



# Writing Data with Excel Libname Engine

Nurefsan (Neffy) Davulcu

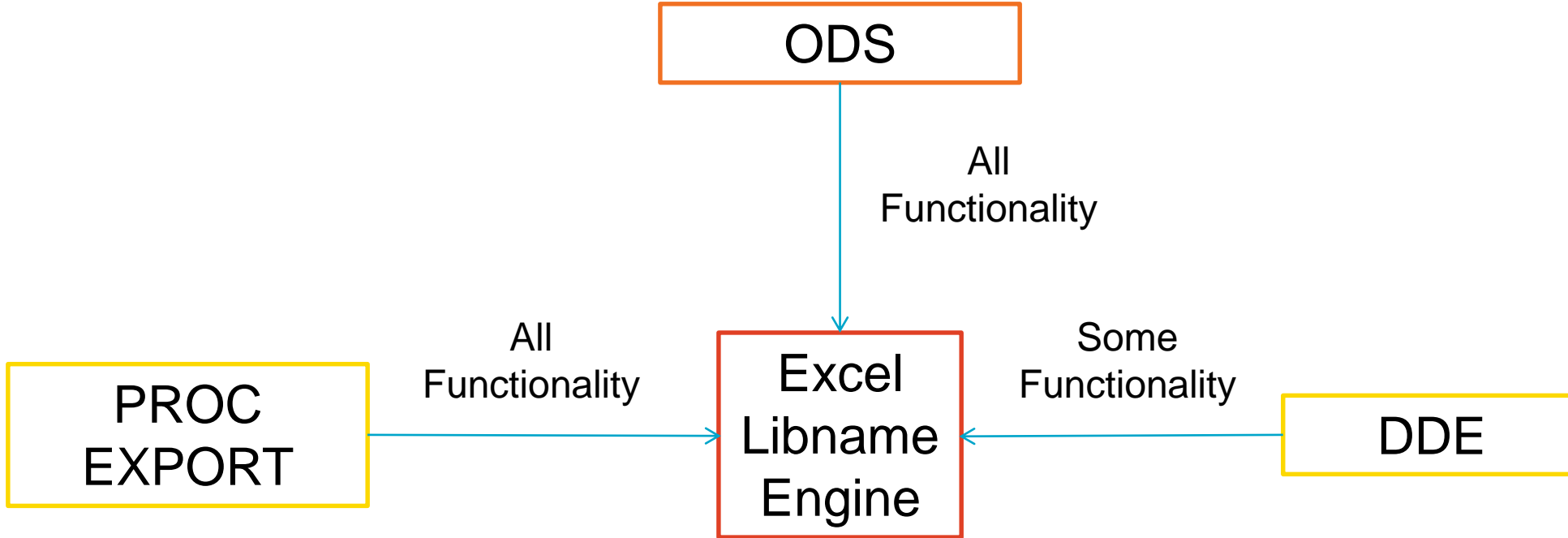
Advanced Analytics Intern, TransUnion Canada

Golden Horseshoe SAS User Group (GHSUG)

Burlington, Ontario, Canada

MAY 27, 2016





# What SAS Libname Engine Can Do

- ✓ **Create new workbooks**
- ✓ **Create a new spreadsheet with a named range**
- ✓ **Write data to an empty existing named range**
- ✓ Append data to spreadsheet data or named range data
- ✓ Read data from a pre-existing spreadsheet
- ✓ Read data from a pre-existing named range
- ✓ **Delete all the data from a spreadsheet**
- ✓ **Delete all the data from a named range**
- ✓ Do the above without Excel and PC

# What SAS Libname Engine Can't Do

- Rename spreadsheets within a workbook
- Delete spreadsheets from a workbook
- Delete workbooks as a whole
- Change or apply formatting
- Delete cells containing formulas
- Write a formula into a cell



# Big Deal – We put some data in Excel

Think: How can we use this to automate procedures?

How can we use this to avoid repeated manual entry and to avoid manual entry error?

How can we use this to create dynamic reports?

