



临床研究SAS高级编程 ——SAS概述



About SAS

- One of the largest software company in the world
- Developed in the early 1970 at North Carolina State University
- Originally intended for management and analysis of agricultural field experiments
- Now the most widely used statistical software
- Used to stand for “Statistical Analysis System”
- CEO: Jim Goodnight

Prerequisites

- **Understand file structures and system commands on your operating systems**
- **Write system commands to create and access system files**
- **Understand programming logic.**



Components of the SAS System

- **Base SAS - data management and basic procedures**
- **SAS/STAT - statistical analysis**
- **SAS/GRAPH - presentation quality graphics**
- **SAS/OR - Operations Research**
- **SAS/ETS - Econometrics and Time Series Analysis**
- **SAS/QC - quality control**



SAS System

- **SAS in an integrated system of software solutions that enables you to perform the following tasks:**
 - ▶ **Data Entry, retrieval, and management**
 - ▶ **Report writing and graphics design**
 - ▶ **Statistical and mathematical analysis**
 - ▶ **Business forecasting and decision support**
 - ▶ **Operational Research and project management**
 - ▶ **Applications development**
- **The core of the SAS System is Base, STAT, Graph**

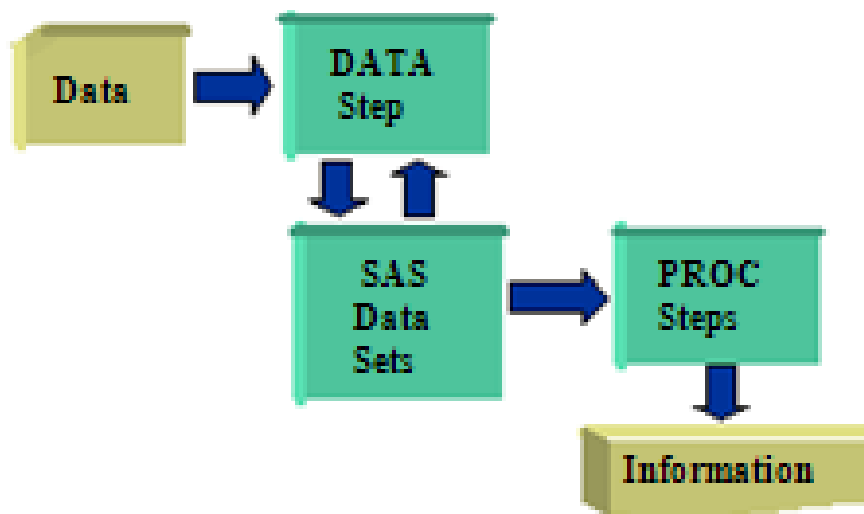


Basic Structure of SAS

- **DATA step and PROC step**
- **DATA step reads data from external sources, manipulates and combines it with other data set and prints reports**
- **PROC step perform analysis on the data, and produce output**
- **The most effective strategy for learning SAS is to concentrate on the details of the DATA step, and learn the details of each procedure as you have a need for them**

