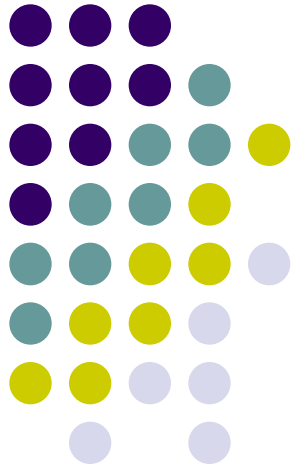
Introduction to SAS Programming

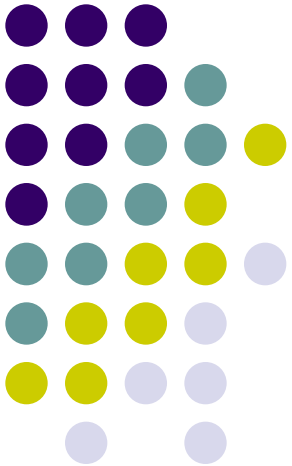
Christina L. Ughrin
Statistical Software Consulting

Some notes pulled from SAS Programming I: Essentials Training

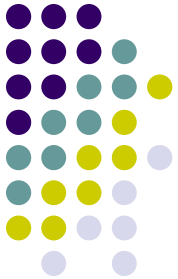


SAS Datasets

Examining the structure of SAS Datasets



SAS Data Sets

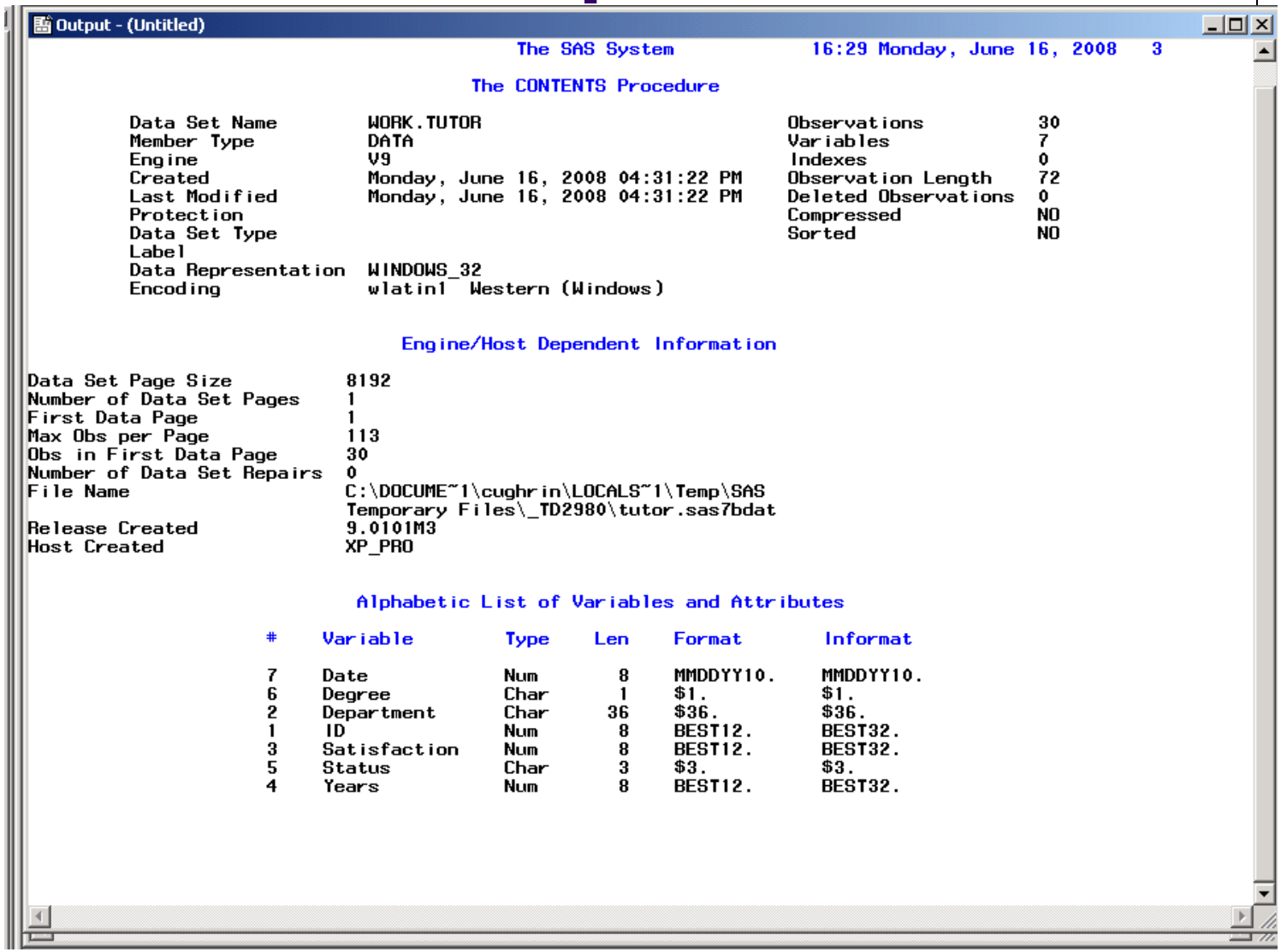


Two Sections

Descriptor Section

Data Section

Data Set Descriptor Section



Output - (Untitled)

The SAS System 16:29 Monday, June 16, 2008 3

The CONTENTS Procedure

Data Set Name	WORK.TUTOR	Observations	30
Member Type	DATA	Variables	7
Engine	V9	Indexes	0
Created	Monday, June 16, 2008 04:31:22 PM	Observation Length	72
Last Modified	Monday, June 16, 2008 04:31:22 PM	Deleted Observations	0
Protection		Compressed	NO
Data Set Type		Sorted	NO
Label			
Data Representation	WINDOWS_32		
Encoding	wlatin1 Western (Windows)		

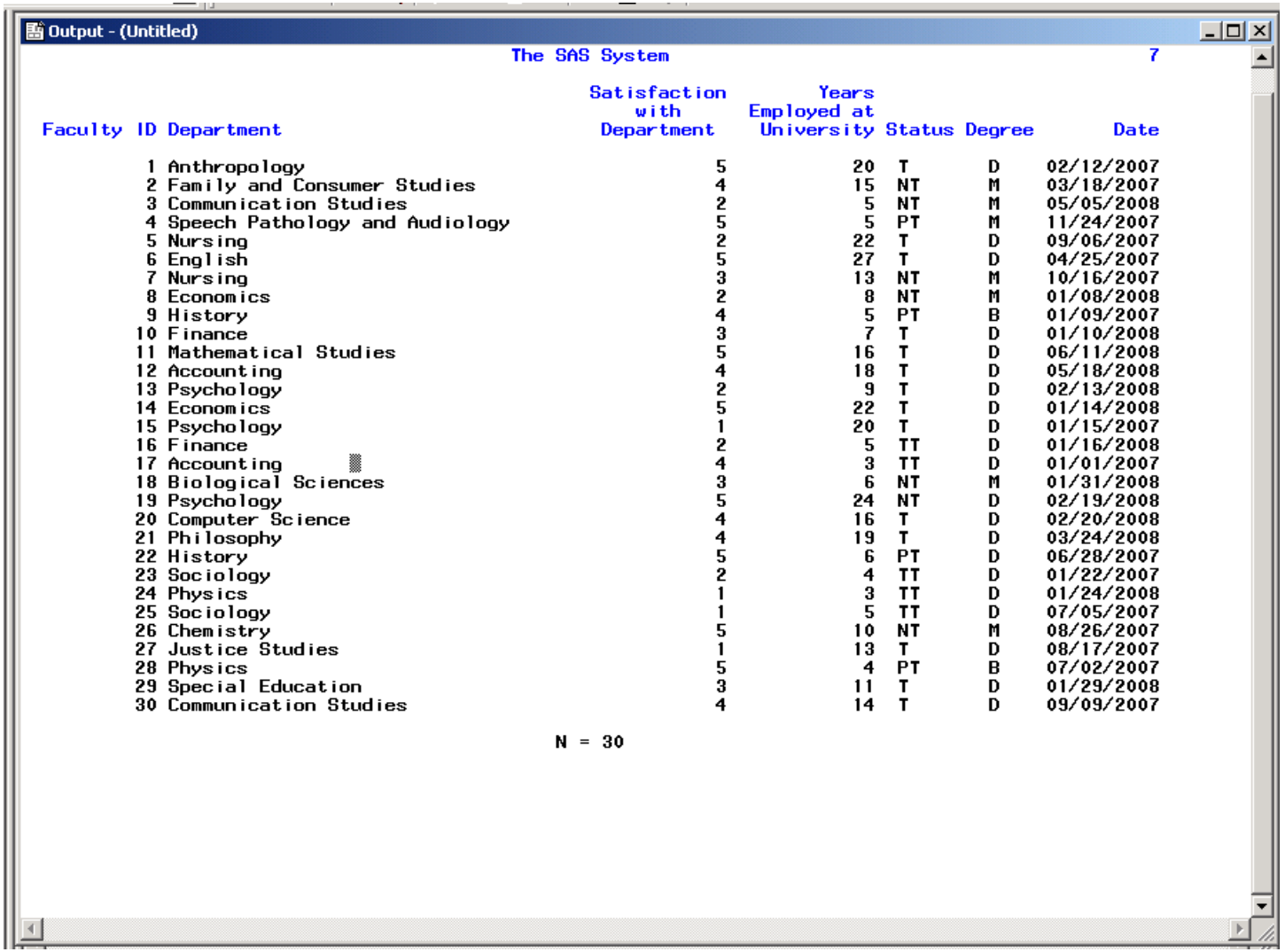
Engine/Host Dependent Information

Data Set Page Size	8192
Number of Data Set Pages	1
First Data Page	1
Max Obs per Page	113
Obs in First Data Page	30
Number of Data Set Repairs	0
File Name	C:\DOCUME~1\cughrin\LOCALS~1\Temp\SAS Temporary Files_TD2980\tutor.sas7bdat
Release Created	9.0101M3
Host Created	XP_PRO

Alphabetic List of Variables and Attributes

#	Variable	Type	Len	Format	Informat
7	Date	Num	8	MMDDYY10.	MMDDYY10.
6	Degree	Char	1	$1.	$1.
2	Department	Char	36	$36.	$36.
1	ID	Num	8	BEST12.	BEST32.
3	Satisfaction	Num	8	BEST12.	BEST32.
5	Status	Char	3	$3.	$3.
4	Years	Num	8	BEST12.	BEST32.

SAS Data Section

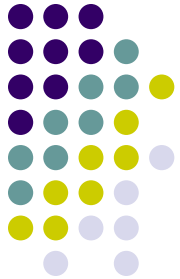

Output - (Untitled)

The SAS System 7

Faculty ID	Department	Satisfaction with Department	Years Employed at University	Status	Degree	Date
1	Anthropology	5	20	T	D	02/12/2007
2	Family and Consumer Studies	4	15	NT	M	03/18/2007
3	Communication Studies	2	5	NT	M	05/05/2008
4	Speech Pathology and Audiology	5	5	PT	M	11/24/2007
5	Nursing	2	22	T	D	09/06/2007
6	English	5	27	T	D	04/25/2007
7	Nursing	3	13	NT	M	10/16/2007
8	Economics	2	8	NT	M	01/08/2008
9	History	4	5	PT	B	01/09/2007
10	Finance	3	7	T	D	01/10/2008
11	Mathematical Studies	5	16	T	D	06/11/2008
12	Accounting	4	18	T	D	05/18/2008
13	Psychology	2	9	T	D	02/13/2008
14	Economics	5	22	T	D	01/14/2008
15	Psychology	1	20	T	D	01/15/2007
16	Finance	2	5	TT	D	01/16/2008
17	Accounting	4	3	TT	D	01/01/2007
18	Biological Sciences	3	6	NT	M	01/31/2008
19	Psychology	5	24	NT	D	02/19/2008
20	Computer Science	4	16	T	D	02/20/2008
21	Philosophy	4	19	T	D	03/24/2008
22	History	5	6	PT	D	06/28/2007
23	Sociology	2	4	TT	D	01/22/2007
24	Physics	1	3	TT	D	01/24/2008
25	Sociology	1	5	TT	D	07/05/2007
26	Chemistry	5	10	NT	M	08/26/2007
27	Justice Studies	1	13	T	D	08/17/2007
28	Physics	5	4	PT	B	07/02/2007
29	Special Education	3	11	T	D	01/29/2008
30	Communication Studies	4	14	T	D	09/09/2007

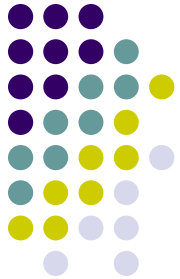
N = 30

Attributes of Variables



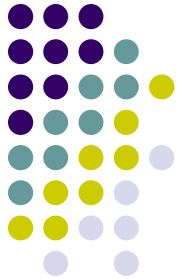
- Name
  - e.g. Status
- Type
  - Numeric or Character
  - e.g. Status in this example is character (T, TT, PT, or NTT) and Satisfaction is numeric (1 to 5).