# Getting Started

SAS 9.0 or later

```
LIBNAME < libref> engine-name <physical-path and filename>  
<SAS/ACCESS engine-connection-options>  
<SAS/ACCESS LIBNAME-options> ;
```

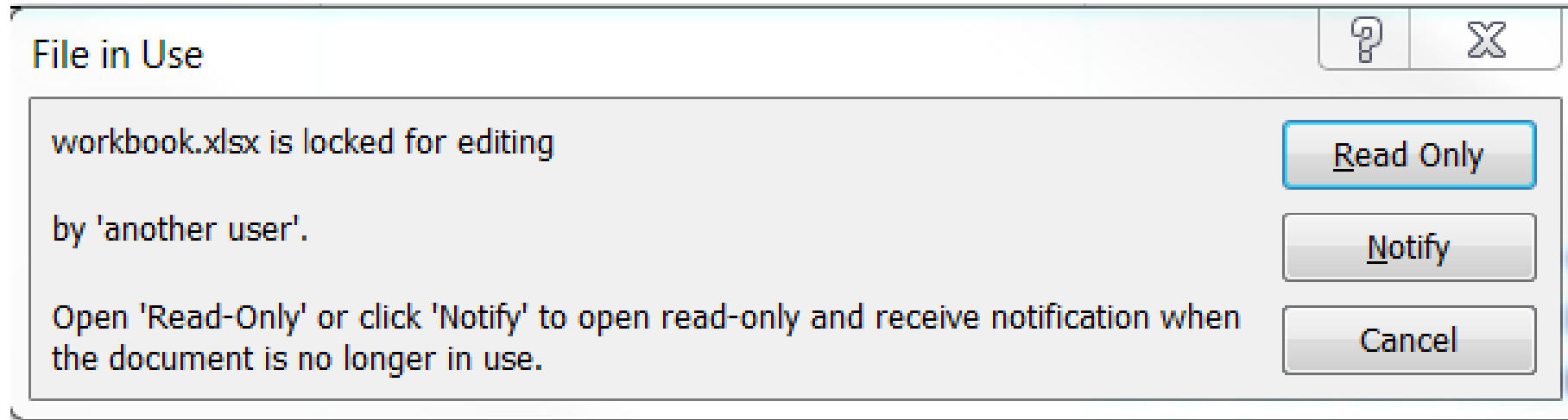
< libref> = Give the library a name

engine-name = excel \*note- this does not need to be specified in most cases, SAS will know from the path name

<physical-path and filename> = "C:\Users\ndavulc\Documents\workbook.xlsx"

Options : VER=year

# Getting Started – Deassign the library!



**LIBNAME** < *libref* > clear;

# Getting Started

Once the library is assigned, SAS will create the global library icon indicating it's not a library with only SAS datasets.

```
libname workbook excel "C:\Users\ndavulc\Documents\workbook.xlsx";
```



Workbook

If the workbook doesn't already exist SAS will create it.



# Excel Named Ranges

In Excel we can assign a group of cells a name called named ranges.

This is what is used to interact with the libname engine and allows data to be placed anywhere inside a spreadsheet.



# Getting Started

The excel libname engine supports many procedures

Commonly Used:

