SAS BASICS

Technology Short Courses: Fall 2008

(October, 2008)

Kentaka Aruga

Object of the course

- Sub-windows in SAS
- Basics of managing data files
- Basic commands in SAS

Introduction: What is SAS?

- What is SAS?
  - Originally an acronym for Statistical Analysis System
  - Provided by SAS institute since the 1970s
  - A software used for statistical analysis, graphing, and presenting data

Introduction: DATA Step

- Two distinct categories
  - DATA step
  - PROC step
- DATA Step
  - Provides data management
  - Use
    - Reading data
    - Data transformation
    - Creating or removing variables

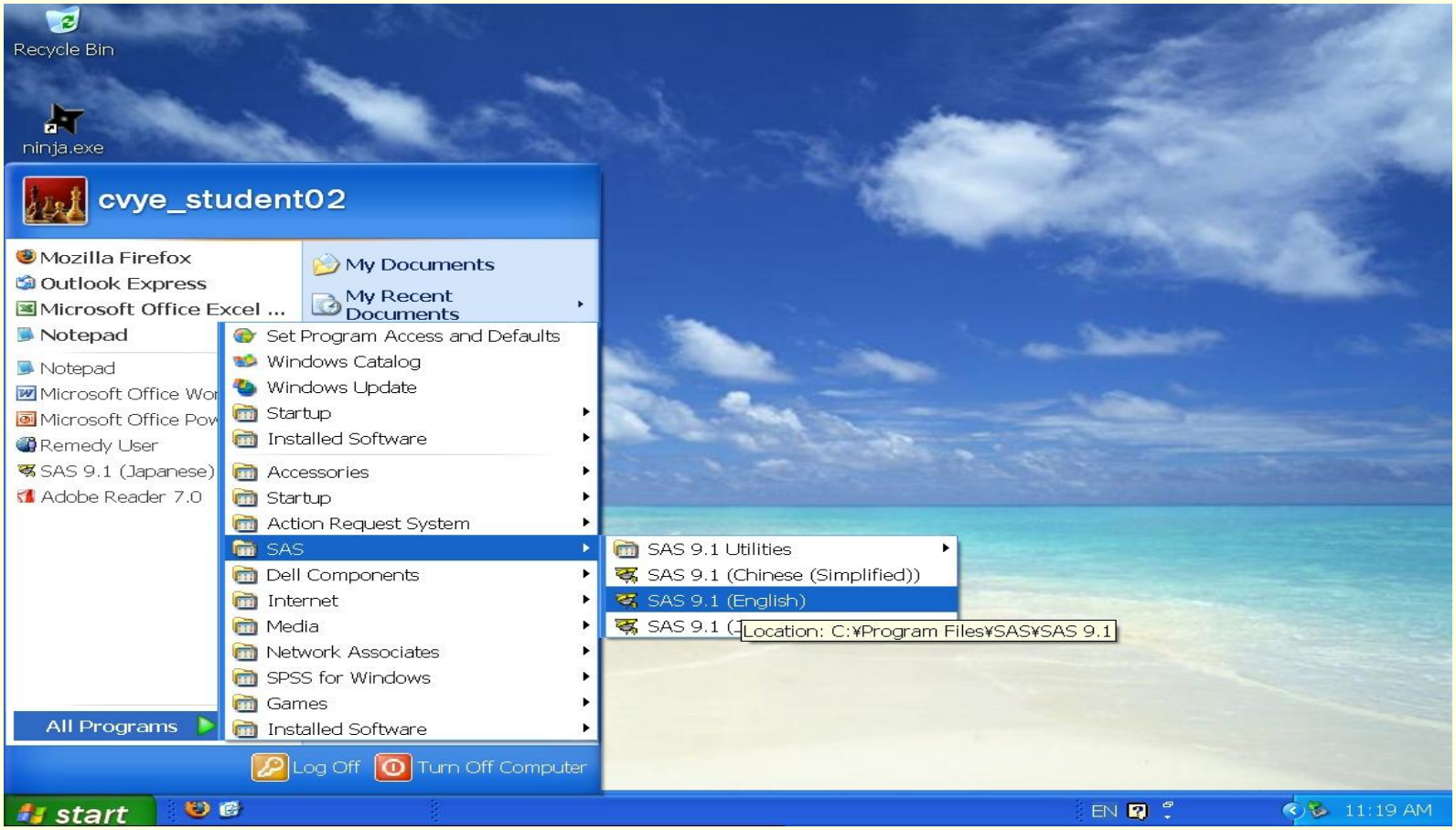
Introduction: PROC Step

- PROC Step
  - Performs a wide variety of analysis on data those are retrieved and transformed from the DATA Step
  - Examples
    - PROC MEANS, CONTENTS, SORT, FREQ, PRINT, PLOT etc.

Section 1
Learning About the Sub-windows

Opening SAS

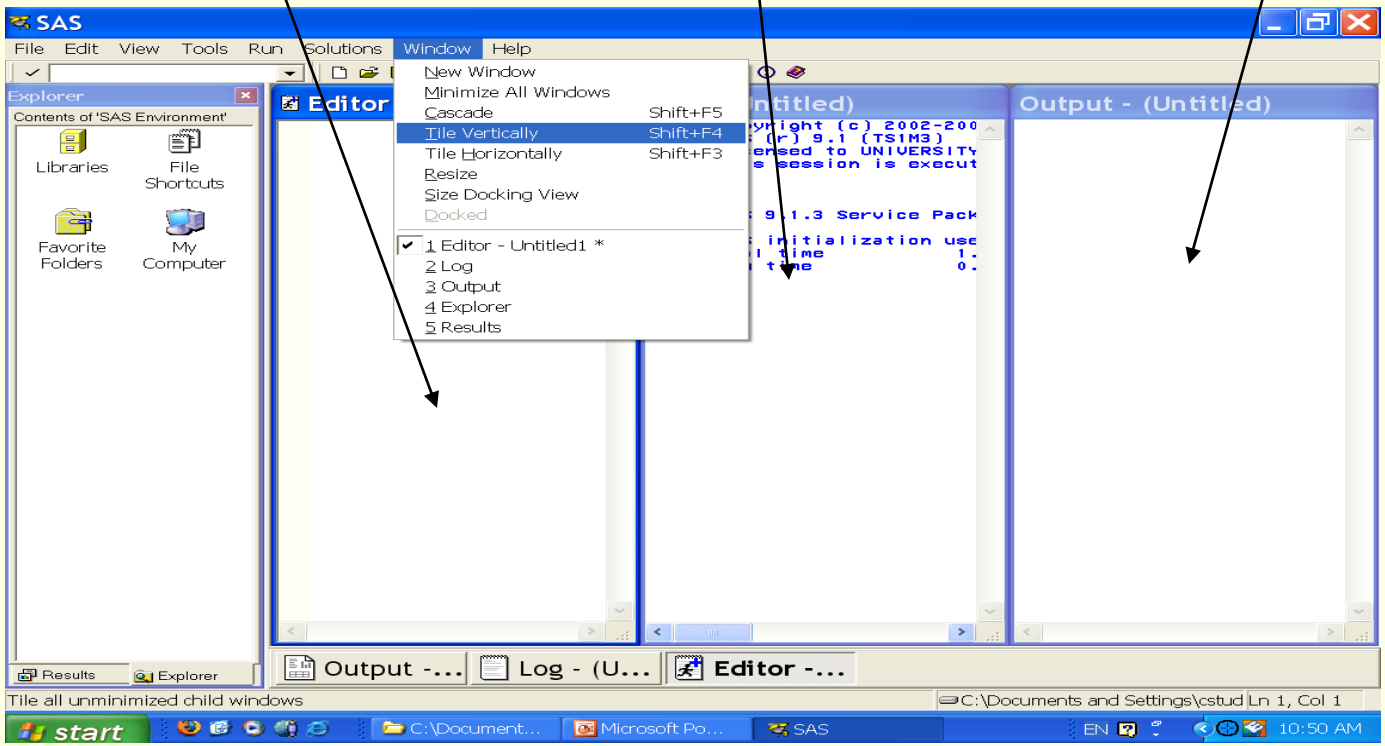
- Start → All Programs → SAS → SAS 9.1



Three main windows: Program editor

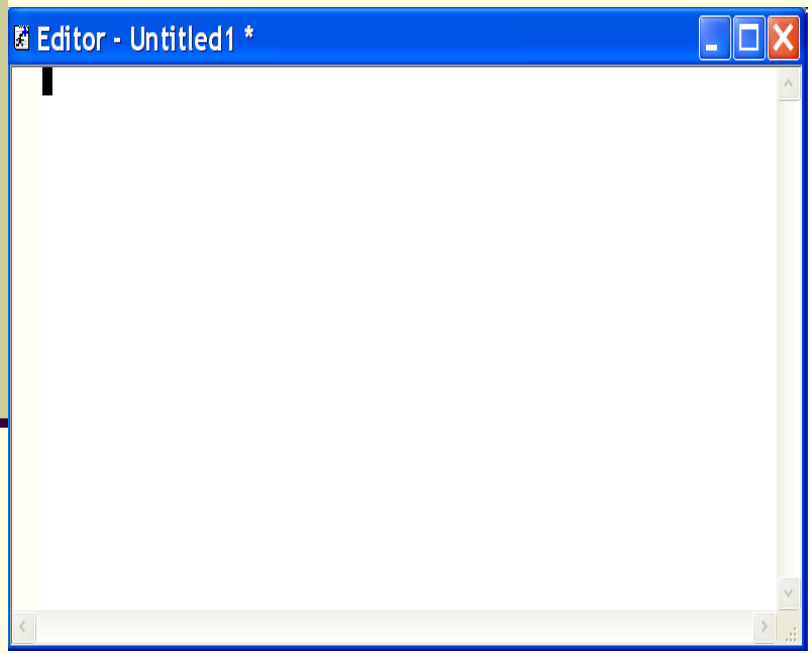
On the top bar click ‘Window’ and then click ‘Tile Vertically.’ You will be able to see three sub-windows.

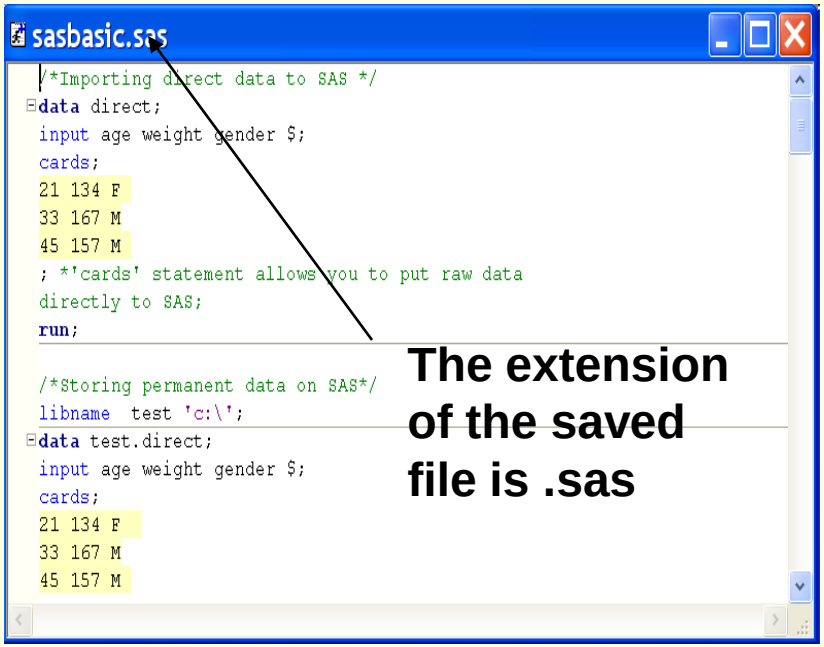
1. Program editor 2. Log window 3. Output window



Three main windows: Program editor

- Program editor
  - Entering and editing SAS command lines

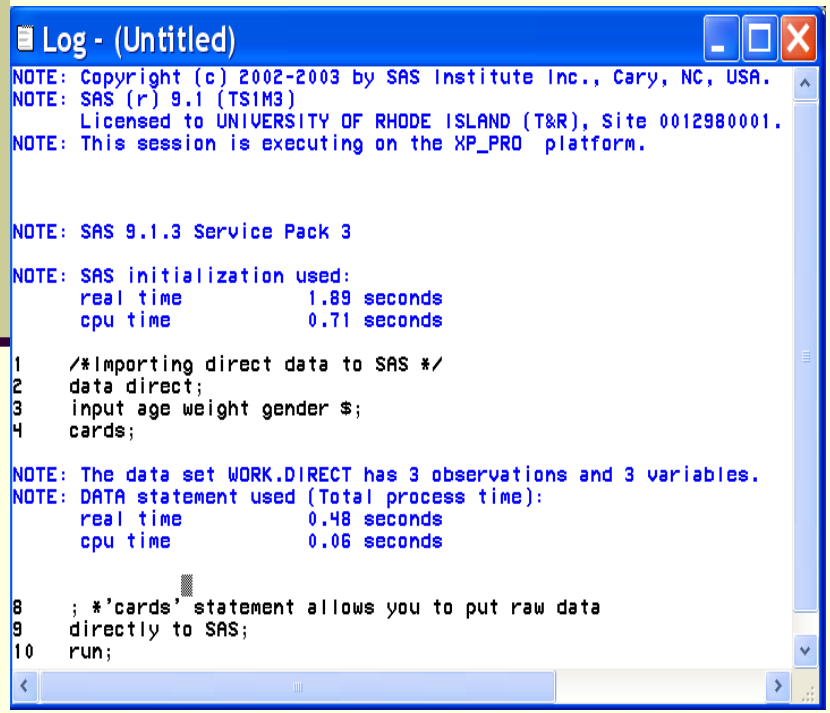




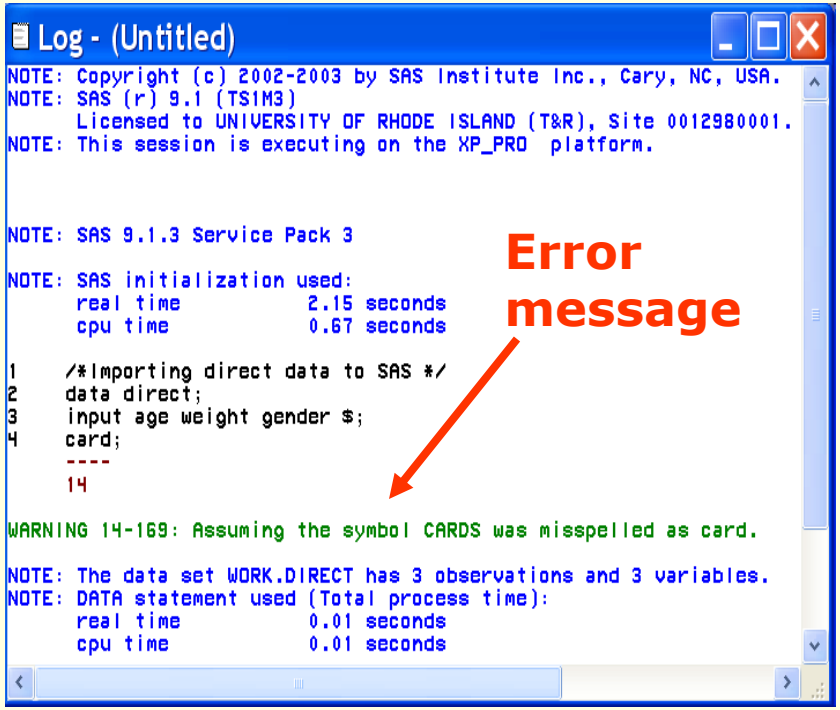
Three main windows: Log window

- Log window
  - This window keeps track of your command runs, and lists SAS notes and error messages (shown in red)

Commands written correctly

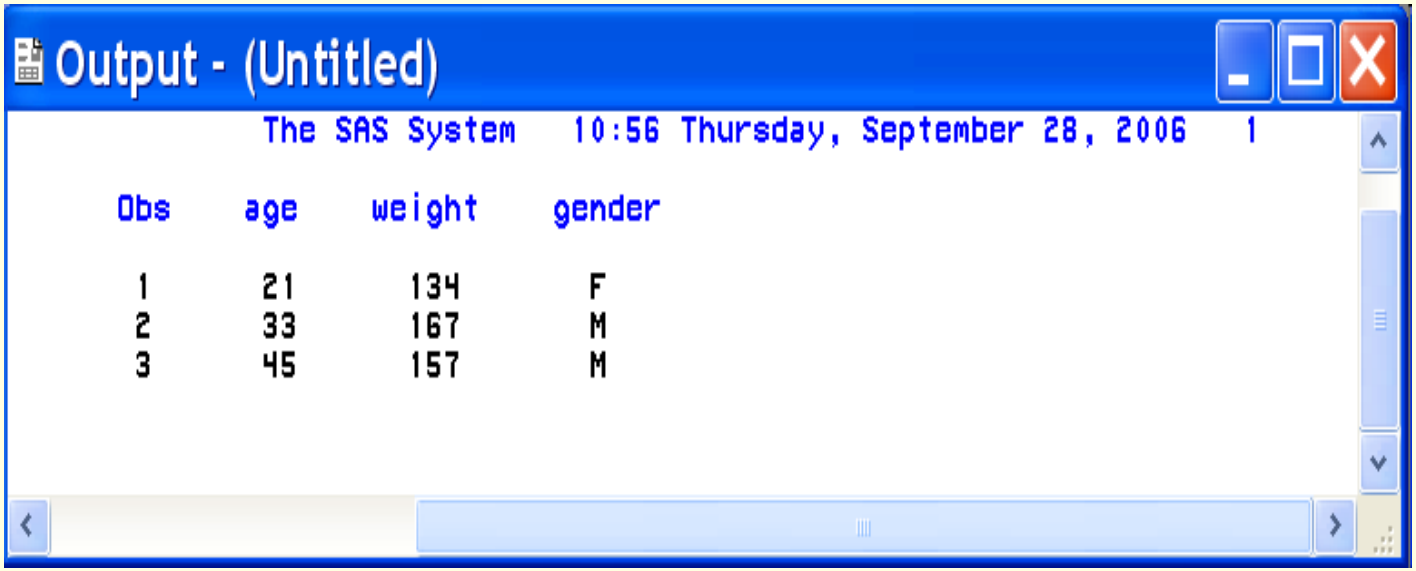


Commands with error



Three main windows: Output window

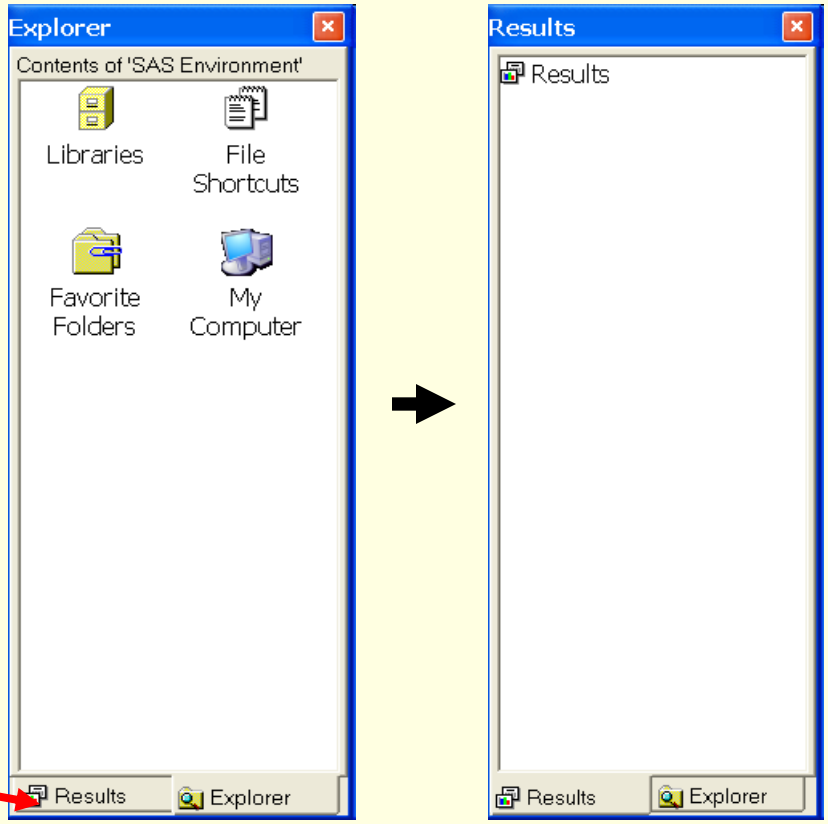
- Output window
  - Shows the results of SAS procedures
  - The extension of the saved file is ".lst"



