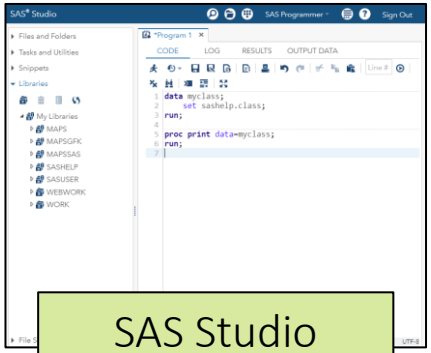
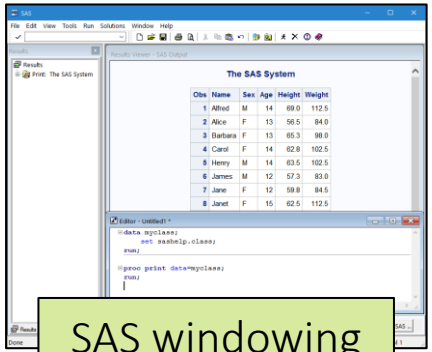
SAS Lesson 01

SAS Programming Interfaces



SAS Studio

SAS Enterprise Guide



SAS windowing environment

All these interfaces have the basic tools that you need for programming.

SAS Programming Interfaces

write and submit code

```
data myclass;
    set sashelp.class;
run;

proc print data=myclass;
run;
```

Editor

view messages from SAS

```
1   data myclass;
2      set sashelp.class;
3   run;

NOTE: There were 19 observe
the data set SASHELP.CLASS.
NOTE: The data set WORK.MYCLASS has 19
observations and 5 variables.
NOTE: DATA statement used:
      real time          0.01 seconds
      cpu time           0.00 seconds

4
5   proc print data=myclass;
NOTE: Writing HTML Body file: sashtml.htm
6   run;

NOTE: There were 19 observations read
from the data set WORK.MYCLASS.
```

Log

view results

Name	Sex	Age	Height	We
Alfred	M	14	69.0	112
Alice	F	13	56.5	84.0
Barbara	F	13	65.3	98.0
Carol	F	14	62.8	102.5
Henry	M	14	63.5	102.5
James	M	12	57.3	83.0
Jan				

	Name	Sex	Age	Height	Weight
1	Alfred	M	14	69	112.5
2	Alice	F	13	56.5	84
3	Barbara	F	13	65.3	98
4	Carol	F	14	62.8	102.5
5	Henry	M	14	63.5	102.5
6	James	M	12	57.3	83
7	Jane	F	12	59.8	84.5
8	Janet	F	15	62.5	112.5

Results and Output Data

PC SAS and SAS on Demand

This demonstration provides an overview of the two programming interfaces that will be used in STAT604.

Setting up the Course Files

- Only required for full (PC) SAS installation
- Download zip file:
http://support.sas.com/content/dam/SAS/support/en/books/data/base-guide-practice-data.zip
- Unzip file in accessible location
- PC SAS
  - C:\Users\myname\Documents\STAT604Data\
- Browse to cert folder under the new unzipped folder
  - Use explorer for PC SAS
- Copy folder path

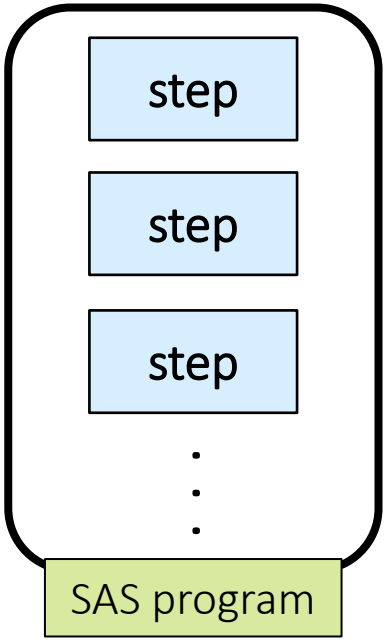
Setting up the Course Files

- Open the program ..cert\cr8data.sas
- Use paste to replace path on the %let path= value
- Run the program 