- PROC DATASETS
- PROC COPY
- PROC SQL
- DATA STEP



## Example 1

Create named ranges in SAS and write data

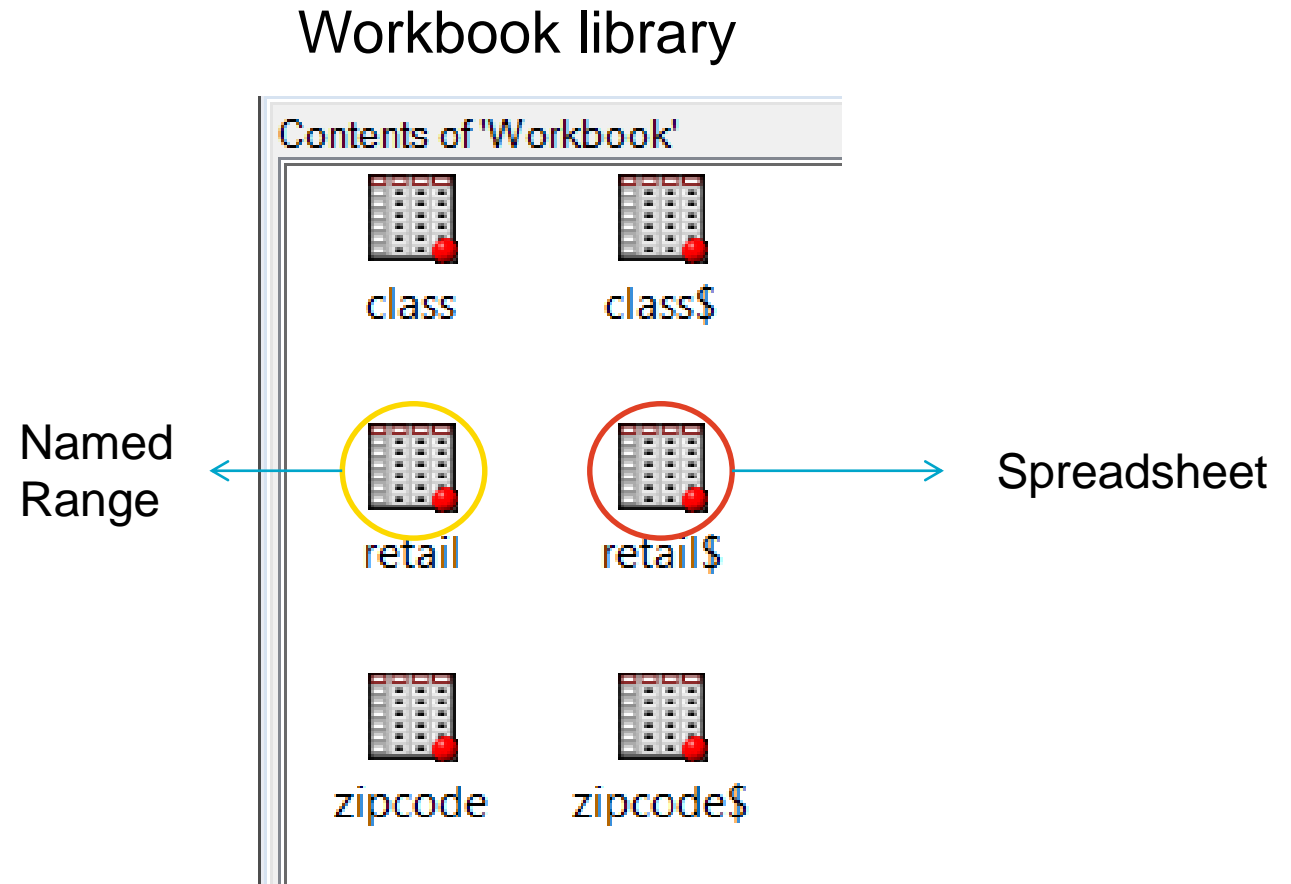
# Create named ranges in SAS and write data



```
DATA workbook.class;  
set sashelp.class;  
run;
```

```
DATA workbook.retail;  
set sashelp.retail;  
run;
```

```
DATA workbook.zipcode;  
set sashelp.zipcode;  
run;
```

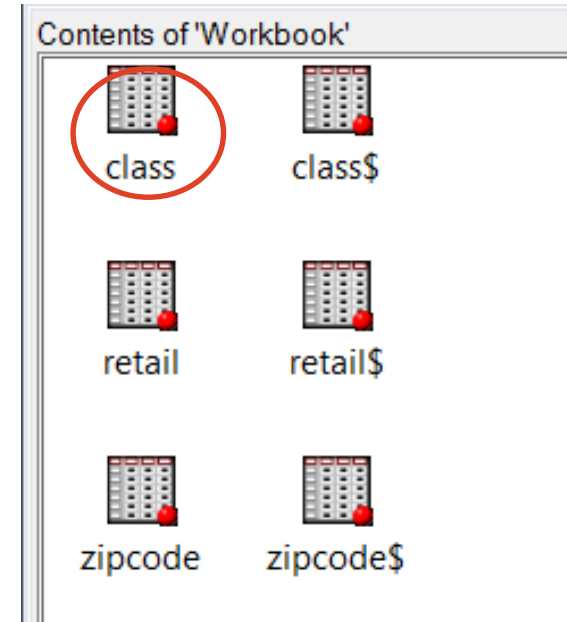


# Create named ranges in SAS and write data



Now I want to add another column and want to put it in the same class sheet.

```
DATA workbook.class;  
    set sashelp.class;  
    teacher='Mrs. Smith';  
  
run;
```