SAS Data Set Terminology



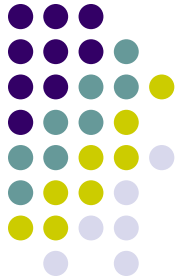
- *Variables* – columns in a SAS data set.
- *Observations* – rows in a SAS data set.
- *Numeric Data* – values that are treated as numeric and may include 8 bytes of floating storage for 16 to 17 significant digits.
- *Character Data* – non numeric data values such as letters, numbers, special characters, and blanks. May be stores with a length of 1 to 32, 767 bytes. One byte is equal to one character.

SAS Data Set and Variable Name Criteria



- Can be 32 characters long.
- Can be uppercase, lowercase, or a mixture of the cases.
- Are not case sensitive
- Cannot start with number and cannot contain special characters or blanks.
- Must start with a letter or underscore.

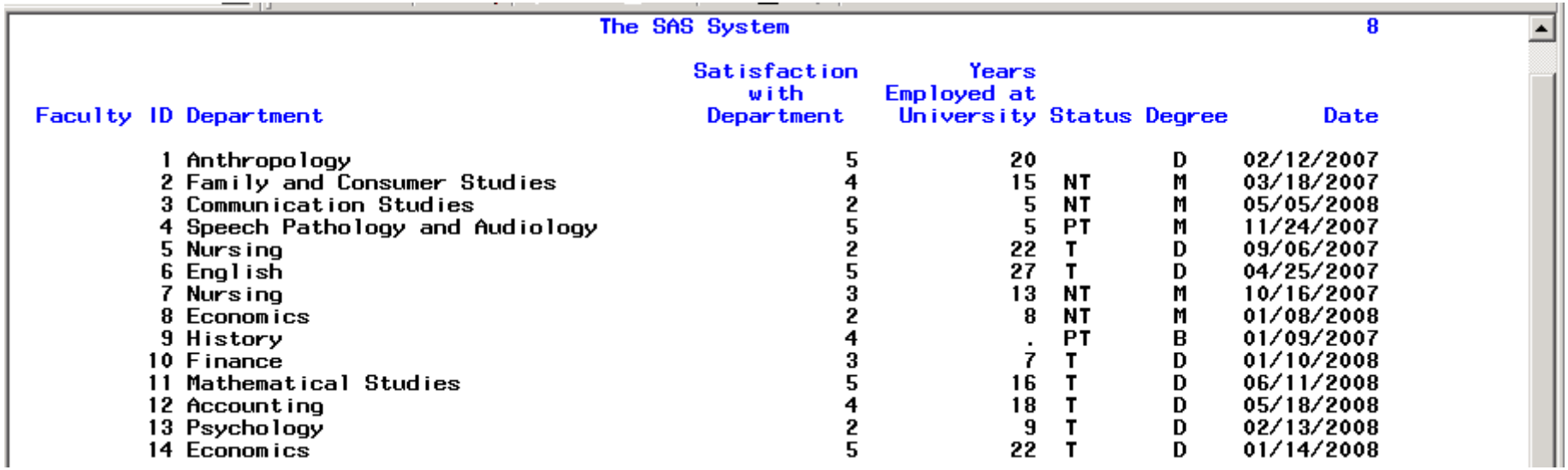
SAS Dates



- Dates are treated as special kind of numeric data.
  - They are the number of days since January 1st, 1960. January 1st 1960 is the 0 point. SAS dates can go back to 1582 (Gregorian Calendar) and forward to the year 20000.
  - Dates are displayed using a format. There are a number of different date formats supported by SAS.
- Time is scored as the number of seconds since midnight. SAS date time is the number of seconds since January 1st, 1960.

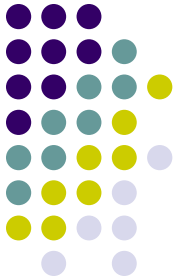
Missing Data in SAS

- Missing values are valid values.
  - For character data, missing values are displayed as blanks.
  - For numeric data, missing values are displayed as periods.



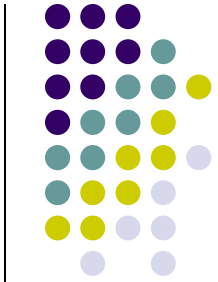
The SAS System 8

Faculty ID	Department	Satisfaction with Department	Years Employed at University	Status	Degree	Date
1	Anthropology	5	20		D	02/12/2007
2	Family and Consumer Studies	4	15	NT	M	03/18/2007
3	Communication Studies	2	5	NT	M	05/05/2008
4	Speech Pathology and Audiology	5	5	PT	M	11/24/2007
5	Nursing	2	22	T	D	09/06/2007
6	English	5	27	T	D	04/25/2007
7	Nursing	3	13	NT	M	10/16/2007
8	Economics	2	8	NT	M	01/08/2008
9	History	4	.	PT	B	01/09/2007
10	Finance	3	7	T	D	01/10/2008
11	Mathematical Studies	5	16	T	D	06/11/2008
12	Accounting	4	18	T	D	05/18/2008
13	Psychology	2	9	T	D	02/13/2008
14	Economics	5	22	T	D	01/14/2008



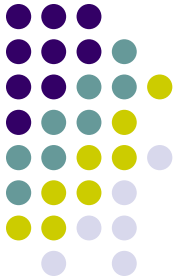
SAS Syntax

SAS Syntax



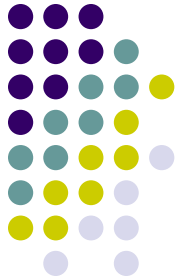
- Statements in SAS are like sentences. The punctuation though is a semicolon(;)rather than a period (.)
- Most Statements (but not all) start with an identifying key word (e.g. proc, data, label, options, format…)
- Statements are strung together into sections similar to paragraphs. These paragraphs in a Windows OS are ended with the word “run” and a semicolon.

Example of SAS Syntax



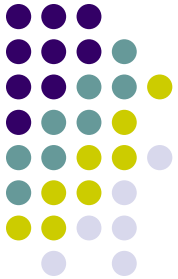
```
proc print data=tutor NOOBS N label;
label ID= Faculty ID
        Department= Department
        Satisfaction= Satisfaction with Department
        Years= Years Employed at University
        Satus= Faculty Status
        Degree= Degree
        Date=Date;
options ls=100 nodate;
run;
```

SAS Syntax Rules



- SAS statements are format free.
- One or more blanks or special characters are used to separate words.
- They can begin and end in any column.
- A single statement can span multiple lines.
- Several statements can be on the same line.

Example of SAS Free Format



```
proc print data=tutor NOOBS N label;
label ID= Faculty ID Department= Department Satisfaction= Satisfaction with Department
Years= Years Employed at University Satus= Faculty Status Degree= Degree Date=Date;
options ls=100 nodate;
run;
```

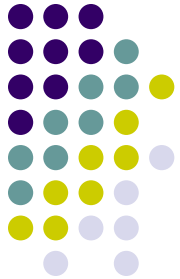
Using the free-format Syntax
rules of SAS though can make it difficult for others (or you) to read your
program. This is akin to
writing a page of text with little attention to line breaks. You may still have
Capital letters and periods, but where a sentence begins and ends may be a bit confusing

Example of SAS Formatted

```
proc print data=tutor NOOBS N label;
label ID= Faculty ID
      Department= Department
      Satisfaction= Satisfaction with Department
      Years= Years Employed at University
      Satus= Faculty Status
      Degree= Degree
      Date=Date;
options ls=100 nodate;
run;
```

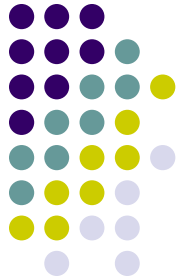
Using the free-format Syntax rules of SAS though can make it difficult for others (or you) to read your program. This is akin to writing a page of text with little attention to line breaks. You may still have capital letters and periods, but where a sentence begins and ends may be a bit confusing. Isn't this paragraph a bit easier to read?

SAS Comments

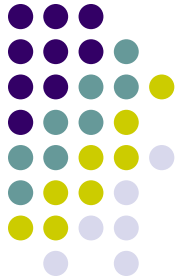


- Type /* to begin a comment.
- Type your comment text.
- Type */ to end the comment.
- Or, type an * at the beginning of a line. Everything between the * and the ; will be commented.
  - e.g. *infile ‘tutor.dat’;
- Alternatively, highlight the text that you would like to comment and use the keys Ctrl / to comment the line. To uncomment a line, highlight and use the Ctrl Shift / keys.

SAS Comments

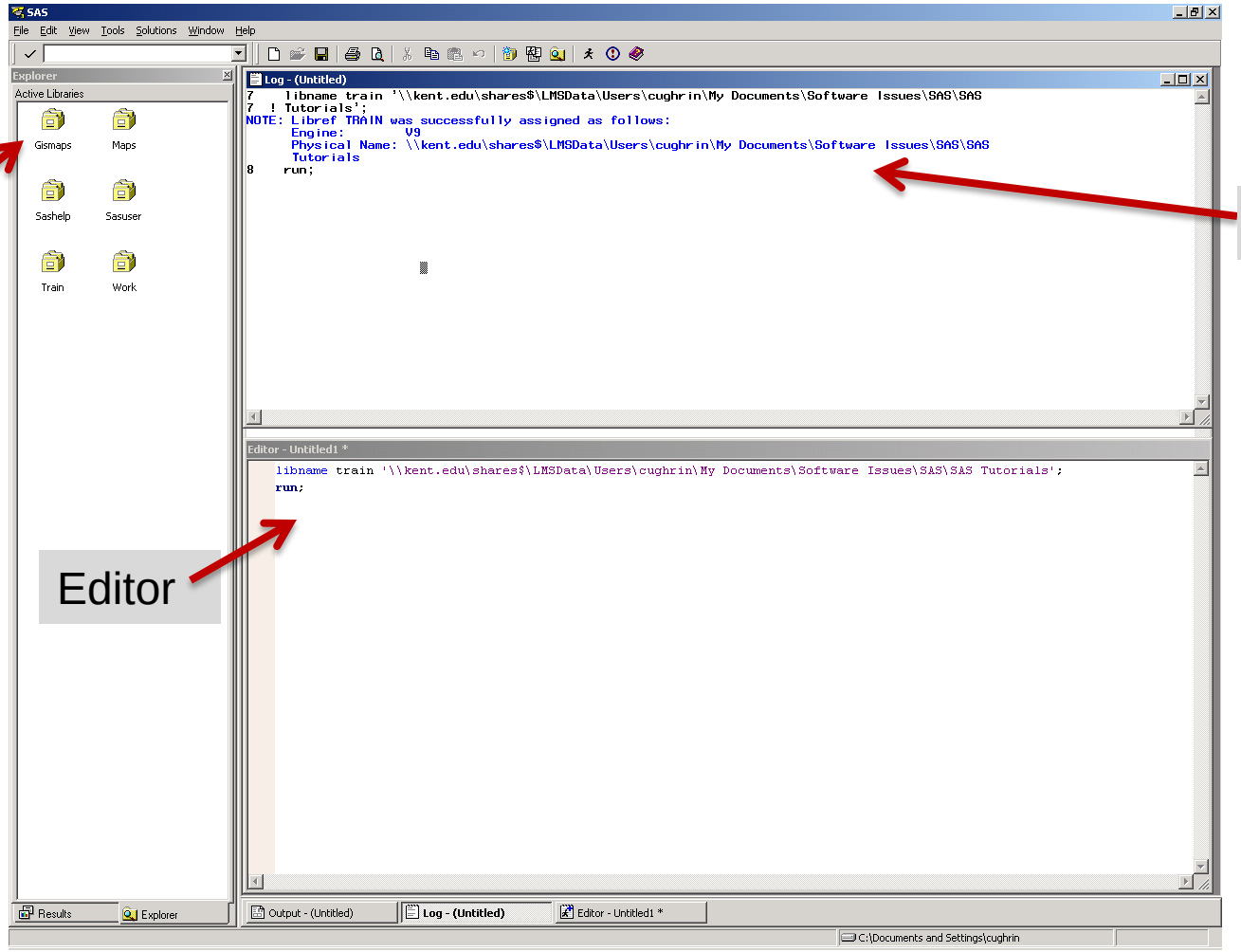


```
proc print data=Tutormissing NOOBS N label;
label ID= Faculty ID
      Department= Department
      Satisfaction= Satisfaction with Department
      Years= Years Employed at University
      Satus= Faculty Status
      Degree= Degree
      Date=Date;
/*options ls=100 nodate;*/
run;
```