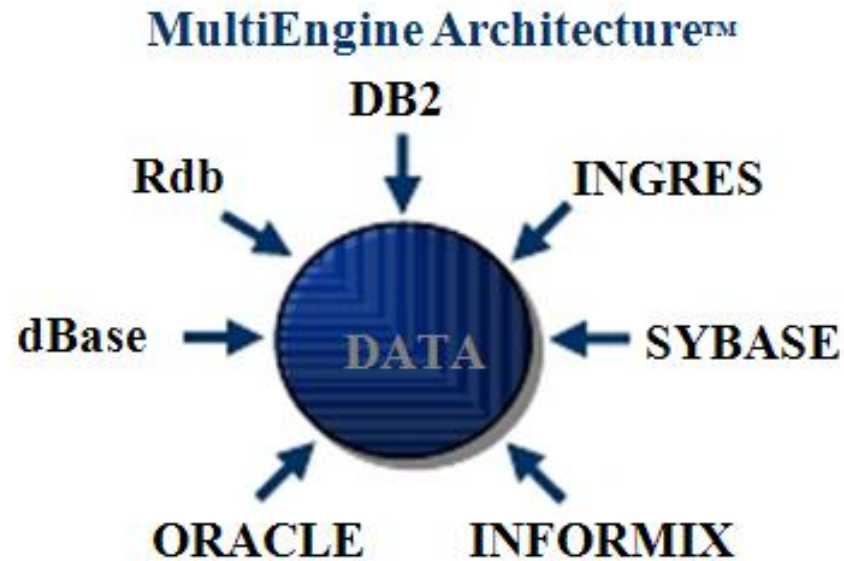


General Work Flow





Design of the SAS System

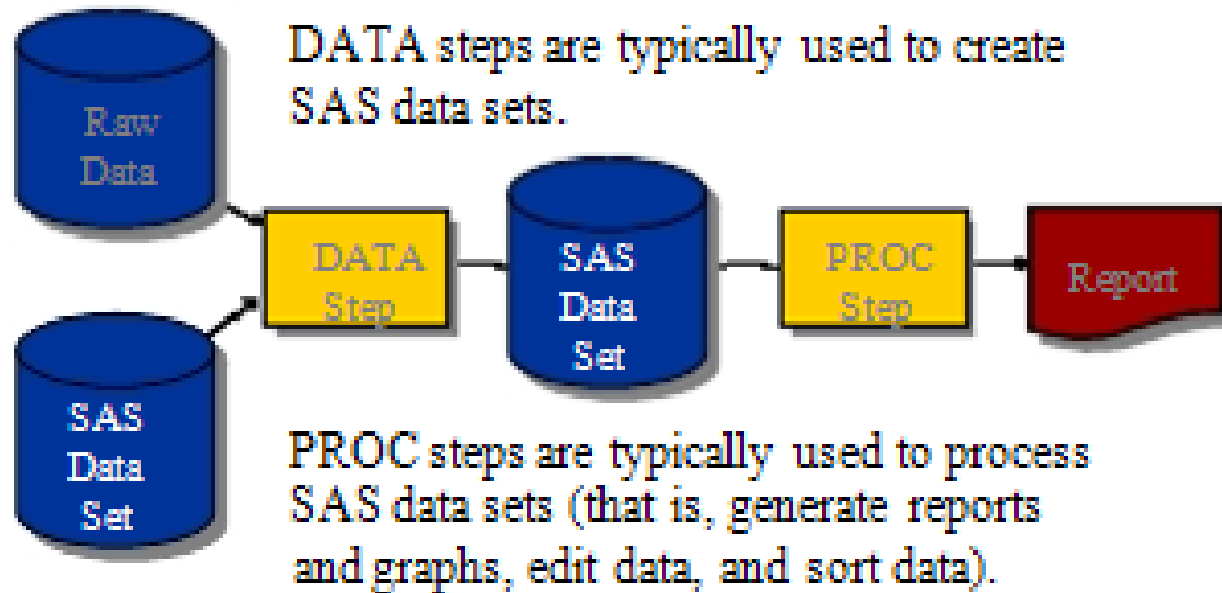


In order to access databases such as ORACLE, the SAS/ACCESS product for your given Database Management System (DBMS) must be licensed, in addition to Base SAS software.



