

Exporting the Data to EXCEL?

***Order the columns the
way you want them!***

Mark A. Martin
Siemens Healthcare Diagnostics Inc.

Boston Area SAS® Users Group (BASUG)
March 28, 2008

Have It Your Way...

... in Two Steps

1. Reorder the variables in the dataset
 - (a) RETAIN statement in DATA step, *or*
 - (b) SELECT statement in PROC SQL.
2. Export the dataset to Excel:
PROC EXPORT

What won't work ...

SAS Usage Note #16111:

In order to reorder the variables/columns when exporting SAS data to Excel, you will need to change the order in the SAS file first and export the reordered data.

You can not perform the reorder in the PROC EXPORT code.

Refer to #8395, "How to reorder variables in a SAS data set"

Futile attempt: KEEP statement does not reorder variables!

```
proc export data=towns(keep=community popn_change popn_2000 popn_1990 county)
  outfile = "C:\BASUG_demo\towns1.xls"
  dbms=Excel replace;
  sheet="Population Change"; * Sheet= is for SAS 9, not recognized by SAS 8;
run;
```

TOWNS Dataset

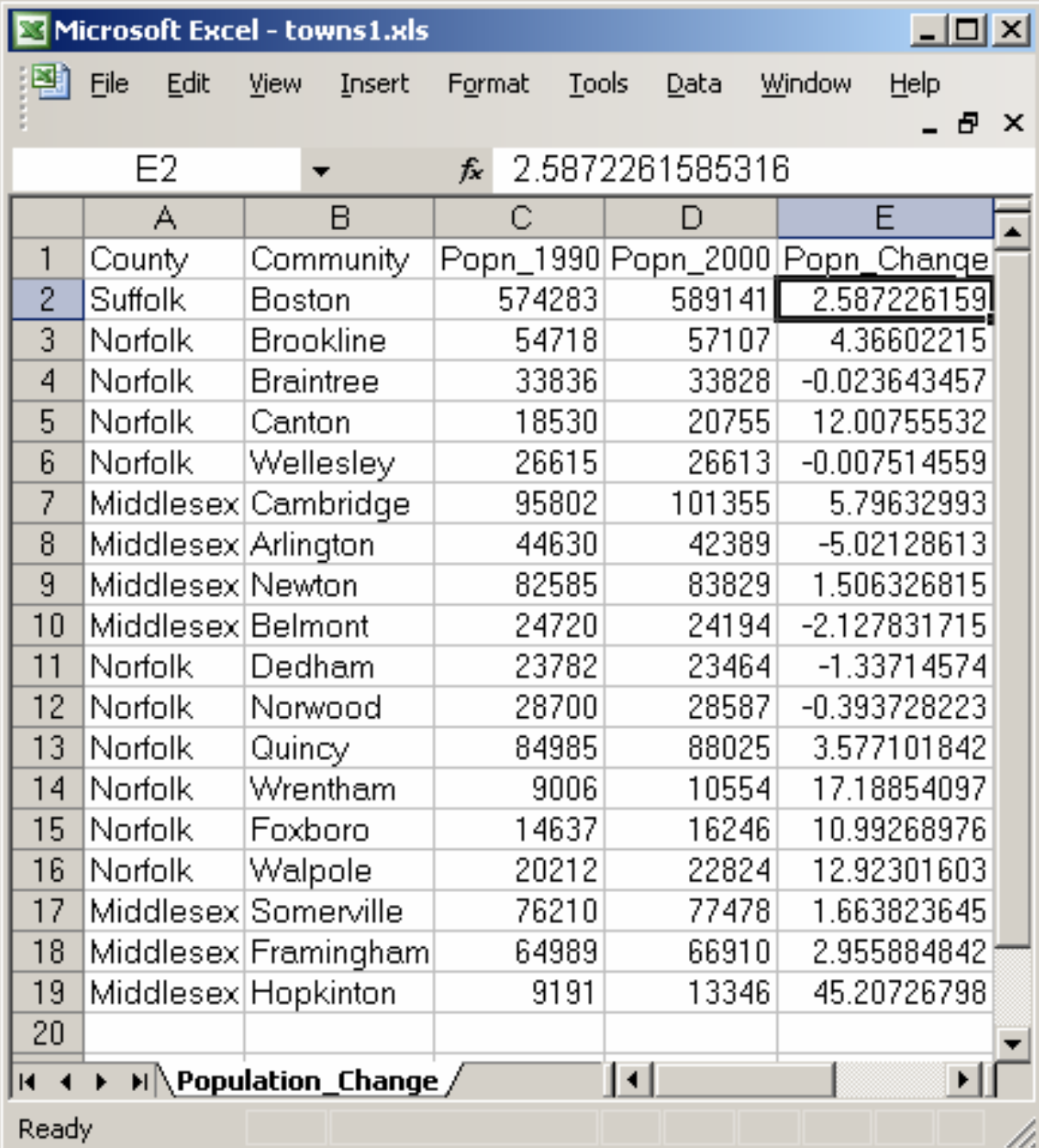
“in order of appearance”
(i.e. variable creation order)