ERROR: The MS Excel table class has been opened for OUTPUT. This table already exists, or there is a name conflict with an existing object. This table will not be replaced. This engine does not support the REPLACE option.

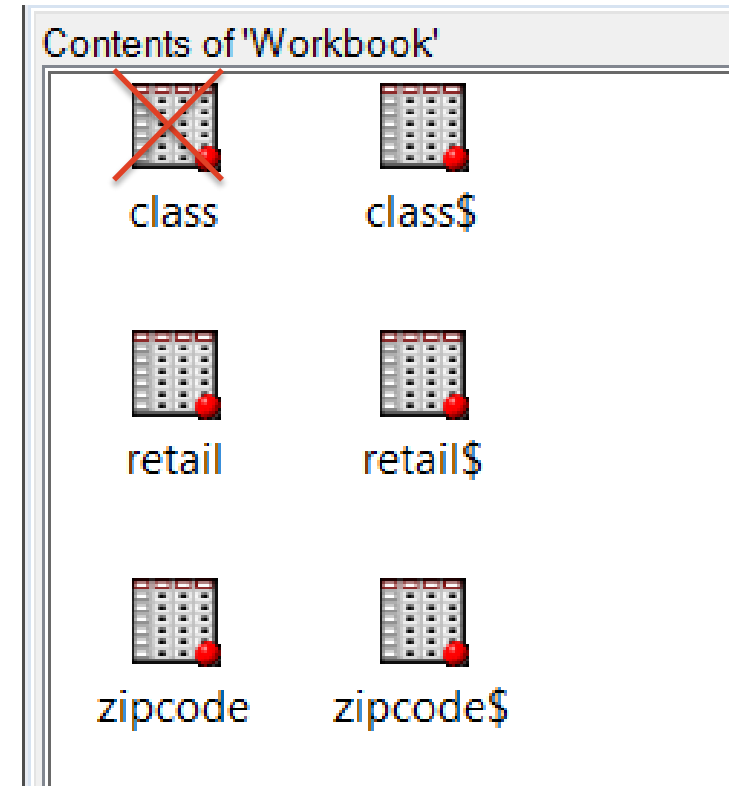
# Create named ranges in SAS and write data



We can't overwrite the data! We must first delete the data.

```
PROC DATASETS lib = workbook nolist;
```

```
    delete class;  
quit;  
run;
```



# Create named ranges in SAS and write data



We can't overwrite the data! We must first delete the data.

```
PROC DATASETS lib = workbook nolist;  
  
    delete class;  
quit;  
run;
```

\*Note that this removes the contents of the Named Range, the assignment of the named Range to a series of cells still exists in the Excel workbook.

# Create named ranges in SAS and write data



Now write the data with the extra teacher column.

```
DATA workbook.class;  
    set sashelp.class;  
    teacher='Mrs. Smith';  
run;
```

Note that the contents of this dataset will not fit in the previous existing class named range. The named range will be expanded in Excel to fit the new data **given the surrounding area is empty.**



# Create named ranges in SAS and write data



Don't forget to de assign the library when you're done!

```
libname workbook clear;
```