‘Explorer’ and ‘Results’ window

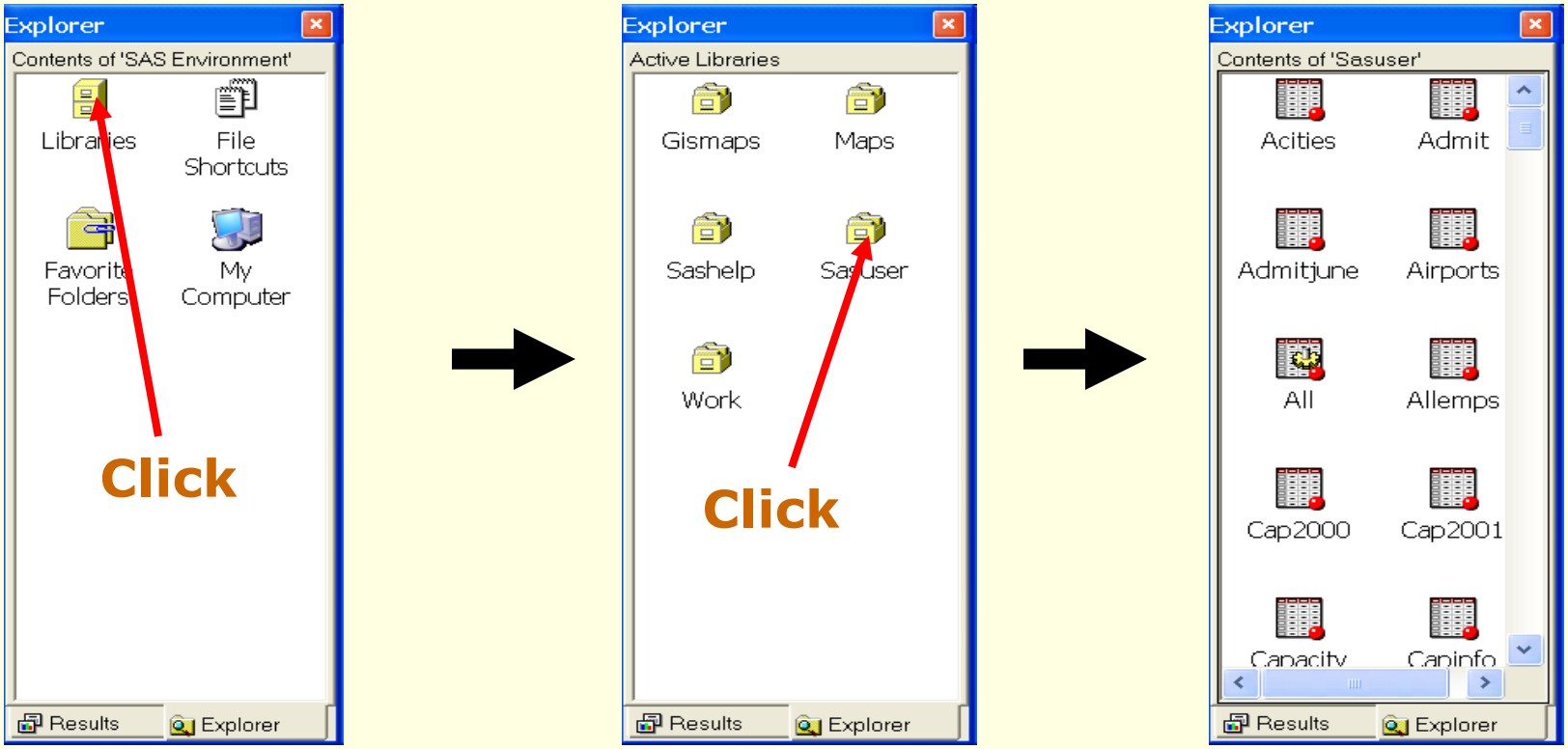
The ‘Explorer’ and ‘Results’ *Windows* will appear on the left side of your screen.

- Explorer window
  - This window is used to explore various default libraries that contain a number of sample SAS data sets
- Results window
  - Organizes the information contained in the Output Window in a hierarchical fashion.



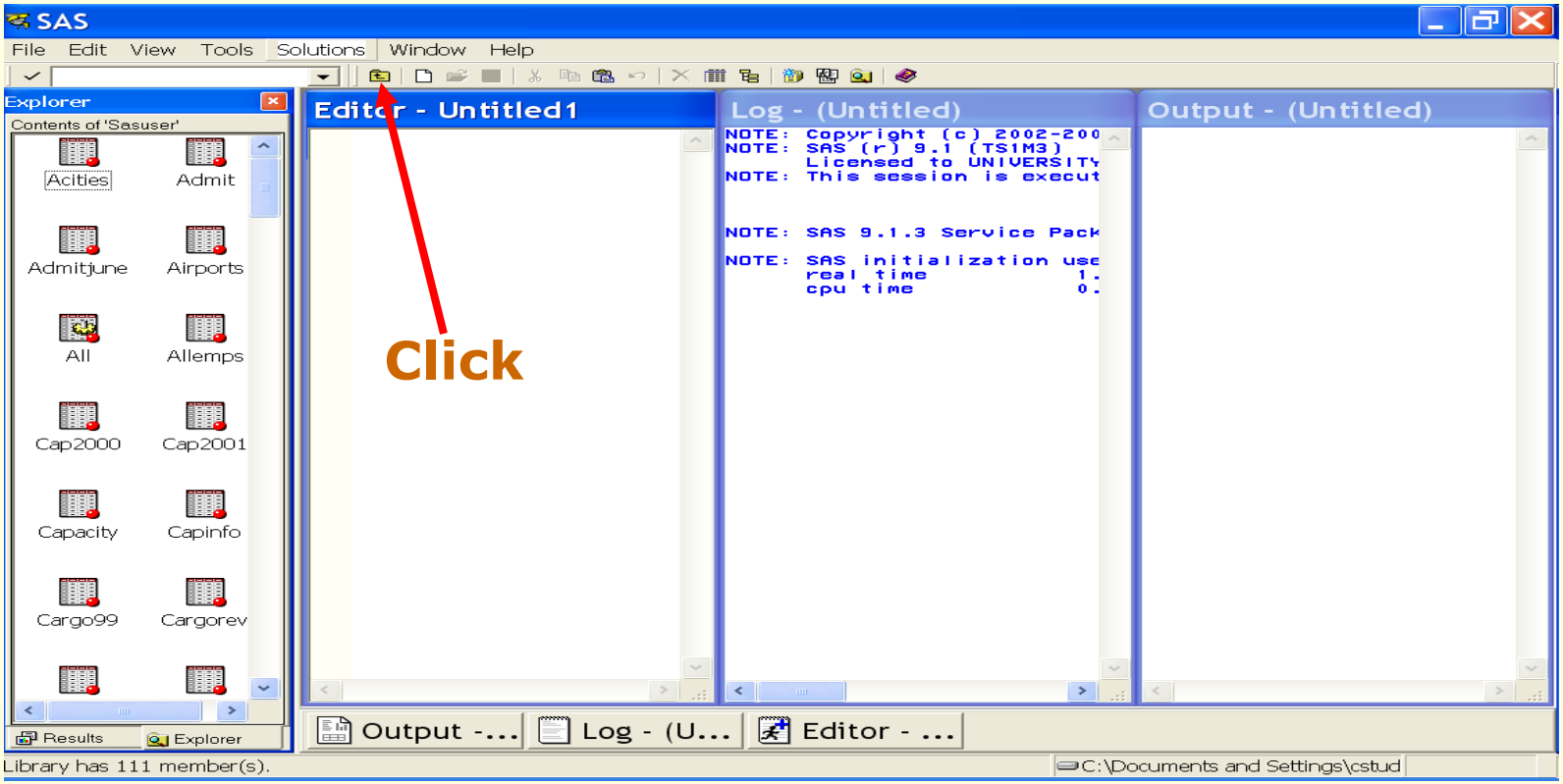
‘Explorer’ window

- Click ‘Libraries’ icon in the Explorer window. Then you will see several subfolders. You can find the raw SAS data in these subfolders.



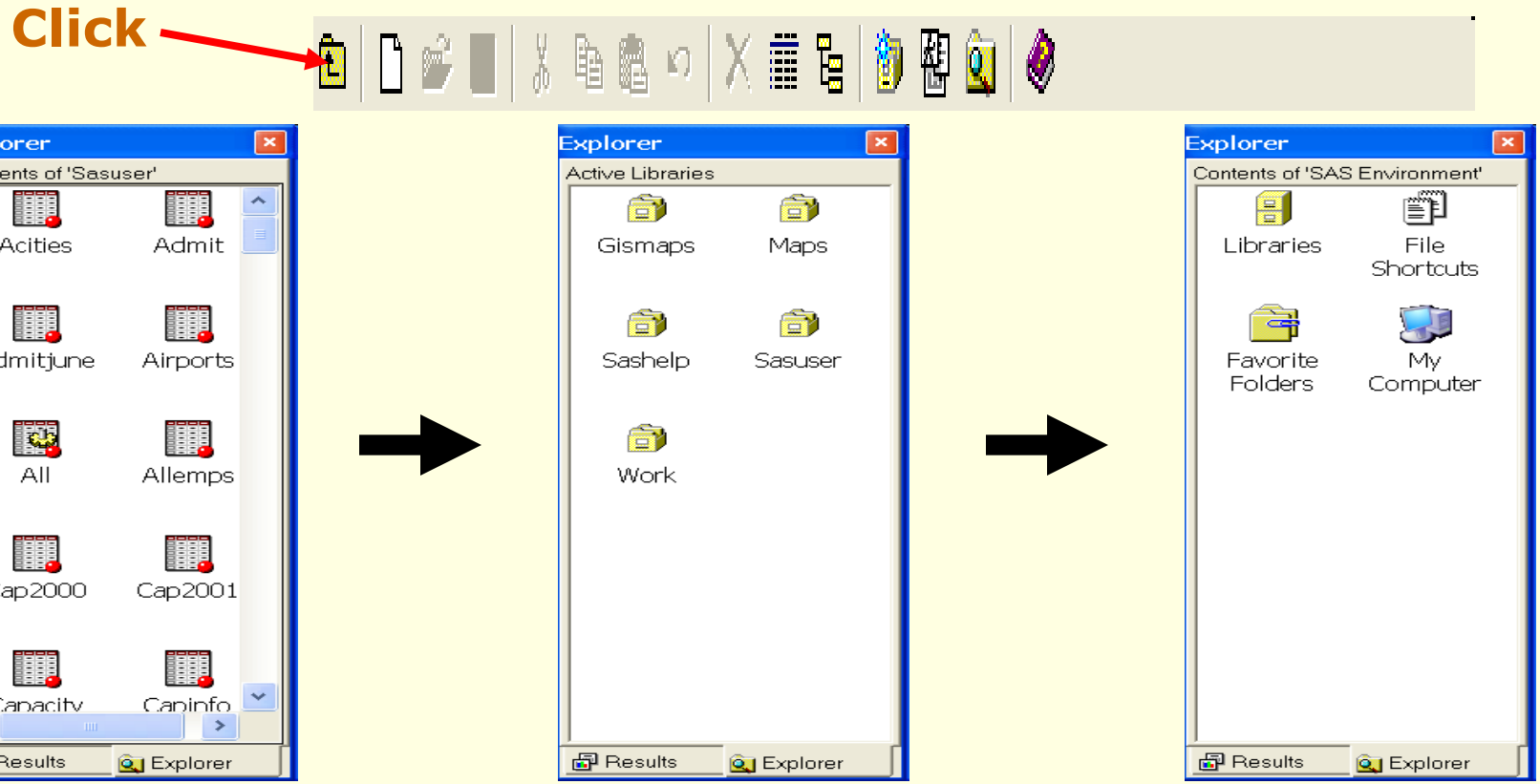
‘Explorer’ window (Cont’d)

- To move backward from one folder to another in the Explorer Window, simply click the left most icon on the toolbar that looks like a folder.



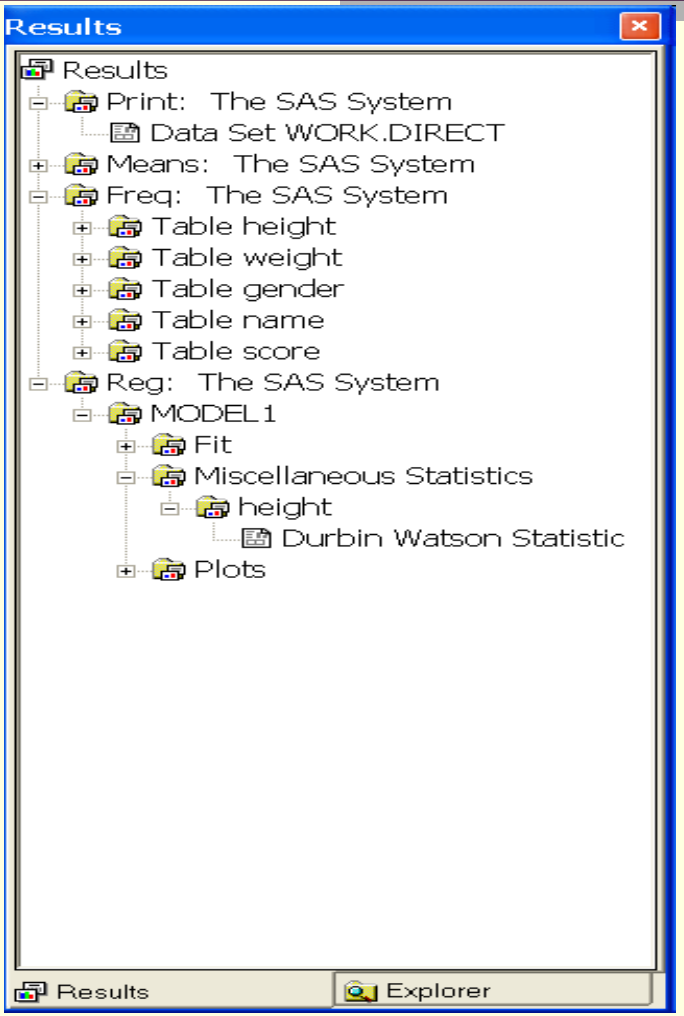
‘Explorer’ window (Cont’d)

- To move backward from one folder to another in the Explorer Window, simply click the left most icon on the toolbar that looks like a folder.



‘Results’ window

- Results window
  - This window allows you to view all the results of procedures you have executed in the program editor.
  - Use the expansion icons (+ or - icons) next to the folder to open or hide its contents.



Points to Remember in SAS program

- All SAS statements begin with a keyword and end with a semicolon (;)
- Except for within the data section, SAS is not sensitive to spacing between words: the amount of space you put between words does not matter.
- Comments are entered in a SAS program using either the following formats:
  - /* comments */ (used for large comment blocks)
  - * comments ; (used for single line comments)

Section 2

Basics of managing data files: DATA step, LIBNAME, PROC export, and PROC import, and data transformation

Practice Round: Getting data

- Go to
  http://www.uri.edu/its/instructional_on-line_materials/sasbasics.html
- Download the SAS program from
  http://www.uri.edu/its/research/basics.txt
- Download two data files from
  http://www.uri.edu/its/research/scores.txt
  http://www.uri.edu/its/research/scores2.txt
- After opening these files, select 'Save As' under File. Save these as C:\ basics.txt, C:\ scores.txt, and C:\ scores2.txt.