SAS Windows

SAS Windows

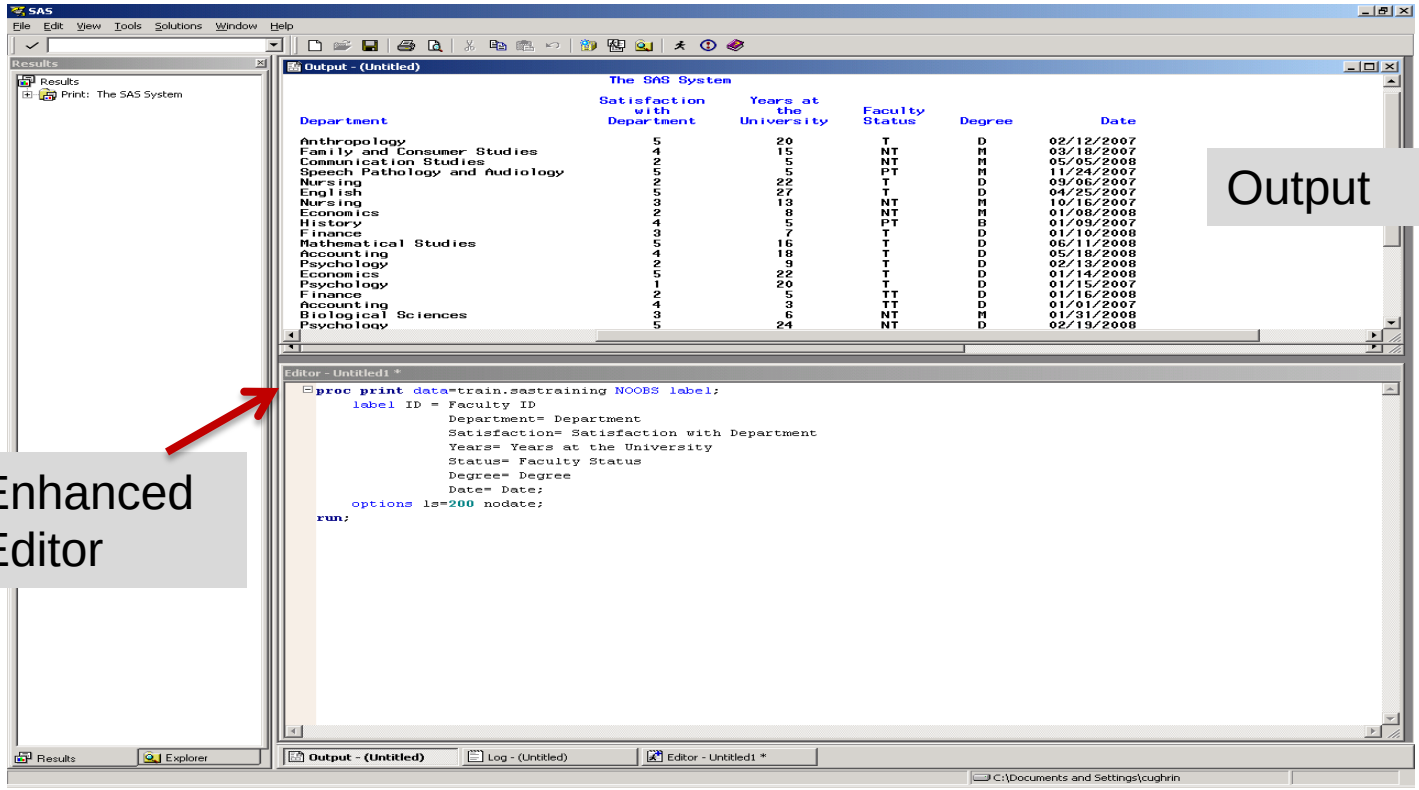


Explorer

Editor

Log

Enhanced Editor Window

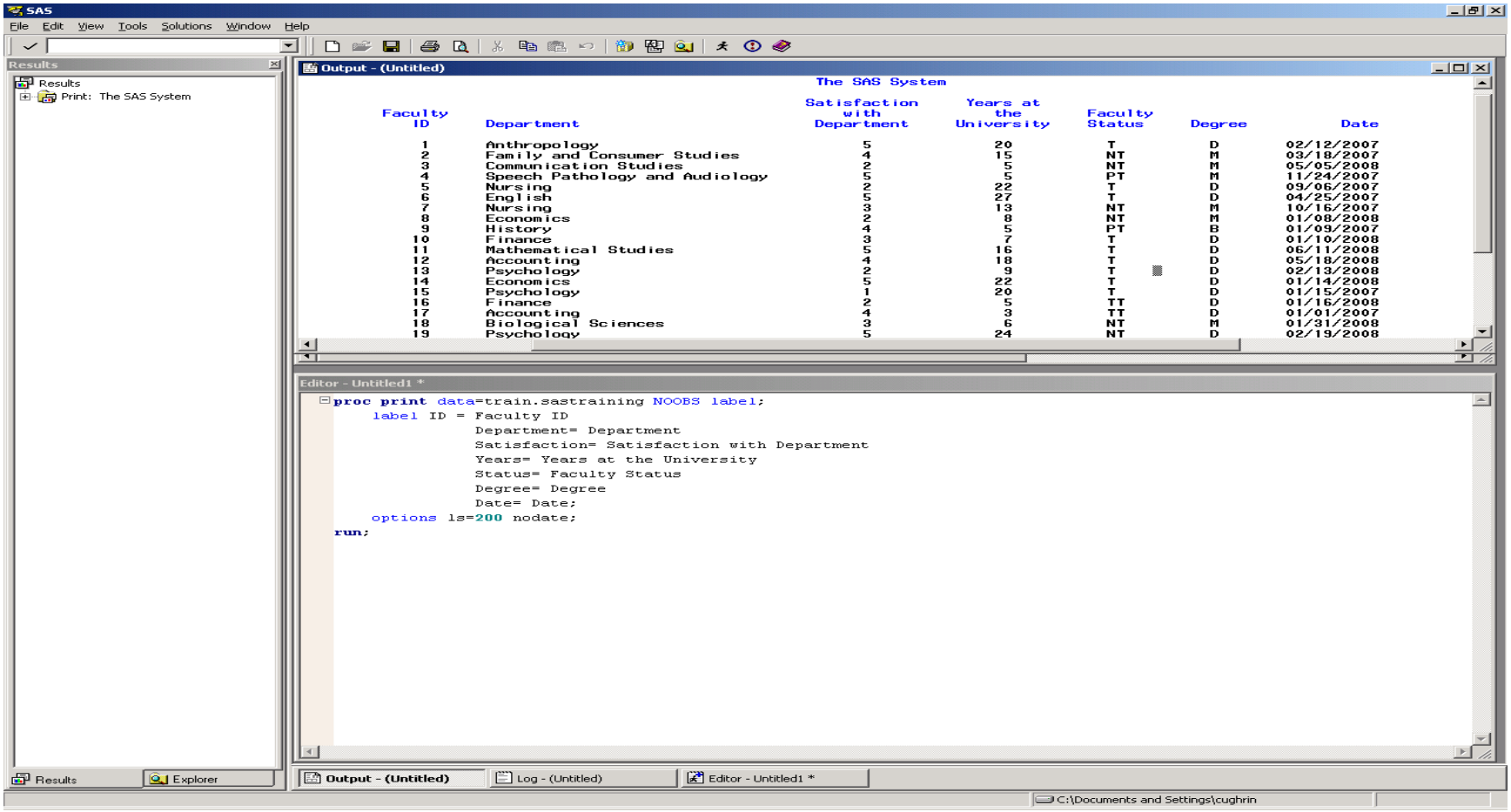


- Your program script appears in this window.
- You can either bring it in from a file or type the program right into the window.
- Once the program is in the window, you can Click Submit (or the running guy).

SAS Log

- SAS Log provides a “blow by blow” account of the execution of your program. It includes how many observations were read and output, as well as, errors and notes.
- Note the errors in red.

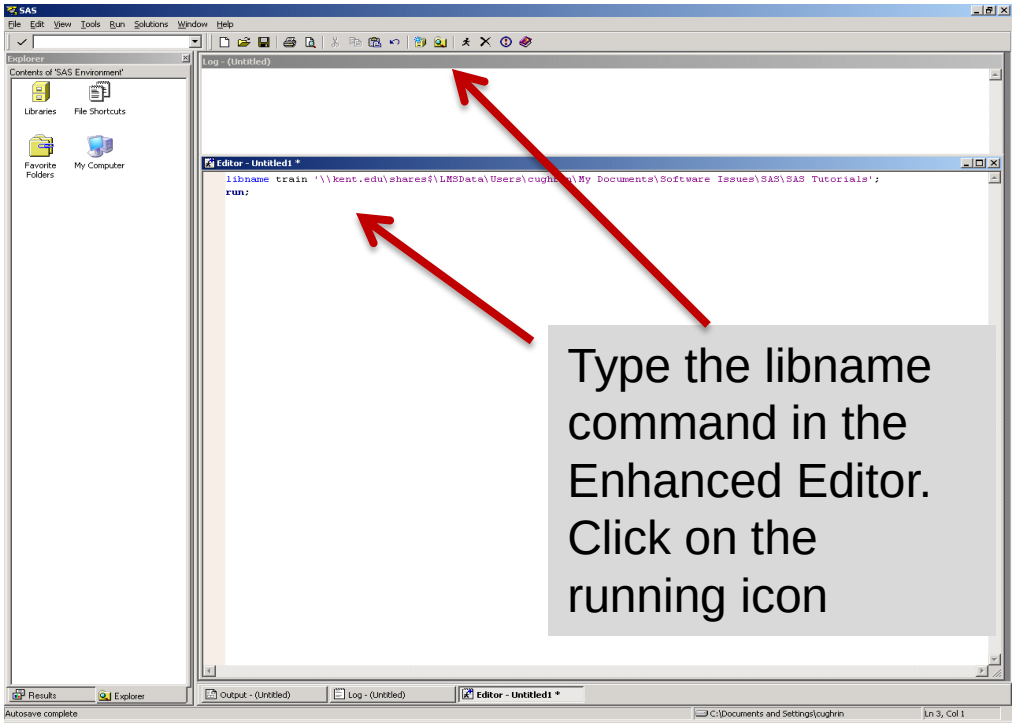
Output Window



SAS Library

- SAS Data Libraries are like drawers in a filing cabinet. The SAS data sets are files within those drawers. Note the icons for the SAS library match that metaphor.
- In order to assign a “drawer”, you assign a library reference name (libref).
- There are two drawers already in your library: work (temporary) and sasuser (permanent).
- You can also create your own libraries (drawers) using the libname statement.

