel, SAS global forum, SAS Ask the Expert Series, SAS Training Post Blog, etc.

Charu loves to support users by teaching at conferences on topics related to SAS, SQL, Efficiencies, PERL, Macros, Python, Viya, etc.

When she’s not coding, Charu is A Yoga Instructor who loves to explore Canadian trails with her husky Miko.

Agenda

PROC IMPORT – Read Excel Into SAS®



SAS Access Engines – Read Excel Into SAS®

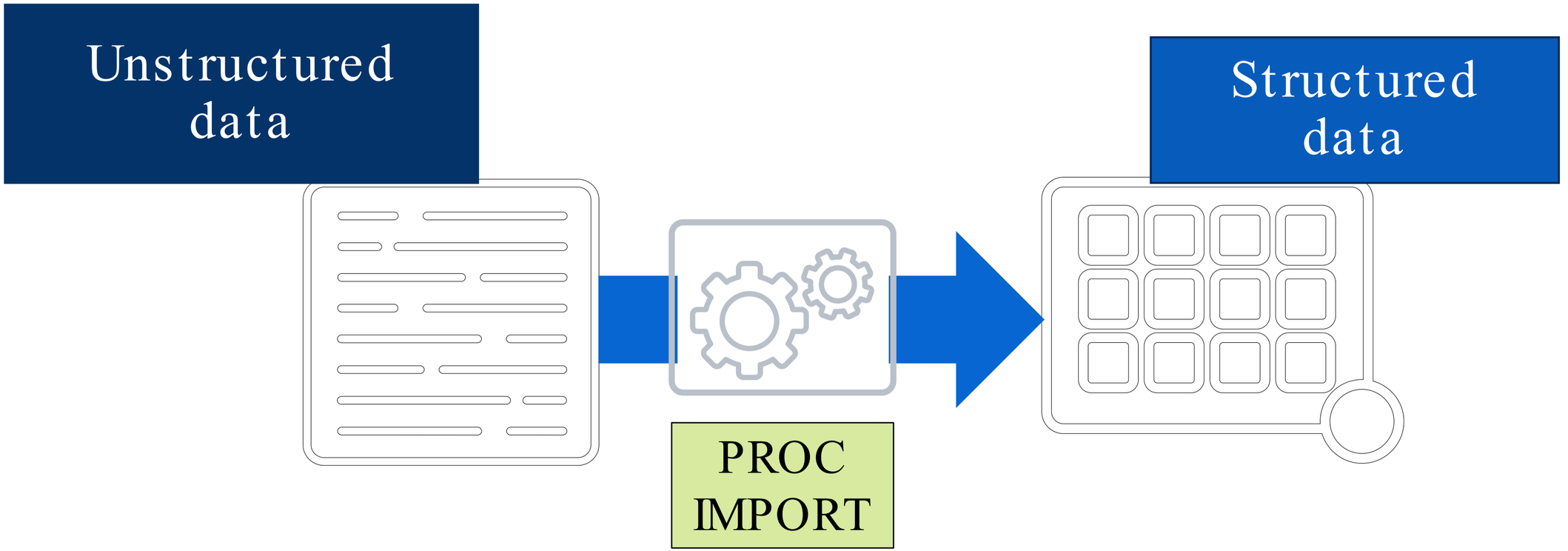
PROC EXPORT – Export SAS to Excel

ODS Tagsets – Take SAS Output to Excel Pivot Tables

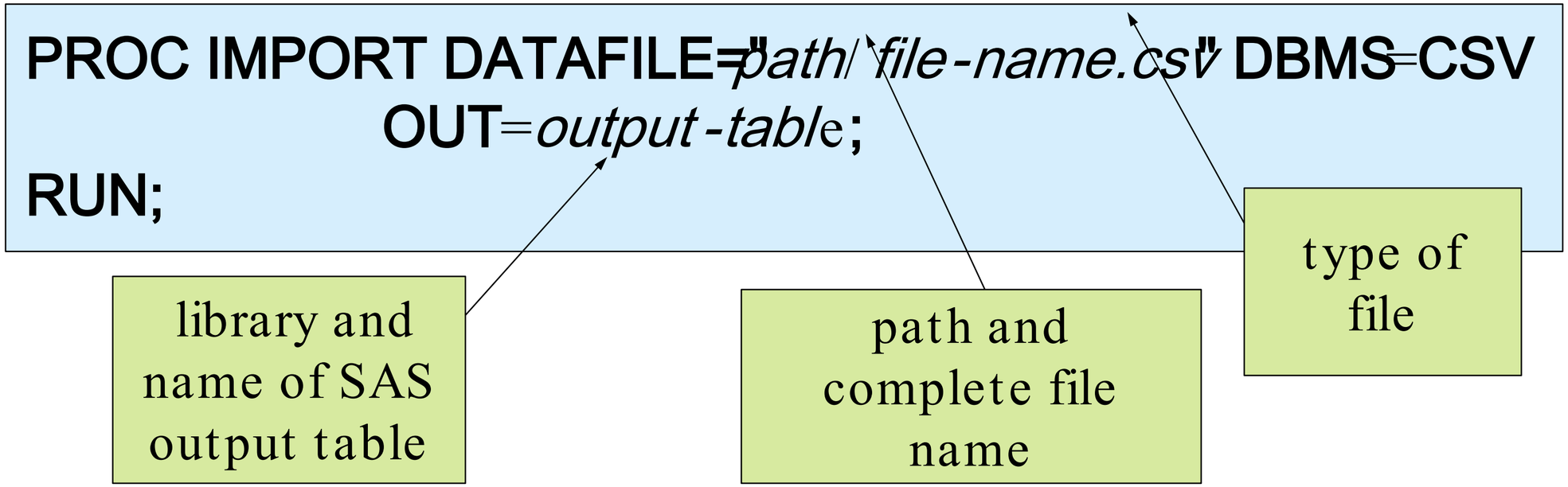
Handy Links

1 PROC Import – Read Excel Into SAS

Importing Unstructured Data



Importing a Comma-Delimited (CSV) File



Importing a Comma-Delimited (CSV) File

```
PROC IMPORT DATAFILE="path/file-name.csv" DBMS=CSV
            OUT=output-table <REPLACE>;
     <GUESSINGROWS=n | MAX;>
RUN;
```

specifies the number of rows used to determine column type and length (default = 20)

replace the output table if it exists

Importing an Excel File

name of sheet that you want to import

```
PROC IMPORT DATAFILE="path/file-name.xlsx" DBMS=XLSX
            OUT=output-table <REPLACE>;
     SHEET=sheet-name;
RUN;
```

type of file

```
proc import datafile="s:/workshop/data/class.xlsx"
            dbms=xlsx
            out=work.class_test_import replace;
     sheet=class_test;
run;
```

2 SAS Access Engines - Read Excel Into SAS®

SAS/ ACCESS LIBNAME Statement

SAS/ ACCESS has multiple LIBNAME engines that access Microsoft Excel workbooks.

```
libname orionx excel "&path\sales.xlsx";
```