Obs	County	Community	Popn_ 1990	Popn_ 2000	Sq_Miles	Income_ per_Capita	Per_ Pupil_ Exp	Popn_ Sq_Mi	Popn_ Change
1	Suffolk	Boston	574283	589141	48.4	15581	8880	12,172	2.6
2	Norfolk	Brookline	54718	57107	6.8	29044	9002	8,398	4.4
3	Norfolk	Braintree	33836	33828	13.9	18624	6882	2,434	-0.0
4	Norfolk	Canton	18530	20755	18.9	22035	6956	1,098	12.0
5	Norfolk	Wellesley	26615	26613	10.2	32253	8563	2,609	-0.0
6	Middlesex	Cambridge	95802	101355	6.4	19879	12625	15,837	5.8
7	Middlesex	Arlington	44630	42389	5.2	21449	7571	8,152	-5.0
8	Middlesex	Newton	82585	83829	18.1	28840	8978	4,631	1.5
9	Middlesex	Belmont	24720	24194	4.7	26793	6938	5,148	-2.1
10	Norfolk	Dedham	23782	23464	10.5	19045	8034	2,235	-1.3
11	Norfolk	Norwood	28700	28587	10.5	18193	7202	2,723	-0.4
12	Norfolk	Quincy	84985	88025	16.8	17436	6823	5,240	3.6
13	Norfolk	Wrentham	9006	10554	22.2	15856	6127	475	17.2
14	Norfolk	Foxboro	14637	16246	20.1	18329	6370	808	11.0
15	Norfolk	Walpole	20212	22824	20.5	20128	6128	1,113	12.9
16	Middlesex	Somerville	76210	77478	4.1	15179	8359	18,897	1.7
17	Middlesex	Framingham	64989	66910	25.1	20407	7086	2,666	3.0
18	Middlesex	Hopkinton	9191	13346	26.6	22497	5810	502	45.2

Futile result:

The *KEEP* statement does *not* reorder variables.

The variable order in the *SAS* dataset determines the Excel column order.



Microsoft Excel - towns1.xls

File Edit View Insert Format Tools Data Window Help

E2 2.5872261585316

	A	B	C	D	E
1	County	Community	Popn_1990	Popn_2000	Popn_Change
2	Suffolk	Boston	574283	589141	2.587226159
3	Norfolk	Brookline	54718	57107	4.36602215
4	Norfolk	Braintree	33836	33828	-0.023643457
5	Norfolk	Canton	18530	20755	12.00755532
6	Norfolk	Wellesley	26615	26613	-0.007514559
7	Middlesex	Cambridge	95802	101355	5.79632993
8	Middlesex	Arlington	44630	42389	-5.02128613
9	Middlesex	Newton	82585	83829	1.506326815
10	Middlesex	Belmont	24720	24194	-2.127831715
11	Norfolk	Dedham	23782	23464	-1.33714574
12	Norfolk	Norwood	28700	28587	-0.393728223
13	Norfolk	Quincy	84985	88025	3.577101842
14	Norfolk	Wrentham	9006	10554	17.18854097
15	Norfolk	Foxboro	14637	16246	10.99268976
16	Norfolk	Walpole	20212	22824	12.92301603
17	Middlesex	Somerville	76210	77478	1.663823645
18	Middlesex	Framingham	64989	66910	2.955884842
19	Middlesex	Hopkinton	9191	13346	45.20726798
20					