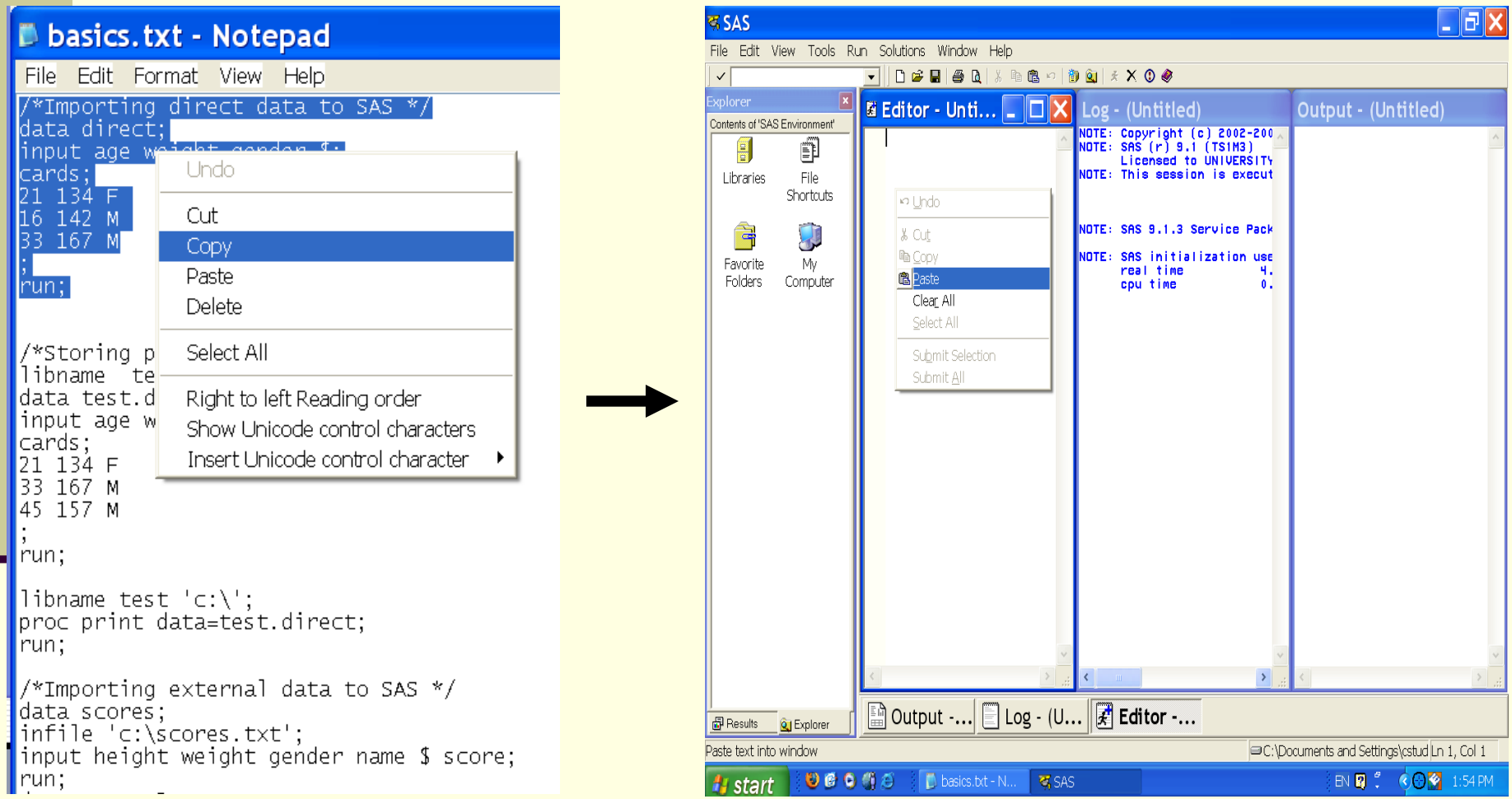
Importing direct data

- Open basics.txt with ‘MS Word’ or ‘Notepad.’
- Drag lines shown below in the file and copy and paste it to the ‘editor ’ window in SAS.

```
data direct;
input age weight gender $;
cards;
21 134 F
33 167 M
45 157 M
;
run;
```

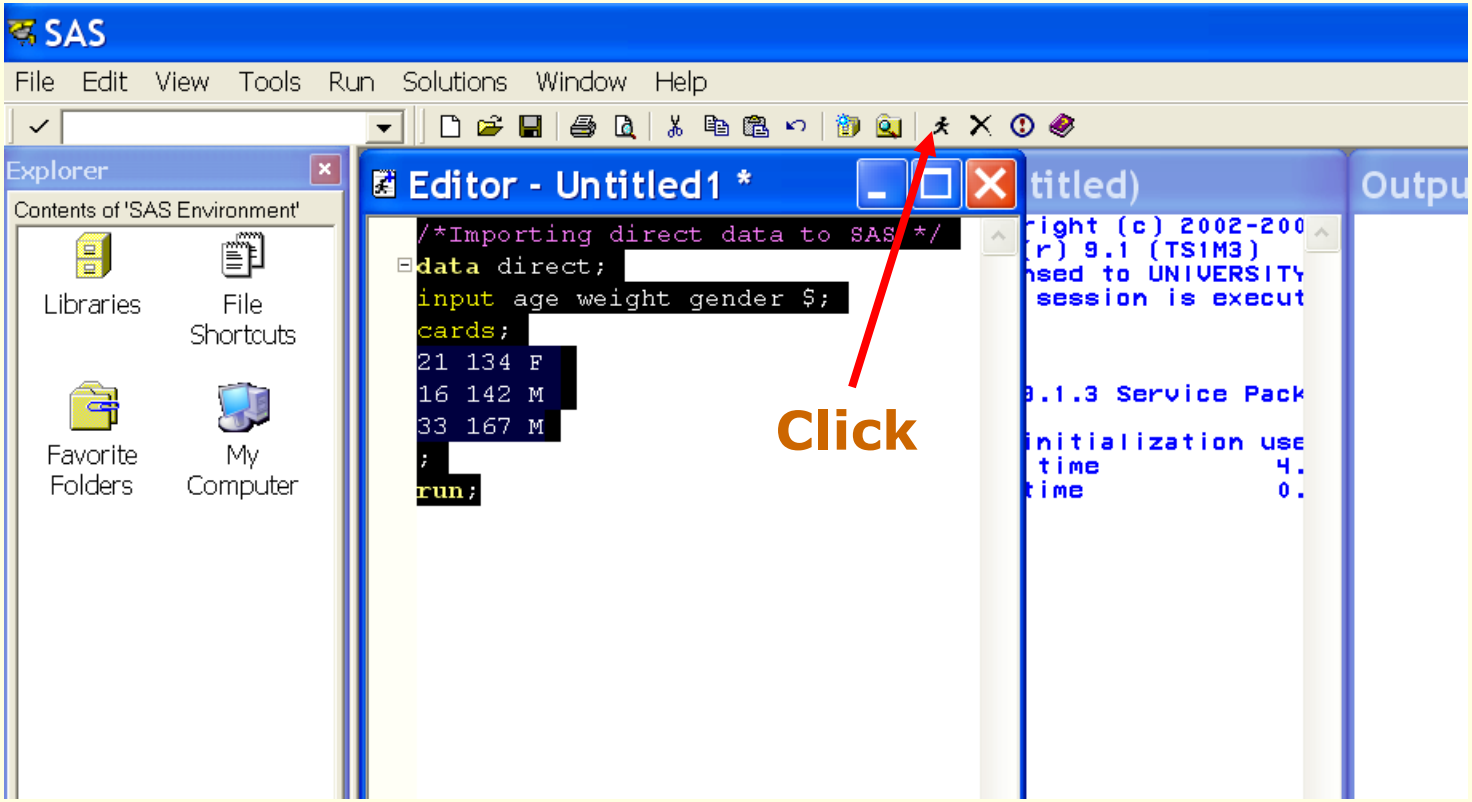
- 'cards' statement allows you to put raw data directly to SAS

You can copy and paste also with your key board. Copy is Ctrl-C and paste is Ctrl-V.



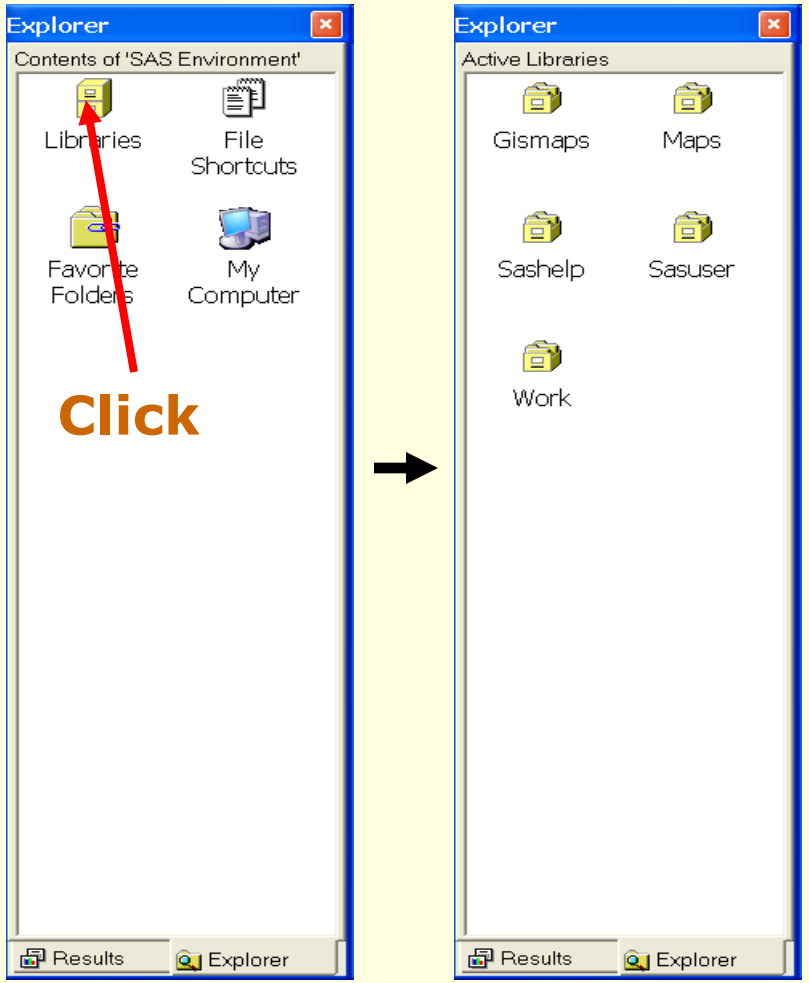
Importing direct data: Executing the commands

- To execute the commands, highlight it and click the 'submit' icon or select 'submit' under the Run menu.



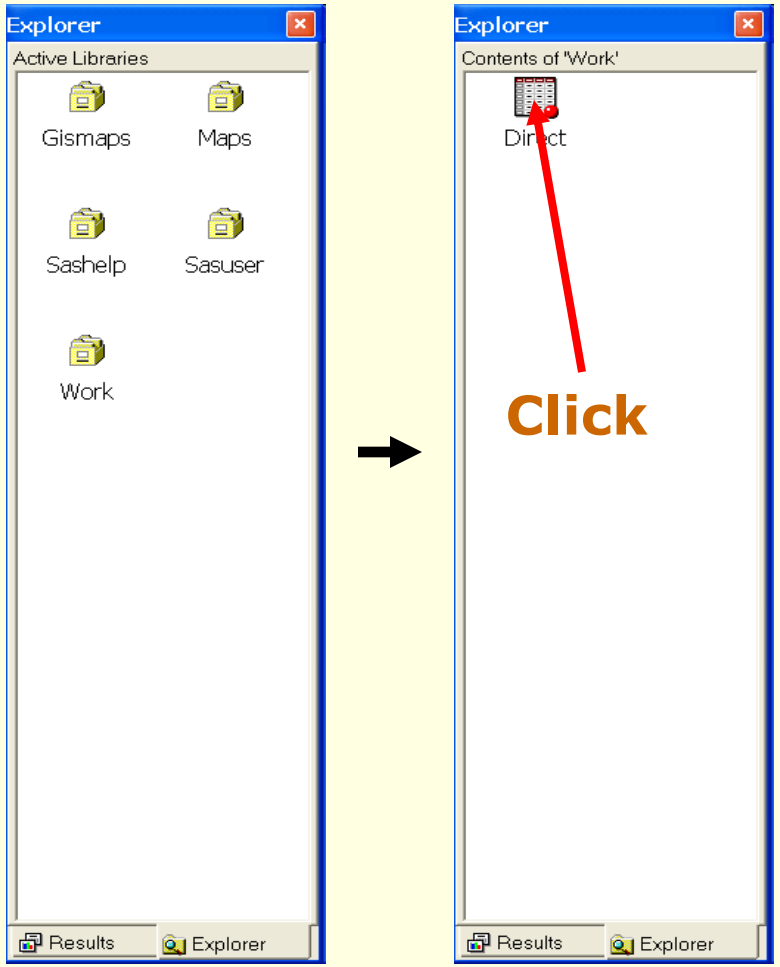
Data command

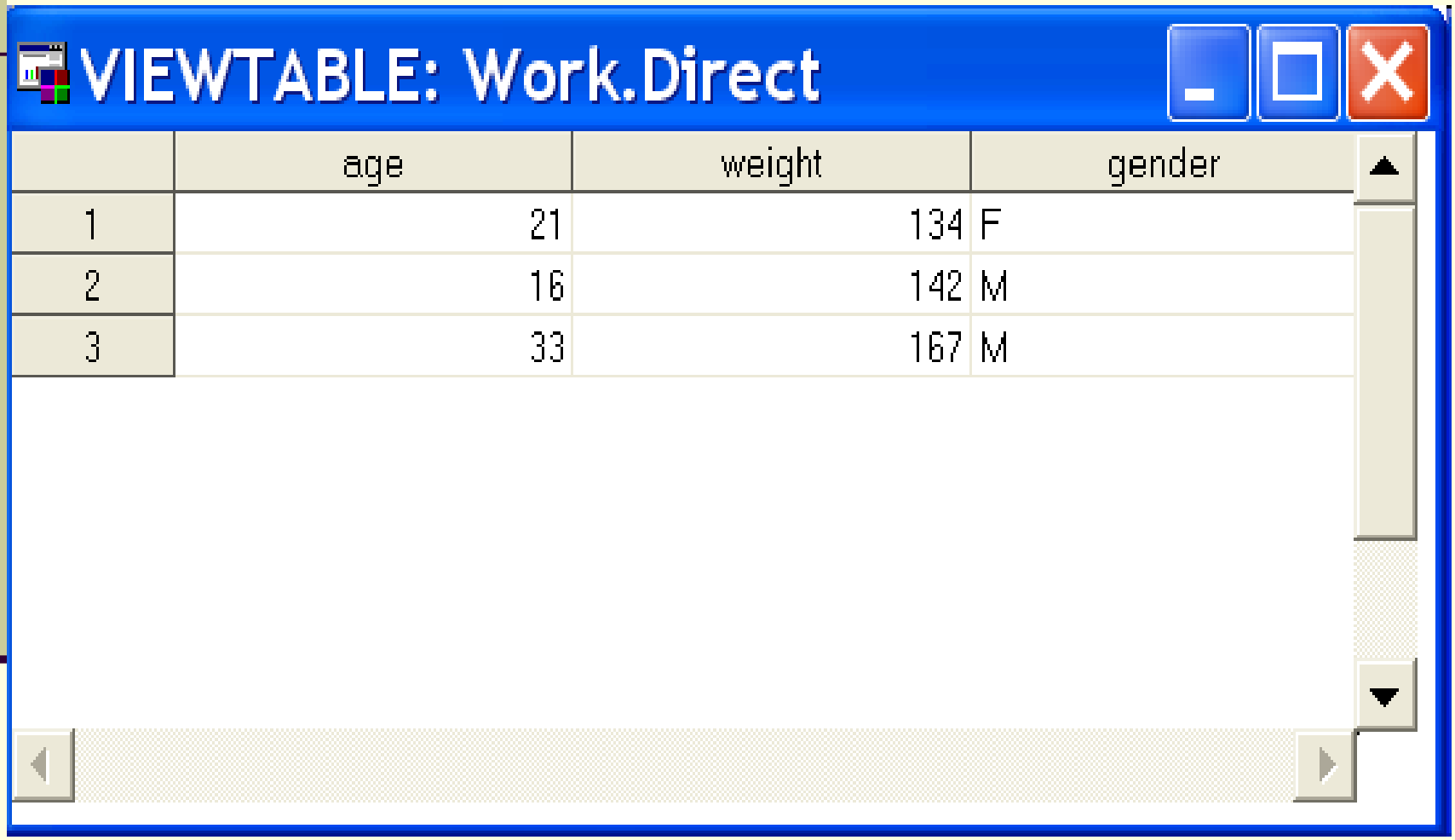
- data direct;
  - Allows SAS to create a temporary SAS data file.
  - In this example the file was named ‘direct’ but you can have your own name by renaming ‘direct.’
  - In the ‘Explorer’ window click Libraries.



Data command: How to see your data in the SAS library

- Now click and go into the ‘Work’ library.
  - You should see the ‘direct’ file you have just created in the library.
- Finally click the ‘direct’ file in the work library.
  - You should be able to see the ‘veiwtable window’

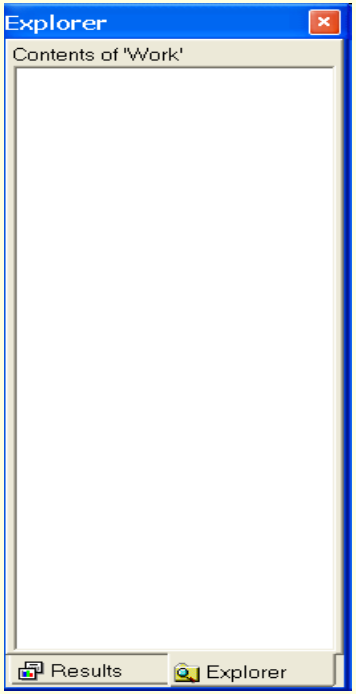




	age	weight	gender
1	21	134	F
2	16	142	M
3	33	167	M

‘Work’ library

- The data in the ‘Work’ library is not stored permanently in SAS. The work folder store files only **temporarily**. Once you exit the SAS program the file will be erased from the folder.
  - End SAS session.
  - Open SAS again and look in the Work library. NO DATA FILES!