Introduction to SAS Programs

- A SAS program is a sequence of steps that the user submits for execution





SAS Programs

```
data work.staff;  
  infile 'raw-data-file';  
  input LastName $ 1-20 FirstName $ 21-30  
         JobTitle $ 36-43 Salary 54-59;  
run;  
proc print data=work.staff;  
run;  
proc means data=work.staff;  
  class JobTitle;  
  var Salary;  
run;
```

DATA
Step

PROC
Steps



Step Boundaries

- **SAS Steps begin with a**
 - ▶ DATA statement
 - ▶ PROC statement
- **SAS Detects the end of a step when it encounters**
 - ▶ A RUN statement (for most steps)
 - ▶ A QUIT statement (for some procedures)
 - ▶ The beginning of another step



Step Boundaries

```
➔ data work.staff;  
   infile 'raw-data-file';  
   input LastName $ 1-20 FirstName $ 21-30  
     Job Title $ 36-43 Salary 54-59;  
➔ run;  
➔ proc print data=work.staff;  
➔ proc means data=work.staff;  
   class Job Title;  
   var Salary;  
➔ run;
```



Running a SAS Program

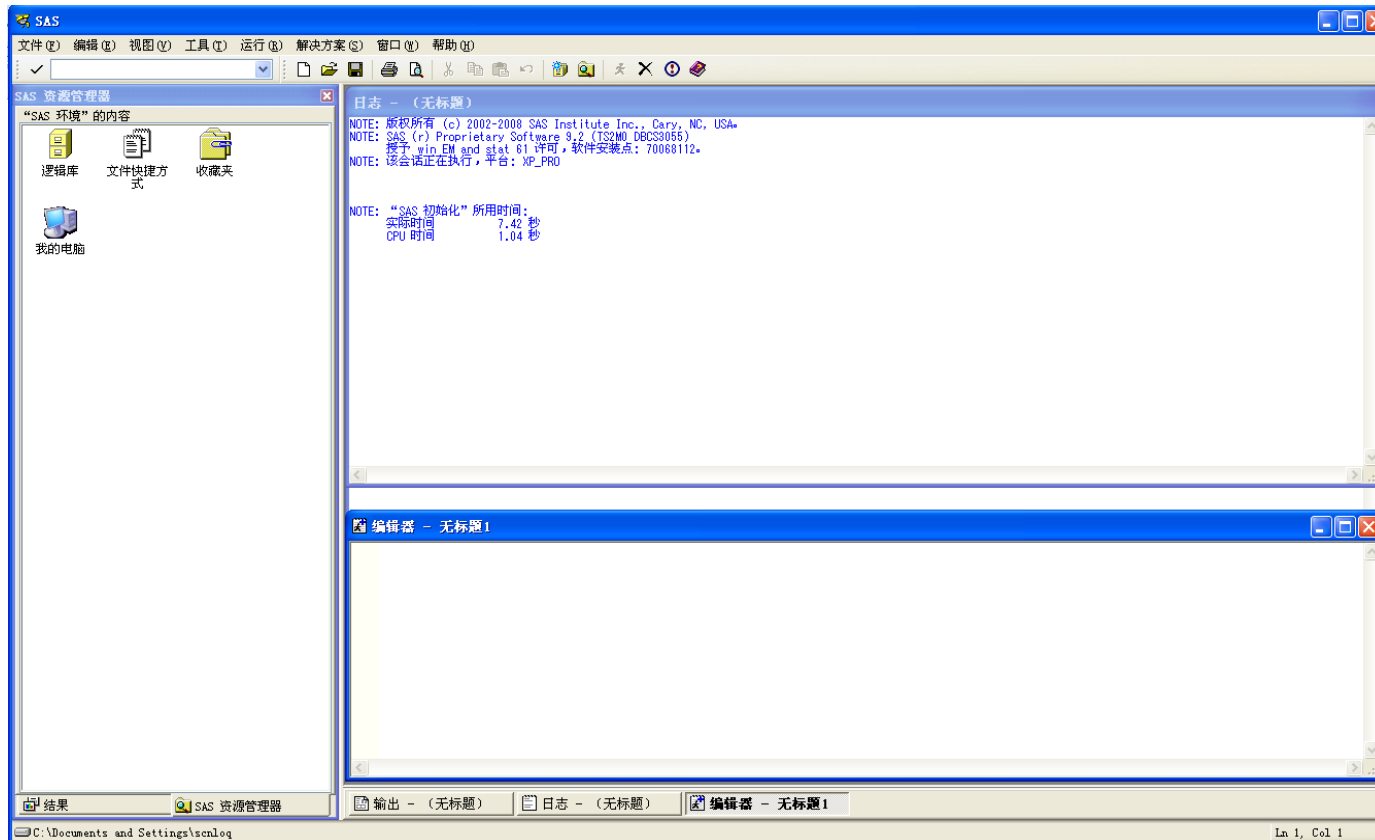


You can invoke SAS in

- ▶ Interactive windowing mode (SAS windowing environment)
- ▶ Interactive menu-driven mode (SAS Enterprise Guide, SAS/ASSIST, SAS/EIT , etc)
- ▶ Batch mode
- ▶ Noninteractive mode

SAS Windowing Environment

Interactive windows enable you to interface with SAS



```
1 data work.staff;
2   infile 'raw-data-file';
3   input LastName $ 1-20 FirstName $ 21-30
4     JobTitle $ 36-43 Salary 34-39;
5   run;
NOTE: The infile 'raw-data-file' is:
      File Name= 'raw-data-file',