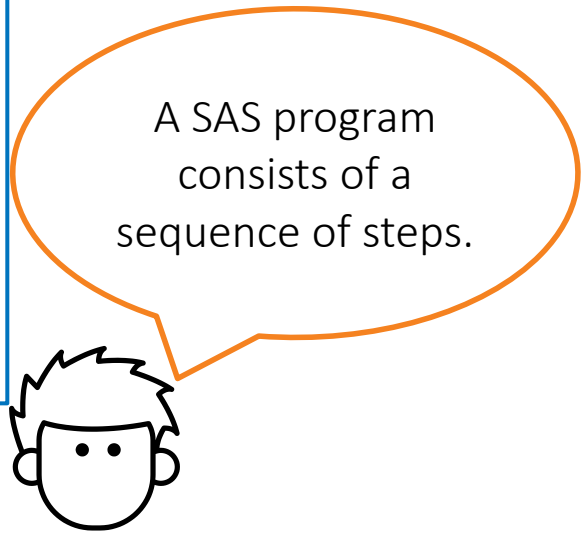
Essentials

Understanding SAS Syntax

SAS Program Structure

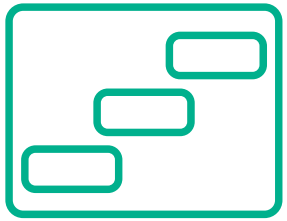



```
data myclass;
     set sashelp.class;
     heightcm=height*2.54;
run;

proc print data=myclass;
run;

proc means data=myclass;
     var age heightcm;
run;
```

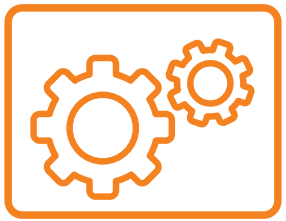


SAS Program Structure

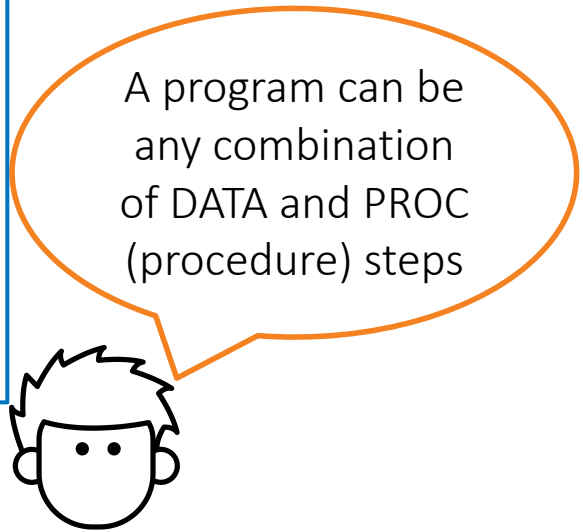
DATA step



PROC step

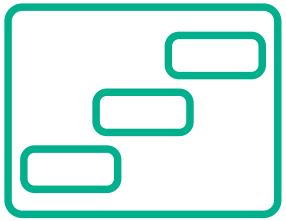


```
data myclass;
     set sashelp.class;
     heightcm=height*2.54;
run;

proc print data=myclass;
run;

proc means data=myclass;
     var age heightcm;
run;
```

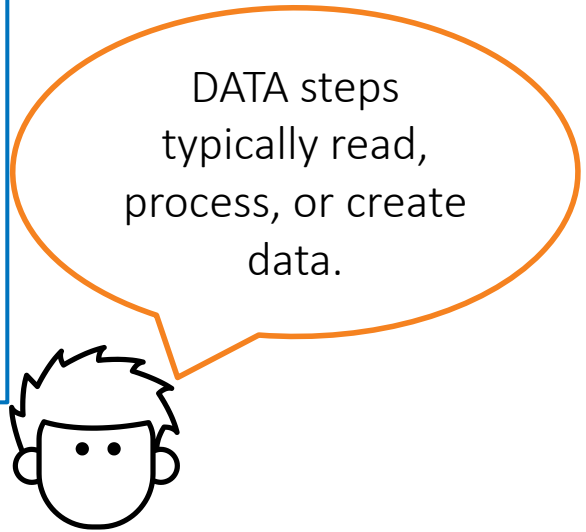


SAS Program Structure

DATA step

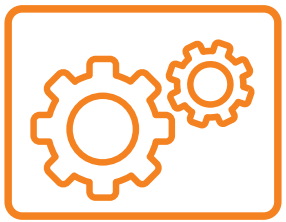


```
data myclass;
     set sashelp.class;
     heightcm=height*2.54;
run;

proc print data=myclass;
run;

proc means data=myclass;
     var age heightcm;
run;
```

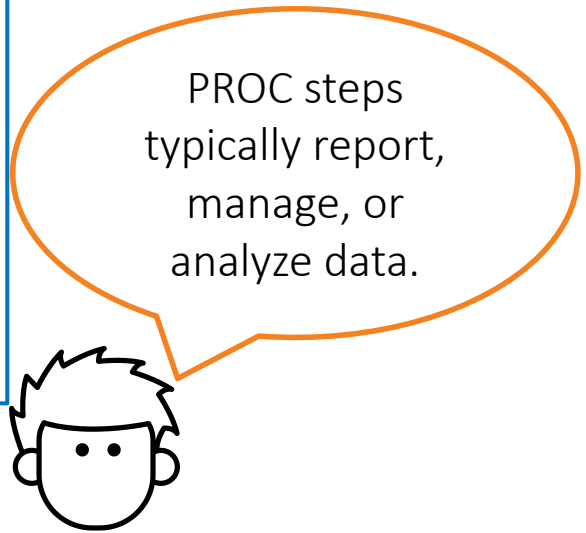


SAS Program Structure

PROC step



```
data myclass;
     set sashelp.class;
     heightcm=height*2.54;
run;

proc print data=myclass;
run;

proc means data=myclass;
     var age heightcm;
run;
```