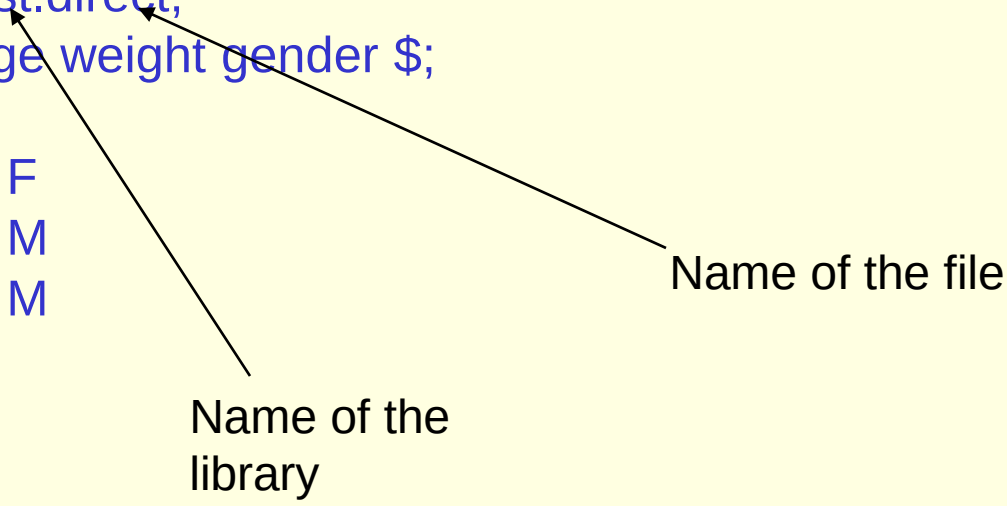
LIBNAME statement

- To store the data permanently, you need to create and reference a library

  ⇒ Use LIBNAME statement

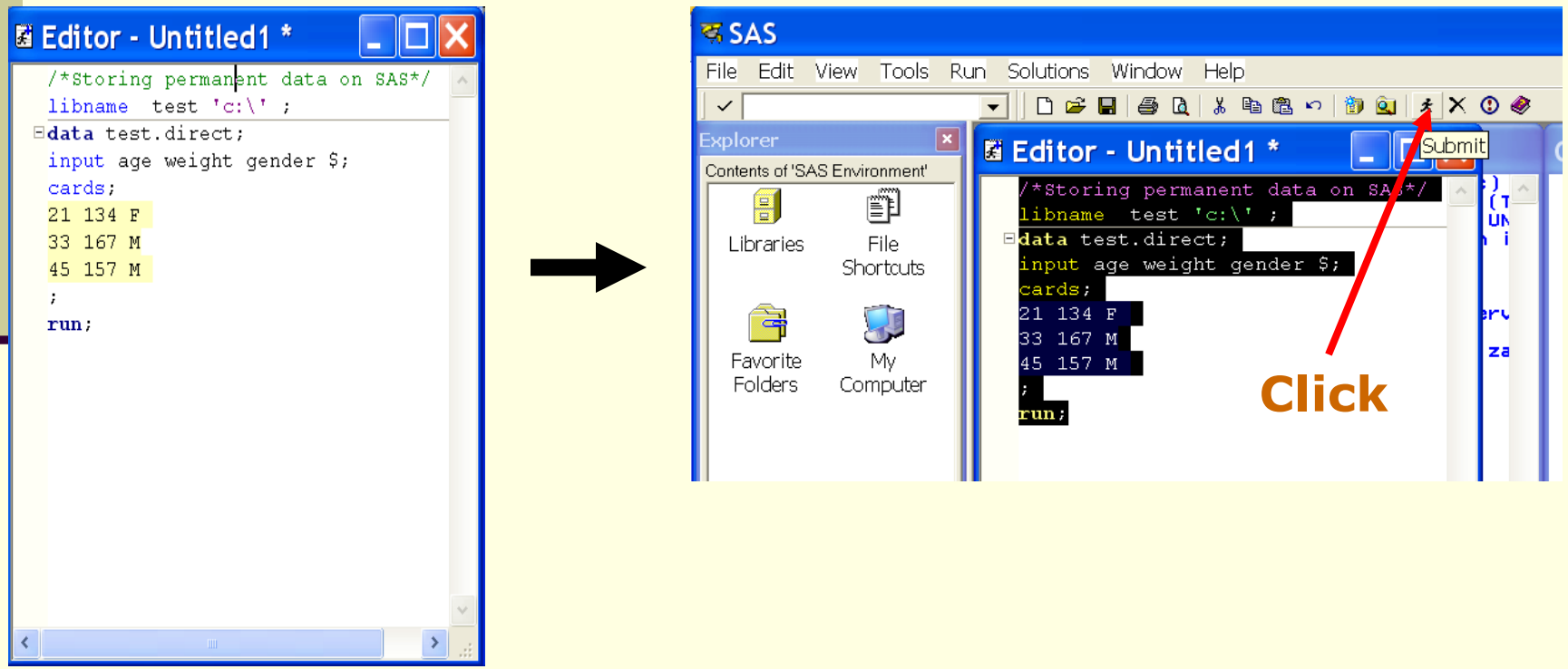
- Drag the lines shown below from the file ‘basics.txt’ and copy and paste it to the ‘editor ’ window in SAS.

```
libname  test 'C:\' ;
data test.direct;
input age weight gender $;
cards;
21 134 F
33 167 M
45 157 M
;
run;
```



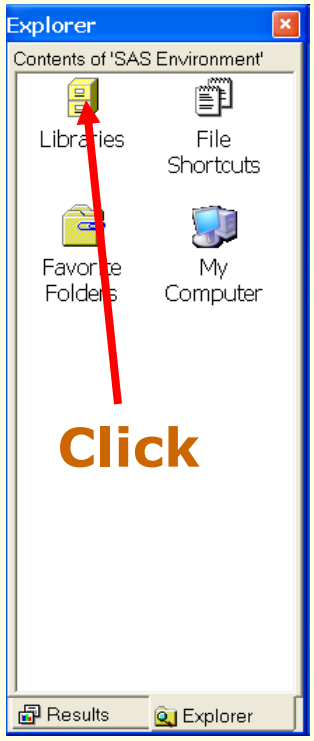
LIBNAME statement (cont'd)

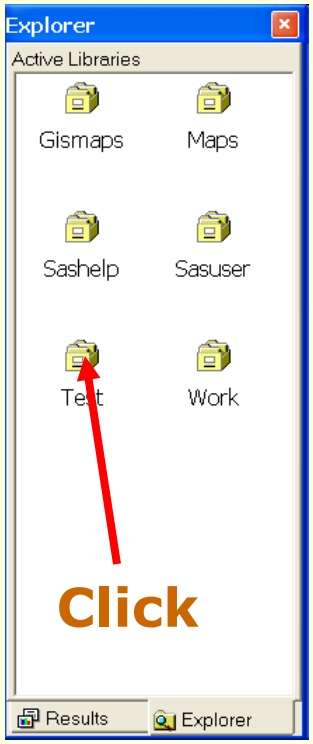
- After pasting the commands to the 'editor' window of SAS, highlight the commands and then click submit.



LIBNAME statement (cont'd)

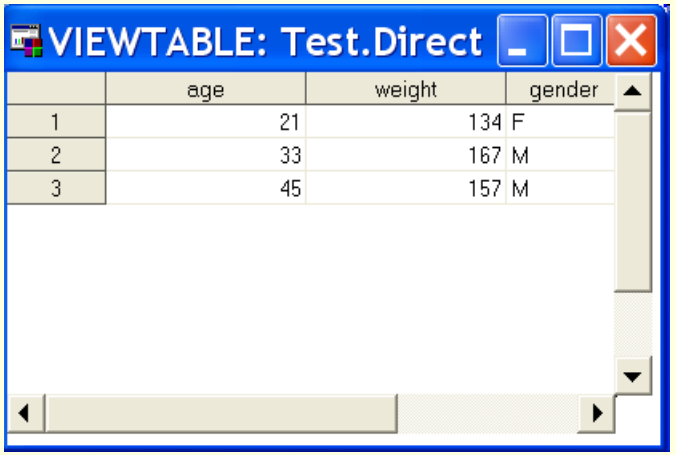
- The command submitted has created a new library named 'test' on SAS, and saved data file 'direct' in this library and in the 'C:\' folder of your computer.
  - In the 'Explorer' window click Libraries. Then go into the 'test' library.
  - Click the 'test library'





LIBNAME statement (cont'd)

- You will now see the 'Direct' file in the 'test' library.
- To view the 'Direct' data file click 'direct
- You will also find the same file in the 'C:\' folder of your computer.



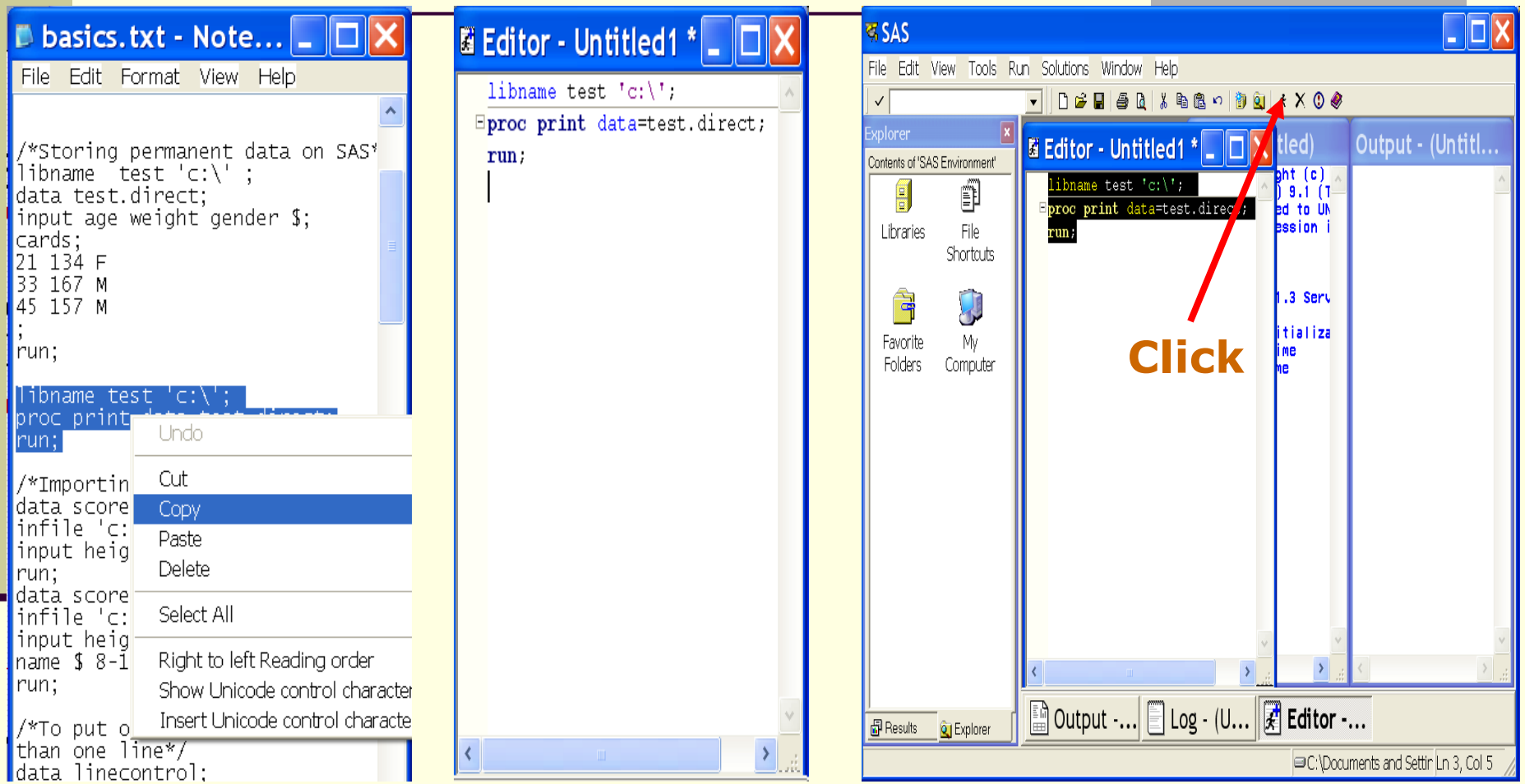
	age	weight	gender
1	21	134	F
2	33	167	M
3	45	157	M

LIBNAME statement (cont'd)

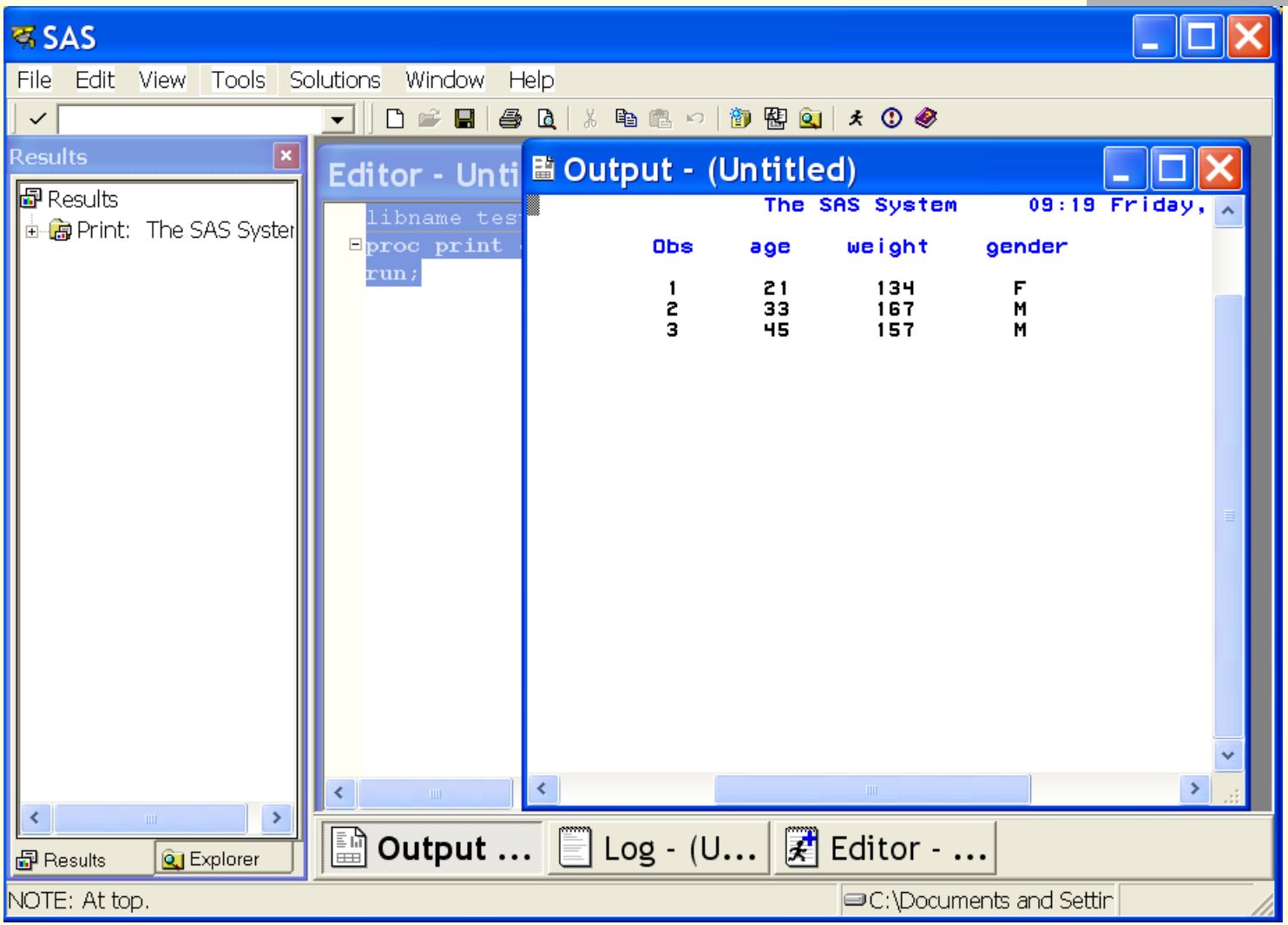
- Once you stored your data file into your C:/ drive with the LIBNAME statement, you can refer to the file without importing the raw data again.
  - Example:
    - Close SAS session, re-open it.
    - Then copy and paste the following commands from 'basics.txt' to the 'editor' window in SAS.

```
libname test 'C:\';
proc print data=test.direct;
run;
```

    - Click the submit icon to execute the command


basics.txt - Note...
File Edit Format View Help
/*Storing permanent data on SAS
libname test 'c:\' ;
data test.direct;
input age weight gender $;
cards;
21 134 F
33 167 M
45 157 M
;
run;
libname test 'c:\';
proc print
run;
Undo
Cut
Copy
Paste
Delete
Select All
Right to left Reading order
Show Unicode control character
Insert Unicode control characte
/*Importin
data score
infile 'c:
input heig
run;
data score
infile 'c:
input heig
name $ 8-1
run;
/*To put o
than one line*/
data linecontrol;
Editor - Untitled1 *
libname test 'c:\';
proc print data=test.direct;
run;
SAS
File Edit View Tools Run Solutions Window Help
Explorer
Contents of 'SAS Environment'
Libraries
File Shortcuts
Favorite Folders
My Computer
Results
Explorer
Editor - Untitled1 *
libname test 'c:\';
proc print data=test.direct;
run;
Click
Output - (Untitl...
Output -...
Log - (U...
Editor -...
C:\Documents and Settir
Ln 3, Col 5

You will see the same data as before!



Forms of INPUT statement

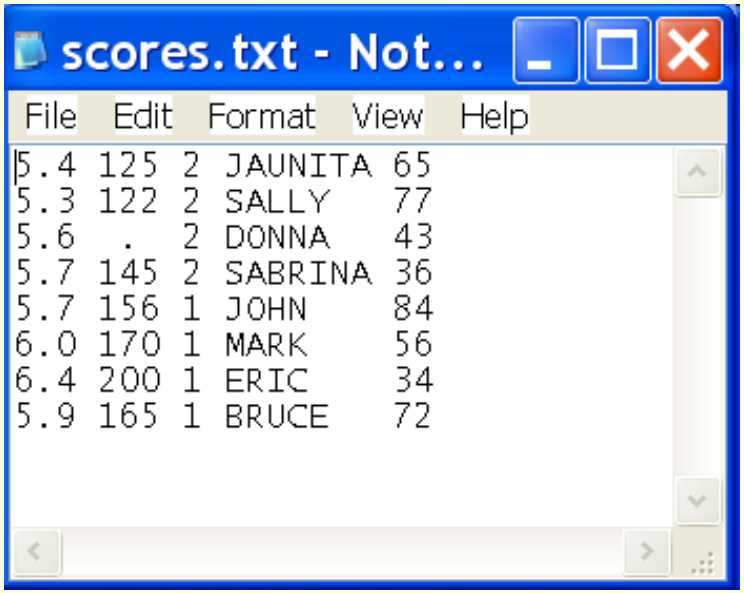
- Example 1
  - input age weight gender $;
    - This statement allows SAS to read the variables used for the raw data.
    - In this example three variables (age, weight, and gender) were put into SAS
    - SAS initially only reads numeric variable so in order to read character values you need to use modifiers:
      - The variable ‘gender’ is a character variable. You need to use ‘$’
        - $: enables SAS to read character values with default size of eight characters with no embedded blanks
      - &: enables SAS to read character values with embedded blanks

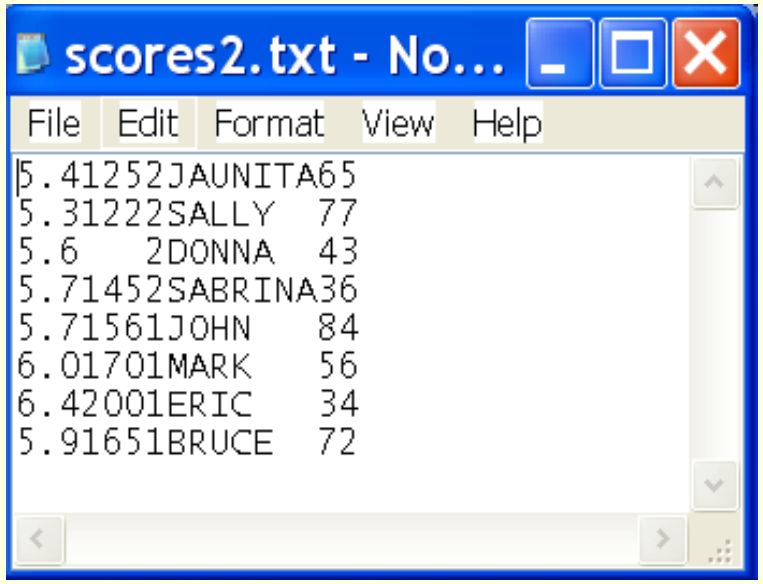
INPUT statement: Example 2

- input height **1-3** weight **4-6** gender **7** name $ **8-14** score **15-16**;
  - If the data contain the followings you need to set up a column input mode to specify the column positions of the pointer
    - Standard character and numeric data
    - Values entered in fixed column positions
    - Character values longer than eight characters
    - Character values with embedded blanks

Importing external data

- Open scores.txt, and scores2.txt from c:/ drive and compare.