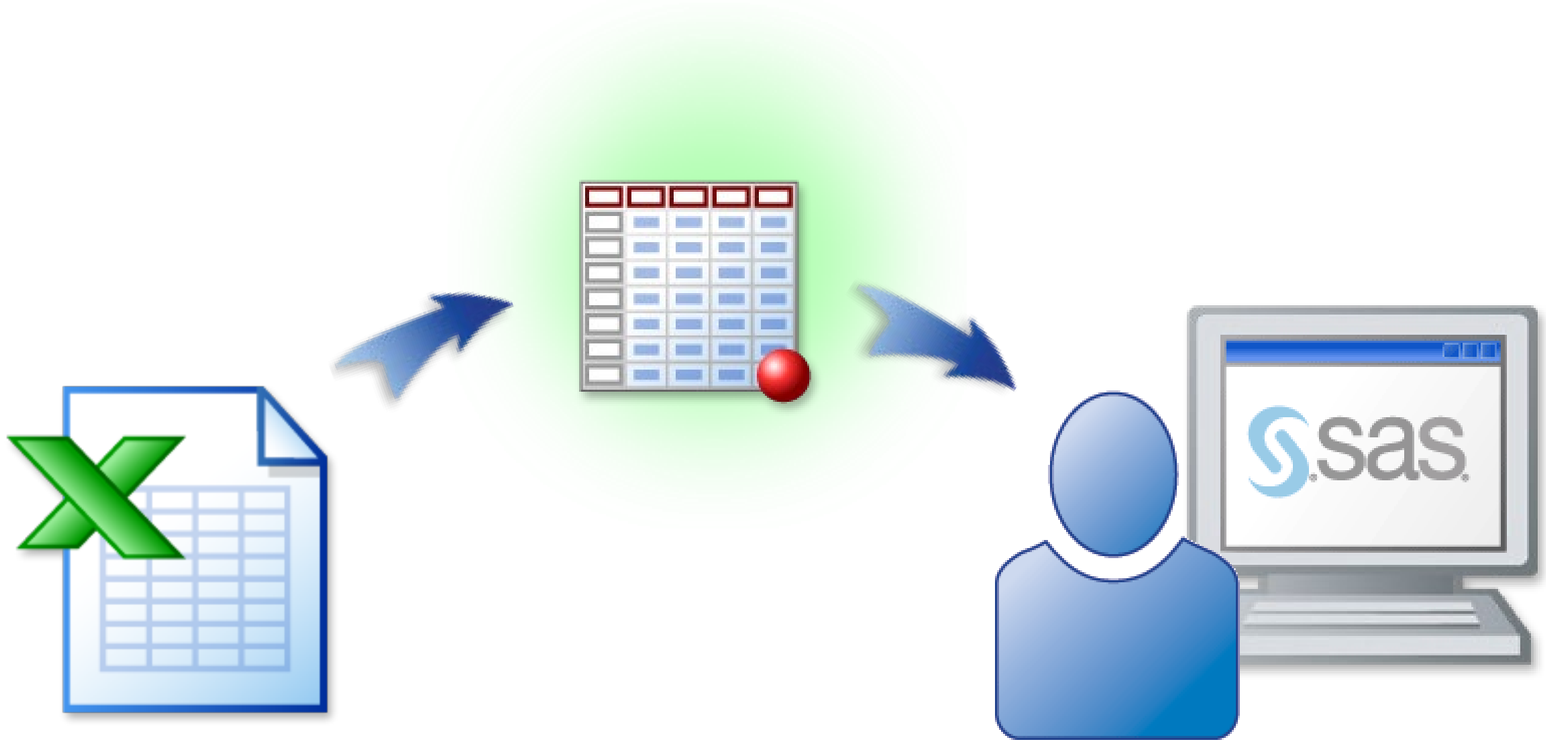
```
libname orionx pcfiles path="&path\sales.xlsx";
```

```
libname orionx xlsx "&path\sales.xlsx";
```