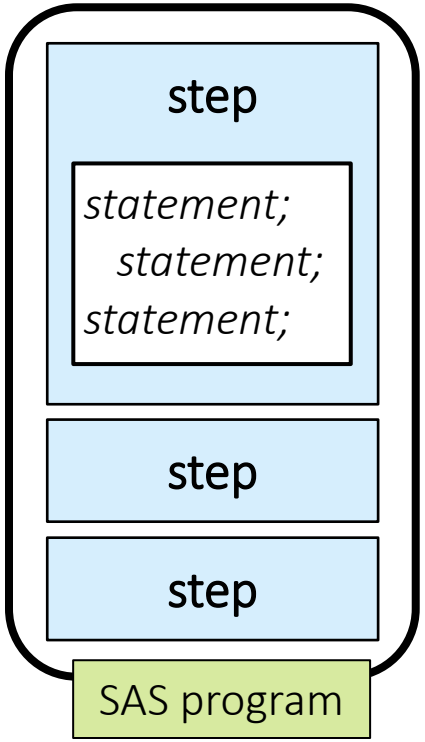
SAS Program Structure

Steps begin with either DATA or PROC.

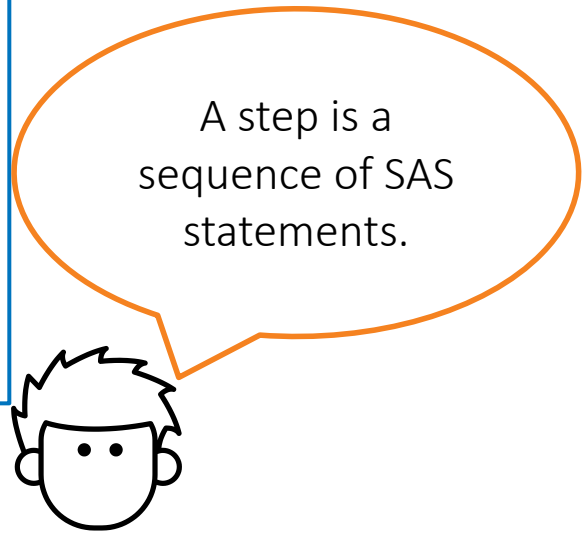
Steps end with RUN. Some PROCs end with QUIT.

```
data myclass;
     set sashelp.class;
     heightcm=height*2.54;
run;

proc print data=myclass;
run;

proc means data=myclass;
     var age heightcm;
run;
```

SAS Program Structure



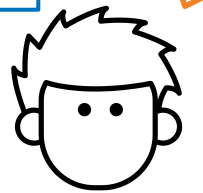

```
data myclass;
     set sashelp.class;
     heightcm=height*2.54;
run;

proc print data=myclass;
run;

proc means data=myclass;
     var age heightcm;
run;
```



SAS Statement Syntax

```
data myclass;
     set sashelp.class;
     heightcm=height*2.54;
run;

proc print data=myclass;
run;

proc means data=myclass;
     var age heightcm;
run;
```

Most statements begin with a keyword, and all statements end with a semicolon.



Global Statements

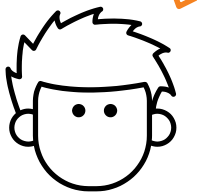
TITLE . . . ;

OPTIONS . . . ;

LIBNAME . . . ;

Run is not a global statement

Global statements are typically outside of steps and do not need a RUN statement.



