

Start with SAS with a few quick easy steps

Zeke Torres, 1016inc.com, Chicago, IL

ABSTRACT

Learn about SAS and basic Data step processing and SAS Procedures. From reading raw data and other SAS datasets to printing, summarizing, reporting, sorting, formats and tabulates. This presentation is ideal for someone who wishes to learn more about SAS and would like a simple introduction to it. The presentation will provide users with different examples of 'projects' or 'problems' that are practical to everyday situations in a variety of businesses and various ways the SAS system can facilitate solving them.

AUTHOR BIOGRAPHY

Zeke Torres is a consultant working in the Chicago area. He has worked with the SAS System for over 10 years and is a certified Base SAS programmer. Zeke has written and presented papers on SAS to fellow programmers at client sites, user conferences (regional and local). Currently Zeke is involved with various SAS User groups in the region and the leader of the Chicago Area SAS Users Group (WCSUG.ORG).

INTRODUCTION

WHAT IS SAS?

SAS is an entire set of integrated software systems for data analysis, processing and reporting.

Some of the common components are:

- Base SAS
- SAS/STAT
- SAS/GRAPH
- SAS/FSP
- SAS/IML
- SAS/ACCESS
- SAS/CONNECT
- SAS/AF

Note:

Not all products are available on all operating systems. All of the above are trademarks of SAS Institute, Cary NC.

Basically SAS is a full featured computer system. It is a 3GL (based on C). Yet it also is essentially its own language.

SAS Code or language is executable on practically any operating system. Minor modifications would be required to specify file locations/sources/output and file types but the actual SAS code/logic in your program can remain the same across the different systems.

You can code on your PC and the code would work on a Mainframe or UNIX system. Just tell SAS where the file(s) are/will be!

SAS provides a full programming language to:

- Calculate mathematical and statistical functions
- Move and manipulate numeric and character data
- Select and transform data
- Rearrange the order of observations (sorting)
- Break datasets into smaller pieces (subsetting)
- Stacking Datasets (concatenation)
- Combining related data values (merging)

Base SAS provides statistical analysis to:

- Calculate correlations
- Univariate statistics and distributions

SAS reporting features provide:

- Charts, Plots, and Graphs using printer graphics
- Summary reports providing totals at several levels
- Tabular reports and mailing labels
- Schedule and summary calendars
- User written custom reports

SAS reducing and summarizing procedures provide:

- Summary statistics and tables
- Descriptive statistics (mean, std, sum, etc)
- Frequencies and cross tabulations

CREATING SAS 'FILES'

Input to SAS can come from:

- Raw files
- NON-SAS database management systems (DBMS)
- Data lines included in the SAS programs
- SAS Datasets (tables) located on remote computers
- SAS Datasets (tables) located on your local computer

COMMAND 'SAS'

SAS has always impressed me by the many ways it is flexible within a variety of operating systems. SAS can run on just about any standard business operating system. Additionally within the SAS code you working on you can execute commands directly to the operating system. With SAS components like AF and WEB/AF, SAS Access and SAS Connect, the ability to create some wonderful applications, solutions is limited only by your imagination, time and budget for the project your on. **With the right examples you can command SAS to ...**

BASIC INPUT

READING IN FILES - TEXT FILES

Lets first look at our source data. This is in a text file.

Raw Data File 1: [class_sample1.sas](#)