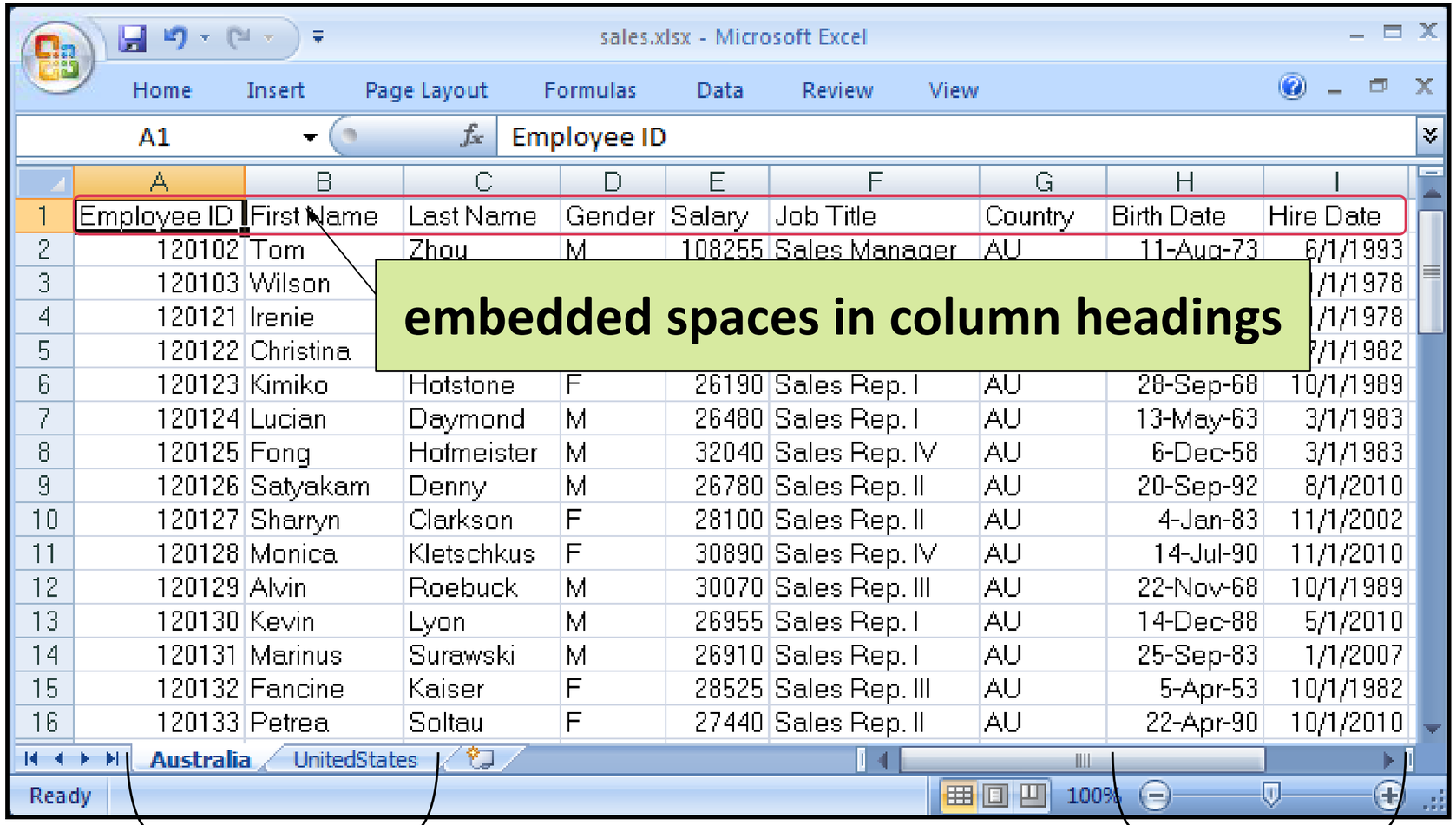
- The engine that you use depends on the operating environment of SAS and the bitness of SAS and Excel.