Population_Change

Ready

What *will* work... (Method A)

1. Reorder the variables in the dataset
 - (a) **RETAIN** statement in **DATA** step, *or*
 - (b) **SELECT** statement in **PROC SQL**.

```
*-----*;  
*----- DATA step, RETAIN statement method -----*;
```

```
data pop_change;  
    retain Community Popn_Change Popn_2000 Popn_1990 ;  
    set towns;  
run;  
proc sort data=pop_change;  
    by county descending popn_change;  
run;
```

What *will* work... (Method B)

1. Reorder the variables in the dataset
 - (a) RETAIN statement in DATA step, *or*
 - (b) SELECT statement in PROC SQL.

```
* -----*;  
* ----- Proc SQL, SELECT statement method -----*;
```

```
proc sql;  
  create table pop_change as  
  select community, popn_change, popn_2000, popn_1990, *  
  from work.towns  
  order by county, popn_change desc ;  
quit;
```

Variable Order has changed!

Original TOWNS Dataset

The CONTENTS Procedure

WORK.TOWNS	Observations	18
DATA	Variables	9

Variables in Creation Order

#	Variable	Type	Len
1	County	Char	9
2	Community	Char	10
3	Popn_1990	Num	8
4	Popn_2000	Num	8
5	Sq_Miles	Num	8
6	Income_per_Capita	Num	8
7	Per_Pupil_Exp	Num	8
8	Popn_Sq_Mi	Num	8
9	Popn_Change	Num	8

POP_CHANGE Dataset
after DATA step RETAIN statement,
or SELECT statement in PROC SQL

The CONTENTS Procedure

WORK.POP_CHANGE	Observations	18
DATA	Variables	9

Variables in Creation Order

#	Variable	Type	Len
1	Community	Char	10
2	Popn_Change	Num	8
3	Popn_2000	Num	8
4	Popn_1990	Num	8
5	County	Char	9
6	Sq_Miles	Num	8
7	Income_per_Capita	Num	8
8	Per_Pupil_Exp	Num	8
9	Popn_Sq_Mi	Num	8

FYI: Other Step 1 methods...

SAS Usage Note #8395:

How to reorder variables in a SAS data set

Any of the following statements can be used to change the order of variables in a SAS data set:

ATTRIB, ARRAY, FORMAT, INFORMAT, LENGTH, and RETAIN.

For details, read the entire Note #8395.

<http://support.sas.com/>

Click on “Samples & SAS Notes”.
Enter “8395” in Search box.

What *will* work... EXPORT step

2. Export the dataset to Excel: PROC EXPORT

```
*-----*;  
*----- SAS 9 Code for Proc Export to Excel -----*;  
  
proc export data=pop_change(keep=county community popn_1990 popn_2000 popn_change)  
    outfile="C:\BASUG_demo\towns2.xls"  
    dbms=Excel replace;  
*    sheet="Population Change";  
run;
```

```
*-----*;  
*----- SAS 8 Code for Proc Export to Excel -----*;  
  
proc export data=pop_change(keep=county community popn_1990 popn_2000 popn_change)  
    outfile="C:\BASUG_demo\towns2.xls"  
    dbms=Excel replace;  
run;
```