## Example 2

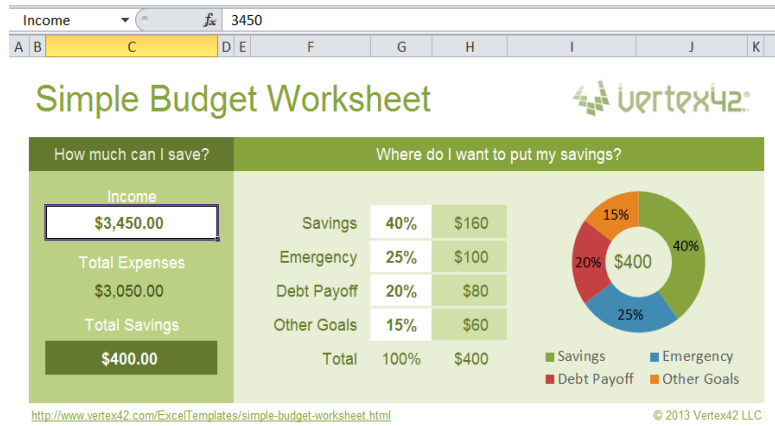
Write data to a premade Excel Template

# Write data to a premade Excel Template

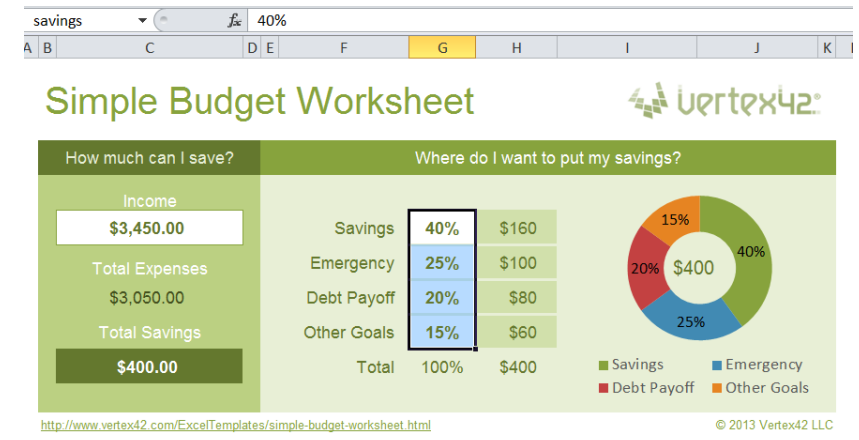
I have a Simple Budget Worksheet where I have already defined three named ranges.

expenses Car

What are my expenses?	
description	amount
Car	\$300.00
Rent	\$1,200.00
Utilities	\$100.00
School Debt	\$200.00
Credit Cards	\$500.00
Groceries	\$400.00
Other	\$350.00
Total Expenses \$3,050.00	



What are my expenses?	
description	amount
Car	\$300.00
Rent	\$1,200.00
Utilities	\$100.00
School Debt	\$200.00
Credit Cards	\$500.00
Groceries	\$400.00
Other	\$350.00



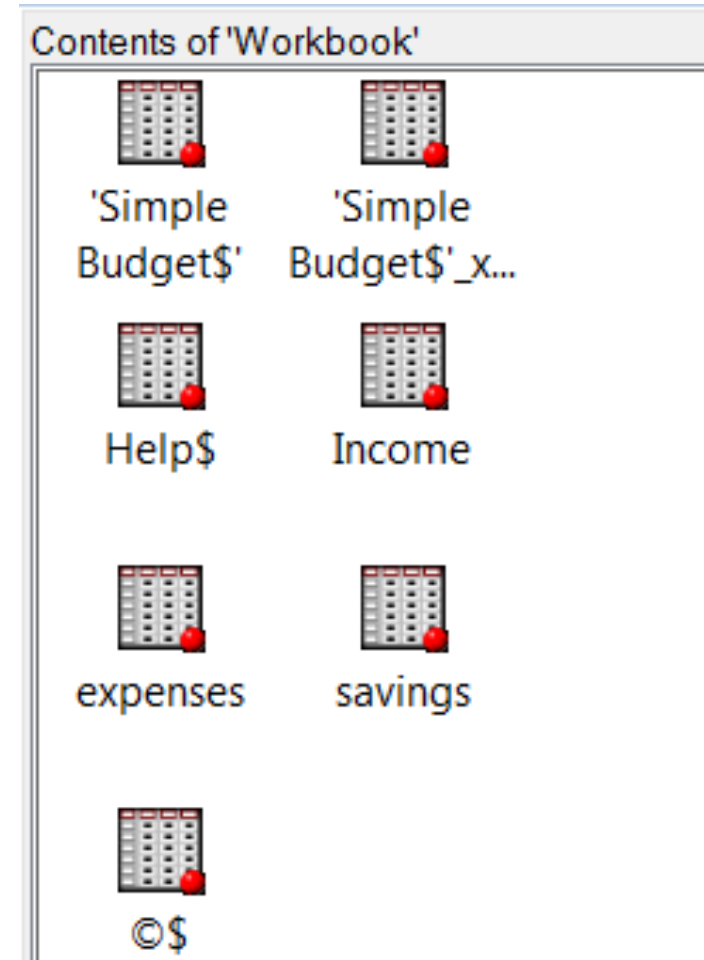
What are my expenses?	
description	amount
Car	\$300.00
Rent	\$1,200.00
Utilities	\$100.00
School Debt	\$200.00
Credit Cards	\$500.00
Groceries	\$400.00
Other	\$350.00