Business Scenario

Use SAS/ACCESS Interface to PC Files to read the Excel worksheet as if it were a SAS data set.



Examining Sales.xlsx



	A	B	C	D	E	F	G	H	I
1	Employee ID	First Name	Last Name	Gender	Salary	Job Title	Country	Birth Date	Hire Date
2	120102	Tom	Zhou	M	108255	Sales Manager	AU	11-Aug-73	6/1/1993
3	120103	Wilson							/1/1978
4	120121	Irenie							/1/1978
5	120122	Christina							/1/1982
6	120123	Kimiko	Hotstone	F	26190	Sales Rep. I	AU	28-Sep-68	10/1/1989
7	120124	Lucian	Daymond	M	26480	Sales Rep. I	AU	13-May-63	3/1/1983
8	120125	Fong	Hofmeister	M	32040	Sales Rep. IV	AU	6-Dec-58	3/1/1983
9	120126	Satyakam	Denny	M	26780	Sales Rep. II	AU	20-Sep-92	8/1/2010
10	120127	Sharryn	Clarkson	F	28100	Sales Rep. II	AU	4-Jan-83	11/1/2002
11	120128	Monica	Kletschkus	F	30890	Sales Rep. IV	AU	14-Jul-90	11/1/2010
12	120129	Alvin	Roebuck	M	30070	Sales Rep. III	AU	22-Nov-68	10/1/1989
13	120130	Kevin	Lyon	M	26955	Sales Rep. I	AU	14-Dec-88	5/1/2010
14	120131	Marinus	Surawski	M	26910	Sales Rep. I	AU	25-Sep-83	1/1/2007
15	120132	Fancine	Kaiser	F	28525	Sales Rep. III	AU	5-Apr-53	10/1/1982
16	120133	Petrea	Soltau	F	27440	Sales Rep. II	AU	22-Apr-90	10/1/2010

embedded spaces in column headings

two worksheets

cells formatted as dates

XLSX Engine

```
LIBNAME libref XLSX "workbook-name.xlsx" <options>;
```

- The XLSX engine requires SAS 9.4M2 or later.
- SAS can be located on Windows, Linux, or UNIX.
- This engine is used with any bitness combination.
- It accesses the Excel file directly without using the Microsoft ACE engine.
- The Excel file must be accessible to Linux or UNIX when you use SAS that is not on Windows.
- The engine reads only .xlsx file formats (Excel 2007 or later).
- It can read more than 255 columns.
- It allows limited LIBNAME options.

Quiz

Does SAS allow embedded blanks and special characters in variable names?

- ☐ Yes
- ☐ No

Quiz - Correct Answer

Does SAS allow embedded blanks and special characters in variable names?