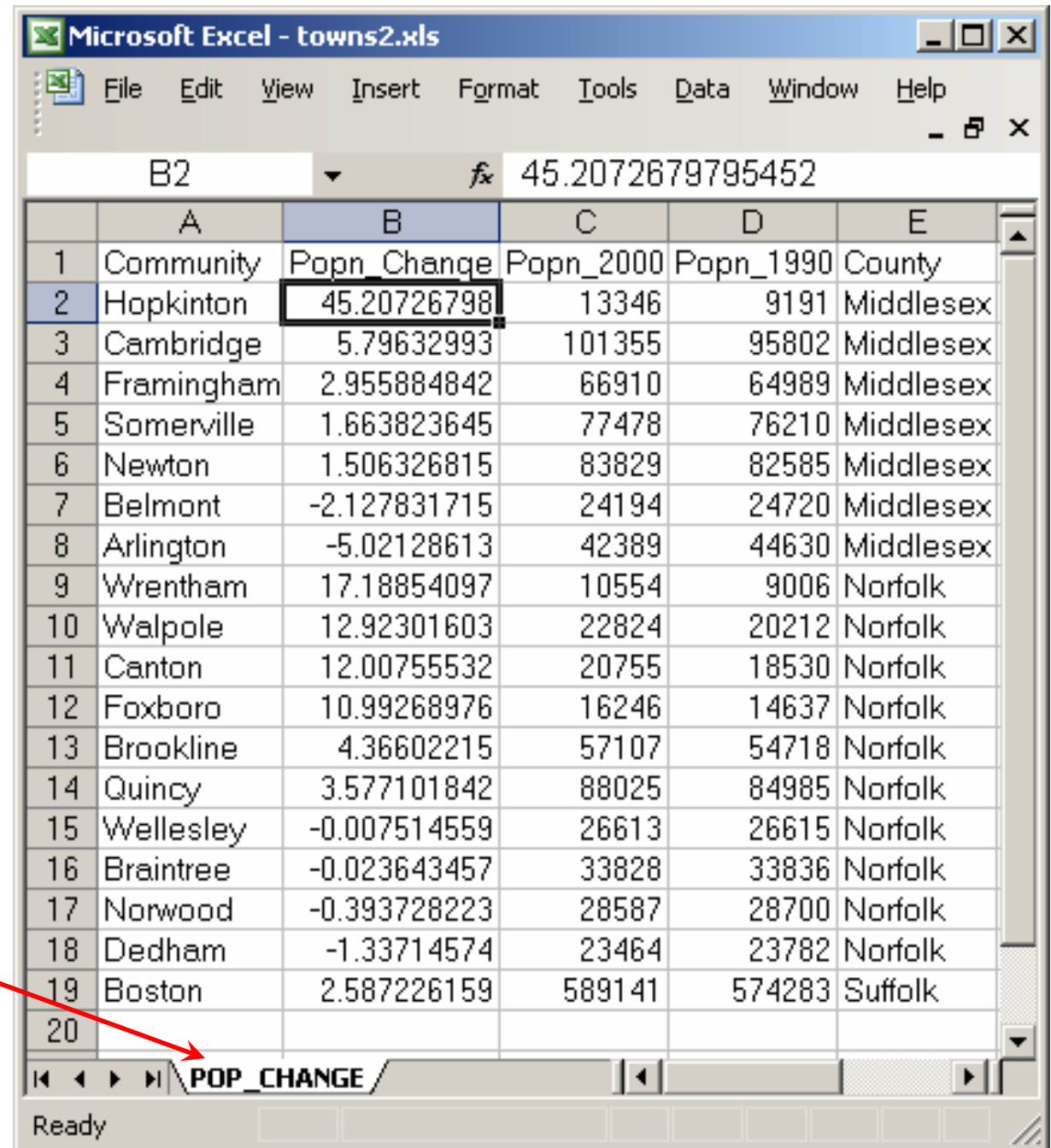
Sheet="*sheetname*" statement became available in SAS 9.

Successful result:

Column order
(i.e. variable order)
matches the *RETAIN*
or *SELECT* statement.

The default Excel
Sheet name is the
SAS dataset name.

(In this case,
POP_CHANGE.)