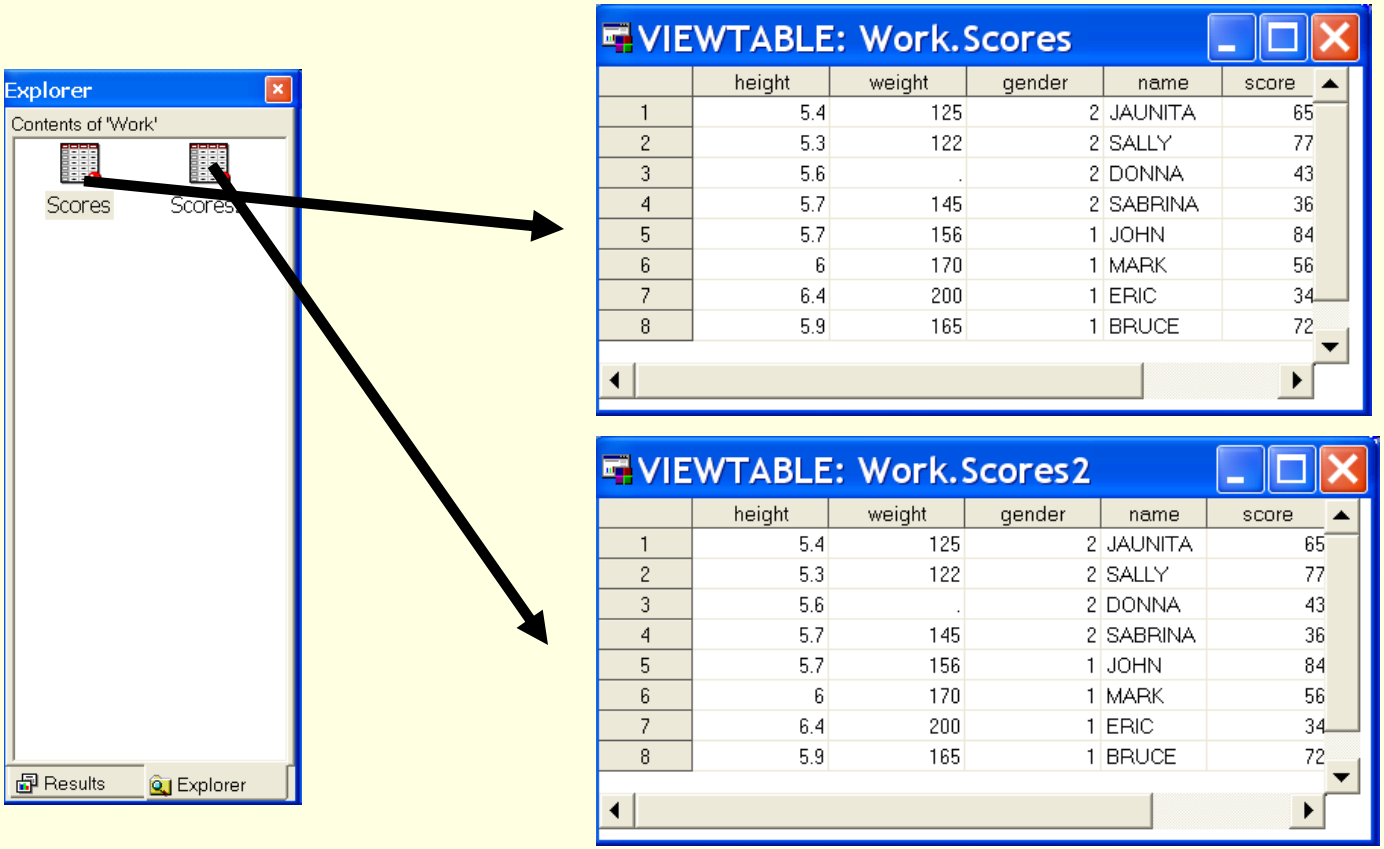
Importing external data (Cont'd)

- Open basics.txt with 'MS Word' or 'Notepad.'
- Drag the lines shown below on the file, copy and paste it to 'editor ' window in SAS, and execute the commands.

```
data scores;
infile 'C:\scores.txt';
input height weight gender name $ score;
run;
data scores2;
infile 'C:\scores2.txt';
input height 1-3 weight 4-6 gender 7
name $ 8-14 score 15-16;
run;
```

Importing external data (Cont'd)

- Go to the 'Explorer' window, click the work folder, and open 'scores' and 'scores2.' You will see exactly the same file.



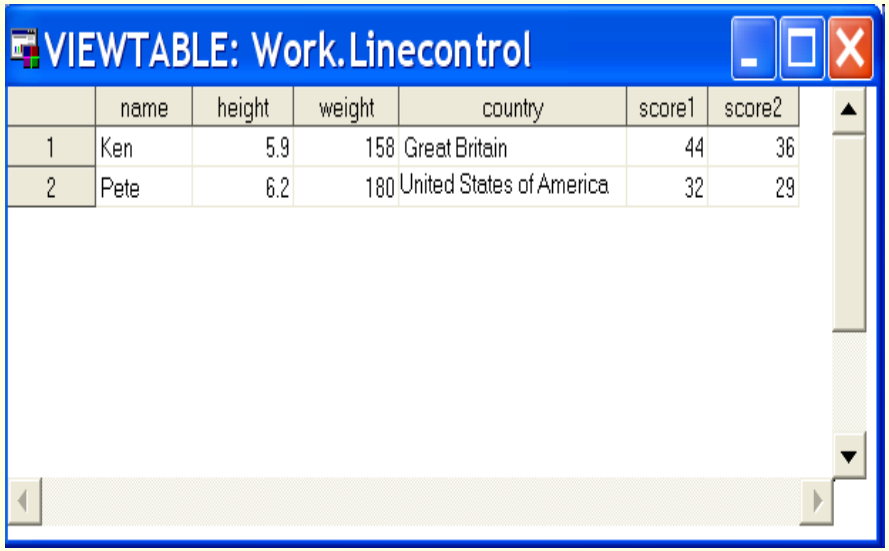
	height	weight	gender	name	score
1	5.4	125	2	JAUNITA	65
2	5.3	122	2	SALLY	77
3	5.6	.	2	DONNA	43
4	5.7	145	2	SABRINA	36
5	5.7	156	1	JOHN	84
6	6	170	1	MARK	56
7	6.4	200	1	ERIC	34
8	5.9	165	1	BRUCE	72

	height	weight	gender	name	score
1	5.4	125	2	JAUNITA	65
2	5.3	122	2	SALLY	77
3	5.6	.	2	DONNA	43
4	5.7	145	2	SABRINA	36
5	5.7	156	1	JOHN	84
6	6	170	1	MARK	56
7	6.4	200	1	ERIC	34
8	5.9	165	1	BRUCE	72

INPUT statement: Example 3

- How to put observations in more than one line
  - #n: moves the pointer to record n.
    - Example

```
data linecontrol;
input #1 name $ height weight #2 country & $24.
#3 score1 score2;
cards;
Ken 5.9 158
Great Britain
44 36
Pete 6.2 180
United States of America
32 29
;
run;
```

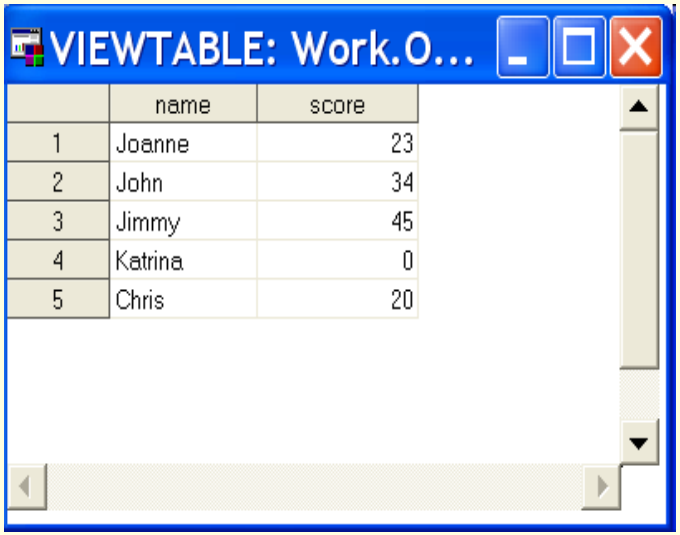


	name	height	weight	country	score1	score2
1	Ken	5.9	158	Great Britain	44	36
2	Pete	6.2	180	United States of America	32	29

INPUT statement: Example 4

- How to put several observations in one line
  - @@: Used when each input line contains values for several observations
    - Example

```
data oneline;
input name $ score @@;
cards;
Joanne 23 John 34 Jimmy 45
Katrina 0 Chris 20
;
run;
```



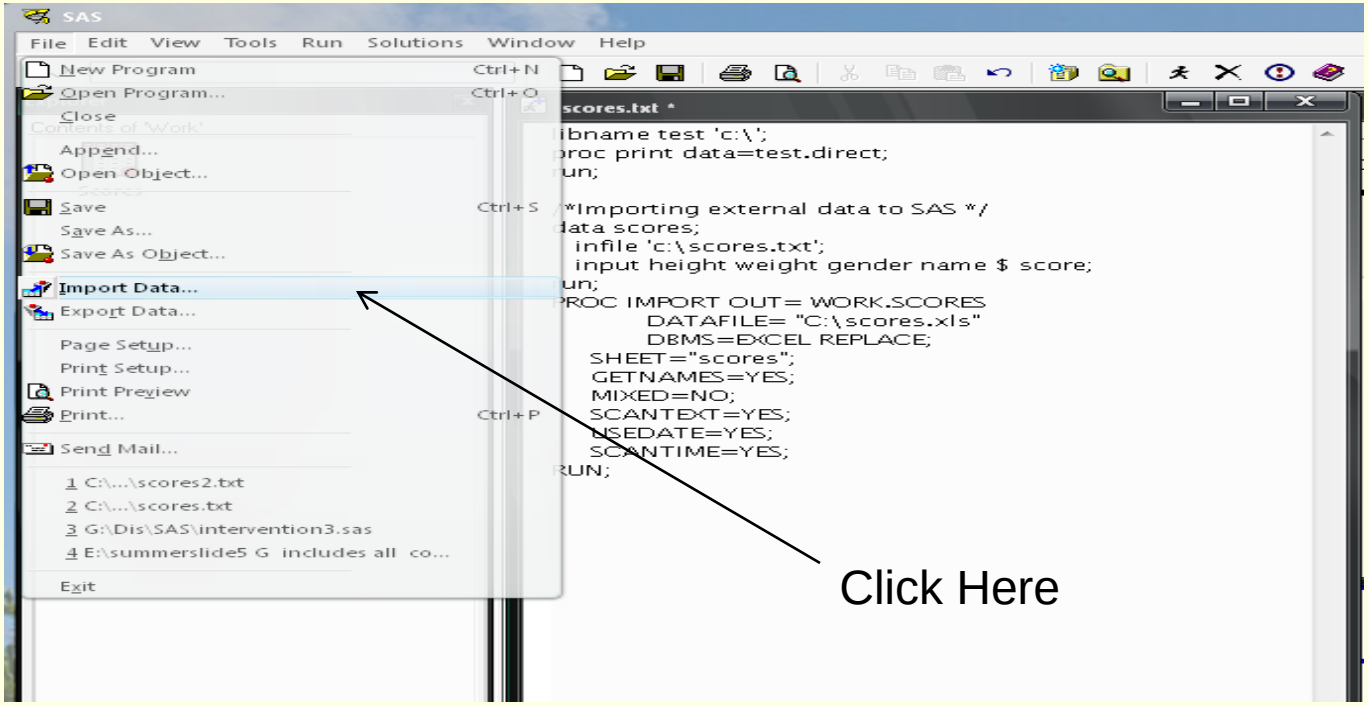
	name	score
1	Joanne	23
2	John	34
3	Jimmy	45
4	Katrina	0
5	Chris	20

Exporting & Importing MS Excel data 1

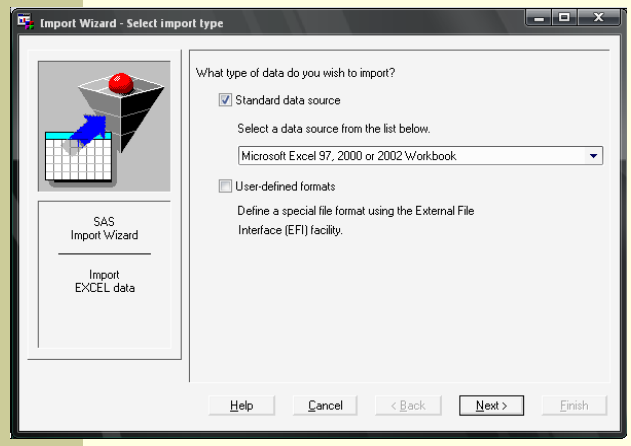
```
/*Exporting data to MS Excel data*/
        proc export data=scores
                    outfile="C:\scores.xls"
                    dbms=excel2000 replace;
                    sheet="scores";
                    run;
/*Importing data from MS Excel*/
        proc import out=impscores
                    datafile="C:\scores.xls"
                    dbms=excel2000  replace;
                    sheet="scores";
                    getnames=yes;
                    mixed=yes;
        run;
```

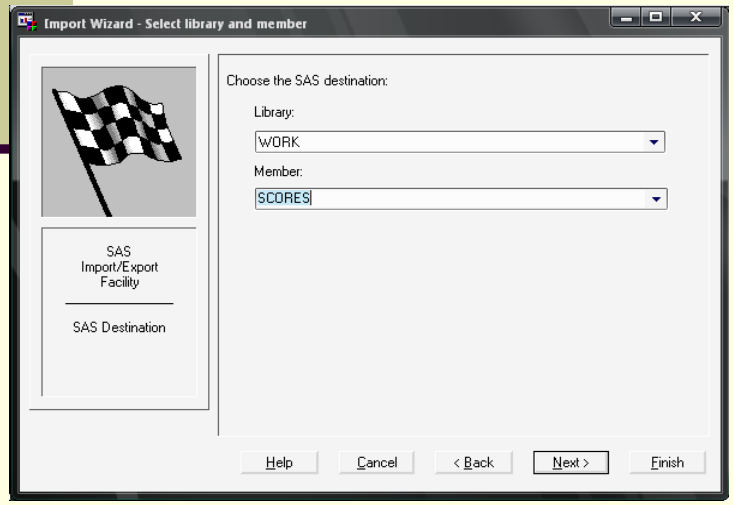
Importing MS Excel data 2-1

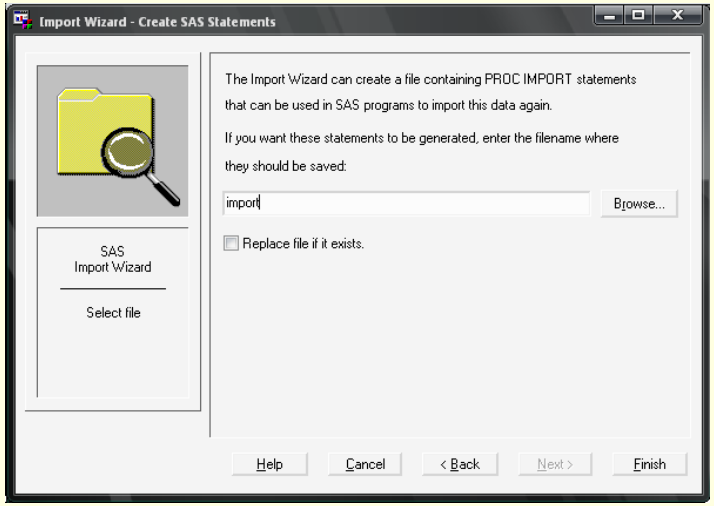
- Click file under the main tab and open “Import Data”



Importing MS Excel data 2-2







Data Transformation

- How to transform data in SAS

```
data trans;
set scores;
* 'Set' command allows reusing created SAS data;
lnheight=log(height);
logheight=log10(height);
index=height/weight;
run;
```

Data Transformation (Cont'd)

VIEWTABLE: Work.Trans

	height	weight	gender	name	score	lnheight	logheight	index
1	5.4	125	2	JAUNITA	65	1.686	0.732	0.043
2	5.3	122	2	SALLY	77	1.668	0.724	0.043
3	5.6	.	2	DONNA	43	1.723	0.748	.
4	5.7	145	2	SABRINA	36	1.74	0.756	0.039
5	5.7	156	1	JOHN	84	1.74	0.756	0.037
6	6	170	1	MARK	56	1.792	0.778	0.035
7	6.4	200	1	ERIC	34	1.856	0.806	0.032
8	5.9	165	1	BRUCE	72	1.775	0.771	0.036

- **Note**
  - **LOG(*x*) : the natural logarithm of *x***
  - **LOG10(*x*) : the log base ten of *x***
  - **LOG2(*x*) : the log base two of *x***

Arithmetic and Comparison Operators

Arithmetic Operators		
Symbol	**Definition**	**Example**
**	exponentiation	a**3
*	multiplication	2*y
/	division	var/5
+	addition	num+3
-	subtraction	sale-discount

Comparison Operators		
Symbol	**Definition**	**Example**
=	equal to	a = 3
^= or NE	not equal to	a ne 3
¬= or NE	not equal to	
~= or NE	not equal to	
> or GT	greater than	num > 5
< or LT	less than	num < 8
>= or GE	greater than or equal to	sales >= 300
<= or LE	less than or equal to	sales <= 100

Data Modification: If / then Statements

- How to delete certain observations from data
  - Example: The following command deletes observations having weight more than 160

```
data modify;
set trans; *'Set' command allows reusing created SAS data;
if weight > 160 then delete;
run;
```

- Open the created data file 'modify' in the 'Work' folder of your library and compare that from the data file 'trans.'