# Write data to a premade Excel Template

Assign a library to the simple budget worksheet exactly as before:

libname workbook excel

"C:\Users\ndavulc\Documents\simple\_budget.xlsx";



# Write data to a premade Excel Template

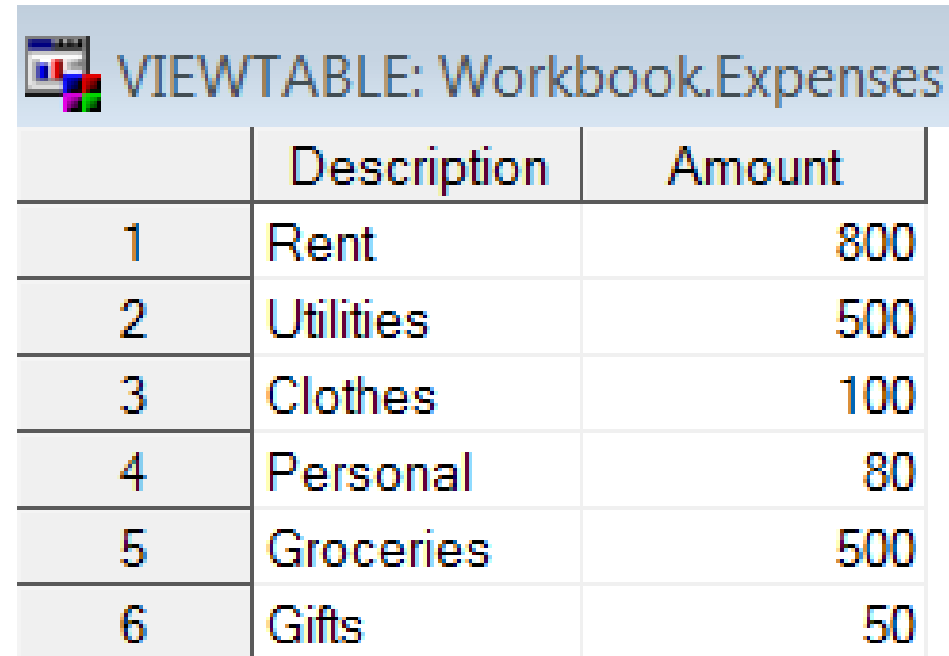
```
PROC DATASETS lib = workbook nolist;
  delete income
    expenses
    savings;
quit;
run;
```

```
PROC SQL;
  create table workbook.income as
  select * from income;

  create table workbook.expenses as
  select * from expenses;

  create table workbook.savings as
  select * from savings;
quit;
```

```
libname workbook clear;
```



	Description	Amount
1	Rent	800
2	Utilities	500
3	Clothes	100
4	Personal	80
5	Groceries	500
6	Gifts	50



# Write data to a premade Excel Template

**Problem #1:** We can only map one data cell to one Excel cell

The merged cells will not be accounted for.