Microsoft Excel - towns2.xls

File Edit View Insert Format Tools Data Window Help

B2 45.2072679795452

	A	B	C	D	E
1	Community	Popn_Change	Popn_2000	Popn_1990	County
2	Hopkinton	45.20726798	13346	9191	Middlesex
3	Cambridge	5.79632993	101355	95802	Middlesex
4	Framingham	2.955884842	66910	64989	Middlesex
5	Somerville	1.663823645	77478	76210	Middlesex
6	Newton	1.506326815	83829	82585	Middlesex
7	Belmont	-2.127831715	24194	24720	Middlesex
8	Arlington	-5.02128613	42389	44630	Middlesex
9	Wrentham	17.18854097	10554	9006	Norfolk
10	Walpole	12.92301603	22824	20212	Norfolk
11	Canton	12.00755532	20755	18530	Norfolk
12	Foxboro	10.99268976	16246	14637	Norfolk
13	Brookline	4.36602215	57107	54718	Norfolk
14	Quincy	3.577101842	88025	84985	Norfolk
15	Wellesley	-0.007514559	26613	26615	Norfolk
16	Braintree	-0.023643457	33828	33836	Norfolk
17	Norwood	-0.393728223	28587	28700	Norfolk
18	Dedham	-1.33714574	23464	23782	Norfolk
19	Boston	2.587226159	589141	574283	Suffolk
20					

POP_CHANGE

Ready

Creating Multiple Sheets in One Excel Workbook

- Must use SAS 9
- Use a PROC EXPORT for each Sheet.
- Use Sheet="*Sheetname*" statement.
- Can add a Sheet to an existing Excel file, or create a new Excel file.
- dbms=Excel **replace**;
Can overwrite a **Sheet** by the same name.
Can *not* replace an existing Excel **file**.

```
*-----*;  
*----- Multiple Sheets in One EXCEL Workbook -----*;  
*----- SAS 9 only -----*;
```

```
proc sql;  
  create table pop_density as  
  select county, community, popn_sq_mi, popn_2000, sq_miles  
  from work.towns  
  order by county, popn_sq_mi descending;  
quit;  
proc export data=pop_density  
  outfile = "C:\BASUG_demo\towns2.xls"  
  dbms=Excel replace;  
  sheet="Population Density 2000";  
run;
```

In SQL, it's officially
desc - but Proc SQL in
SAS also recognizes
descending

```
proc sql;  
  create table educ_spend as  
  select community, per_pupil_exp, income_per_capita  
  from work.towns  
  order by per_pupil_exp;  
quit;  
proc export data=educ_spend  
  outfile = "C:\BASUG_demo\towns2.xls"  
  dbms=Excel replace;  
  sheet="Education Spending - 2000";  
run;
```

Success:

Sheets were
added to
existing file
(Towns2.XLS)

Sheet Name
Nuances:

Note that
SAS replaces
spaces and
some
characters
with "_"

Microsoft Excel - towns2.xls

File Edit View Insert Format Tools Data Window Help

B2 5810

	A	B	C	D	E	F
1	Community	Per Pupil Exp	Income_per_Capita			
2	Hopkinton	5810	22497			
3	Wrentham	6127	15856			
4	Walpole	6128	20128			
5	Foxboro	6370	18329			
6	Quincy	6823	17436			
7	Braintree	6882	18624			
8	Belmont	6938	26793			
9	Canton	6956	22035			
10	Framingham	7086	20407			
11	Norwood	7202	18193			
12	Arlington	7571	21449			
13	Dedham	8034	19045			
14	Somerville	8359	15179			
15	Wellesley	8563	32253			
16	Boston	8880	15581			
17	Newton	8978	28840			
18	Brookline	9002	29044			
19	Cambridge	12625	19879			
20						