You can see that observations for ‘Mark,’ ‘Eric’, and ‘Bruce,’ have been deleted in ‘modify.’

VIEWTABLE: Work.Trans

	height	weight	gender	name	score	lnheight	logheight	index
1	5.4	125	2	JAUNITA	65	1.686	0.732	0.043
2	5.3	122	2	SALLY	77	1.668	0.724	0.043
3	5.6	.	2	DONNA	43	1.723	0.748	.
4	5.7	145	2	SABRINA	36	1.74	0.756	0.039
5	5.7	156	1	JOHN	84	1.74	0.756	0.037
6	6	170	1	MARK	56	1.792	0.778	0.035
7	6.4	200	1	ERIC	34	1.856	0.806	0.032
8	5.9	165	1	BRUCE	72	1.775	0.771	0.036

VIEWTABLE: Work.Modify

	height	weight	gender	name	score	lnheight	logheight	index
1	5.4	125	2	JAUNITA	65	1.686	0.732	0.043
2	5.3	122	2	SALLY	77	1.668	0.724	0.043
3	5.6	.	2	DONNA	43	1.723	0.748	.
4	5.7	145	2	SABRINA	36	1.74	0.756	0.039
5	5.7	156	1	JOHN	84	1.74	0.756	0.037

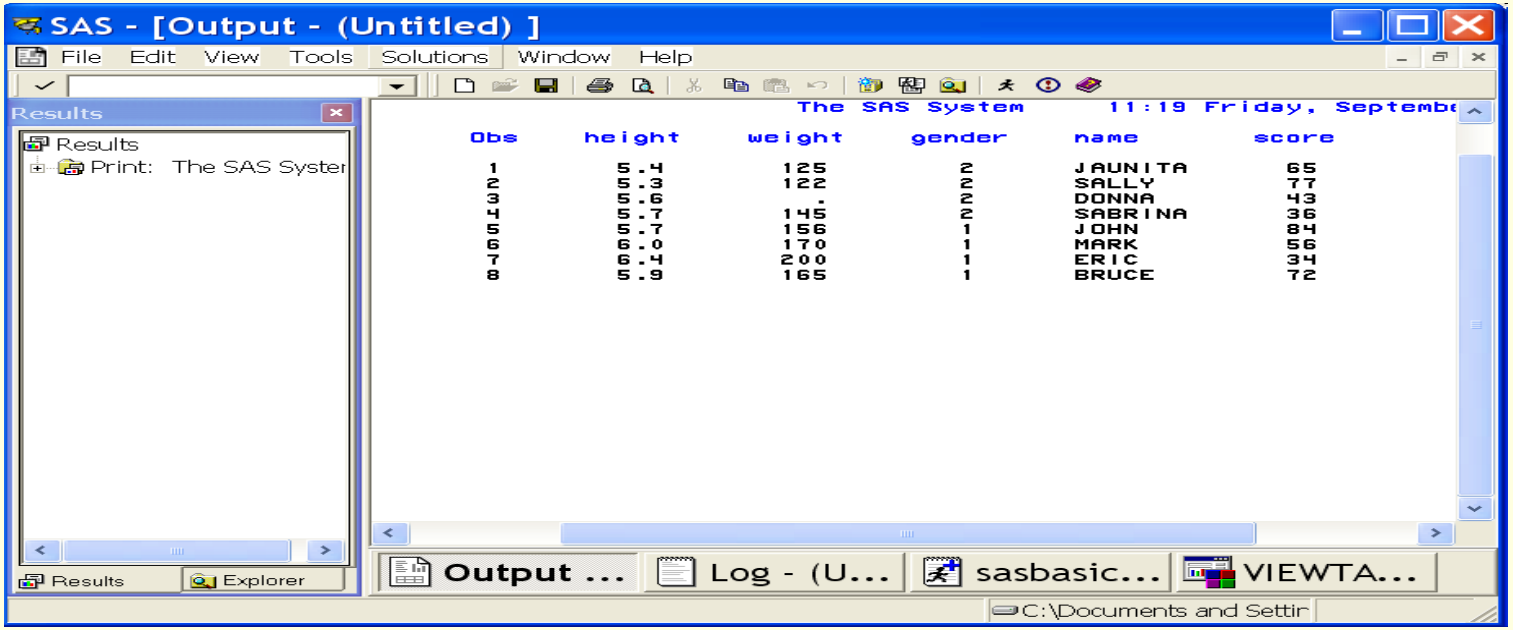
Section 3

Basic commands in SAS:
PROC step

Proc Steps: proc print

- Use: to see the SAS data file in the output window

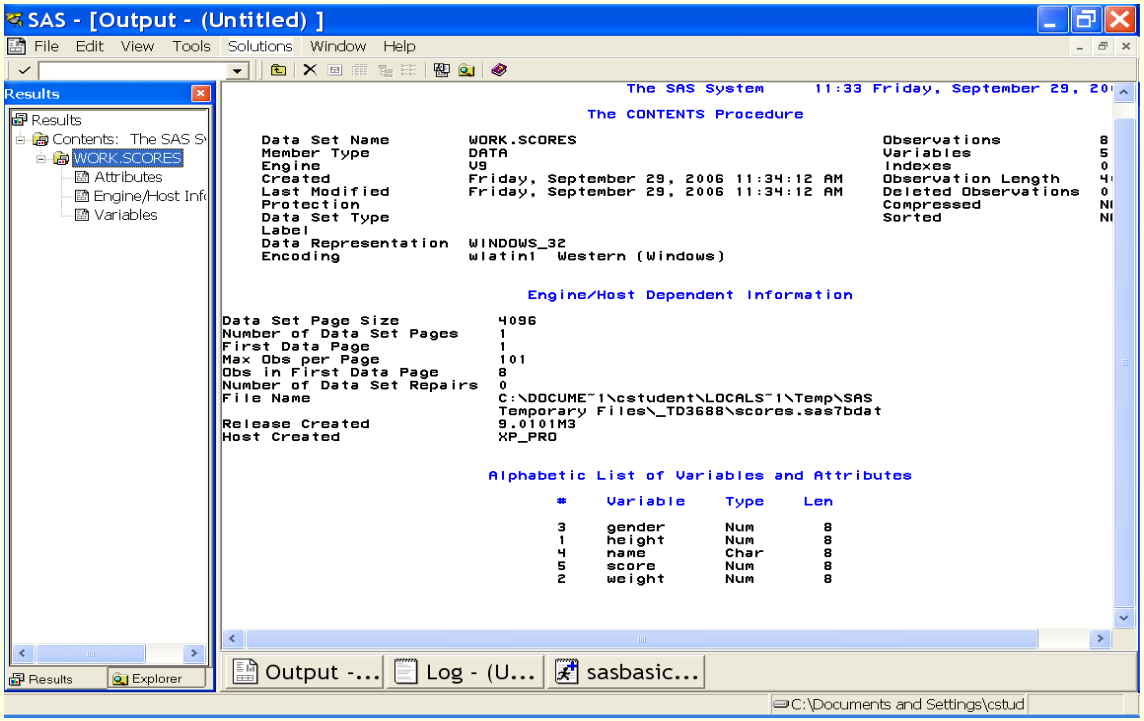
```
proc print data=scores;
run;
```



Proc Steps: proc contents

- Use: to see the contents of SAS data file

```
proc contents data=scores;
run;
```



Proc Steps: proc sort

- Use: to sort SAS data file

proc sort data=scores out=name
by name; *Sorts the data by name
in alphabetical orders
run;

proc sort data=scores out=height
by height; *Sorts the data by height
in ascending orders;
Run;

proc sort data=scores out=height2
by **descending** height;
*Sorts the data by height
in descending orders;
run;

VIEWTABLE: Work.Name

	height	weight	gender	name	score
1	5.9	165	1	BRUCE	72
2	5.6	.	2	DONNA	43
3	6.4	200	1	ERIC	34
4	5.4	125	2	JAUNITA	65
5	5.7	156	1	JOHN	84
6	6	170	1	MARK	56
7	5.7	145	2	SABRINA	36
8	5.3	122	2	SALLY	77

VIEWTABLE: Work.Height

	height	weight	gender	name	score
1	5.3	122	2	SALLY	77
2	5.4	125	2	JAUNITA	65
3	5.6	.	2	DONNA	43
4	5.7	145	2	SABRINA	36
5	5.7	156	1	JOHN	84
6	5.9	165	1	BRUCE	72
7	6	170	1	MARK	56
8	6.4	200	1	ERIC	34

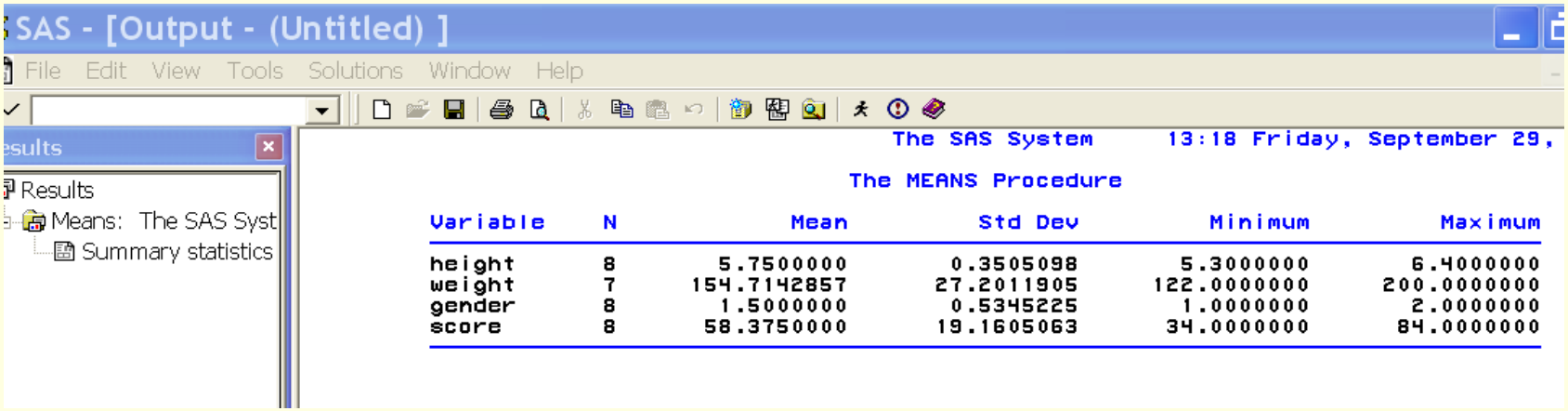
VIEWTABLE: Work.H...

	height	weight	gender	name	score
1	6.4	200	1	ERIC	34
2	6	170	1	MARK	56
3	5.9	165	1	BRUCE	72
4	5.7	145	2	SABRINA	36
5	5.7	156	1	JOHN	84
6	5.6	.	2	DONNA	43
7	5.4	125	2	JAUNITA	65
8	5.3	122	2	SALLY	77

Proc Steps: proc means

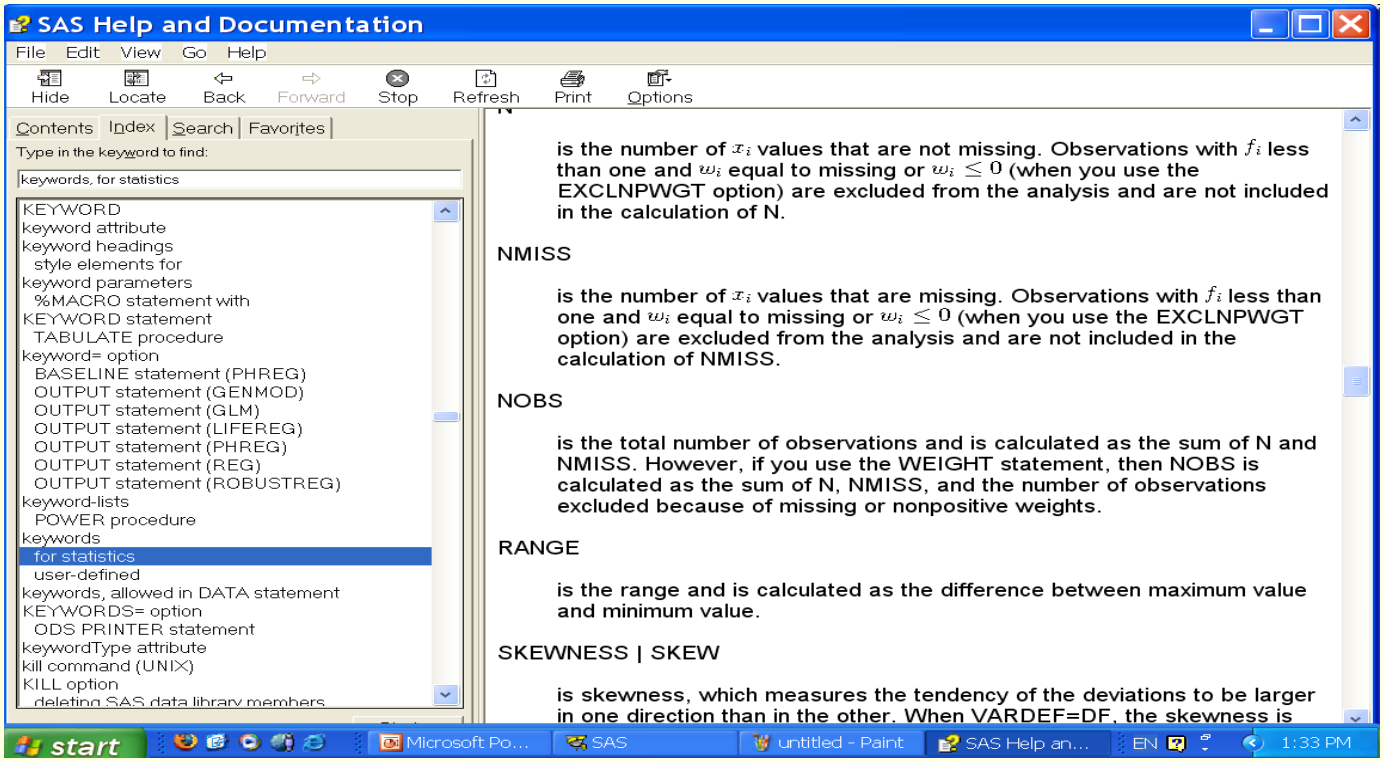
- Use: to see basic simple statistics of data

```
proc means data=scores;
run;
   *This provides the number of obvs, mean,
   std, min, and max of all numeric variables;
```



Proc means: How to see other simple statistics

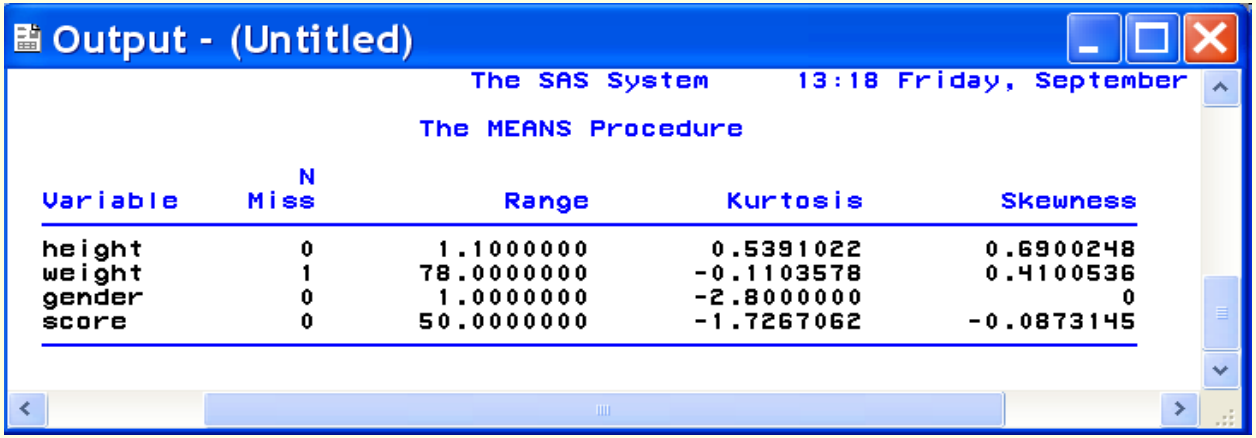
- To find out the commands for other simple statistics click the help icon and then click index. Then type in 'keywords' in the search box and enter. Finally, click 'for statistics';



Proc Steps: How to see other simple statistics (Cont'd)

- Insert the commands for the simple statistics you want to calculate with SAS before the command 'data="file name" ' :