Activity

```
data mycars;
    set sashelp.cars;
    AvgMPG=mean(mpg_city, mpg_highway);
run;

title "Cars with Average MPG Over 35";
proc print data=mycars;
    var make model type avgmpg;
    where AvgMPG > 35;
run;

title "Average MPG by Car Type";
proc means data=mycars mean min max maxdec=1;
    var avgmpg;
    class type;
run;

title;
```

Steps? 3

Statements in PROC PRINT? 4

Global Statements? 3

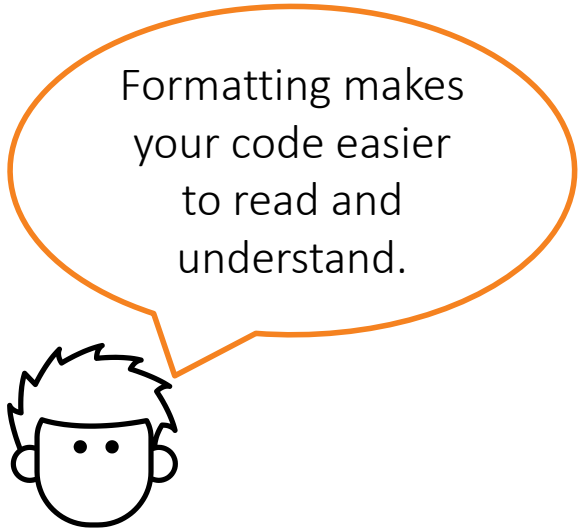
↳ title

SAS Program Syntax: Format

These are the same to SAS.

```
data myclass;set sashelp.class;run;
proc print data=myclass;run;
```

```
data myclass;
    set sashelp.class;
run;

proc print data=myclass;
run;
```



SAS Program Syntax: Case

```
data under13;
     set sashelp.class;
     where AGE<13;
     drop heIGht Weight;
run;
```

Unquoted values can be in any case.

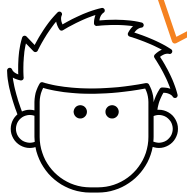
SAS Program Syntax: Comments

```
/* students under 13 yo */

data under13;
     set sashelp.class;
     where Age<13;
     *drop Height Weight;
run;
```

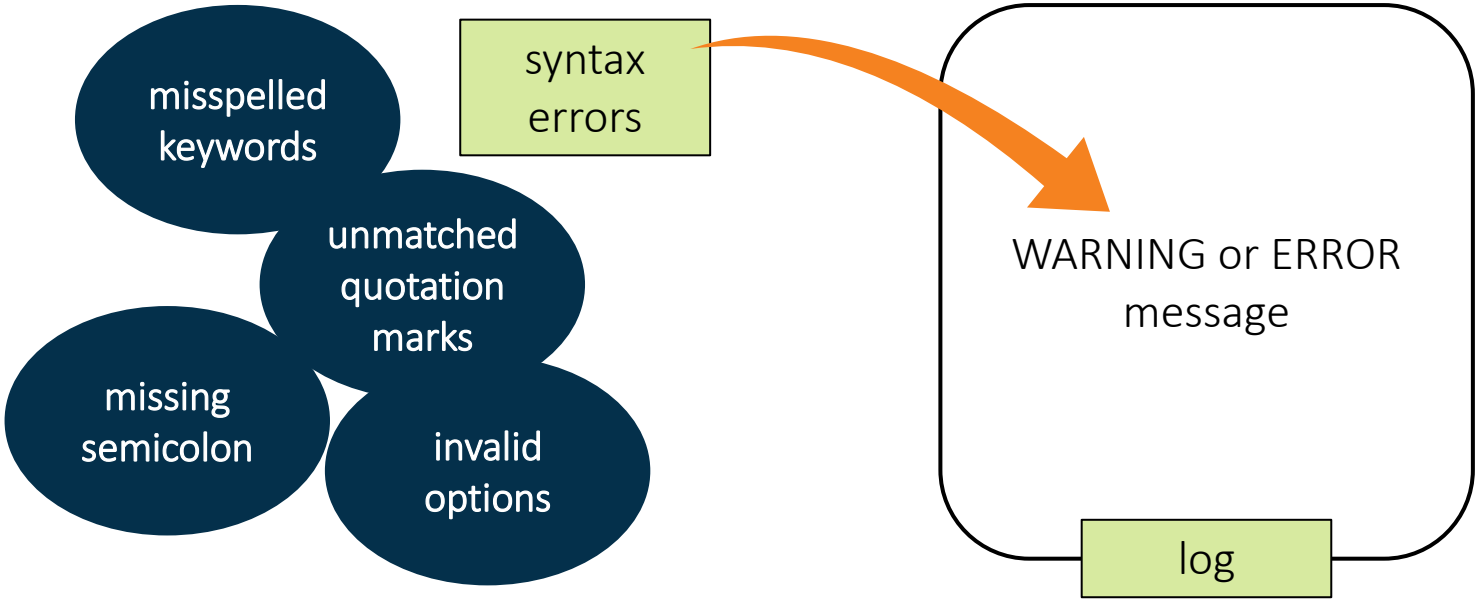
comments out everything between /* and */

comments out a single statement ending in a semicolon

Comments are ignored when a program executes.



Finding and Resolving Syntax Errors



Finding and Resolving Syntax Errors Using comments

This demonstration illustrates finding and resolving common syntax errors. It also demonstrates comment types.