1	Bill	I	3	791.23	19.32	171.92	60630	RM	11/23/2003	01/03/23
2	Tim	I	3	4929.44	92.44	106.24	60601	OH	11/23/2003	02/01/44
3	Pat	I	3	954.31	45.13	104.63	60601	OH	11/23/2003	03/01/31
4	John	I	6	273.14	37.41	83.71	60647	LA	11/23/2003	04/01/14
5	JANE	I	3	591.73	19.37	61.97	60601	ES	11/23/2003	05/01/73
6	Linda	I	2	778.91	87.19	68.79	60618	IL	11/23/2003	06/01/91
7	Rose	S	1	3912.72	21.27	52.17	60630	IL	11/23/2003	07/01/72
8	Alice	I	2	5886.13	68.31	56.81	60601	IN	11/23/2003	08/01/13
9	ED	I	1	737.84	73.48	57.38	60601	HI	11/23/2003	09/01/84
10	DANIEL	S	5	991.35	19.35	41.35	60647	IA	11/23/2003	10/01/35
11	Jim	I	6	6928.96	82.69	48.29	60601	HI	11/23/2003	11/01/96
12	William	I	17	254.27	45.72	34.52	60618	RM	11/23/2003	12/01/27
13	Andrew	S	5	573.59	37.85	33.75	60630	OH	11/23/2003	01/01/59
14	Benjamin	S	3	791.79	19.97	31.97	60601	OK	11/23/2003	02/01/79
15	Janet	S	4	978.91	87.19	38.79	60601	LA	11/23/2003	03/01/91
16	Janis Lee	S	3	3512.92	21.29	32.19	60647	ES	11/23/2003	04/01/92
17	Joy	I	1	8086.23	68.32	26.82	60601	RM	11/23/2003	05/01/23
18	Lois	I	1	837.54	73.48	27.38	60618	RM	11/23/2003	01/01/54
19	LOUIS	S	1	499.75	99.67	29.97	60630	OH	11/23/2003	02/01/75
20	Sarah	I	8	4029.96	92.69	29.29	60601	OK	11/23/2003	03/01/96
21	Tom	I	2	862.51	55.15	22.55	60601	LA	11/23/2003	04/01/51
22	Sanya	S	3	575.72	57.27	25.77	60647	ES	11/23/2003	05/01/72
23	Ian	S	2	497.93	79.39	17.99	60601	IL	11/23/2003	06/01/93
24	Ira	I	4	379.94	97.49	19.79	60618	IL	11/23/2003	07/01/94
25	Mary	I	1	3315.25	51.52	15.12	60630	OH	11/23/2003	09/01/25
26	Steve	S	2	7086.56	68.65	16.85	60601	HI	11/23/2003	09/01/56
27	Deke	S	10	637.77	79.77	17.97	60601	IA	11/23/2003	10/01/77
28	TIM	S	2	9943.92	34.29	19.49	60647	HI	11/23/2003	11/01/92

Raw Data File 2: [class_sample2.sas](#)

1	Computer	I	12	791.23	19.32	171.92	60630	RM	11/23/2003	01/03/23
2	Printer	I	3	4929.44	92.44	106.24	60601	OH	11/23/2003	02/01/44
3	Paper	I	5	954.31	45.13	104.63	60601	OH	11/23/2003	03/01/31
4	Lapcok	I	6	273.14	37.41	83.71	60647	LA	11/23/2003	04/01/14
5	MousePad	I	3	591.73	19.37	61.97	60601	ES	11/23/2003	05/01/73
6	Head Phones	I	2	778.91	87.19	68.79	60618	IL	11/23/2003	06/01/91
7	Clock	S	1	3912.72	21.27	52.17	60630	IL	11/23/2003	07/02/72
8	Coat	I	2	5886.13	68.31	56.81	60601	IN	11/23/2003	08/01/13
9	Pants	I	1	737.84	73.48	57.38	60601	HI	11/23/2003	09/01/84
10	Cash	S	5	991.35	19.35	41.35	60647	IA	11/23/2003	10/01/35
11	Piston	I	6	6928.96	82.69	48.29	60601	HI	11/23/2003	11/06/96
12	Spark Plug	I	17	254.27	45.72	34.52	60618	RM	11/23/2003	12/07/27
13	Oil Filter	S	5	573.59	37.85	33.75	60630	OH	11/23/2003	01/08/59
14	10w40 Oil	S	3	791.79	19.97	31.97	60601	OK	11/23/2003	02/09/79
15	Pagebleed	I	3	978.91	87.19	38.79	60601	LA	11/23/2003	03/01/91
16	Plasma Screen	S	3	3512.92	21.29	32.19	60647	ES	11/23/2003	04/02/92
17	Light Bulb	I	2	8086.23	68.32	26.82	60601	RM	11/23/2003	05/03/23
18	Blank CD's	I	1	837.54	73.45	27.35	60618	RM	11/23/2003	01/04/54
19	Tires	S	1	499.75	99.57	29.97	60630	OH	11/23/2003	02/05/75
20	Hlas	I	8	4029.96	92.69	29.29	60601	OK	11/23/2003	03/06/96
21	Bubcaps	S	1	862.51	25.15	22.55	60601	LA	11/23/2003	04/01/51
22	Computer	S	3	575.72	57.27	25.77	60647	ES	11/23/2003	05/02/72
23	Computer	S	2	497.93	79.39	17.99	60601	IL	11/23/2003	06/03/93
24	Computer	I	4	379.94	97.49	19.79	60618	IL	11/23/2003	07/04/94

These files have 'fixed' columns. Nothing separates the columns. The layout is:
Name, Department, Years, Sales Amount,
Cellphoneexpense, rent, Zip, State, Date1, Date2
We should always have some layout for data we wish to read.
SAS provides us with different ways to read data. Even if the data was in CSV, TXT, HTML, Excel.