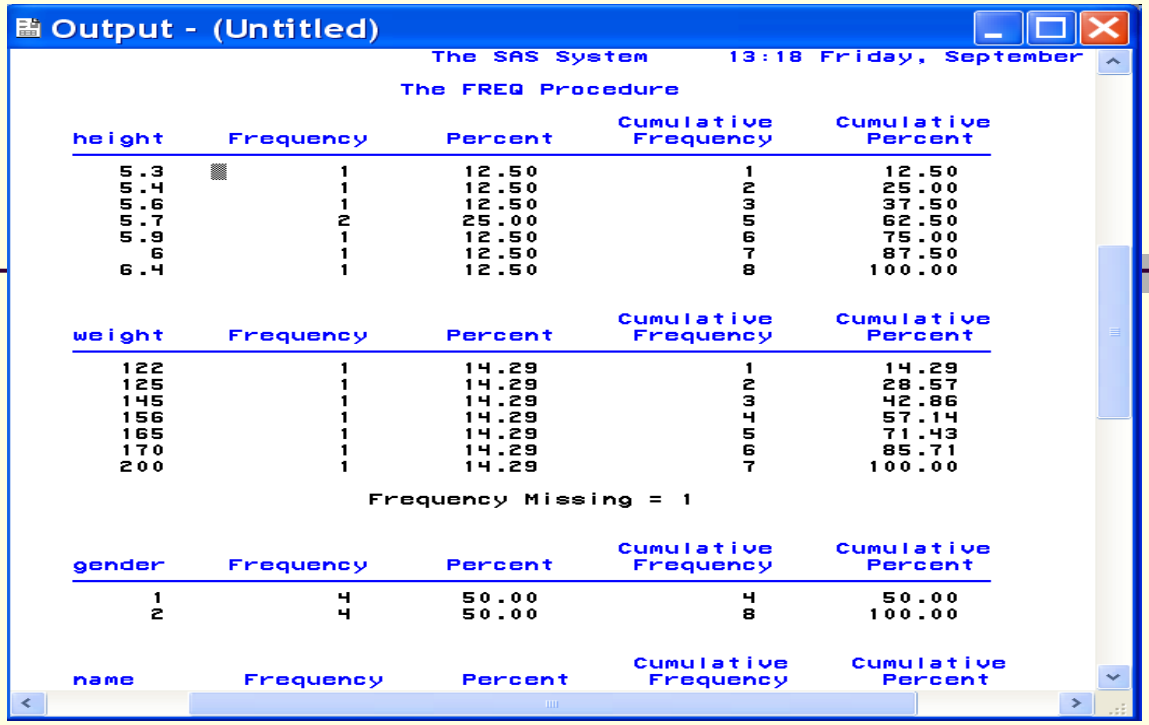
```
proc means nmiss range kurt skew data=scores;
 run;
```

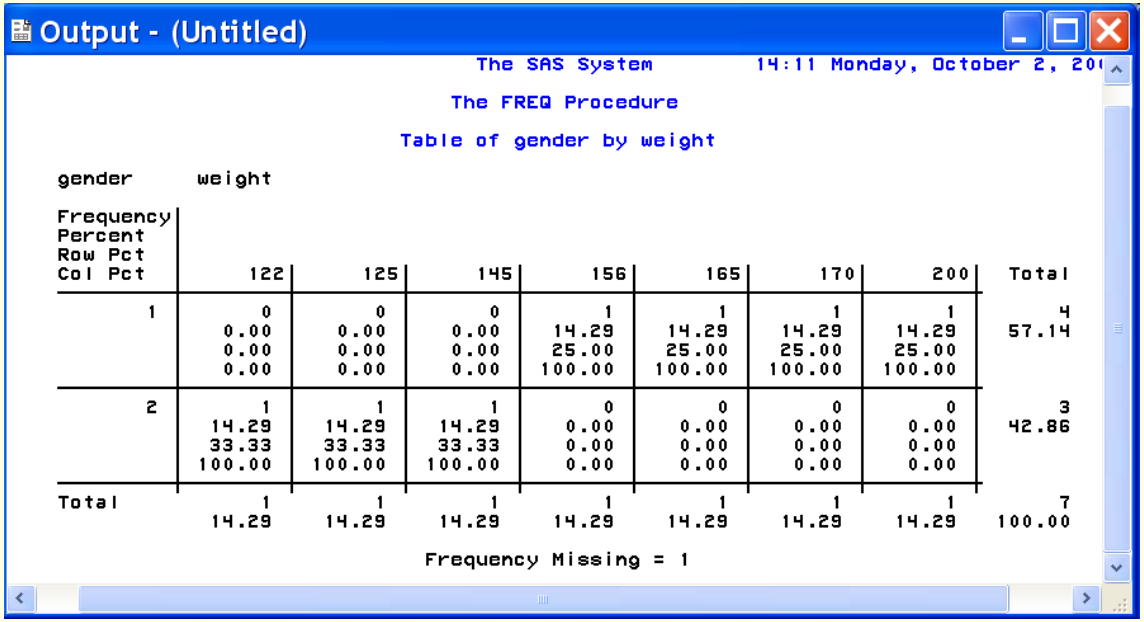


Proc Steps: proc freq

- proc freq
  - Use: to analyze frequency of the variables and to create frequency tables for variables

```
proc freq data=scores;
run; *shows one-way frequencies;

proc freq data=scores;
tables gender*weight;
run; *creates cross-tabulation table;
```


Output - (Untitled)
The SAS System 13:18 Friday, September
The FREQ Procedure
height Frequency Percent Cumulative Frequency Cumulative Percent
5.3 1 12.50 1 12.50
5.4 1 12.50 2 25.00
5.6 1 12.50 3 37.50
5.7 2 25.00 5 62.50
5.9 1 12.50 6 75.00
6 1 12.50 7 87.50
6.4 1 12.50 8 100.00
weight Frequency Percent Cumulative Frequency Cumulative Percent
122 1 14.29 1 14.29
125 1 14.29 2 28.57
145 1 14.29 3 42.86
156 1 14.29 4 57.14
165 1 14.29 5 71.43
170 1 14.29 6 85.71
200 1 14.29 7 100.00
Frequency Missing = 1
gender Frequency Percent Cumulative Frequency Cumulative Percent
1 4 50.00 4 50.00
2 4 50.00 8 100.00
name Frequency Percent Cumulative Frequency Cumulative Percent

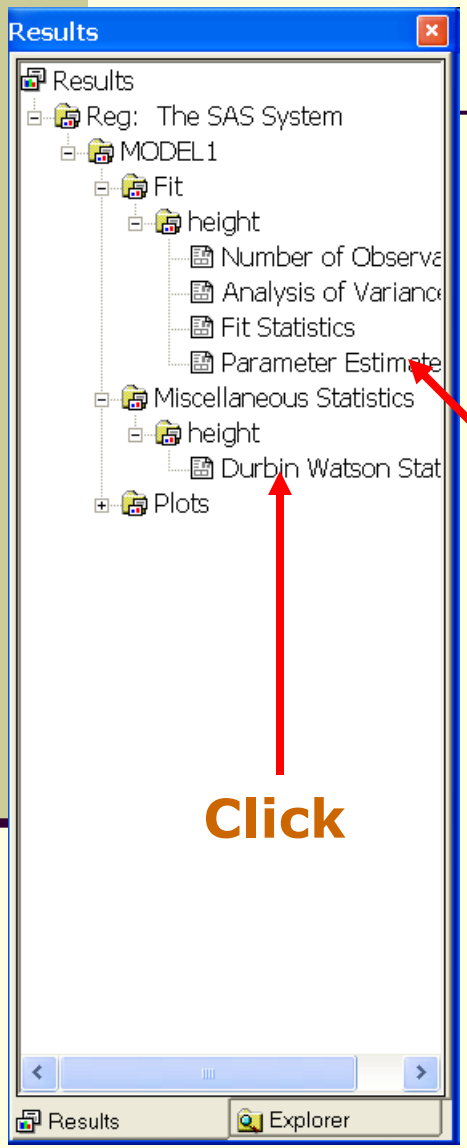

Output - (Untitled)
The SAS System 14:11 Monday, October 2, 20
The FREQ Procedure
Table of gender by weight
gender weight
Frequency
Percent
Row Pct
Col Pct 122 125 145 156 165 170 200 Total
1 0 0 0 1 1 1 1 4
0.00 0.00 0.00 14.29 14.29 14.29 14.29 57.14
0.00 0.00 0.00 25.00 25.00 25.00 25.00
0.00 0.00 0.00 100.00 100.00 100.00 100.00
2 1 1 1 0 0 0 0 3
14.29 14.29 14.29 0.00 0.00 0.00 0.00 42.86
33.33 33.33 33.33 0.00 0.00 0.00 0.00
100.00 100.00 100.00 0.00 0.00 0.00 0.00
Total 1 1 1 1 1 1 1 7
14.29 14.29 14.29 14.29 14.29 14.29 14.29 100.00
Frequency Missing = 1

Proc Steps: proc reg

- proc reg
  - One of a general-purpose procedures for regression analysis in SAS

```
proc reg data=scores;
model height=weight      / dw alpha=0.01 clb    ;
plot height*weight  / cframe=ligr conf pred ;
run;
```

$$\text{height} = \alpha + \beta\text{weight} + \varepsilon$$



Click

Click

Output - (Untitled)

Parameter Estimates

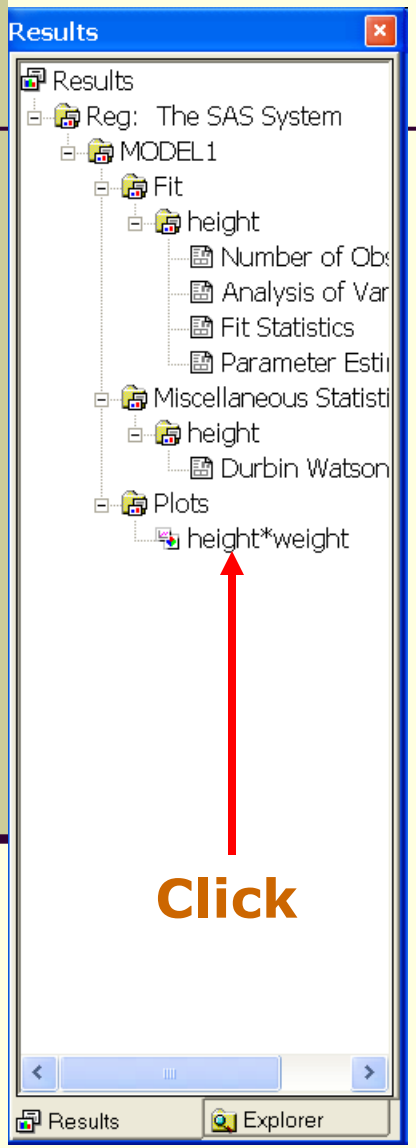
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|	99% Confidence Limits	
Intercept	1	3.66848	0.12483	29.39	<.0001	3.16516	4.17180
weight	1	0.01359	0.00079634	17.07	<.0001	0.01038	0.01680

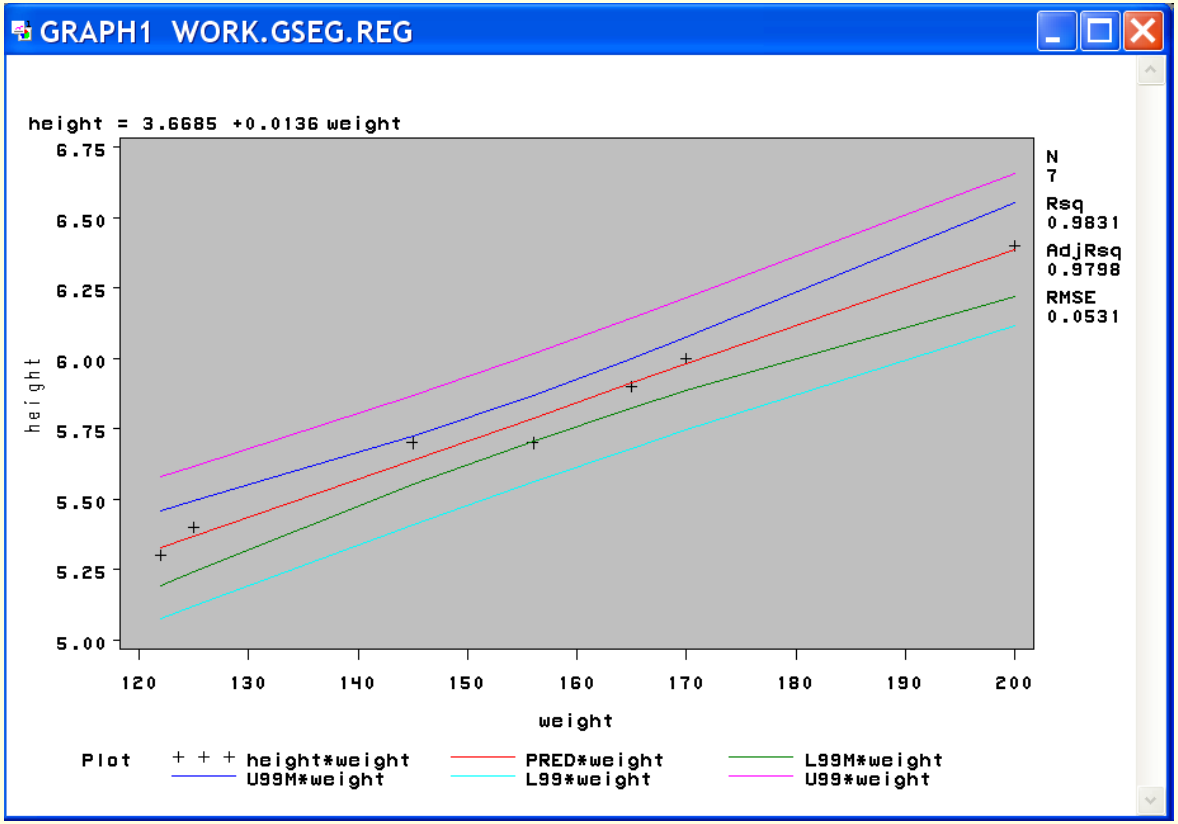
Output - (Untitled)

The SAS System 13:18 Frida

The REG Procedure
Model: MODEL1
Dependent Variable: height

Durbin-Watson D	3.281
Number of Observations	7
1st Order Autocorrelation	-0.682

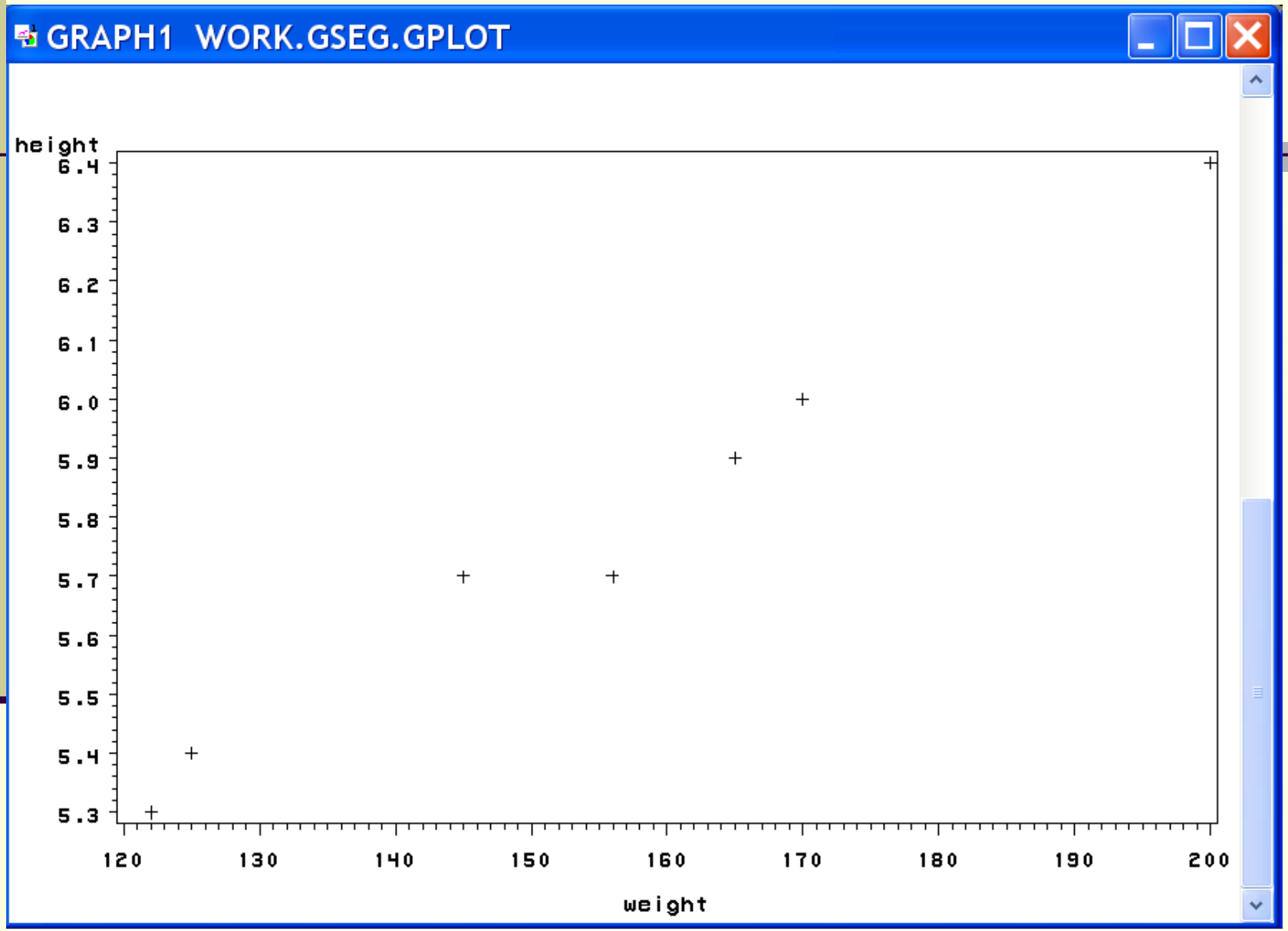

Results
Results
Reg: The SAS System
MODEL1
Fit
height
Number of Obs
Analysis of Var
Fit Statistics
Parameter Esti
Miscellaneous Statisti
height
Durbin Watson
Plots
height*weight
Click
Results
Explorer


GRAPH1 WORK.GSEG.REG
height = 3.6685 +0.0136 weight
6.75
6.50
6.25
6.00
5.75
5.50
5.25
5.00
height
120
130
140
150
160
170
180
190
200
weight
N
7
Rsq
0.9831
AdjRsq
0.9798
RMSE
0.0531
Plot
height*weight
U99M*weight
PRED*weight
L99*weight
L99M*weight
U99*weight