\ehs\ehs02.sas

Referencing SAS Files

SAS Libraries Overview – Prep Guide Chapter 2

SAS Data Libraries

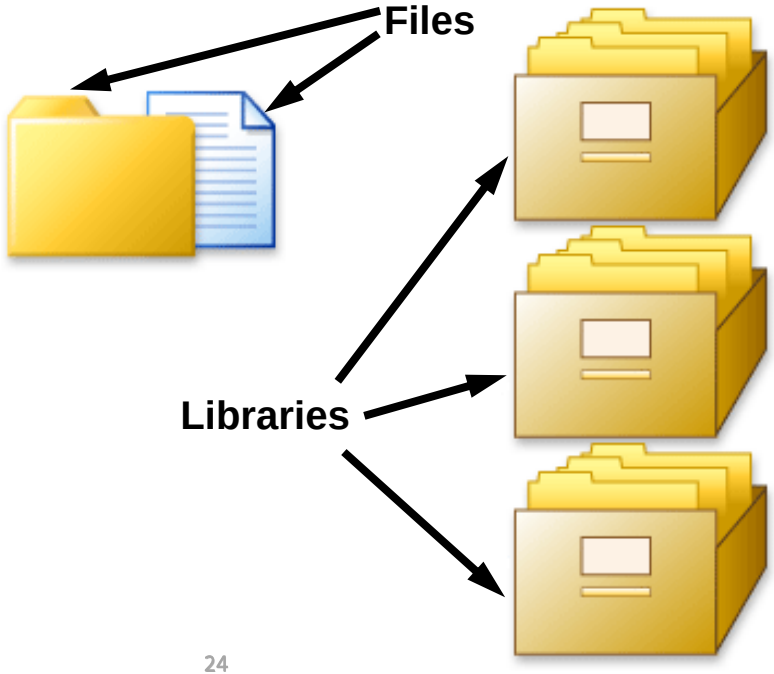
A *SAS data library* is a collection of SAS files or tables that are recognized as a unit by SAS. (Excludes raw data files.)

Directory-based System	A SAS data library is a directory.
Windows Example:	c:\users*userid*\cert
UNIX Example:	/users/*userid*/cert

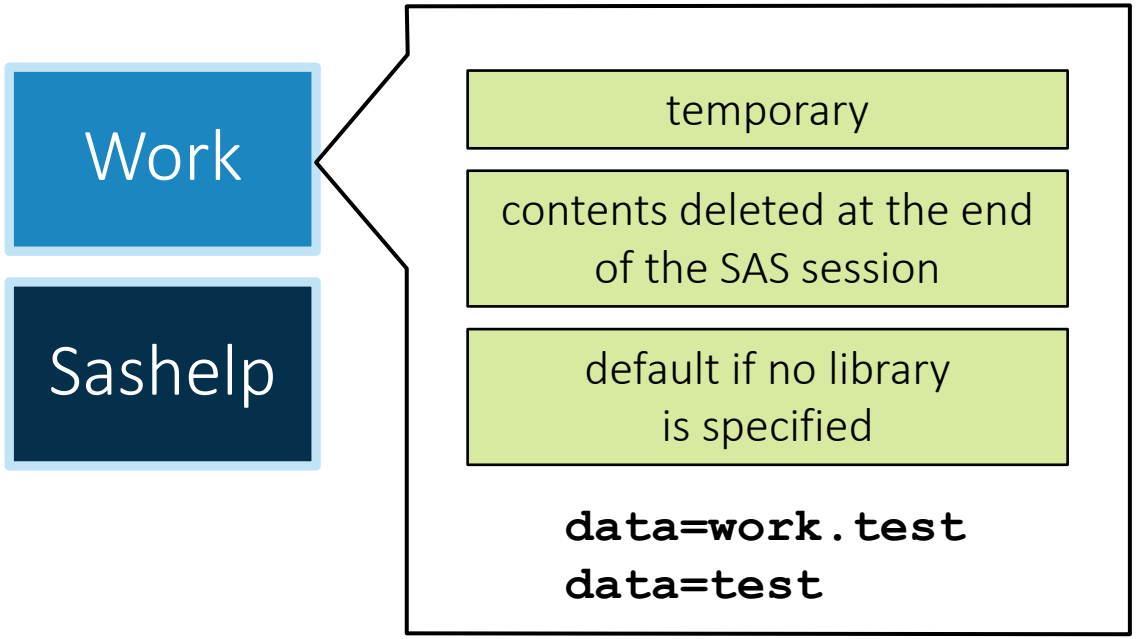
z/OS (OS/390)	A SAS data library is an operating system file.
z/OS (OS/390) Example:	*userid*.workshop.sasdata