- ○ Yes
- ○ No

SAS variable names

- can be 1 to 32 characters long.
- must start with a letter or underscore. Subsequent characters can be letters, underscores, or numerals.
- can be uppercase, lowercase, or mixed case.
- are not case sensitive.

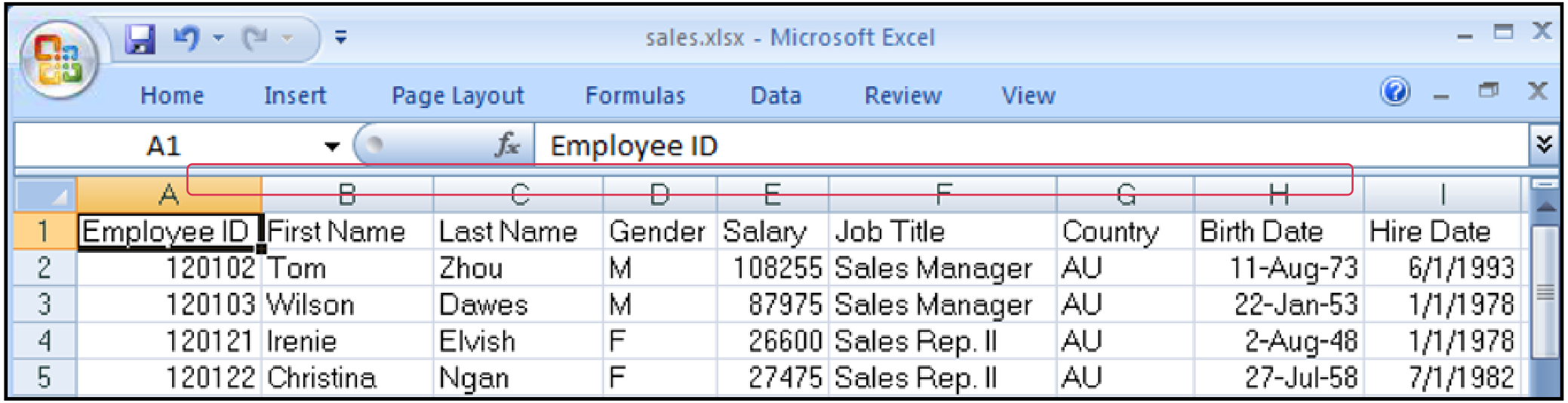
VALIDVARNAME= Option

Set the VALIDVARNAME= option to **V7** to enforce SAS naming rules in SAS Studio and SAS Enterprise Guide.

```
options validvarname=V7;
```

- Up to 32 mixed-case alphanumeric characters are allowed.
- Names must begin with an alphabetic character or an underscore.
- Invalid characters are changed to underscores.
- Any column name that is not unique is made unique by appending a counter to the name.

SAS Variable Names



	A	B	C	D	E	F	G	H	I
1	Employee ID	First Name	Last Name	Gender	Salary	Job Title	Country	Birth Date	Hire Date
2	120102	Tom	Zhou	M	108255	Sales Manager	AU	11-Aug-73	6/1/1993
3	120103	Wilson	Dawes	M	87975	Sales Manager	AU	22-Jan-53	1/1/1978
4	120121	Irenie	Elvish	F	26600	Sales Rep. II	AU	2-Aug-48	1/1/1978
5	120122	Christina	Ngan	F	27475	Sales Rep. II	AU	27-Jul-58	7/1/1982

Excel column headings are used as variable names.

- The SAS windowing environment replaces blanks and special characters with underscores.
- By default, SAS Studio and SAS Enterprise Guide allow blanks and special characters in variable names.

Exploring the Library

Regardless of the LIBNAME engine that is used, you can use the CONTENTS procedure to explore the library.

```
libname orionx pcfiles path="&path\sales.xlsx";

proc contents data=orionx._all_;
run;

libname orionx clear;
```

- When SAS has a libref that is assigned to an Excel workbook, the workbook cannot be opened in Excel.
- To disassociate the libref, use a LIBNAME statement with the CLEAR option.

p107d01a

Exploring the Library

Partial PROC CONTENTS Output

XLSX Engine

The CONTENTS Procedure

#	Name	Member Type	DBMSTYPE
1	AUSTRALIA	DATA	TABLE
2	UNITEDSTATES	DATA	TABLE

For the EXCEL and PCFILES engines, worksheet names end with a dollar sign.

p107d01a

Subsetting a Worksheet

Regardless of the LIBNAME engine used, the PRINT procedure can be used to display a subset of the worksheet.

```
proc print data=orionx.Australia noobs;
    where Job_Title contains 'IV';
    var Employee_ID Last_Name
        Job_Title Salary;
run;
```