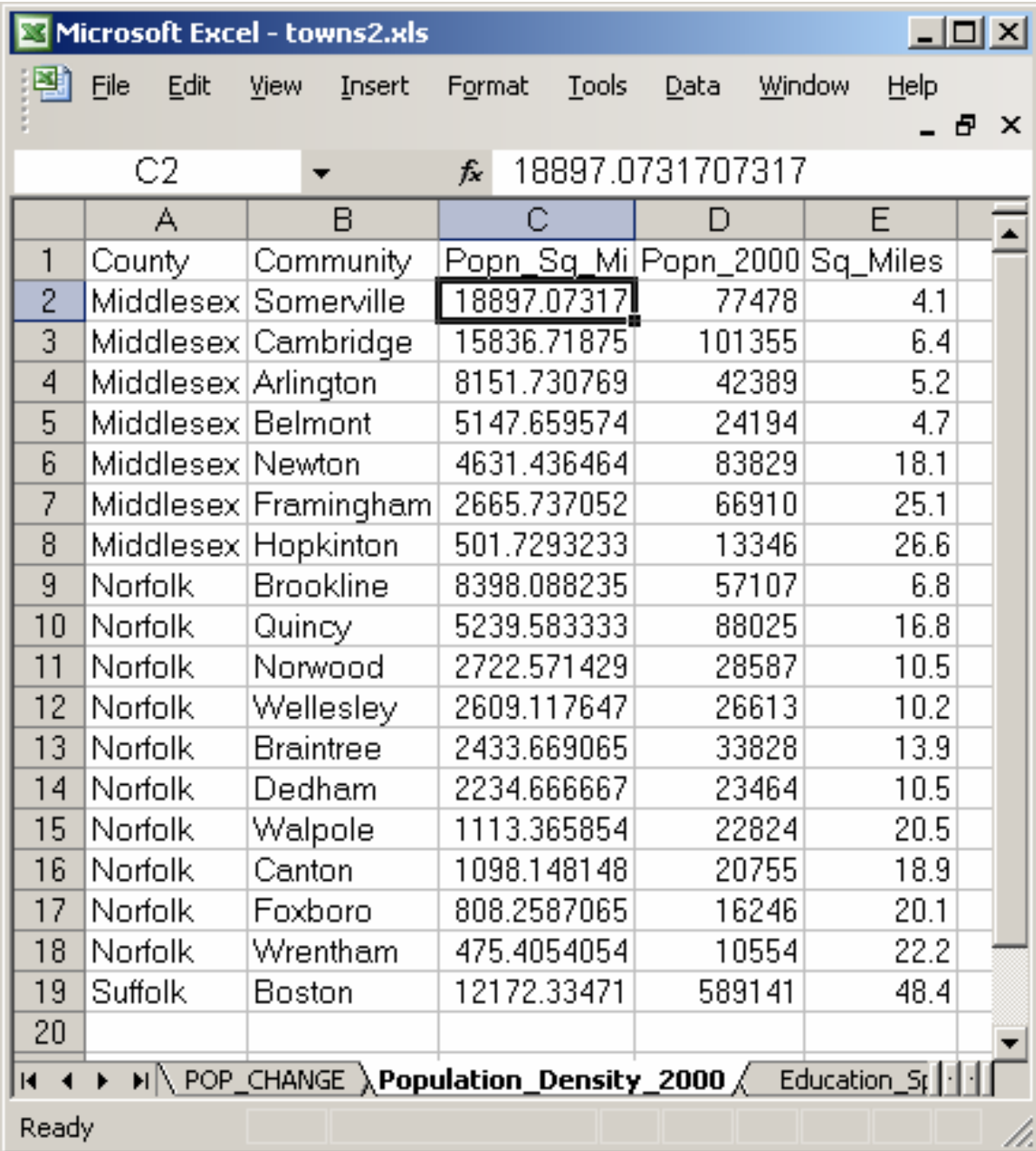
POP_CHANGE Population_Density_2000 Education_Spending 2000

Ready

Success (continued):

Same file
(Towns2.XLS)