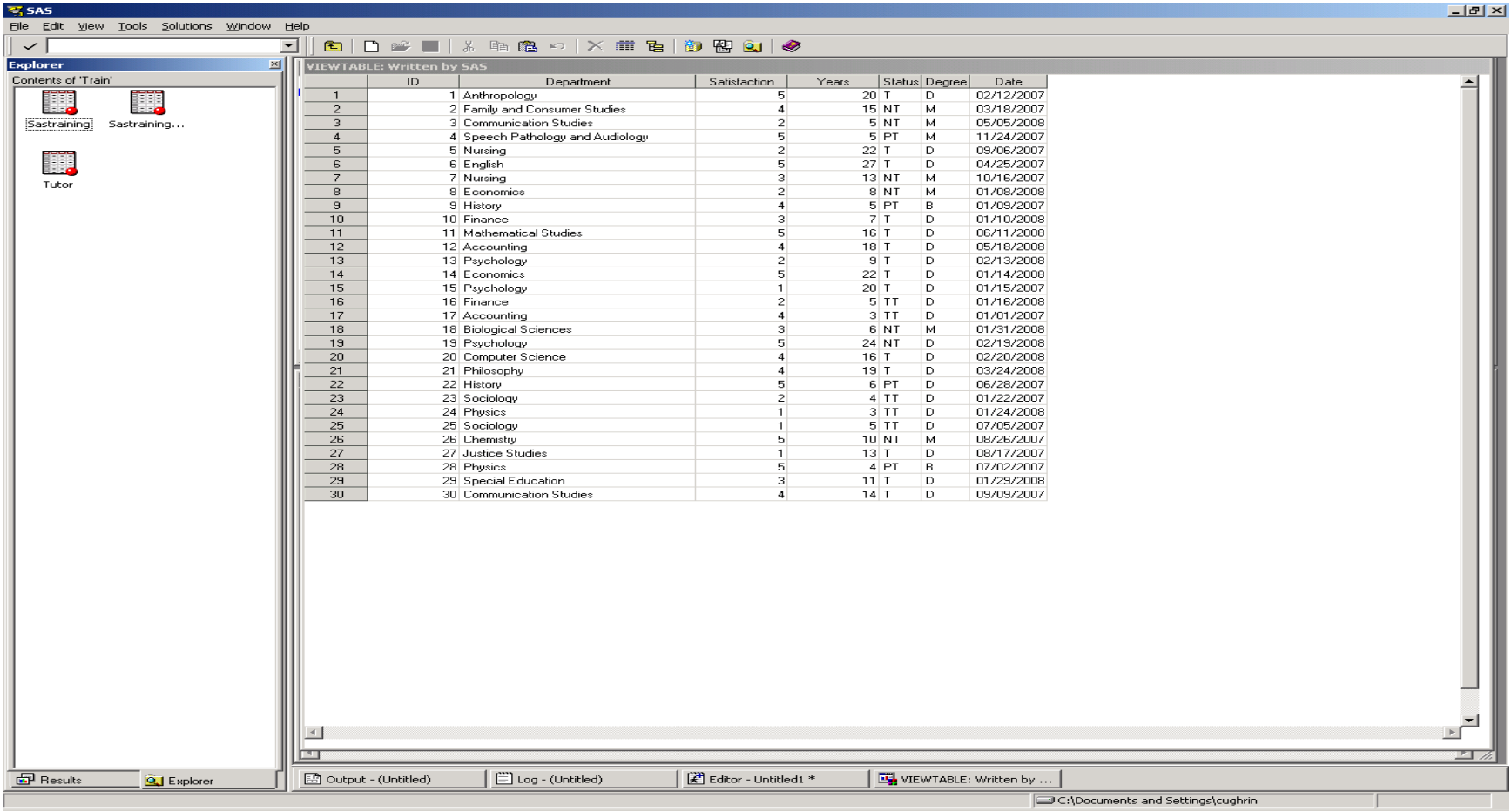
Establishing the libname



libname Tina ‘E:\Trainings\JMP Training’;

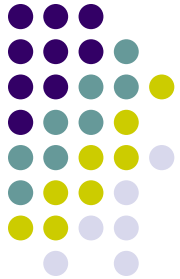
run;

Viewtable Window



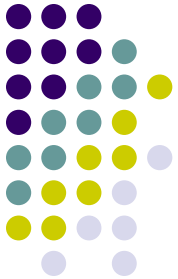
	ID	Department	Satisfaction	Years	Status	Degree	Date
1	1	Anthropology	5	20	T	D	02/12/2007
2	2	Family and Consumer Studies	4	15	NT	M	03/18/2007
3	3	Communication Studies	2	5	NT	M	05/05/2008
4	4	Speech Pathology and Audiology	5	5	PT	M	11/24/2007
5	5	Nursing	2	22	T	D	09/06/2007
6	6	English	5	27	T	D	04/25/2007
7	7	Nursing	3	13	NT	M	10/16/2007
8	8	Economics	2	8	NT	M	01/08/2008
9	9	History	4	5	PT	B	01/09/2007
10	10	Finance	3	7	T	D	01/10/2008
11	11	Mathematical Studies	5	16	T	D	06/11/2008
12	12	Accounting	4	18	T	D	05/18/2008
13	13	Psychology	2	9	T	D	02/13/2008
14	14	Economics	5	22	T	D	01/14/2008
15	15	Psychology	1	20	T	D	01/15/2007
16	16	Finance	2	5	TT	D	01/16/2008
17	17	Accounting	4	3	TT	D	01/01/2007
18	18	Biological Sciences	3	6	NT	M	01/31/2008
19	19	Psychology	5	24	NT	D	02/19/2008
20	20	Computer Science	4	16	T	D	02/20/2008
21	21	Philosophy	4	19	T	D	03/24/2008
22	22	History	5	6	PT	D	06/28/2007
23	23	Sociology	2	4	TT	D	01/22/2007
24	24	Physics	1	3	TT	D	01/24/2008
25	25	Sociology	1	5	TT	D	07/05/2007
26	26	Chemistry	5	10	NT	M	08/26/2007
27	27	Justice Studies	1	13	T	D	08/17/2007
28	28	Physics	5	4	PT	B	07/02/2007
29	29	Special Education	3	11	T	D	01/29/2008
30	30	Communication Studies	4	14	T	D	09/09/2007

Data Step Programming



- SAS data set can be created using another SAS data set as input or raw data
- To create a SAS data set using another SAS data set, the DATA and SET statements are used.
- To create a SAS data set from raw data, you use INFILE and INPUT statements.
- DATA and SET cannot be used for raw data and INFILE and INPUT cannot be used for existing SAS datasets.

Reading a SAS Dataset



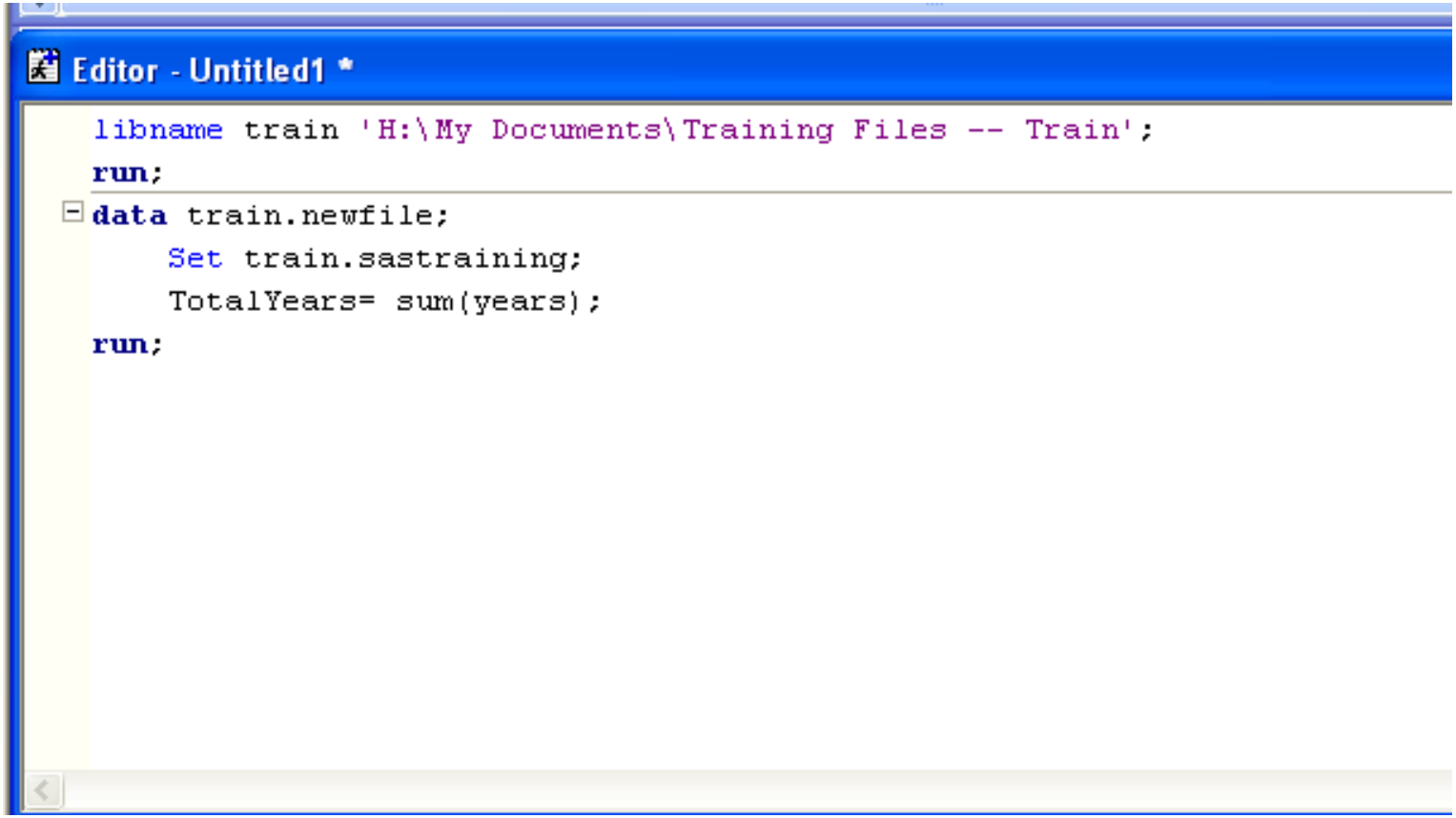
DATA (name of new SAS dataset)

SET (name of existing SAS dataset)

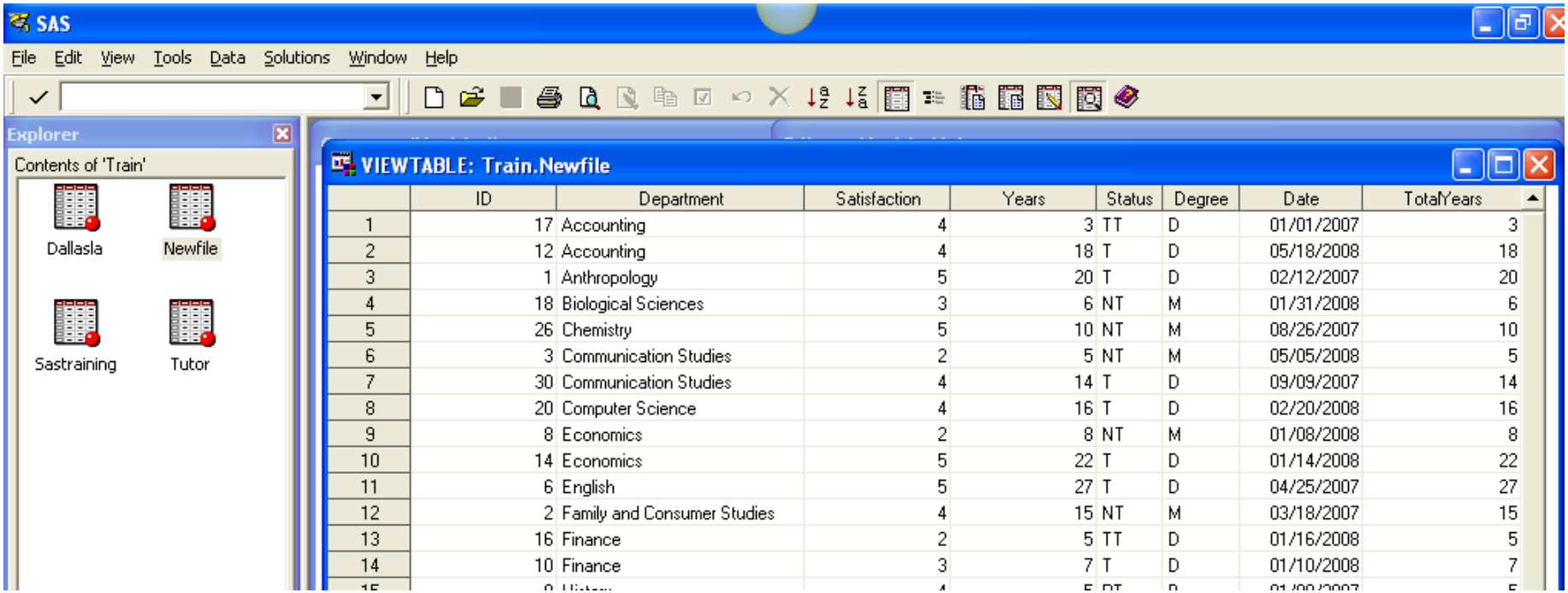
Additional statements

Run;