      RECFM=V,LRECL=256
NOTE: 18 records were read from the infile 'raw-data-file'.
      The minimum record length was 39.
      The maximum record length was 59.
NOTE: The data set WORK.STAFF has 18 observations and 4 variables.
6   proc print data=work.staff;
7   run;
NOTE: There were 18 observations read from the dataset WORK.STAFF.
8   proc means data=work.staff;
9     class JobTitle;
10    var Salary;
11   run;
NOTE: There were 18 observations read from the dataset WORK.STAFF.
```



SAS Output example

PROC PRINT Output

The SAS System

Obs	LastName	First Name	JobTitle	Salary
1	TORRES	JAN	Pilot	50000
2	LANGKAMM	SARAH	Mechanic	80000
3	SMITH	MICHAEL	Mechanic	40000
4	LEISTNER	COLIN	Mechanic	36000
5	WADE	KIRSTEN	Pilot	85000
6	TOMAS	HARALD	Pilot	105000
7	WAUGH	TIM	Pilot	70000
8	LEHMANN	DAGMAR	Mechanic	64000
9	TRETTAHN	MICHAEL	Pilot	100000
10	TIETZ	OTTO	Pilot	45000
11	ODONOGHUE	ART	Mechanic	52000
12	WALKER	THOMAS	Pilot	95000
13	NOROVITA	JOACHIM	Mechanic	78000
14	OESTERBERG	ANIA	Mechanic	80000
15	LAUFFER	CRAIG	Mechanic	40000
16	TORR	JUGDISH	Pilot	45000
17	WAGSCHAL	NADIA	Pilot	77500
18	TOERMOEN	JOCHEN	Pilot	65000



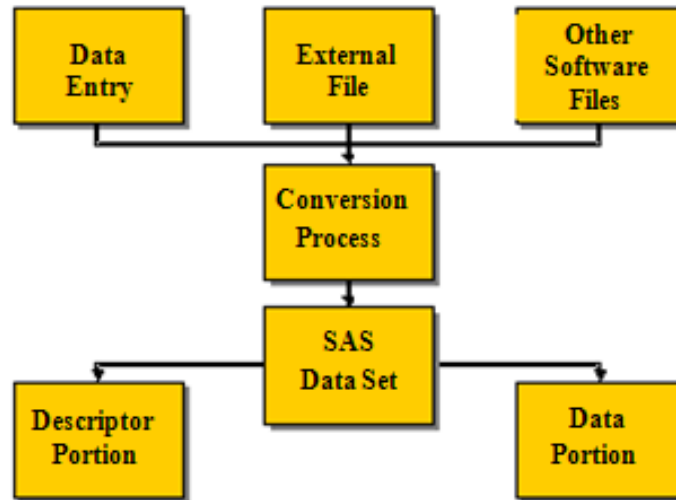
Mastering Fundamental Concepts

- Define the components of a SAS data set
- Define a SAS Variable
- Identify a missing value and a SAS date value
- State the naming conventions for SAS data sets and variables
- Explain SAS Syntax rules
- Investigate a SAS data set using the CONTENTS and PRINT procedures