Showing the
Population Density
sheet



	A	B	C	D	E
	County	Community	Popn_Sq_Mi	Popn_2000	Sq_Miles
2	Middlesex	Somerville	18897.07317	77478	4.1
3	Middlesex	Cambridge	15836.71875	101355	6.4
4	Middlesex	Arlington	8151.730769	42389	5.2
5	Middlesex	Belmont	5147.659574	24194	4.7
6	Middlesex	Newton	4631.436464	83829	18.1
7	Middlesex	Framingham	2665.737052	66910	25.1
8	Middlesex	Hopkinton	501.7293233	13346	26.6
9	Norfolk	Brookline	8398.088235	57107	6.8
10	Norfolk	Quincy	5239.583333	88025	16.8
11	Norfolk	Norwood	2722.571429	28587	10.5
12	Norfolk	Wellesley	2609.117647	26613	10.2
13	Norfolk	Braintree	2433.669065	33828	13.9
14	Norfolk	Dedham	2234.666667	23464	10.5
15	Norfolk	Walpole	1113.365854	22824	20.5
16	Norfolk	Canton	1098.148148	20755	18.9
17	Norfolk	Foxboro	808.2587065	16246	20.1
18	Norfolk	Wrentham	475.4054054	10554	22.2
19	Suffolk	Boston	12172.33471	589141	48.4
20					

Overwrite an Existing Excel Sheet

Let's re-do the "Population Density 2000" sheet,
using the REPLACE option.

This time ignore County and sort in ascending order.

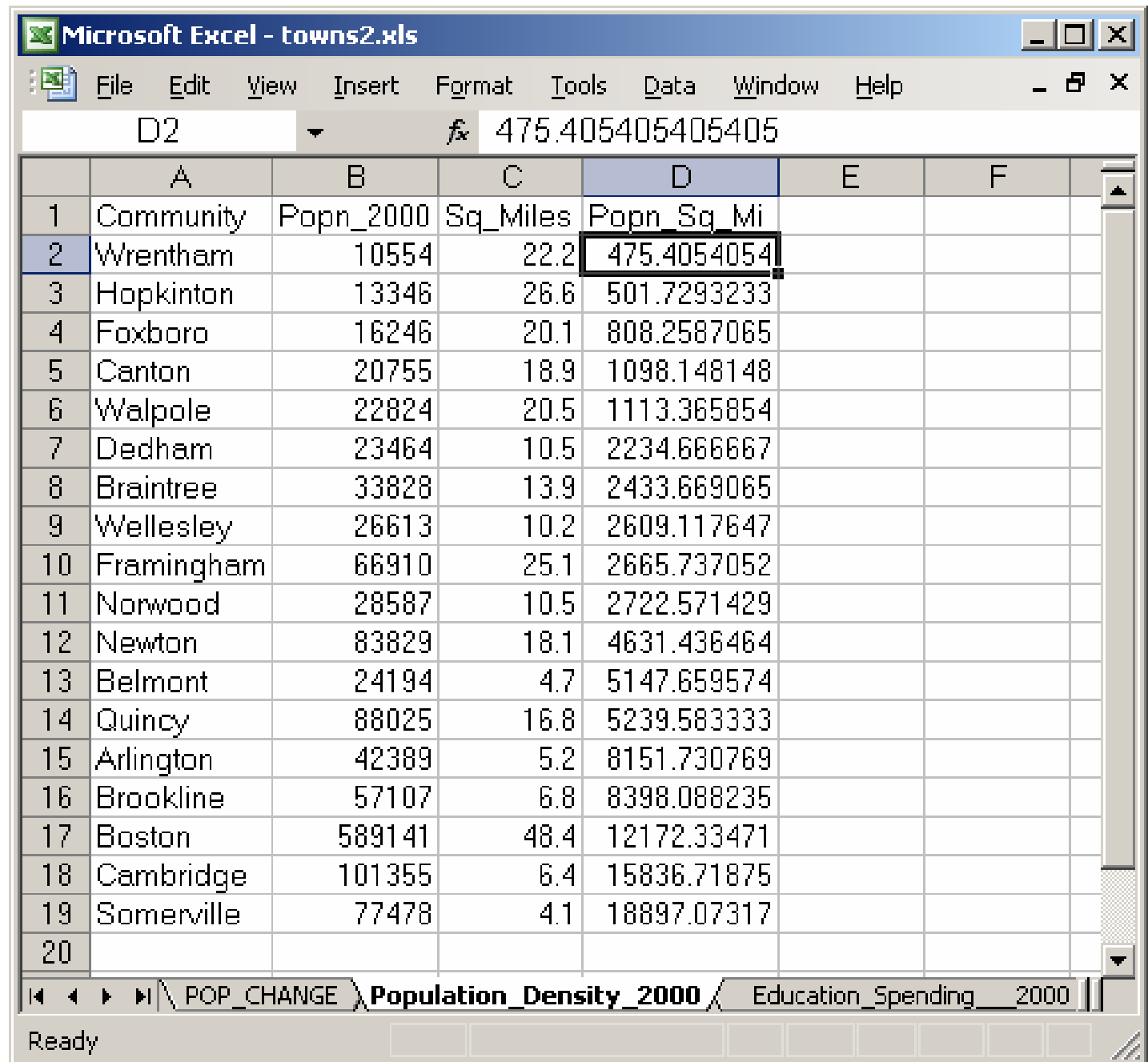
```
*-----*;  
*----- Overwrite an Existing EXCEL Worksheet -----*;  
*----- SAS 9 only -----*;
```

```
proc sql;  
  create table popn_density as  
  select community, popn_2000, sq_miles, popn_sq_mi  
  from work.towns  
  order by popn_sq_mi;  
quit;  
proc export data=popn_density  
  outfile = "C:\BASUG_demo\towns2.xls"  
  dbms=Excel replace;  
  sheet="Population Density 2000";  
run;
```