SAS Data Libraries

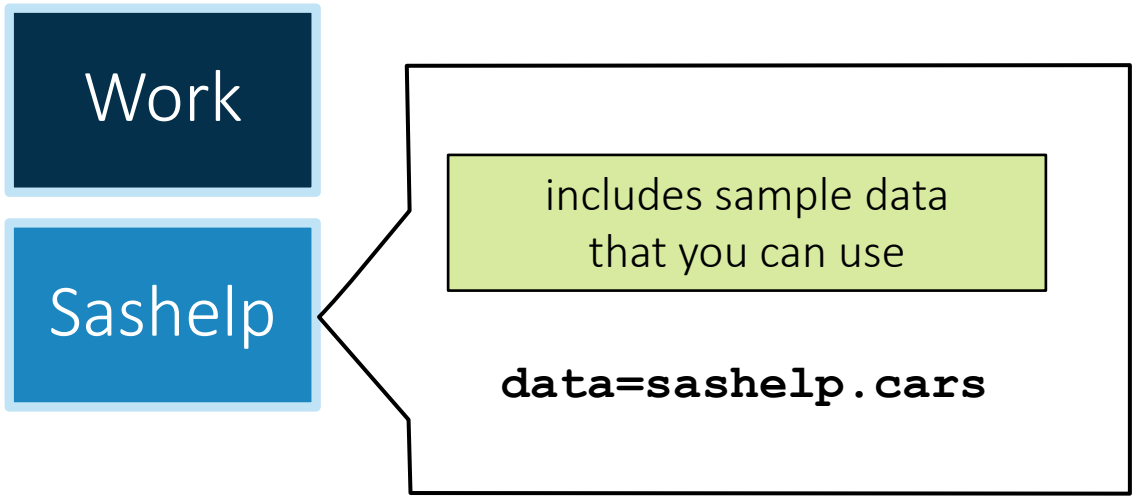
You can think of a SAS data library as a drawer in a filing cabinet and a SAS data set as one of the file folders in the drawer.