SAS Data Sets

SAS Data Sets



22

Data must be in the form of a SAS data set to be processed by many SAS procedures and some DATA steps statements

A SAS program is a file that contains SAS code

A SAS Data Set is a specially structured file that contains data values



SAS Data Sets

- SAS data sets have a descriptor portion and a data portion
- Descriptor portion contains: data set name, data set label, data/time created, storage information, number of observations, and information for each variable: for example, name ,type, length, position, format, informat, label
- Data Portion: the value for each variable

```
PROC CONTENTS DATA=sas-data-set; RUN;
```

or

```
PROC DATASETS LIBRARY=sas-library;
```

```
CONTENTS sas-data-set; Run;
```



Example

Partial PROC CONTENTS Output

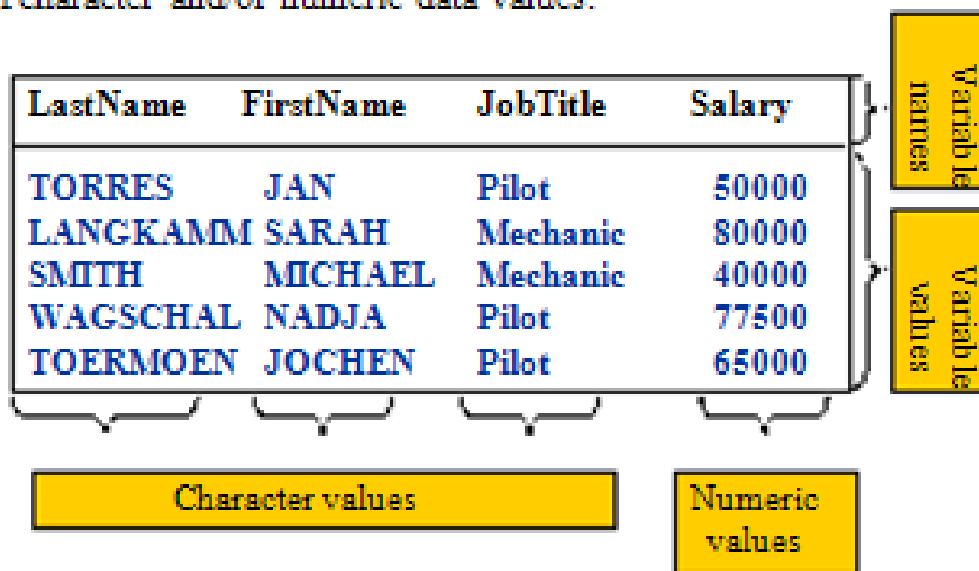
The SAS System			
The CONTENTS Procedure			
Data Set Name:	WORK.STAFF	Observations:	18
Member Type:	DATA	Variables:	4
Engine:	V9	Indexes:	0
Created:	Monday, December 01, 2003 10:36:59 AM	Observation Length:	48
Last Modified:	Monday, December 01, 2003 10:36:59 AM	Deleted Observations:	0
Protection:		Compressed:	NO
Data Set Type:		Sorted:	NO
Label:			
Alphabetic List of Variables and Attributes			
#	Variable	Type	Len
2	FirstName	Char	10
3	JobTitle	Char	8
1	LastName	Char	20
4	Salary	Num	8

This is partial view of the default PROC CONTENT output.
PROC CONTENTS output also contains information about the physical location of the file and other information



SAS Data sets: Data Portion

The *data portion* of a SAS data set is a rectangular table of character and/or numeric data values.