Reading a SAS Dataset

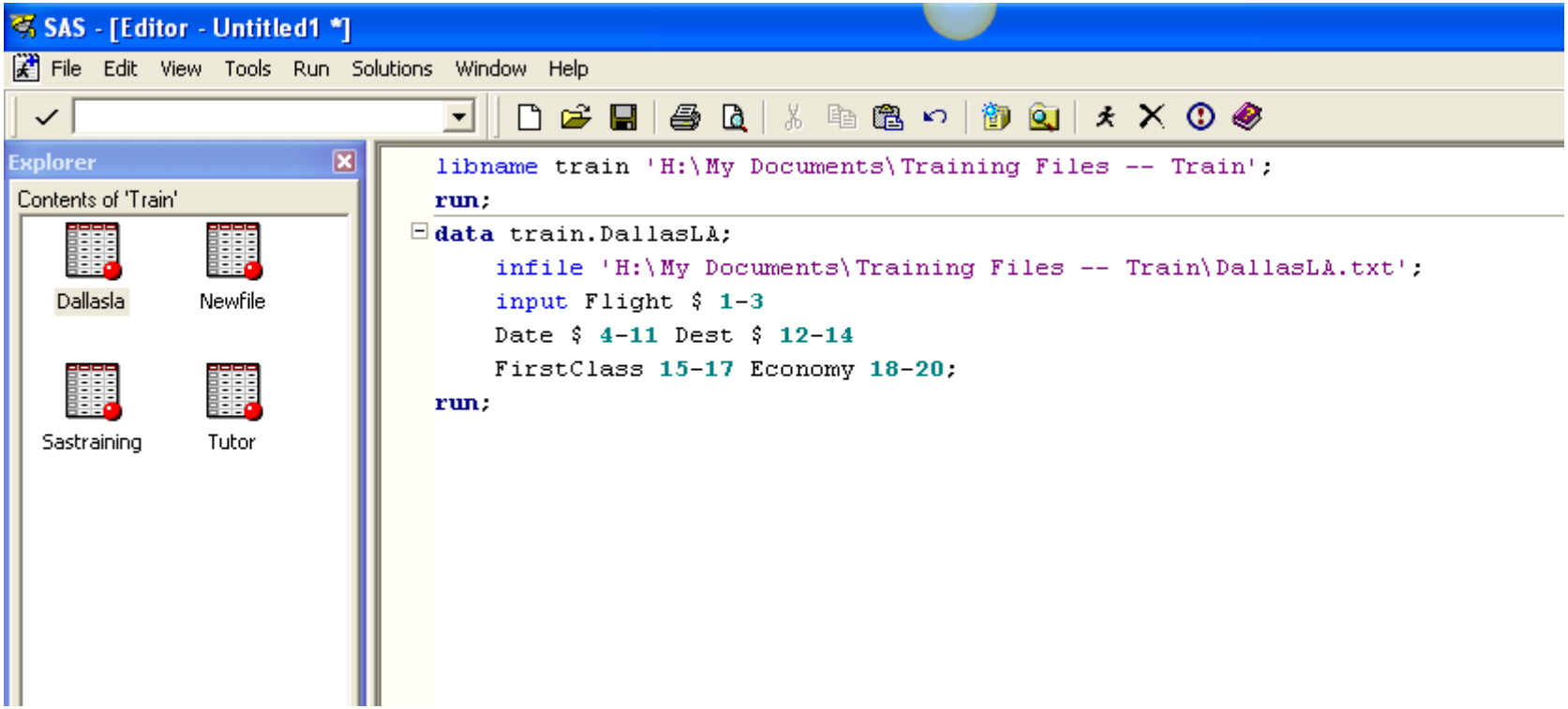


Reading SAS Dataset

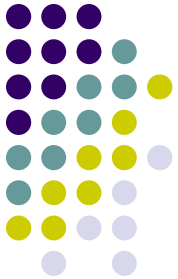


	ID	Department	Satisfaction	Years	Status	Degree	Date	TotalYears
1	17	Accounting	4	3	TT	D	01/01/2007	3
2	12	Accounting	4	18	T	D	05/18/2008	18
3	1	Anthropology	5	20	T	D	02/12/2007	20
4	18	Biological Sciences	3	6	NT	M	01/31/2008	6
5	26	Chemistry	5	10	NT	M	08/26/2007	10
6	3	Communication Studies	2	5	NT	M	05/05/2008	5
7	30	Communication Studies	4	14	T	D	09/09/2007	14
8	20	Computer Science	4	16	T	D	02/20/2008	16
9	8	Economics	2	8	NT	M	01/08/2008	8
10	14	Economics	5	22	T	D	01/14/2008	22
11	6	English	5	27	T	D	04/25/2007	27
12	2	Family and Consumer Studies	4	15	NT	M	03/18/2007	15
13	16	Finance	2	5	TT	D	01/16/2008	5
14	10	Finance	3	7	T	D	01/10/2008	7

Reading Raw Data



Selecting Variables



- You can use a DROP or KEEP statement in a DATA step to control which variables are written to a new SAS data set.

Selecting Variables

```
libname train 'H:\My Documents\Training Files -- Train';
run;
data train.newfile;
    Set train.sastraining;
    TotalYears= sum(years);
    Drop Years;
run;
```

	ID	Department	Satisfaction	Status	Degree	Date	TotalYears
1	17	Accounting	4	TT	D	01/01/2007	3
2	12	Accounting	4	T	D	05/18/2008	18
3	1	Anthropology	5	T	D	02/12/2007	20
4	18	Biological Sciences	3	NT	M	01/31/2008	6
5	26	Chemistry	5	NT	M	08/26/2007	10
6	3	Communication Studies	2	NT	M	05/05/2008	5
7	30	Communication Studies	4	T	D	09/09/2007	14
8	20	Computer Science	4	T	D	02/20/2008	16
9	8	Economics	2	NT	M	01/08/2008	8
10	14	Economics	5	T	D	01/14/2008	22
11	6	English	5	T	D	04/25/2007	27
12	2	Family and Consumer Studies	4	NT	M	03/18/2007	15
13	16	Finance	2	TT	D	01/16/2008	5
14	10	Finance	3	T	D	01/10/2008	7
15	9	History	4	PT	B	01/09/2007	5
16	22	History	5	PT	D	06/28/2007	6
17	27	Justice Studies	1	T	D	08/17/2007	13
18	11	Mathematical Studies	5	T	D	06/11/2008	16
19	7	Nursing	3	NT	M	10/16/2007	13
20	5	Nursing	2	T	D	09/06/2007	22
21	21	Philosophy	4	T	D	03/24/2008	19
22	24	Physics	1	TT	D	01/24/2008	3
23	28	Physics	5	PT	B	07/02/2007	4
24	13	Psychology	2	T	D	02/13/2008	9
25	15	Psychology	1	T	D	01/15/2007	20
26	19	Psychology	5	NT	D	02/19/2008	24
27	23	Sociology	2	TT	D	01/22/2007	4
28	25	Sociology	1	TT	D	07/05/2007	5
29	29	Special Education	3	T	D	01/29/2008	11
30	4	Speech Pathology and Audiology	5	PT	M	11/24/2007	5

Selecting Variables

Editor - Untitled1 *

```
libname train 'H:\My Documents\Training Files -- Train';
run;
data train.newfile;
    Set train.sastraining;
    TotalYears= sum(years);
    Keep Department TotalYears;
run;
```

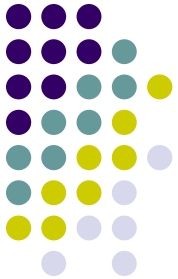
VIEWTABLE: Train.Newfile

	ID	Department	Satisfaction	Status	Degree	Date	TotalYears
1	17	Accounting	4	TT	D	01/01/2007	3
2	12	Accounting	4	T	D	05/18/2008	18
3	1	Anthropology	5	T	D	02/12/2007	20
4	18	Biological Sciences	3	NT	M	01/31/2008	6
5	26	Chemistry	5	NT	M	08/26/2007	10
6	3	Communication Studies	2	NT	M	05/05/2008	5
7	30	Communication Studies	4	T	D	09/09/2007	14
8	20	Computer Science	4	T	D	02/20/2008	16
9	8	Economics	2	NT	M	01/08/2008	8
10	14	Economics	5	T	D	01/14/2008	22
11	6	English	5	T	D	04/25/2007	27
12	2	Family and Consumer Studies	4	NT	M	03/18/2007	15
13	16	Finance	2	TT	D	01/16/2008	5
14	10	Finance	3	T	D	01/10/2008	7
15	9	History	4	PT	B	01/09/2007	5
16	22	History	5	PT	D	06/28/2007	6
17	27	Justice Studies	1	T	D	08/17/2007	13
18	11	Mathematical Studies	5	T	D	06/11/2008	16

VIEWTABLE: Train.Newfile

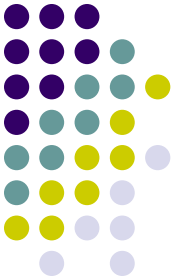
	Department	TotalYears
1	Accounting	3
2	Accounting	18
3	Anthropology	20
4	Biological Sciences	6
5	Chemistry	10
6	Communication Studies	5
7	Communication Studies	14
8	Computer Science	16
9	Economics	8
10	Economics	22
11	English	27
12	Family and Consumer Studies	15
13	Finance	5
14	Finance	7
15	History	5
16	History	6
17	Justice Studies	13
18	Mathematical Studies	16

Date Functions



- Create SAS date values
  - TODAY() – obtains the date value from the system clock
  - MDY(month,day,year) – uses numeric month, day, and year values to return the corresponding SAS date value.
- Extract information from SAS date values
  - YEAR (SAS-date) – extracts the year from a SAS date and returns a four-digit value for year
  - QTR (SAS-date) – extracts the quarter from a SAS date and returns a number from 1-4
  - MONTH (SAS-date) extracts the month from a SAS date and returns a number from 1 to 12
  - WEEKDAY (SAS-date) – extracts the day of the week and returns a number from 1 to 7

Date Function – Weekday Function

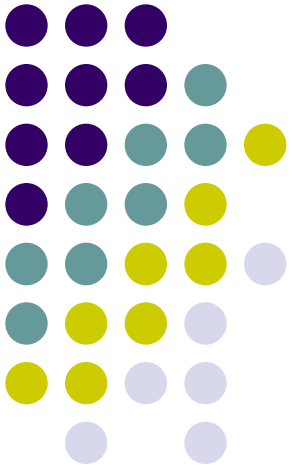


```
libname train 'H:\My Documents\Training Files -- Train';
run;
data train.newfile;
    Set train.sastraining;
    DayofWeek=weekday(Date);
run;
```