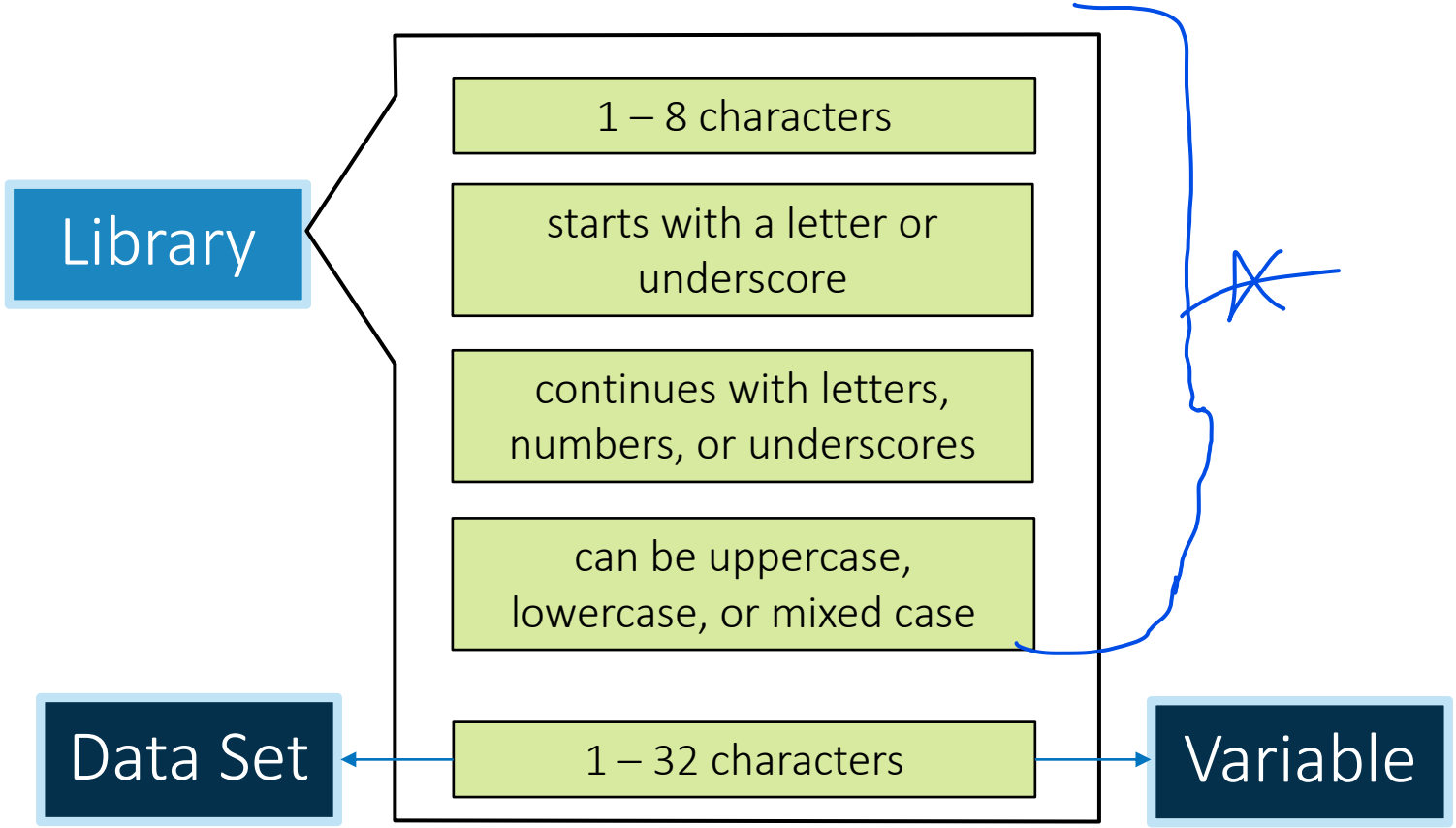
Automatic SAS Libraries



Automatic SAS Libraries



Default SAS Naming Rules



Changing the Default Behavior

```
options option1=optionvalue option2;
```

system option with value

additional options (may or may not use a value)

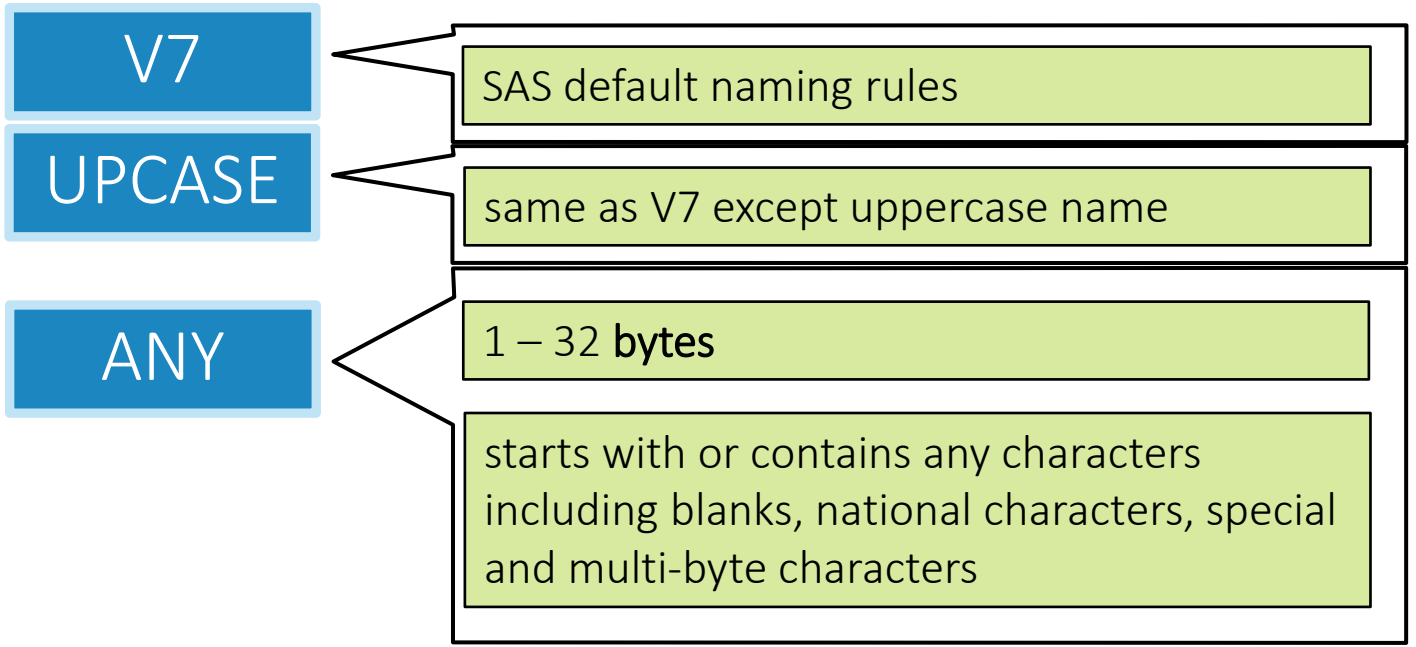
```
proc options;
run;
```

write current options to SAS log

System options are global!