READING IN THE SOURCE DATA

Now lets look at different ways we can read this data.

SAS Code File A: [class_code_readin1.sas](#)

If we look at our first source code we can see one method of input for the file.

```
1 **** READING FILES 001;
2 *** libname rawin 'c:\class_sample';
3 filename rawin '\\swmfs01\%$torref\class_sample.sas';
4
5 data spiegy_1;
6   infile rawin;
7   input
8     First_name      $1-10
9     Division        $20
10    Years           23-24
11    Sales           25-35
12    CellPhoneExpense 36-44
13    Rent            45-53
14    ;
15  run;
16
17 proc print data=spiegy_1 uniform;
18 run;
```

Let's run this code and see what we get...

```
SAS - [Log - (Untitled)]
1 **** READING FILES 001;
2 *** libname rawin 'c:\class_sample';
3 filename rawin '\\swmfs01\%$torref\class_sample.sas';
4
5 data spiegy_1;
6   infile rawin;
7   input
8     First_name      $1-10
9     Division        $20
10    Years           23-24
11    Sales           25-35
12    CellPhoneExpense 36-44
13    Rent            45-53
14    ;
15  run;

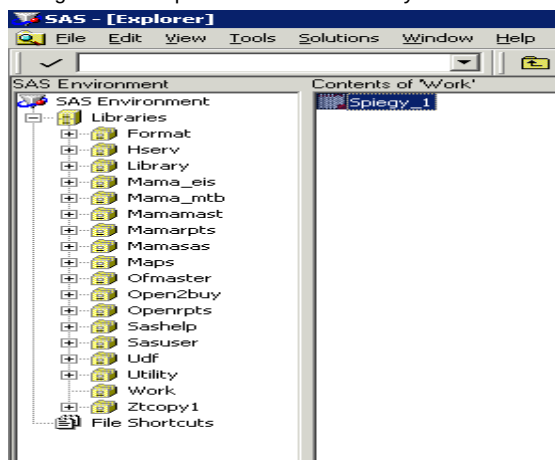
NOTE: The infile RAWIN is:
File Name=\\swmfs01\%$torref\class_sample.sas,
RECFM=V,LRECL=256

NOTE: 28 records were read from the infile RAWIN.
The minimum record length was 84.
The maximum record length was 84.
NOTE: The data set WORK.SPIEGY_1 has 28 observations and 6
variables.
NOTE: Compressing data set WORK.SPIEGY_1 increased size by
100.00 percent.
Compressed is 2 pages; un-compressed would require 1
pages.
NOTE: DATA statement used:
real time      0.18 seconds
cpu time       0.04 seconds

16
17 proc print data=spiegy_1 uniform;
18 run;

NOTE: There were 28 observations read from the data set
WORK.SPIEGY_1.
NOTE: PROCEDURE PRINT used:
real time      0.11 seconds
cpu time       0.07 seconds
```

Let's actually take a look at the file in a SAS Dataset.
First let's get to the Explorer then 'Work' Library.



Next Click on the SAS Dataset we are working with.