Replaced:

This new
version of
the sheet is
sorted by
Population per
Square Mile.

County is
now omitted.



Microsoft Excel - towns2.xls

File Edit View Insert Format Tools Data Window Help

D2 = 475.405405405405

	A	B	C	D	E	F
1	Community	Popn_2000	Sq_Miles	Popn_Sq_Mi		
2	Wrentham	10554	22.2	475.4054054		
3	Hopkinton	13346	26.6	501.7293233		
4	Foxboro	16246	20.1	808.2587065		
5	Canton	20755	18.9	1098.148148		
6	Walpole	22824	20.5	1113.365854		
7	Dedham	23464	10.5	2234.666667		
8	Braintree	33828	13.9	2433.669065		
9	Wellesley	26613	10.2	2609.117647		
10	Framingham	66910	25.1	2665.737052		
11	Norwood	28587	10.5	2722.571429		
12	Newton	83829	18.1	4631.436464		
13	Belmont	24194	4.7	5147.659574		
14	Quincy	88025	16.8	5239.583333		
15	Arlington	42389	5.2	8151.730769		
16	Brookline	57107	6.8	8398.088235		
17	Boston	589141	48.4	12172.33471		
18	Cambridge	101355	6.4	15836.71875		
19	Somerville	77478	4.1	18897.07317		
20						

POP_CHANGE Population_Density_2000 Education_Spending_2000

Ready

What if I'm Running SAS 8 ?

- Only one Sheet per Excel file.
Sorry, you'll have to deal with more files!
- Sheet name will automatically be the SAS dataset name.
- Use a PROC EXPORT for each dataset
Specify a different Outfile="*Filename*" for each.
- dbms=Excel **replace**;
Overwrites an existing Excel **file** if the Outfile name already exists.

SUMMARY

You *can* have it your way... in two steps!

1. Reorder the variables in the dataset
 - (a) RETAIN statement in DATA step, *or*
 - (b) SELECT statement in PROC SQL.
2. Export the dataset to Excel:
PROC EXPORT

Advantage of PROC SQL:
you can sort without a PROC SORT!

Acknowledgement

Thanks to *Vanessa Hayden* for the PROC SQL variable reordering tip that led to this presentation.

You never know what new tip you'll pick up at a BASUG meeting!

Mark A. Martin, Siemens Healthcare Diagnostics Inc.
mark.martin@siemens.com