Proc Steps: proc gplot

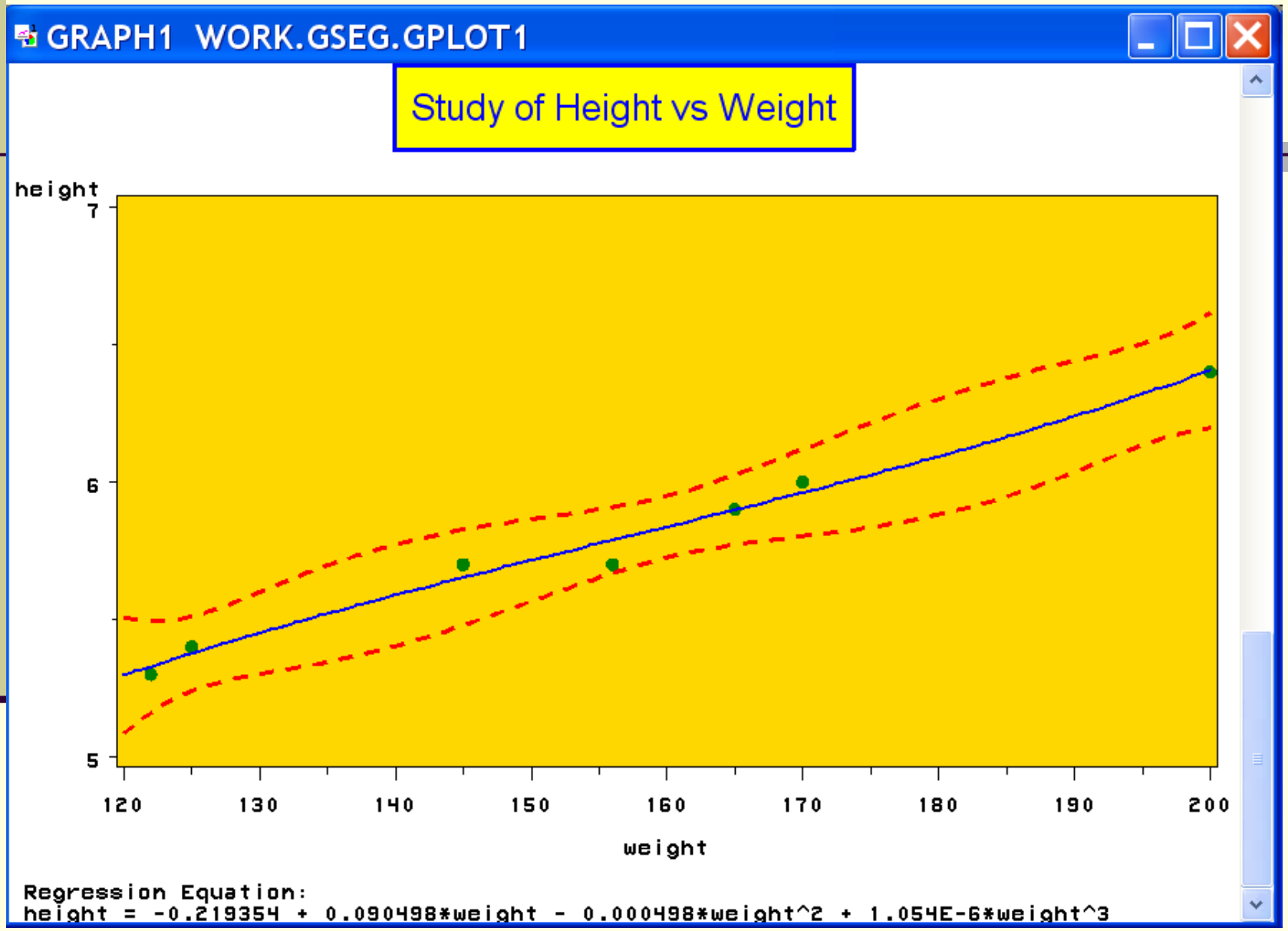
- proc gplot
  - Use: to plot the values of two or more variables on a set of coordinate axes

```
proc gplot data=scores;
 plot height*weight;
*height=vertical axis,
 weight=horizontal axis;
 run;
```


GRAPH1 WORK.GSEG.GPLOT
height
6.4
6.3
6.2
6.1
6.0
5.9
5.8
5.7
5.6
5.5
5.4
5.3
120
130
140
150
160
170
180
190
200
weight

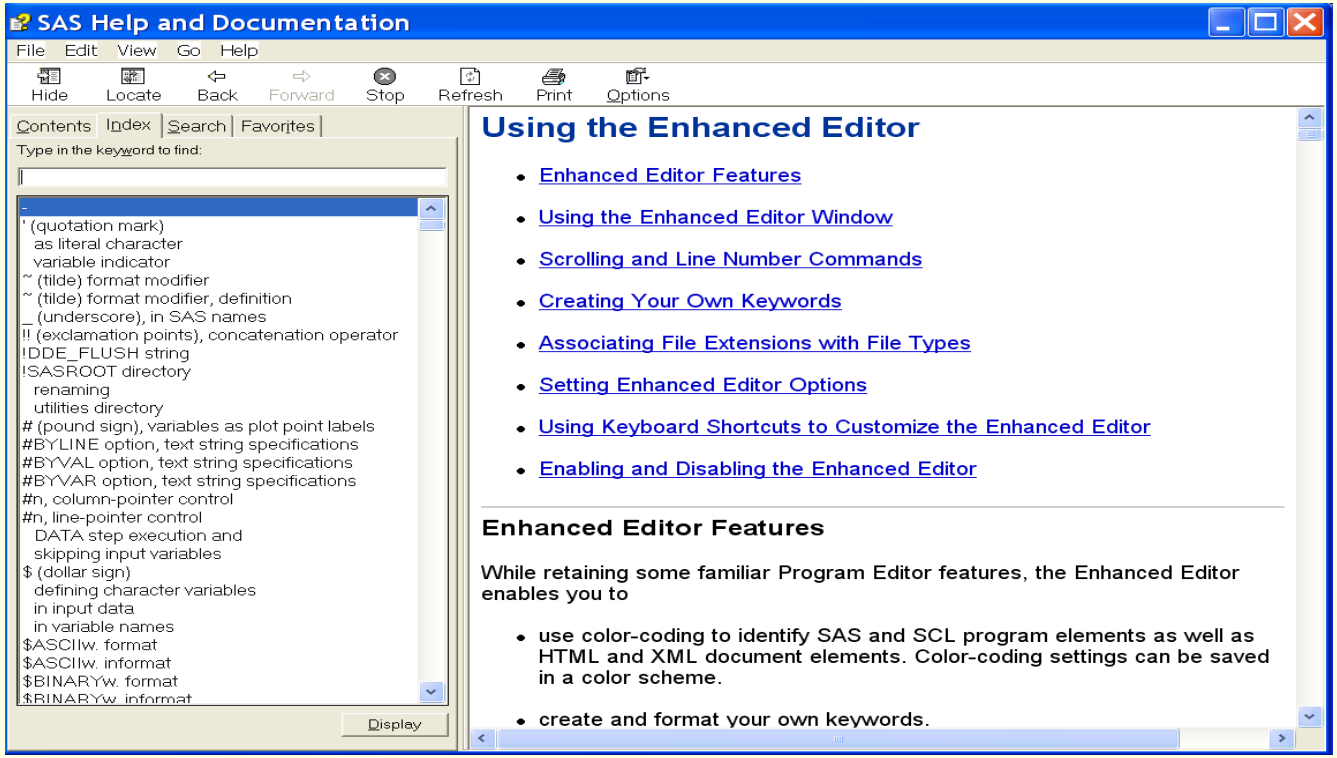
Using advanced options in SAS

```
proc gplot data=scores;
plot height*weight
/skipmiss haxis=120 to 200 by 10 hminor=1
vaxis=5.0 to 7.0 by 1.0 vminor=1
Regeqn cframe=gold; *Options for the plot statement;

title font=arial c=blue box=3 bcolor=yellow
'Study of Height vs Weight'; *Putting a title
for your graph;

symbol i=rcclm95 value=dot height=1
cv=green ci=blue co=red width=2;
*Setting the colors and size for the plot symbol
and lines. i= can be also expressed as interpol=;
run;
```


GRAPH1 WORK.GSEG.GPLOT1
Study of Height vs Weight
height
7
6
5
120
130
140
150
160
170
180
190
200
weight
Regression Equation:
height = -0.219354 + 0.090498*weight - 0.000498*weight^2 + 1.054E-6*weight^3

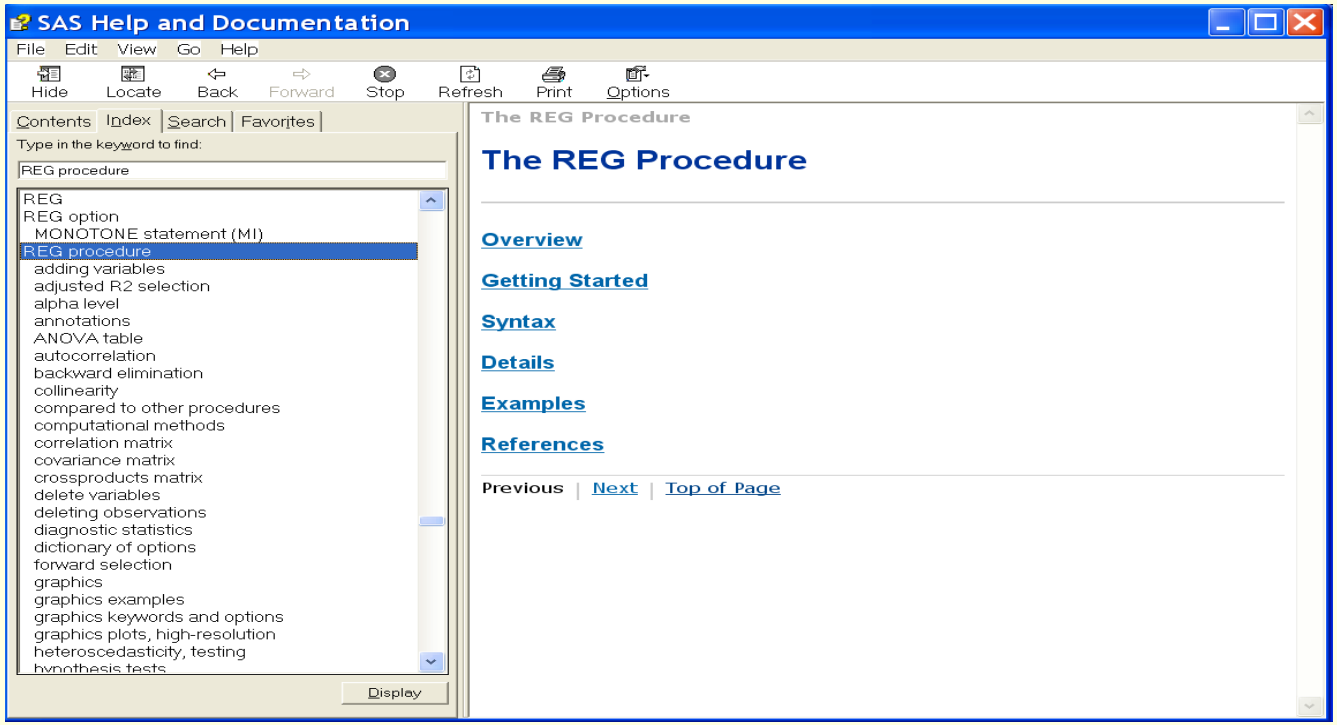
Useful supports

- In the tool bar click the help menu or the help icon



Useful supports: using the Help in SAS

- Example: click index and type ‘reg.’ Then double click ‘REG procedure’



Useful supports: other useful sites

- Online SAS manuals

  http://www.uri.edu/sasdoc

  This will automatically link you to http://support.sas.com/documentation/onlinedoc/sas9doc.html

- Statbookstore: useful site for finding program examples

  http://www.geocities.com/statbookstore/

Exercise

- Import the following data and use the libname statement to save the data to your ‘c:/’ drive of the computer.
- Use SAS to determine the mean and variance of ‘height’ and ‘score’ of the data.
- Determine the intercept (b1) and the coefficient (b2) of the model,

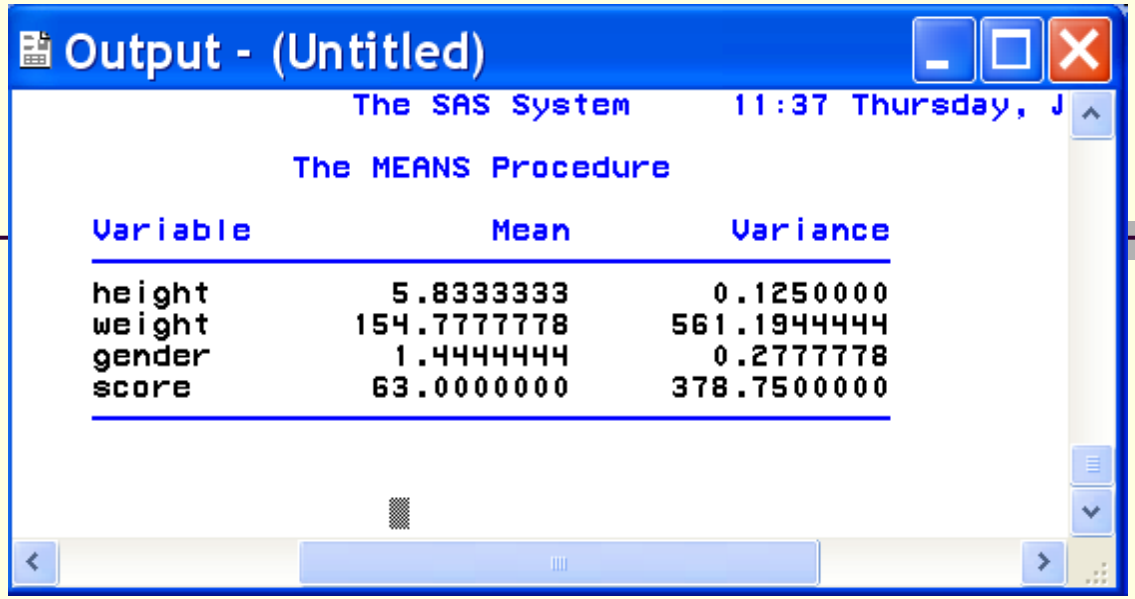
  height = b1 + b2 * weight + e

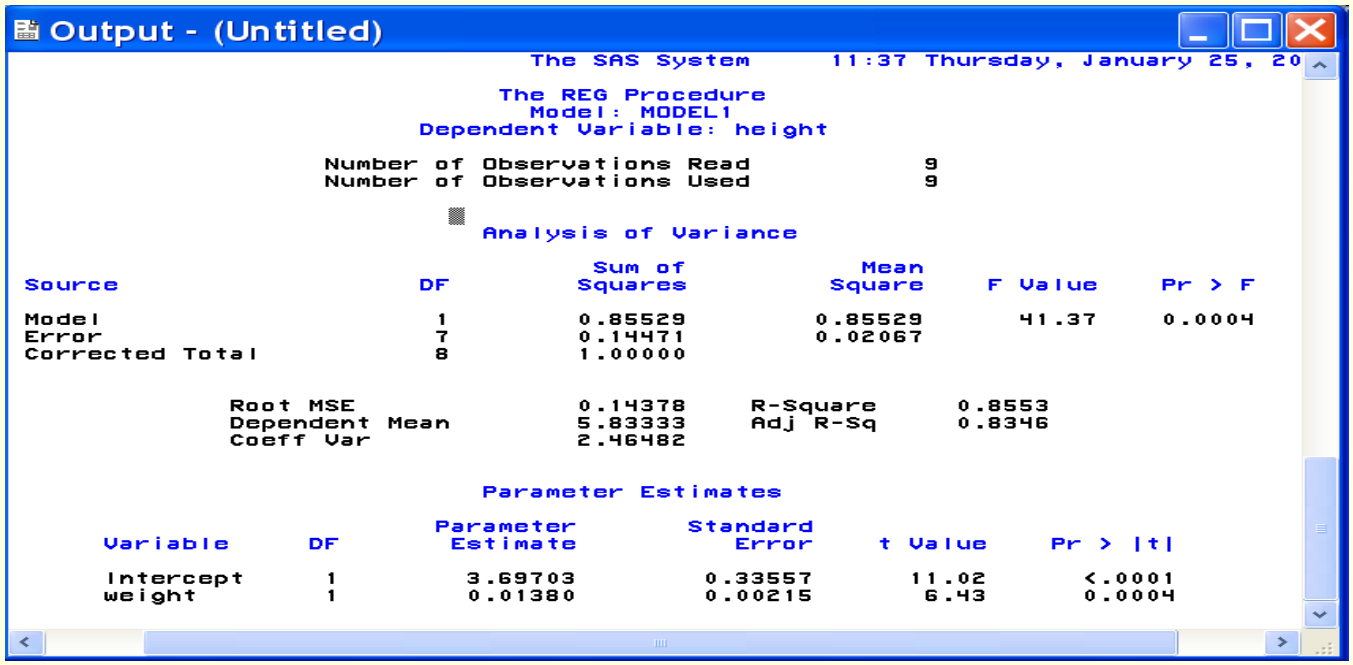
  using the data.

height	weight	gender	name	score
5.4	125	2	JAUNITA	65
5.3	122	2	SALLY	77
5.7	145	2	SABRINA	36
5.9	150	2	KATE	55
5.7	156	1	JOHN	84
6	170	1	MARK	56
6.4	200	1	ERIC	34
5.9	165	1	BRUCE	72
6.2	160	1	TOM	88

Solution

```
libname  test 'c:\';
data test.scores3;
input height weight gender name $ score;
cards;
5.4 125 2   JAUNITA 65
5.3 122 2   SALLY   77
5.7 145 2   SABRINA 36
5.9 150 2   KATE    55
5.7 156 1   JOHN    84
6   170 1   MARK    56
6.4 200 1   ERIC    34
5.9 165 1   BRUCE   72
6.2 160 1   TOM     88
;
run;
proc means mean var data=test.scores3;
run;

proc reg data=test.scores3;
model height = weight;
run;
```


Output - (Untitled)
The SAS System 11:37 Thursday, J
The MEANS Procedure
Variable Mean Variance
height 5.8333333 0.1250000
weight 154.7777778 561.1944444
gender 1.4444444 0.2777778
score 63.0000000 378.7500000


Output - (Untitled)
The SAS System 11:37 Thursday, January 25, 20
The REG Procedure
Model: MODEL1
Dependent Variable: height
Number of Observations Read 9
Number of Observations Used 9
Analysis of Variance
Source DF Sum of Squares Mean Square F Value Pr > F
Model 1 0.85529 0.85529 41.37 0.0004
Error 7 0.14471 0.02067
Corrected Total 8 1.00000
Root MSE 0.14378 R-Square 0.8553
Dependent Mean 5.83333 Adj R-Sq 0.8346
Coeff Var 2.46482
Parameter Estimates
Variable DF Parameter Estimate Standard Error t Value Pr > |t|
Intercept 1 3.69703 0.33557 11.02 <.0001
weight 1 0.01380 0.00215 6.43 0.0004

For further Questions:
kentaka@mail.uri.edu