	First_name	Division	Years	Sales	CellPhoneExpense	Rent
1	Sally	I	12	791.23	19.32	171.92
2	Tim	I	3	4928.44	82.44	128.24
3	WILL	I	5	954.31	45.13	104.53
4	John	I	6	273.14	37.41	83.71
5	JANE	I	3	591.73	19.37	61.97
6	Linda	I	2	778.91	87.19	68.79
7	Rose	S	1	3912.72	21.27	52.17
8	Alice	I	2	9586.13	68.91	56.81
9	ED	I	1	737.84	73.48	57.38
10	DANIEL	S	5	991.35	19.53	41.93
11	Jim	I	6	6328.96	82.69	48.29
12	William	I	17	254.27	45.72	34.52
13	Andrew	S	5	573.58	37.85	33.75
14	Benjamin	S	3	791.79	19.97	31.97
15	Janet	S	4	978.91	87.19	38.79
16	Jennifer	S	3	3512.92	21.29	32.19
17	Joy	I	2	5086.23	68.32	26.82
18	Lou	I	1	937.54	73.45	27.35
19	LOUIS	S	1	499.75	99.57	29.97
20	Sarah	I	8	4029.96	92.69	29.29
21	Tom	S	1	952.51	25.15	22.55
22	Sanga	S	3	575.72	57.27	25.77
23	Ian	S	2	497.93	79.39	17.99
24	Ina	I	4	379.94	97.49	19.79
25	Mary	I	1	3315.25	51.52	15.12
26	Steve	S	2	7086.56	68.65	16.85
27	Zeke	S	10	637.77	73.77	17.37
28	TIM	S	2	9943.92	34.29	13.49

SAS Code File B: [class_code_readin2.sas](#)

```
1 **** READING FILES 002;
2 data spiegy_1;
3   infile rawin;
4   input
5     @01 First_name      $10.
6     @20 Division        $1.
7     @23 Years           2.
8     @25 Sales           11.2
9     @36 CellPhoneExpense 9.2
10    @45 Rent            9.2
11    ;
12    label First_name="Employee";
13    annual_sales=sales*12;
14
15  run;
16
17 proc print data=spiegy_1 label uniform;
18 run;
```

	1	2	3	4	5	6	7	8	9	
	123456789012345									

SAS Code File P: class_code_printto1.sas

```

1  PROC PRINTTO
2  FILENAME TEST PRINTER 'LPT1: ';
3  PROC PRINTTO LOG=TEST;
4  RUN;
5
6  FILENAME MYFILE 'C:\MYDIR\MYLOG.LOG';
7  PROC PRINTTO LOG=MYFILE;
8  RUN;
9
10 PROC PRINTTO PRINT=MYPRINT;
11 RUN;
12
13 PROC PRINTTO PRINT='LPT1: ';
14 RUN;
15
16
17 PROC PRINTTO;
18 RUN;

```

MANIPULATE VARIABLES

COMMON VARIABLE FUNCTIONS

SAS FUNCTIONS: Character/Date and Time Functions

SAS Code File C: class_code_varmanip1.sas

```

1  *** Basic Variable Manipulation 001 ;
2  data spiegy_1;
3  infile rawin;
4  input
5
6      @01 First_name $10.
7      @20 Division $1.
8      @23 Years 2.
9      @25 Sales 11.2
10     @36 CellPhoneExpense 9.2
11     @45 Rent 9.2
12
13     label First_name="Employee";
14     annual_sales=sales*12;
15     New_name=upcase(first_name);
16     if division="I" then division="Information Systems"; else
17     if division="S" then division="Sales Department";
18
19 run;
20
21 proc print data=spiegy_1 label uniform;
22 run;

```

CHARACTER

Employee	Division	Years	Sales	CellPhoneExpense	Rent	annual_sales	New_name
1 Sally	I	12	791.23	19.32	171.92	9494.76	SALLY
2 Tim	I	3	4928.44	82.44	128.24	59141.28	TIM
3 WILL	I	5	954.31	45.13	104.53	11451.72	WILL
4 John	I	6	273.14	37.41	83.71	3277.68	JOHN
5 JANE	I	3	591.73	19.37	61.97	7100.76	JANE
6 Linda	I	2	778.91	87.19	68.79	9346.92	LINDA
7 Rose	S	1	3912.72	21.27	52.17	46952.64	ROSE
8 Alice	I	2	5586.13	68.21	56.81	67032.56	ALICE
9 ED	I	1	737.84	73.48	57.38	8854.08	ED
10 DANIEL	S	5	991.35	19.53	41.93	11896.2	DANIEL
11 Jim	I	6	6528.96	82.69	48.29	83147.52	JIM
12 William	I	17	254.27	45.72	34.52	3051.24	WILLIAM
13 Andrew	S	5	573.58	37.85	33.75	6882.96	ANDREW
14 Benjamin	S	3	791.79	19.97	31.97	9501.48	BENJAMIN
15 Janet	S	4	978.91	87.19	38.79	11746.92	JANET
16 Jennifer	S	3	3512.92	21.29	32.19	42195.04	JENNIFER
17 Joy	I	2	5086.23	68.32	26.82	61034.76	JOY
18 Lou	I	1	837.54	73.45	27.35	10050.48	LOU
19 LOUIS	S	1	499.75	99.57	29.97	5997	LOUIS
20 Sarah	I	8	4029.96	92.69	29.29	48359.52	SARAH
21 Tom	S	1	962.51	25.15	22.95	11430.12	TOM
22 Sarva	S	3	575.72	57.27	25.77	6908.64	SANYA
23 Ian	S	2	497.93	79.39	17.99	5975.16	IAN
24 Ina	I	4	379.94	97.49	13.79	4559.28	INA
25 May	I	1	3315.25	51.52	15.12	39783	MARY
26 Steve	S	2	7086.56	68.65	16.85	85038.72	STEVE
27 Zeke	S	10	637.77	73.77	17.37	7653.24	ZEKE
28 TIM	S	2	9943.92	34.29	13.49	119327.04	TIM