**Fix #1a:** Format the spreadsheet accordingly.

The screenshot shows an Excel spreadsheet with a table. The header row is merged across columns B to I. The table content is as follows:

What are my expenses?	
description	amount
Description	
Rent	
Utilities	
Clothes	
Personal	
Groceries	
Gifts	



# Write data to a premade Excel Template

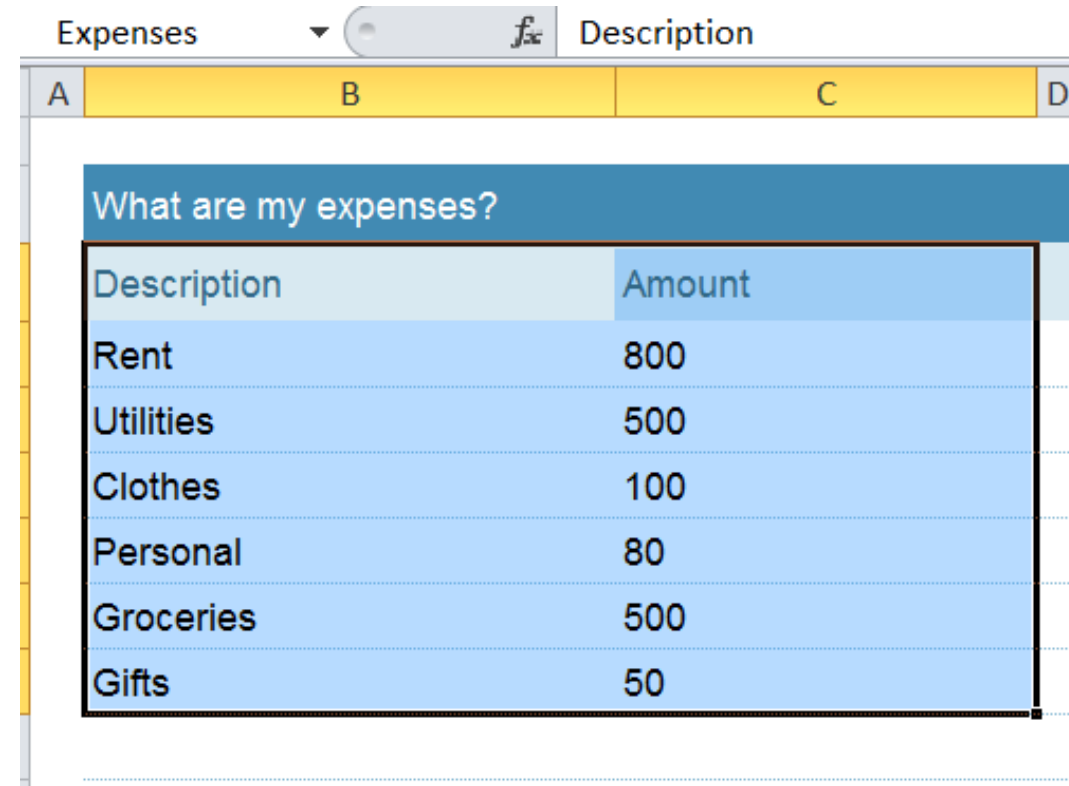
**Problem #2:** SAS will write the variable names.

What are my expenses?	
description	amount
Description	Amount
Rent	800
Utilities	500
Clothes	100
Personal	80
Groceries	500
Gifts	50

# Write data to a premade Excel Template

**Problem #2:** SAS will write the variable names.

**Fix #2a:** Account for this in the named range so that the headings in the spreadsheet are populated according to the variable names in SAS.



The screenshot shows an Excel spreadsheet with a named range 'Expenses' and a formula bar containing 'Description'. The spreadsheet has columns A, B, C, and D, and rows 1 through 8. The data is as follows:

What are my expenses?	
Description	Amount
Rent	800
Utilities	500
Clothes	100
Personal	80
Groceries	500
Gifts	50