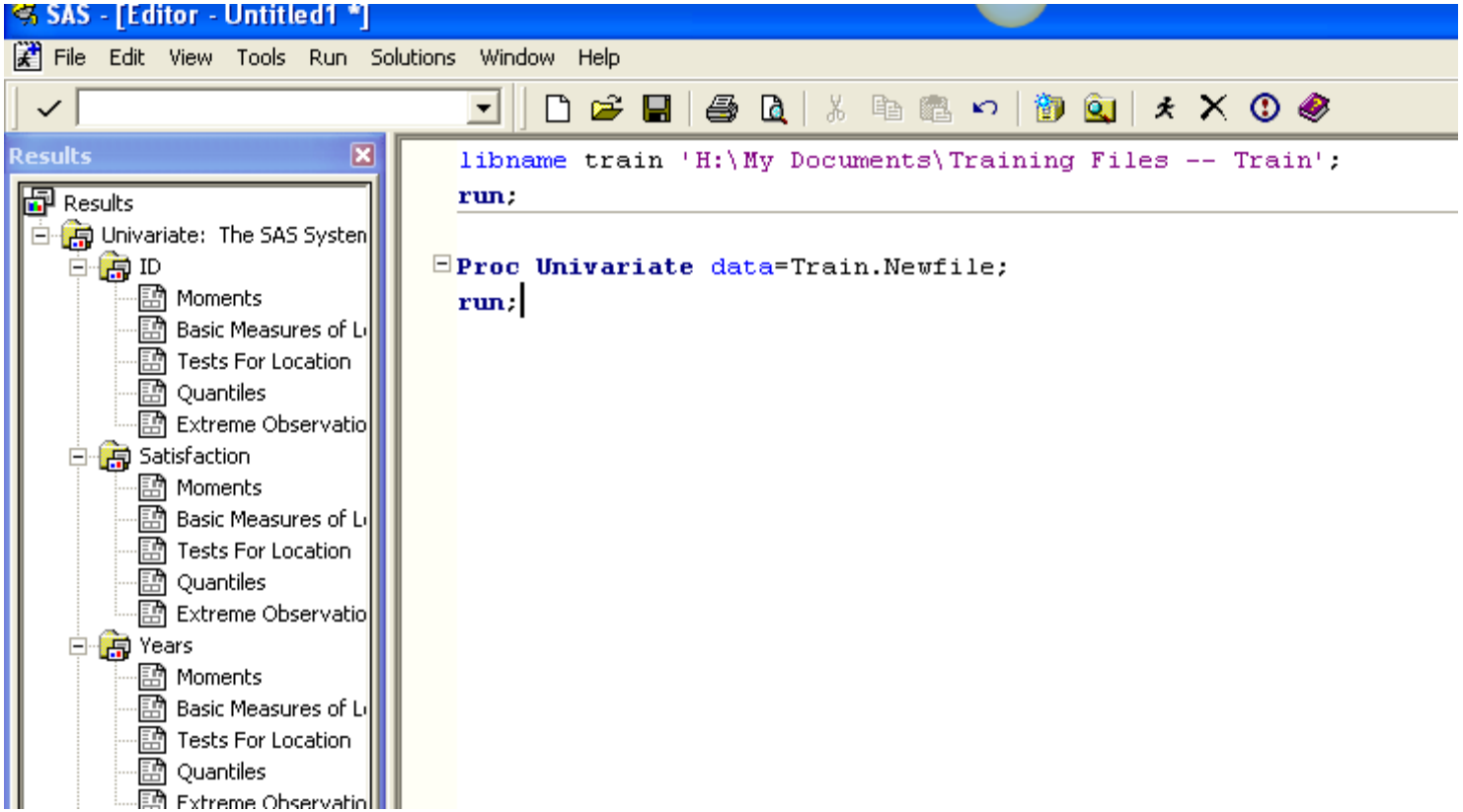
	ID	Department	Satisfaction	Years	Status	Degree	Date	DayofWeek
1	17	Accounting	4	3	TT	D	01/01/2007	2
2	12	Accounting	4	18	T	D	05/18/2008	1
3	1	Anthropology	5	20	T	D	02/12/2007	2
4	18	Biological Sciences	3	6	NT	M	01/31/2008	5
5	26	Chemistry	5	10	NT	M	08/26/2007	1
6	3	Communication Studies	2	5	NT	M	05/05/2008	2
7	30	Communication Studies	4	14	T	D	09/09/2007	1
8	20	Computer Science	4	16	T	D	02/20/2008	4
9	8	Economics	2	8	NT	M	01/08/2008	3
10	14	Economics	5	22	T	D	01/14/2008	2
11	6	English	5	27	T	D	04/25/2007	4
12	2	Family and Consumer Studies	4	15	NT	M	03/18/2007	1
13	16	Finance	2	5	TT	D	01/16/2008	4
14	10	Finance	3	7	T	D	01/10/2008	5
15	9	History	4	5	PT	B	01/09/2007	3
16	22	History	5	6	PT	D	06/28/2007	5
17	27	Justice Studies	1	13	T	D	08/17/2007	6
18	11	Mathematical Studies	5	16	T	D	06/11/2008	4
19	7	Nursing	3	13	NT	M	10/16/2007	3
20	5	Nursing	2	22	T	D	09/06/2007	5
21	21	Philosophy	4	19	T	D	03/24/2008	2
22	24	Physics	1	3	TT	D	01/24/2008	5
23	28	Physics	5	4	PT	B	07/02/2007	2
24	13	Psychology	2	9	T	D	02/13/2008	4
25	15	Psychology	1	20	T	D	01/15/2007	2
26	19	Psychology	5	24	NT	D	02/19/2008	3
27	23	Sociology	2	4	TT	D	01/22/2007	2
28	25	Sociology	1	5	TT	D	07/05/2007	5
29	29	Special Education	3	11	T	D	01/29/2008	3
30	4	Speech Pathology and Audiology	5	5	PT	M	11/24/2007	7

Proc Univariate

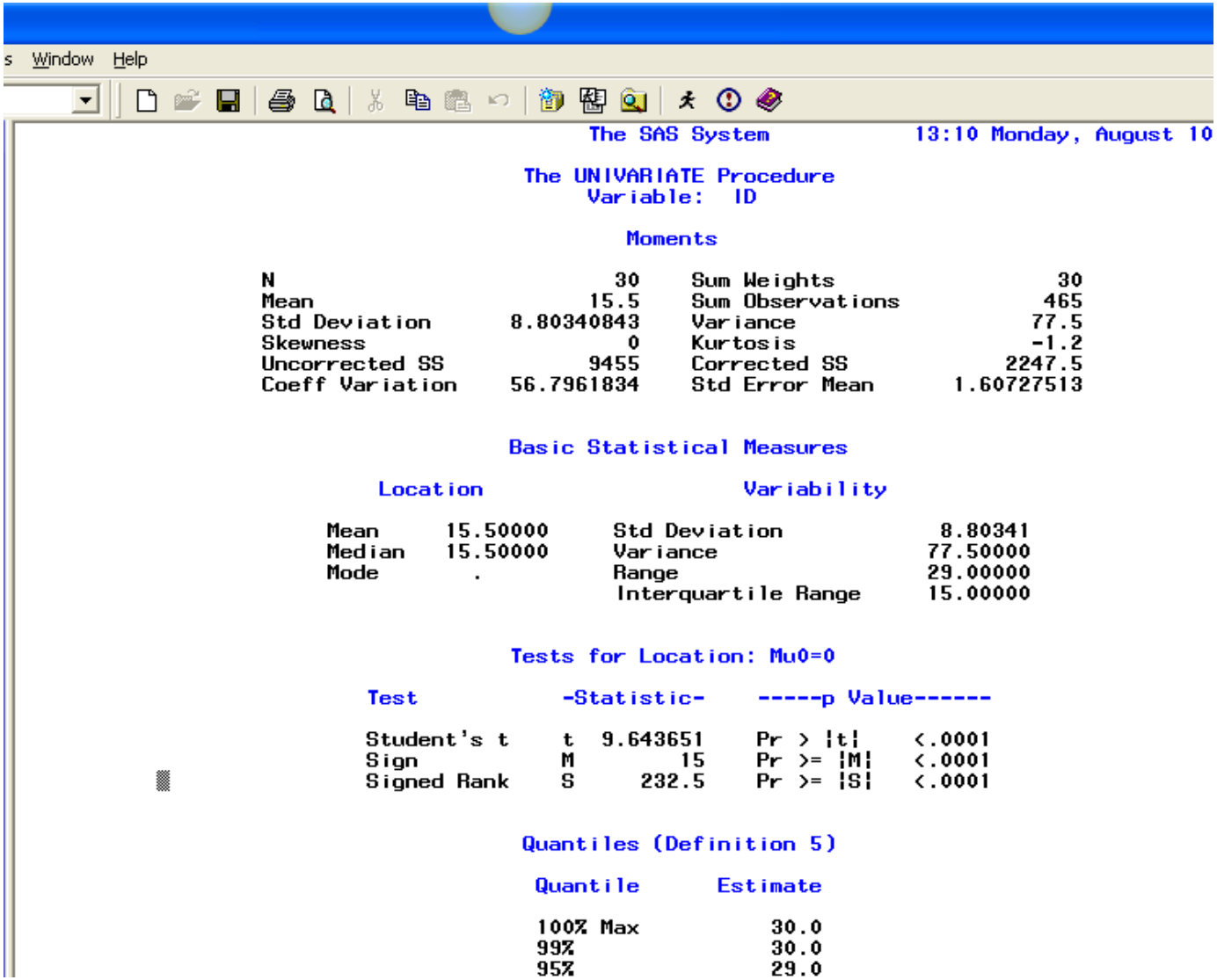
Proc Univariate



Proc Univariate



Proc Univariate



s Window Help

The SAS System 13:10 Monday, August 10

The UNIVARIATE Procedure
Variable: ID

Moments

N	30	Sum Weights	30
Mean	15.5	Sum Observations	465
Std Deviation	8.80340843	Variance	77.5
Skewness	0	Kurtosis	-1.2
Uncorrected SS	9455	Corrected SS	2247.5
Coeff Variation	56.7961834	Std Error Mean	1.60727513

Basic Statistical Measures

Location		Variability	
Mean	15.50000	Std Deviation	8.80341
Median	15.50000	Variance	77.50000
Mode	.	Range	29.00000
		Interquartile Range	15.00000

Tests for Location: Mu0=0

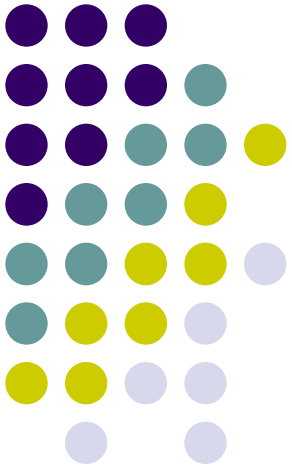
Test	-Statistic-		-----p Value------	
Student's t	t	9.643651	Pr > |t|	<.0001
Sign	M	15	Pr >= |M|	<.0001
Signed Rank	S	232.5	Pr >= |S|	<.0001

Quantiles (Definition 5)

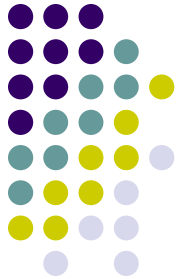
Quantile	Estimate
100% Max	30.0
99%	30.0
95%	29.0

Getting started with programming

Proc Print



Proc Print – Beginning Procedures

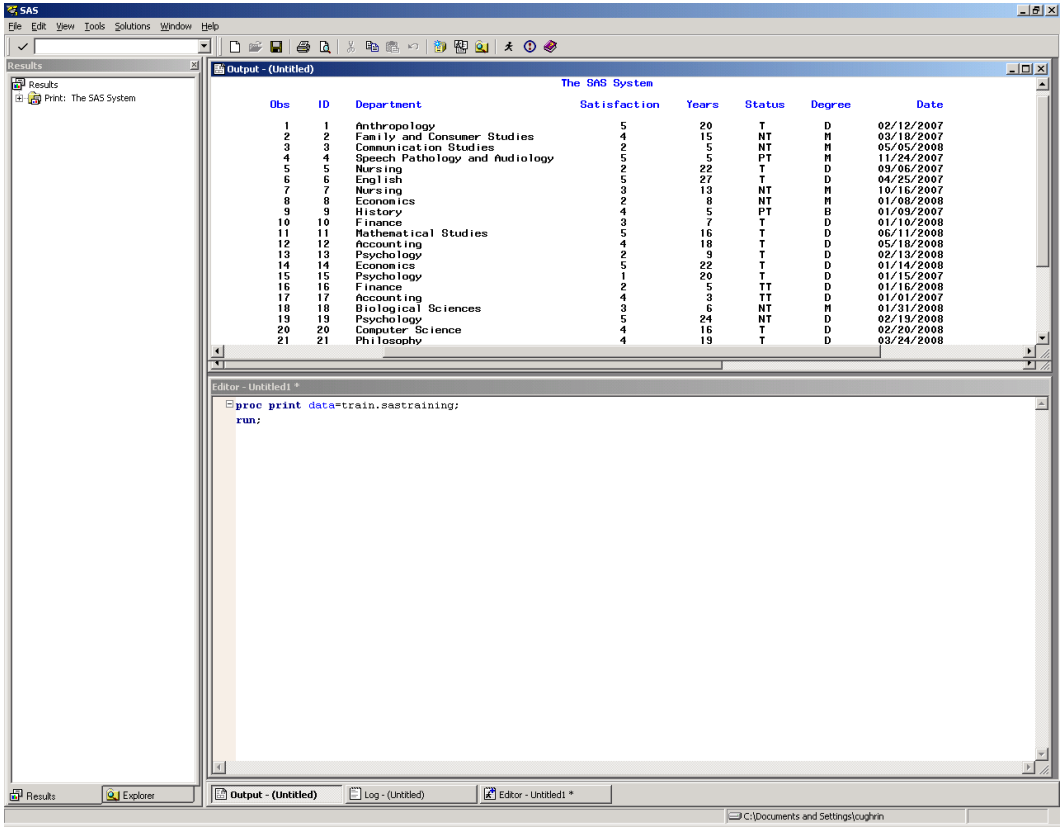


- Examining data using proc print procedure.
- Display particular variables of interest.
- Display particular observations.
- Display a list report with column totals.