SAS Code File D: class_code_varmanip2.sas

```

1  **** Basic Variable Manipulation 002 ;
2  data spiegy_1;
3  infile rawin;
4  length division $20.;
5  input
6
7      @01 First_name $10.
8      @20 Division $1.
9      @23 Years 2.
10     @25 Sales 11.2
11     @36 CellPhoneExpense 9.2
12     @45 Rent 9.2
13
14     label First_name="Employee";
15     annual_sales=sales*12;
16     New_name=upcase(first_name);
17     if division="I" then division="Information Systems"; else
18     if division="S" then division="Sales Department";
19
20 run;
21
22 proc print data=spiegy_1 label uniform;
23 run;

```

Employee	Division	Years	Sales	CellPhoneExpense	Rent	annual_sales	New_name
1 Sally	I	12	791.23	19.32	171.9	9494.76	SALLY
2 Tim	I	3	4928.44	82.44	128.2	59141.28	TIM
3 WILL	I	5	954.31	45.13	104.5	11451.72	WILL
4 John	I	6	273.14	37.41	83.71	3277.68	JOHN
5 JANE	I	3	591.73	19.37	61.97	7100.76	JANE
6 Linda	I	2	778.91	87.19	68.79	9346.92	LINDA
7 Rose	S	1	3912.72	21.27	52.17	46952.64	ROSE
8 Alice	I	2	5586.13	68.21	56.81	67032.56	ALICE
9 ED	I	1	737.84	73.48	57.38	8854.08	ED
10 DANIEL	S	5	991.35	19.53	41.93	11896.2	DANIEL
11 Jim	I	6	6528.96	82.69	48.29	83147.52	JIM
12 William	I	17	254.27	45.72	34.52	3051.24	WILLIAM
13 Andrew	S	5	573.58	37.85	33.75	6882.96	ANDREW
14 Benjamin	S	3	791.79	19.97	31.97	9501.48	BENJAMIN
15 Janet	S	4	978.91	87.19	38.79	11746.92	JANET
16 Jennifer	S	3	3512.92	21.29	32.19	42195.04	JENNIFER
17 Joy	I	2	5086.23	68.32	26.82	61034.76	JOY
18 Lou	I	1	837.54	73.45	27.35	10050.48	LOU
19 LOUIS	S	1	499.75	99.57	29.97	5997	LOUIS
20 Sarah	I	8	4029.96	92.69	29.29	48359.52	SARAH
21 Tom	S	1	962.51	25.15	22.95	11430.12	TOM
22 Sarva	S	3	575.72	57.27	25.77	6908.64	SANYA
23 Ian	S	2	497.93	79.39	17.99	5975.16	IAN
24 Ina	I	4	379.94	97.49	13.79	4559.28	INA
25 May	I	1	3315.25	51.52	15.12	39783	MARY
26 Steve	S	2	7086.56	68.65	16.85	85038.72	STEVE
27 Zeke	S	10	637.77	73.77	17.37	7653.24	ZEKE
28 TIM	S	2	9943.92	34.29	13.49	119327.04	TIM