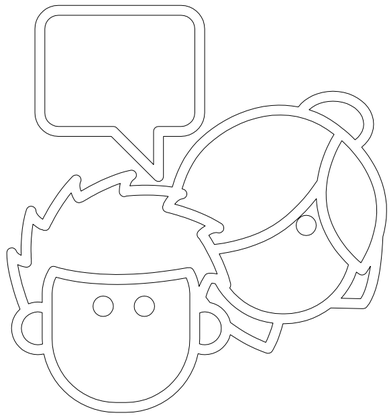
XLSX engine

Subsetting a Worksheet

PROC PRINT Output

Employee_ ID	Last_Name	Job_Title	Salary
120125	Hofmeister	Sales Rep. IV	32040
120128	Kletschkus	Sales Rep. IV	30890
120135	Platts	Sales Rep. IV	32490
120159	Phoumirath	Sales Rep. IV	30765
120166	Nowd	Sales Rep. IV	30660

Quiz



What is the difference between using the XLSX LIBNAME engine and PROC IMPORT to read Excel data in a SAS program?

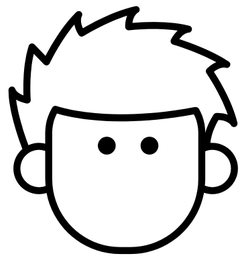
3 PROC Export - Export SAS to Excel®

Exporting Data Using PROC EXPORT

```
PROC EXPORT DATA=input-table OUTFILE="output-file"
            <DBMS=identifier> <REPLACE>;
RUN;
```

tells SAS how to format the output

Column names are automatically written as the first row of the output file.

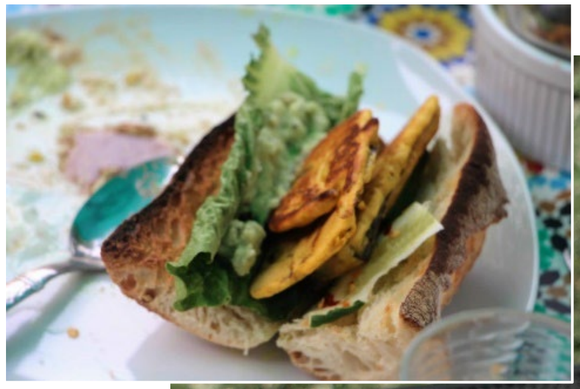


Exporting Data Using Code

```
proc export data=sashelp.cars
     outfile="C:/HOW/excel in sas 42/data/cars.xlsx"
     dbms=xlsx replace;
run;
```

3 ODS Tagsets -Take SAS®to Excel Pivot Tables

ODS: The Sandwich Technique



ODS Tagsets Table Editor to Excel Pivot Tables

A *pivot table* is an interactive, cross-tabulated Excel report that summarizes and analyzes data from various sources, including those external to Excel.

Benefits:

- Drill-down capabilities
- User-friendly, hands-on reports
- Pivot tables can be performed on multiple subsets of data.

Update the Template Item Store

- By default, templates are stored in the SASUSER.TEMPLAT item store unless the ODS PATH is modified.
- Errors might occur when writing to the SASUSER location if the access mode for the template store has been changed to (read).
- To eliminate error messages, use the ODS PATH statement to write to an alternate location.
- The example below adds a temporary item store in the Work directory where the templates will be written.
- In this example, the work.templat item store is added before the default path.

```
Ods path(Prepend) work.templat(update);
```

Install the Tagsets Table Editor

Install the tagsets Table Editor by simply using the %INCLUDE statement to run it.

```
%let path = C:\ugdemo\code\code for tagsets tableeditor;
%include "&path\tableeditor.tpl";
```

Use the Tagsets Table Editor to Add a Pivot Table

```
ods tagsets.tableeditor
file="%sysfunc(getoption(work))\iasug.html"
```

%sysfunc(getoption(work)) points to location of the **Work** library.

Add Row and Column Options

```
ods tagsets.tableeditor
file="%sysfunc(getoption(work))\iasug.html"
options(
pivotrow="product_line"
pivotcol="year"
pivotdata="profit"
```

Use options to add one or more columns to row, column, page, and data areas of the layout in Excel.

PIVOTPAGE= option - creates the report layout.

PIVOTROW= option - adds one or more variables to the row.