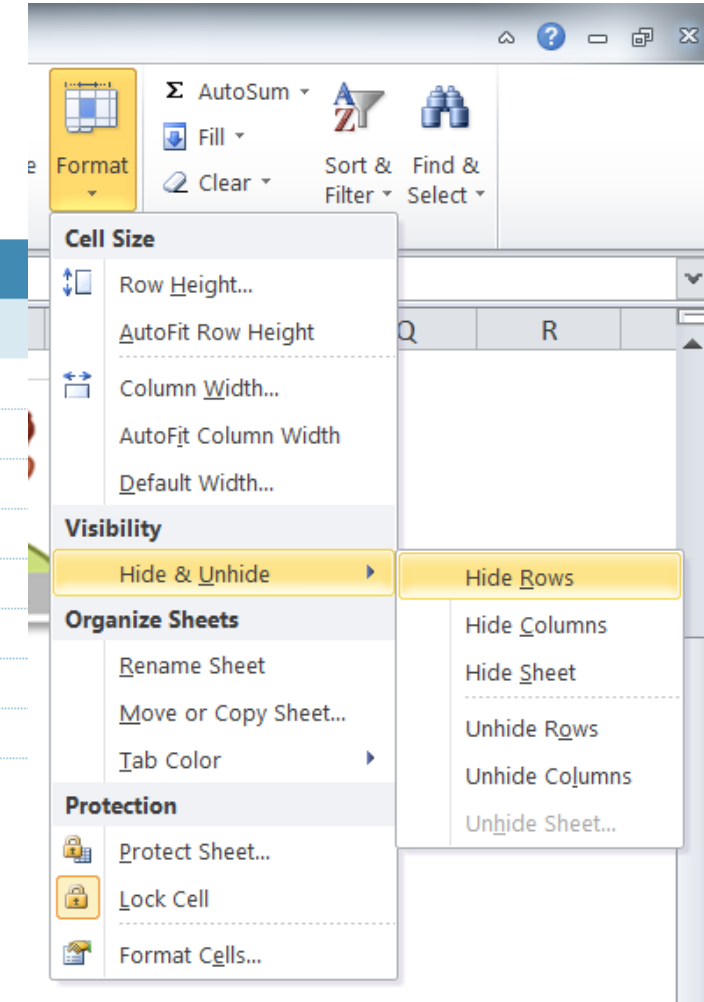


# Write data to a premade Excel Template

**Problem #2:** SAS will write the variable names.

**Fix #2b:** Use Hidden cells.

What are my expenses?	
description	amount
Description	Amount
Rent	800
Utilities	500
Clothes	100
Personal	80
Groceries	500
Gifts	50





# Write data to a premade Excel Template

Let's say we had 30 expenses listed in our SAS dataset and tried to write to Excel.

ERROR: Error attempting to CREATE a DBMS table.

ERROR: Execute: Too many fields defined..

WARNING: File deletion failed for WORKBOOK.expenses.data

**Remember that we can't delete cells containing formulas.**



# Wrap Up

- Utilize Excel's visual capabilities and SAS analytic capabilities by separating the visual and data aspects
- Increase efficiency by reducing manual entry and repetition

## Contact Information

Nurefsan (Neffy) Davulcu

*Advanced Analytics Intern, TransUnion Canada*

Work phone: 905-340-1000 Ext 2049

Work email: [ndavulc@transunion.com](mailto:ndavulc@transunion.com)

*McMaster University, Honours Math & Stats (Level 4, Co-op)*

Personal Cell: 289-689-7383

School Email: [davulcn@mcmaster.ca](mailto:davulcn@mcmaster.ca)

# References

Choate, Paul A and Martell, Carol A. “De-Mystifying the SAS LIBNAME Engine in Microsoft Excel: A Practical Guide” Accessed May 25, 2016.

<http://www2.sas.com/proceedings/sugi31/024-31.pdf>.

Heaton, Ed. “So, Your Data are in Excel!” Accessed May 25, 2016.

<http://www2.sas.com/proceedings/sugi31/020-31.pdf>.

Droogendyk, Harry. “Customized Excel Output Using the Excel Libname” Accessed May 25, 2016. [http://www.stratia.ca/papers/excel\\_libname.pdf](http://www.stratia.ca/papers/excel_libname.pdf).

“Libname Statement Syntax” SAS Institute. Accessed May 25, 2016.

<http://support.sas.com/documentation/cdl/en/acpcref/63181/HTML/default/viewer.htm#n1wvmggexroxgyn17rp61jml3cvn.htm>.