SAS Code File E: class_code_varmanip3.sas

NUMERIC

PROC'S YOU SHOULD KNOW ABOUT

PROC FREQ

SAS Code File J: class_code_freq1.sas

```

1  PROC FREQ DATA=SPIEGY_1;
2  WHERE FIRST_NAME IN : ("S");
3  TITLE 'REVIEW OF THOSE WITH FIRST NAME STARTING WITH S';
4  TABLE YEARS*DIVISION;
5  TABLE DIVISION;
6  RUN;
7
8  PROC FREQ DATA=SPIEGY_1;
9  TABLE YEARS/NOCOL NOROW NOPERCENT;
10 RUN;
11
12 PROC FREQ DATA=SPIEGY_1;
13 TABLE DIVISION /NOCOL NOROW NOPERCENT;
14 TABLE FIRST_NAME /NOCOL NOROW NOPERCENT;
15 RUN;

```

PROC SORT

SAS Code File I: class_code_sort1.sas

```

1  PROC SORT
2  PROC SORT DATA=SPIEGY_1 OUT= SPIEGY_2 NODUPKEY;
3  BY FIRST_NAME;
4  RUN;
5
6  PROC SORT DATA=SPIEGY_2;
7  BY DESCENDING SALES;
8  RUN;
9
10 PROC SORT DATA=SPIEGY_2 NODUPKEY;
11 BY YEARS;
12 RUN;

```

PROC SUMMARY/MEAN

SAS Code File K: class_code_summary1.sas

```

1  PROC SUMMARY DATA=SPIEGY_1;
2  CLASS division;
3  VAR sales rent;
4  OUTPUT OUT=SPIEGY_3 SUM=;
5  RUN;
6
7  PROC SUMMARY DATA=SPIEGY_2;
8  WHERE years=10;
9  CLASS years;
10 VAR sales rent;
11 OUTPUT OUT=SPIEGY_3 SUM=;
12 RUN;
13
14 PROC SUMMARY DATA=SPIEGY_3 MISSING;
15 CLASS division;
16 VAR sales rent;
17 OUTPUT OUT=SPIEGY_4
18 SUM(sales)=sales_total
19 SUM(rent)=rent_total
20 mean(sales)=sales_avg
21 mean(rent)=rent_avg
22 ;
23 RUN;
24

```

PROC FORMAT

SAS Code File L: class_code_format1.sas

```

1  PROC FORMAT;
2  VALUE $divis 'I'='Info Services'
3              'S'='Sales'
4  ;
5  RUN;

```

PROC COPY

SAS Code File M: class_code_copy1.sas

```

1  PROC COPY INLIB =INPUT02 OUTLIB=OUTPT02;
2  RUN;
3
4  LIENAME SASDATAL XPORT 'C:\DATA\SPIEGEL_PORT.DAT';
5  PROC COPY IN=SASDATAL OUT=WORK ;
6  RUN;

```

PROC DOWN(UP)LOAD

PROC COPY

SAS Code File M: [class_code_copy1.sas](#)

```

1  PROC COPY INLIB =INPUT02 OUTLIB=OUTPT02;
2  RUN;
3
4  LIENAME SASDATAL XPORT 'C:\DATA\SPIEGEL_PORT.DAT';
5  PROC COPY IN=SASDATAL OUT=WORK ;
6  RUN;

```

PROC TABULATE

SAS Code File O: class_code_tabulate1.sas

CONCLUSION

REFERENCES

ACKNOWLEDGMENTS

The author would like to thank the following individuals for thier invaluable assistance in the preparation, and technical review of this paper.

Dana Laskowski
Transamerica
Hoffman Estates IL

AUTHORS CONTACT INFORMATION

Your comments and questions are valued and encouraged.
Contact the author at:



Ezequiel 'Zeke' Torres
1016Inc.com
Chicago, Illinois 60630
Work Phone: (773)467-8492
Fax: (773)467-8492
Email: info@1016inc.com
Web: www.1016inc.com

OTHER USEFUL SAS INFO

TEXTPAD / ULTRAEDIT

Useful editors not just for TEXT but also for SAS or any other syntax.

www.textpad.com
www.ultraedit.com

RECOMMENDED BOOKS FOR SAS

Rick Aster – Professional SAS Programming Secrets

SAS HELP RESOURCES

SAS.COM

Call with your site number and problem description.

SAS LEARNING EDITION