PIVOTCOL= option - adds one or more variables to the column.

PIVOTDATA= option - supplies one or more variables to the analysis.

Add Labeling Options

```
ods tagsets.tableeditor
file="%sysfunc(getoption(work))\iasug.html"
options(
pivotrow="product_line"
pivotcol="year"
pivotdata="profit"
pivot_sheet_name="Profit Analysis Pivot Table"
sheet_name="Profit Report"
```

PIVOT_SHEET_NAME :

Provides the ability to specify the worksheet names for the pivot tables.

Add Statistics to Pivot Table

```
ods tagsets.tableeditor
file="%sysfunc(getoption(work))\iasug.html"
options(
pivotrow="product_line"
pivotcol="year"
pivotdata="profit"
pivot_sheet_name="Profit Analysis Pivot Table"
sheet_name="Profit Report"
pivotdata_stats="average"
```

PIVOTTDATA_STATS :

Provides statistics for the data fields of the pivot tables.

By default, the SUM statistic is added for the analysis. Use the PIVOTDATA_STATS= tagset option to change statistics. You can use any statistic that can be generated by Excel in the PIVOTDATA_STATS option.

Available Statistics in Excel

Here are the available statistics in Excel:

- Average – the average of the data items
- Count – the number of data items
- Countnums – the number of data items with numeric values Max, the data item with the largest value
- Min – the data item with the smallest value
- Product – the result of multiplying the data items
- StdDev – the sample standard deviation
- StdDevP – the population standard deviation
- Sum – the result of adding the data items
- Var – the sample variance
- Varp – the population variance

Create Pivot Charts

```
ods tagsets.tableeditor
file="%sysfunc(getoption(work))\pharmasug.html"
options(
pivotrow="product_line"
pivotcol="year"
pivotdata="profit"
pivot_sheet_name="Profit Analysis Pivot Table"
sheet_name="Profit Report"
pivotdata_stats="average"
pivotcharts="yes"
chart_type="conecol"
```

Pivot charts can be added to an existing workbook as well. Simply add the PIVOTCHARTS="yes" option along with the CHART_TYPE option.

Ask for Help

Most of the MARKUP destinations and a few of the non-MARKUP destinations are considered self-documenting.

Use the DOC='HELP' suboption to retrieve documentation in the SAS log.

ODS *destination* . . . **OPTIONS(DOC='HELP');**

Ask for Help

```
ods tagsets.tableeditor
file="%sysfunc(getoption(work))\pharmasug.html"
options(
pivotrow="product_line"
pivotcol="year"
pivotdata="profit"
pivot_sheet_name="Profit Analysis Pivot Table"
sheet_name="Profit Report"
pivotdata_stats="average"
pivotcharts="yes"
chart_type="conecol"
 doc='help' );
```