Default List Report

Proc print data=train.sastraining;

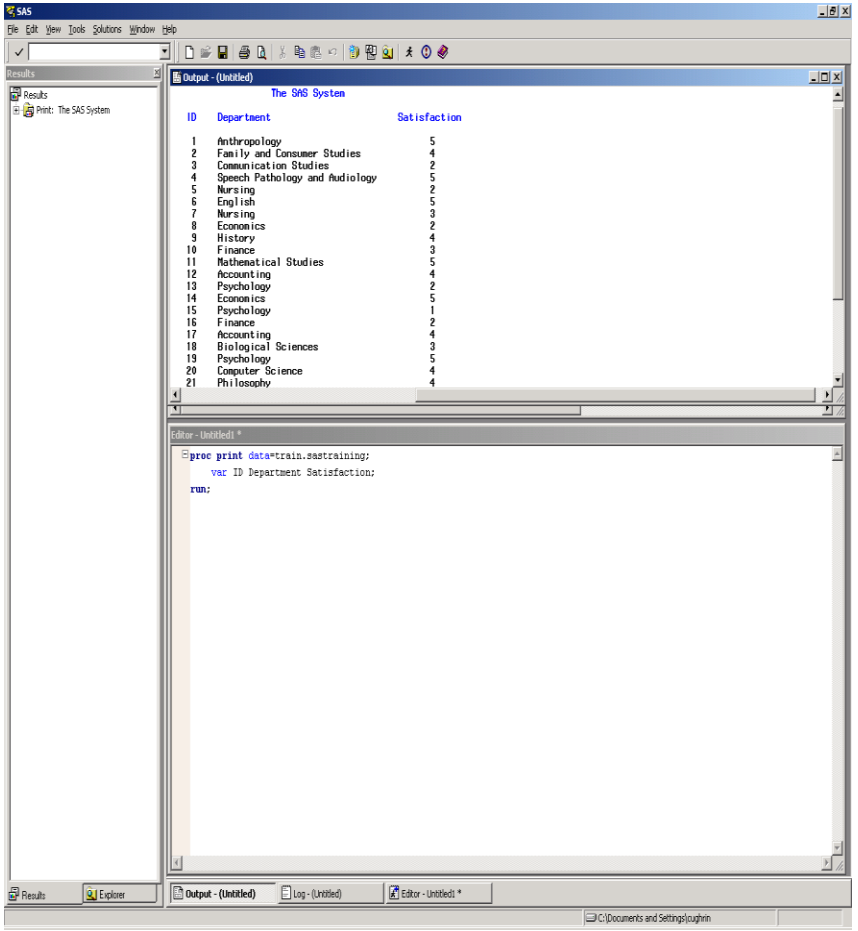
Run;



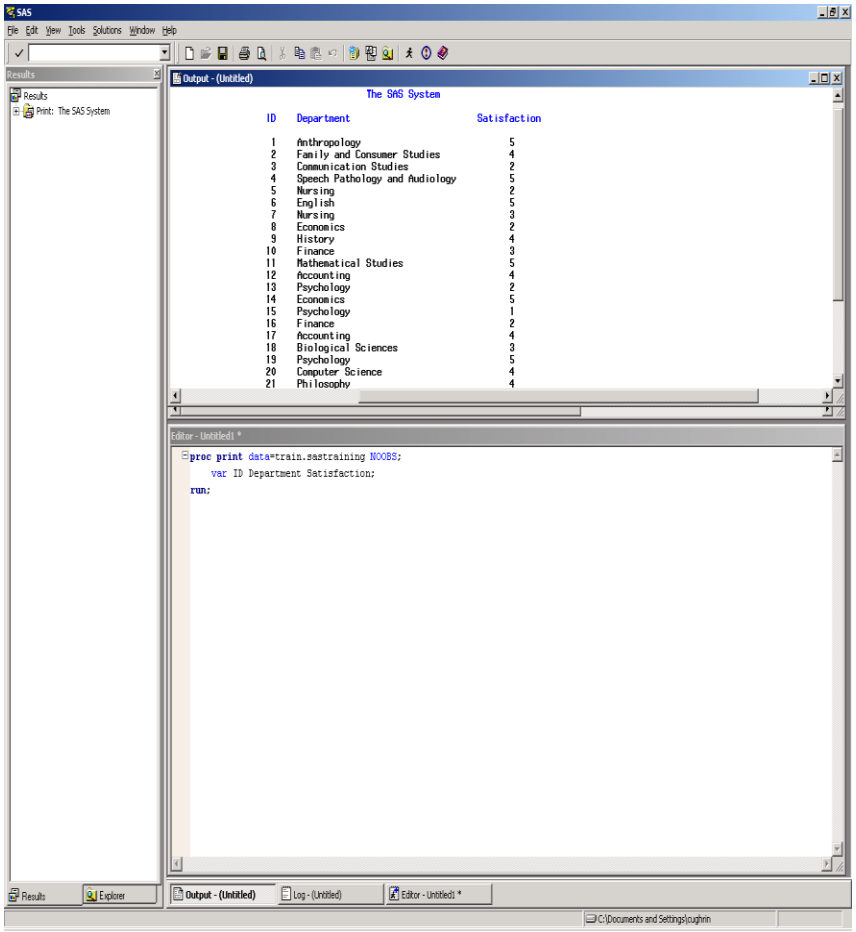
Printing Particular Variables

- Use the **VAR** statement which allows you to:
  - Select variables for your proc print
  - Define the order of the variables in the proc print.

Proc print
    data=train.sastraining;
    var ID Department Satisfaction;
Run;



Suppressing Obs Column

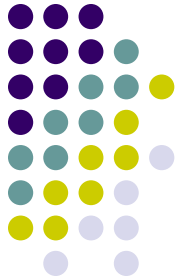


- The NOOBS option suppresses the number of observations column that shows up on the left hand side of a proc print output.

Proc print
data=train.sastraining
NOOBS;

Run;

Subsetting Data with the WHERE Statement



- Allows you to select particular observations based on criteria.
- Can be used with most SAS procedures (“IF” statements are generally used in the Data step though).
- Operands
  - Variables and Observations
- Operators
  - Comparisons
  - Logical,
  - Special
  - Functions

Comparison Operators

Mneumonic	Symbol	Definition
EQ	=	equal to
NE	^= or ~=	not equal to
GT	>	greater than
LT	<	less than
GE	>=	greater than or equal to
LE	<=	less than or equal to
IN		equal to one of a list

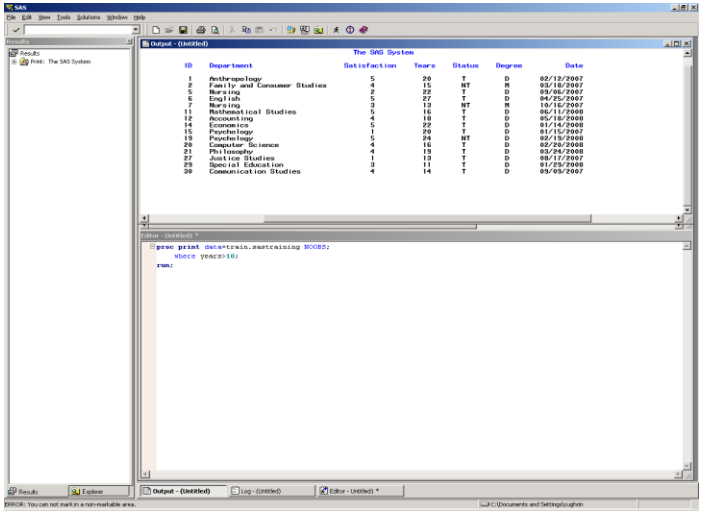
Examples of WHERE Comparison Operators

Proc print

data=train.sastraining
NOOBS;

where department=
'Psychology';

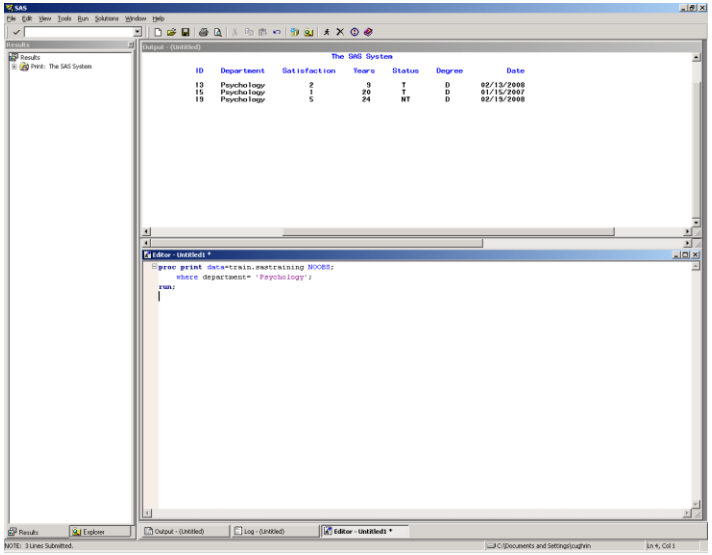
Run;



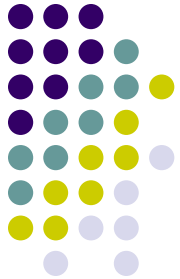
proc print data=train.sastraining
NOOBS;

where department=
'Psychology';

run;

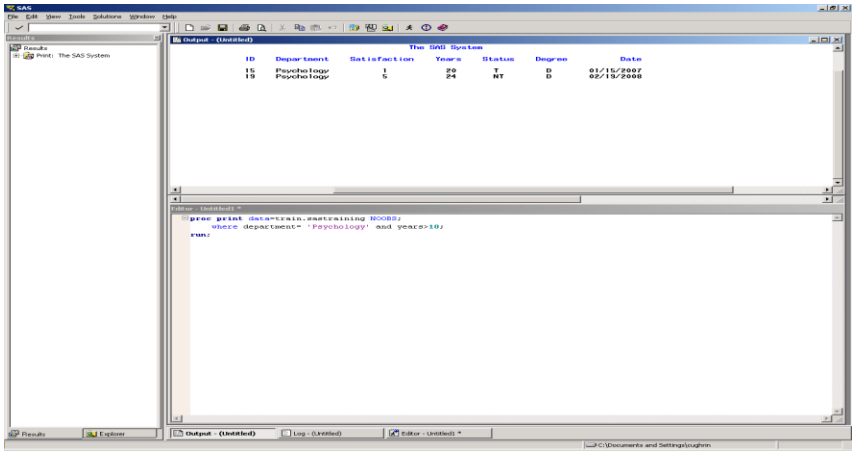


WHERE Logical Operators

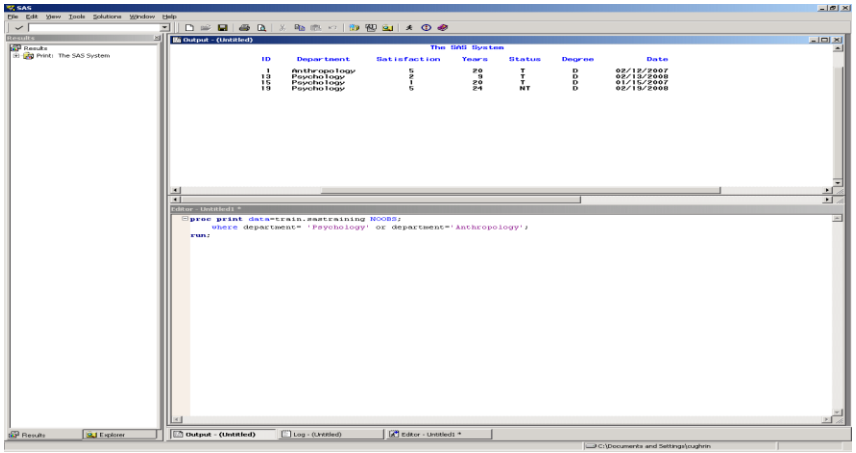


- And (&) Used if both expressions are true, then the compound expression is true.
- OR (|) Used if either expression is true, then the compound expression is true.
- Not (^) Can be combined with other operators to reverse the logic of a comparison.

Examples of WHERE Logical Operators

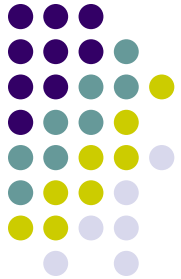


```
proc print data=train.sastraining NOOBS;
    where department= 'Psychology' and
    years>10;
run;
```



```
proc print data=train.sastraining NOOBS;
    where department= 'Psychology' or
    department='Anthropology';
run;
```

WHERE Special Operators

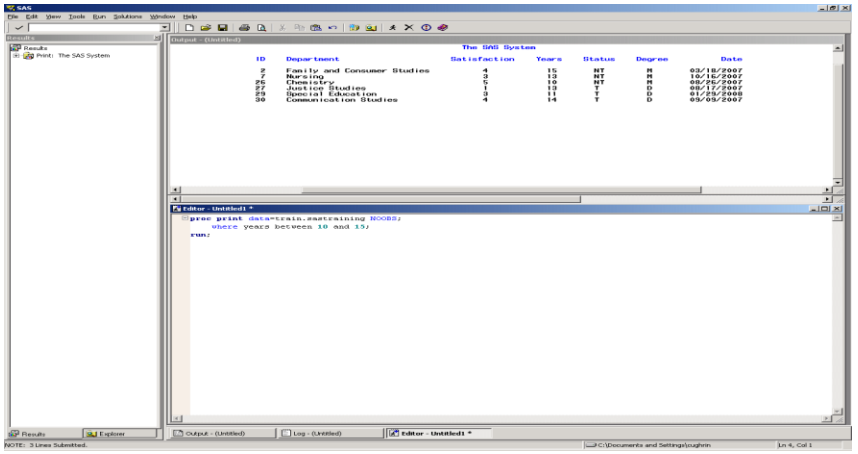


- BETWEEN-AND – Used to select observations in which the value of the variable falls within a range of values.