Changing the Default Behavior for Variables

```
options validvarname=V7|UPCASE|ANY;
```



Changing the Default Behavior for Tables

```
options validmemname=COMPATIBLE|EXTEND;
```

COMPATIBLE

- SAS default naming rules

EXTEND

- 1 – 32 **bytes**
- includes national characters
- special characters except / \ * ? " < > |: -
- cannot begin with blank or period

Lesson Quiz

1. How many steps does this program contain?

a. one
b. two
c. four
d. eight

```
data national;
    set sashelp.baseball;
    BatAvg=nHits/nAtBat;
run;

proc contents data=national;
run;

proc print data=national;
run;

proc means data=national;
    var BatAvg;
run;
```

1. How many steps does this program contain?

a. one
b. two
c. four
d. eight

```
data national;
    set sashelp.baseball;
    BatAvg=nHits/nAtBat;
run;

proc contents data=national;
run;

proc print data=national;
run;

proc means data=national;
    var BatAvg;
run;
```

2. Running a SAS program can create which of the following?

a. log
b. output data
c. results
d. all of the above

2. Running a SAS program can create which of the following?

a. log
b. output data
c. results
d. all of the above

3. Which of the following is a SAS syntax requirement?

a. Begin each statement in column one.
b. Put only one statement on each line.
c. Separate each step with a line space.
d. End each statement with a semicolon.

3. Which of the following is a SAS syntax requirement?

a. Begin each statement in column one.
b. Put only one statement on each line.
c. Separate each step with a line space.
d. End each statement with a semicolon.

4. How many statements does this program contain?

a. five
b. six
c. seven
d. eight

```
*Create a cars report;

title "European Cars Priced Over 30K";
footnote "Internal Use Only";

proc print data=sashelp.cars;
   where Origin='Europe'
         and MSRP>30000;
   var Make Model Type
       Mpg_City Mpg_Highway;
run;
```

4. How many statements does this program contain?

a. five
b. six
c. seven
d. eight

```
*Create a cars report;

title "European Cars Priced Over 30K";
footnote "Internal Use Only";

proc print data=sashelp.cars;
   where Origin='Europe'
         and MSRP>30000;
   var Make Model Type
       Mpg_City Mpg_Highway;
run;
```

5. Which of the following steps is typically used to generate reports and graphs?

   a. DATA
   b. PROC
   c. REPORT
   d. RUN

5. Which of the following steps is typically used to generate reports and graphs?

a. DATA
b. PROC
c. REPORT
d. RUN

6. Does this comment contain syntax errors?

```
 /*
Report created for budget
presentation; revised October 15.
  */
proc print data=work.newloan;
run;
```

a. No. The comment is correctly specified.

b. Yes. Every comment line must end with a semicolon.

c. Yes. The comment is on more than one line.

d. Yes. There is a semicolon in the middle of the comment.

6. Does this comment contain syntax errors?

```
 /*
Report created for budget
presentation; revised October 15.
  */
proc print data=work.newloan;
run;
```

a. No. The comment is correctly specified.

b. Yes. Every comment line must end with a semicolon.

c. Yes. The comment is on more than one line.

d. Yes. There is a semicolon in the middle of the comment.

7. What result would you expect from submitting this step?

```
proc print data=work.newsalesemps
run;
```

a. a report of the **work.newsalesemps** data set
b. an error message in the log
c. the creation of a table named **work.newsalesemps**

7. What result would you expect from submitting this step?

```
proc print data=work.newsalesemps
run;
```

a. a report of the **work.newsalesemps** data set
b. an error message in the log
c. the creation of a table named **work.newsalesemps**

8. What happens if you submit the following program?

```
porc print data=work.newsalesemps;
run;
```

a. SAS does not execute the step.

b. SAS assumes that PROC is misspelled and executes the step.

8. What happens if you submit the following program?

```
porc print data=work.newsalesemps;
run;
```

a. SAS does not execute the step.

b. SAS assumes that PROC is misspelled and executes the step.

9. This program contains a syntax error because **National** is in different cases.

```
data national;
     set sashelp.baseball;
     BatAvg=nHits/nAtBat;
run;

proc means data=NATIONAL;
     var BatAvg;
run;
```

a. True

b. False

9. This program contains a syntax error because **National** is in different cases.

```
data national;
     set sashelp.baseball;
     BatAvg=nHits/nAtBat;
run;

proc means data=NATIONAL;
     var BatAvg;
run;
```

a. True

b. False

10. Which of the following is not a SAS programming interface?

a. SAS Enterprise Guide
b. SAS Manager
c. SAS Studio
d. SAS windowing environment

10. Which of the following is not a SAS programming interface?

a. SAS Enterprise Guide
b. SAS Manager
c. SAS Studio
d. SAS windowing environment