ODS *destination* . . . **OPTIONS(DOC='HELP');**

Final Step: PROC PRINT

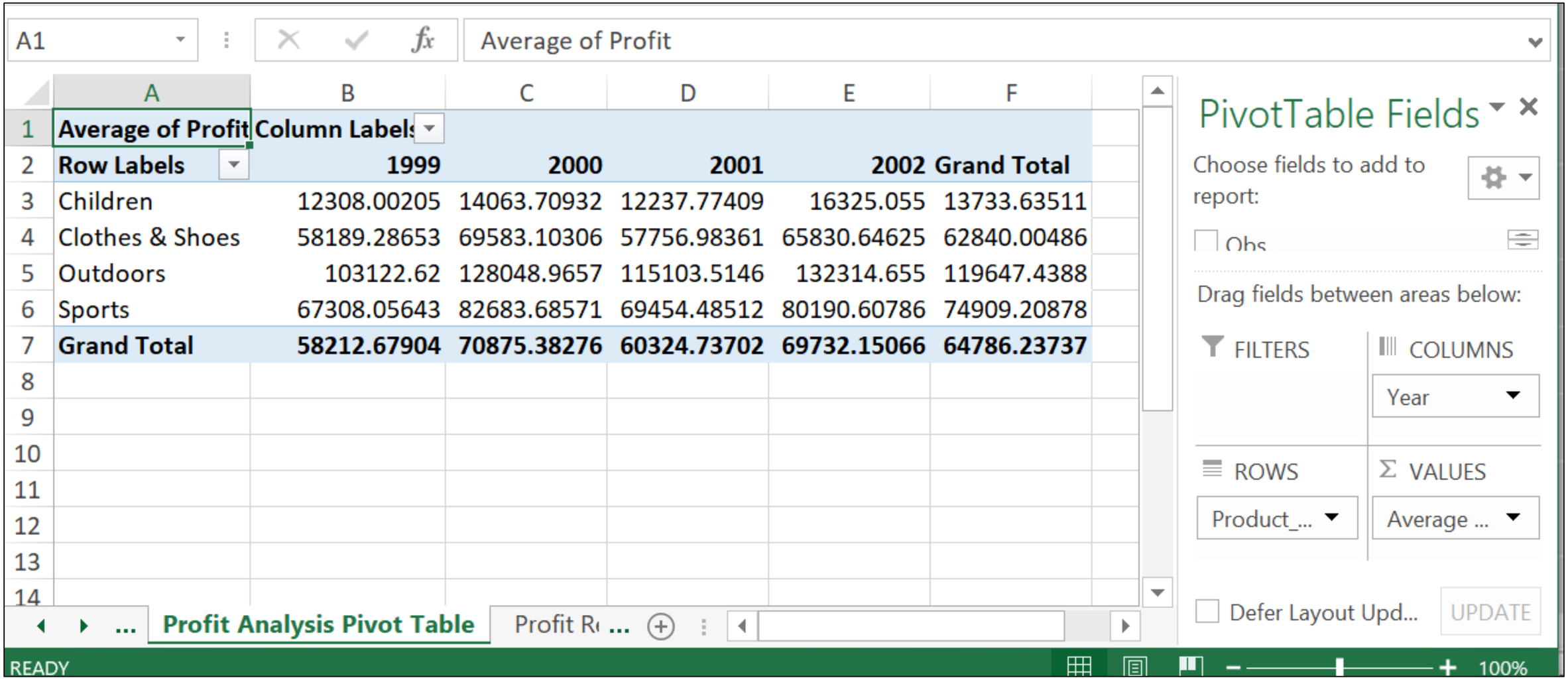
```
ods tagsets.tableeditor
file="%sysfunc(getoption(work))\pharmasug.html"
options(
pivotrow="product_line"
pivotcol="year"
pivotdata="profit"
pivot_sheet_name="Profit Analysis Pivot Table"
sheet_name="Profit Report"
pivotdata_stats="average"
pivotcharts="yes"
chart_type="conecol"
 doc='help' );

proc print data=sashelp.orsales;
run;

ods tagsets.tableeditor close;
```

Select the data to be printed using PROC PRINT

Pivot Table Ready



Average of Profit	Column Labels				
Row Labels	1999	2000	2001	2002	Grand Total
Children	12308.00205	14063.70932	12237.77409	16325.055	13733.63511
Clothes & Shoes	58189.28653	69583.10306	57756.98361	65830.64625	62840.00486
Outdoors	103122.62	128048.9657	115103.5146	132314.655	119647.4388
Sports	67308.05643	82683.68571	69454.48512	80190.60786	74909.20878
Grand Total	58212.67904	70875.38276	60324.73702	69732.15066	64786.23737

Tip Sheet of Pivot Table Options

- Add pivot tables to an Excel worksheet using the TableEditor tagset.
- Use options such as PIVOTROW=, PIVOTCOL=, PIVOTPAGE, and PIVOTDATA= to add one or more columns to a row, a column, a page, and data areas of the layout in Excel.
- Add multiple variables with one of these options by separating each variable with a comma.
- Use additional options to modify statistics from the default of SUM.
- A button is added by default to the Web page.
- When using Microsoft Internet Explorer, the button can be selected to create the pivot table on Microsoft Windows.

The TableEditor tagset can be downloaded from this location .

Handy Links

LIBNAME Statement

Importing a .xlsx file into SAS

SAS/ ACCESS libname engines

Download tableeditor Zip file

ODS Excel destination tip sheet

PROC IMPORT". SAS®9.4 and SAS®Viya®3.4 Programming Documentation

How SAS engines can read a range of data in Excel". Shankar, Charu. Nov 20 10

Using SAS Output delivery system and ODS markup to generate custom Pivot tables

Using SAS to add Pivot Tables to your Excel Workbook". Parker, Chevell. March 20 15

Thank You

✓ Did you enjoy this session, Let us know in the **evaluation**

Charu Shankar
SAS Institute Toronto

EMAIL Charu.shankar@sas.com
BLOG https://blogs.sas.com/content/author/charushankar/
TWITTER CharuYogaCan
LINKEDIN https://www.linkedin.com/in/charushankar/