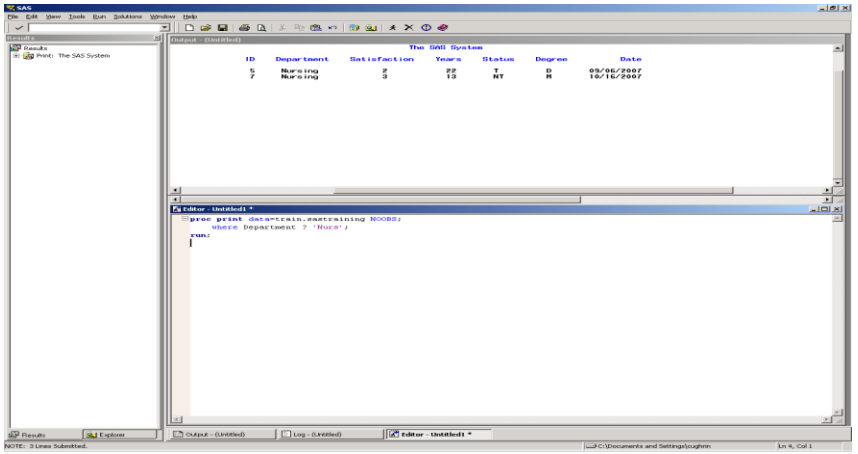
- CONTAINS ? – Used when one wants to select observations that include the specified substring.

Examples of WHERE Special Operators

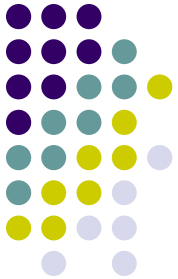
```
proc print data=train.sastraining NOOBS;
    where years between 10 and 15;
run;
```



```
proc print data=train.sastraining NOOBS;
    where Department ? 'Nurs';
run;
```



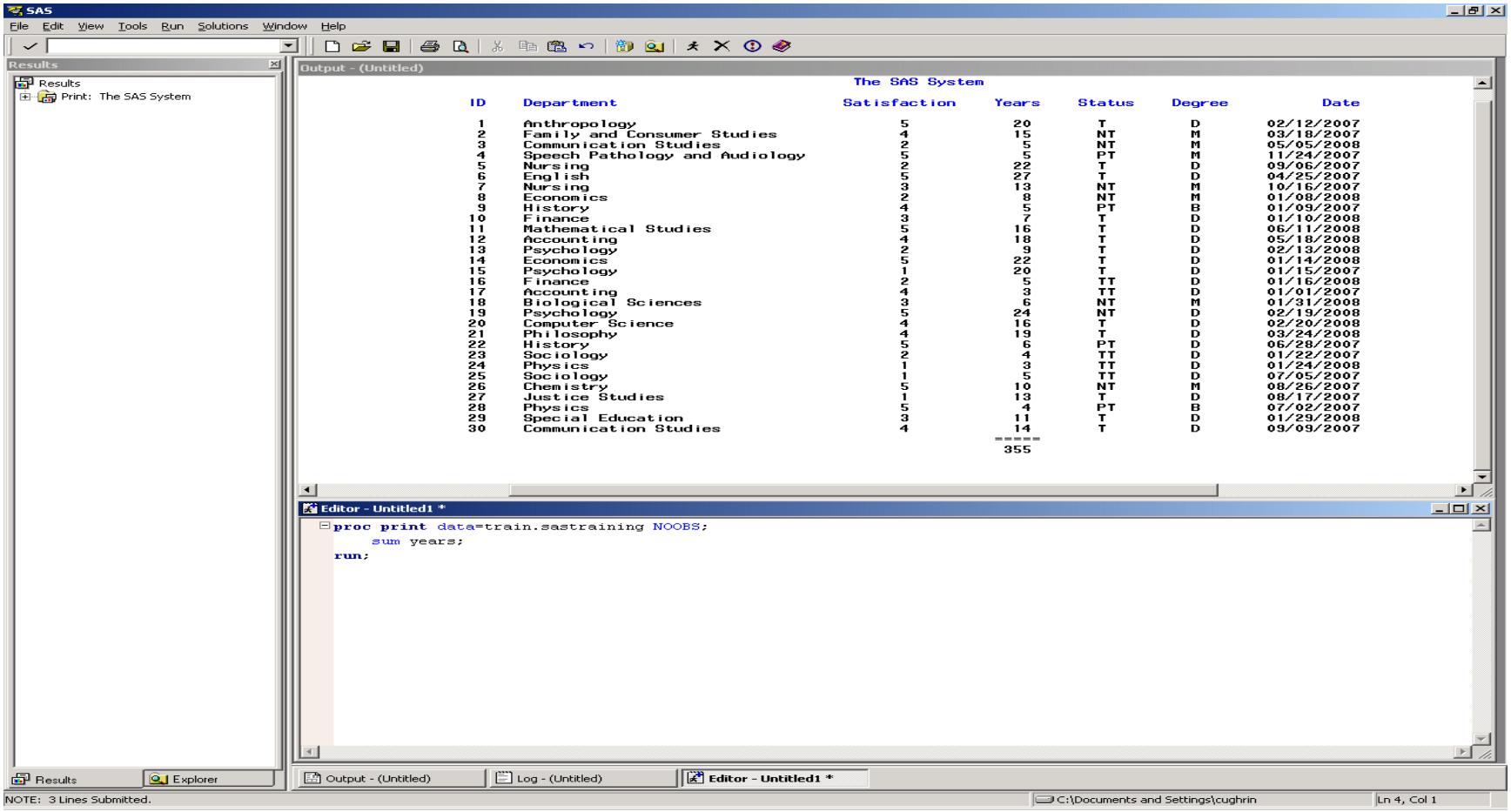
Column Totals

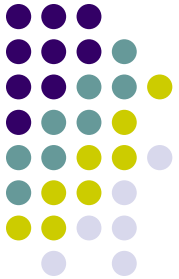


- Can provide a Total

- Can also provide subtotals if data is printed in groups.

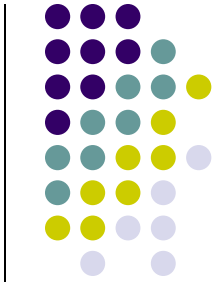
Example of Column Total





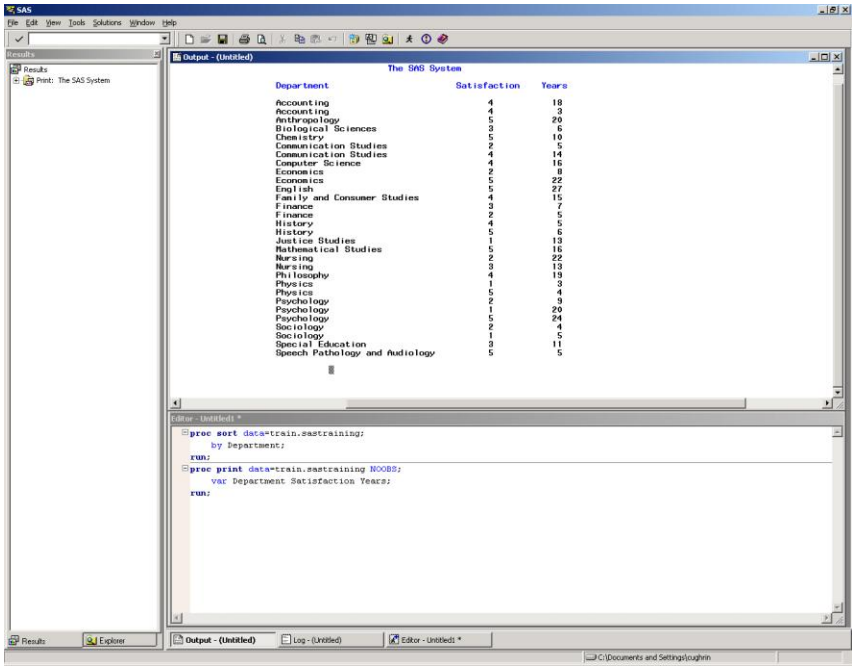
Proc Sort

Overview of Proc Sort



- Sorts (arranges) observations of the data set.
- Can create a new SAS data set containing rearranged observations.
- Can sort on more than one variable at a time.
- Sorts ascending (default) and descending.
- Does not provide printed output (that requires the proc print statements).
- Treats missing data as smallest possible value.

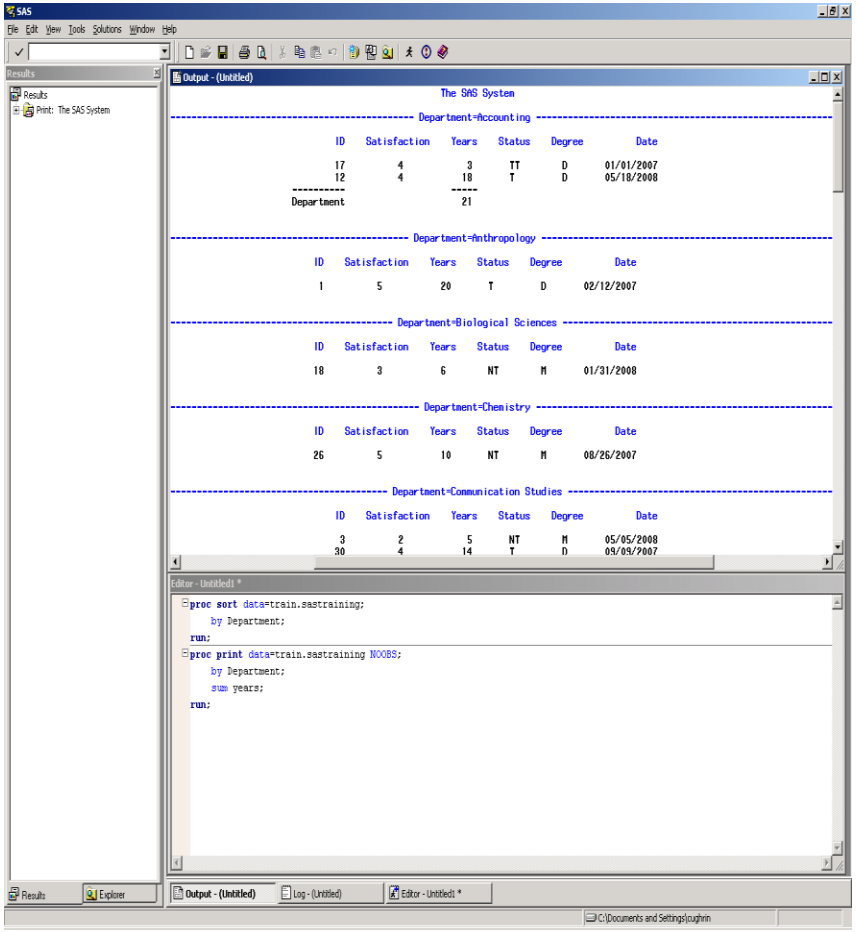
Proc Sort Example



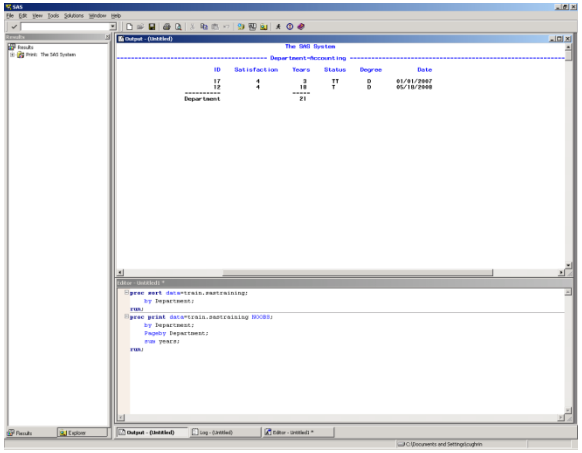

```
proc sort data=train.sastraining;
    by Department;
run;
proc print data=train.sastraining NOOBS;
    var Department Satisfaction Years;
run;
```

Printing Totals and Subtotals Proc Sort and Proc Print Example

```
proc sort data=train.sastraining;
    by Department;
run;
proc print data=train.sastraining NOOBS;
    by Department;
    sum years;
run;
```



Page Breaks with Proc Sort and Proc Print



```
proc sort data=train.sastraining;
    by Department;
run;
proc print data=train.sastraining NOOBS;
    by Department;
    Pageby Department;
    sum years;
run;
```