SAS Variable Values



There are two types of variables



Character

- Contains any value: letter, numbers, special characters, and blanks. Character values are stored with a length of 1 to 32767 bytes. One byte equals one character



Numeric

- Stored floating point numbers in 8 bytes of storage by default. Eight bytes of floating point storage provide space for 16 or 17 significant digits. You are not restricted to 8 digits



SAS Data Set and Variable Names

SAS Names

- ▶ Can be 32 characters long
- ▶ Can be uppercase, lowercase, or mixed-case
- ▶ Must start with a letter or underscore, subsequent characters can be letters, underscore, or numeric digits

In version 6 and earlier, data set and variable names can only be a maximum of 8 characters long.

Starting in Version 8, special characters can be used in data set and variable names if you put the name in quotes followed immediately by letter N

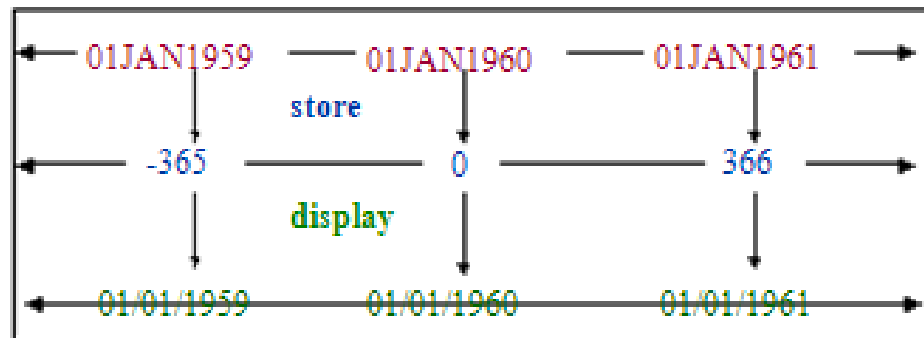
Example: 'Flight#'`n`;

In order to use special characters in variables names, the `VALIDVARNAME` option must be set to `ANY`;

Example: `options validvarname=any;`

SAS Data Values

- SAS stores date values as numeric values
- A SAS date value is stored as the number of days between Jan 1st, 1960, and a special date



Missing Data Values

- A value must exist for every variable for each observation.
- Missing values are valid values
 - ▶ A numeric missing value is displayed as a period
 - ▶ A character missing value is displayed as a blank

LastName	FirstName	JobTitle	Salary
TORRES	JAN	Pilot	50000
LANGKAMM	SARAH	Mechanic	80000
SMITH	MICHAEL	Mechanic	.
WAGSCHAL	NADJA	Pilot	77500
TOERMOEN	JOCHEN		65000

SAS Syntax Rules

- **SAS statements usually begin with an identifying keyword, always end with a semicolon;**
- **SAS statements are free-format**
- **One or more blanks or special characters can be 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**

SAS programming statements are easier to read if you begin DATA, PROC, and RUN statements in column one and indent the other statements



SAS Comments

- **Type `/*` to begin a comment**
- **Type `*/` to end the comment**
- **An additional method for commenting one line of code is to use the asterisk at the beginning of the comment**

Exercises

- SAS statements usually begin with a _____
- Every SAS statement ends with a _____
- Character variable values can be up to _____ characters long, and use _____byte(s) of storage per character
- A SAS variable name has _____to _____characters and begin with a _____or an _____
- By default, numeric variables are stored in _____bytes of storage
- The internally stored SAS date value for Jan 23rd, 1960, is _____



Exercises

- A missing character value is displayed as a _____
- A missing numeric value is displayed as a _____
- What are the two kinds of Steps?
- What are the two portions of SAS data set?
- What are the two types of variables?



Explore SAS Environment under Windows



Enhanced Editor

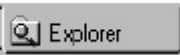
- ▶ It provides many helpful features, including color coding and automatically retaining the program after each submit, eliminating the need to recall your program

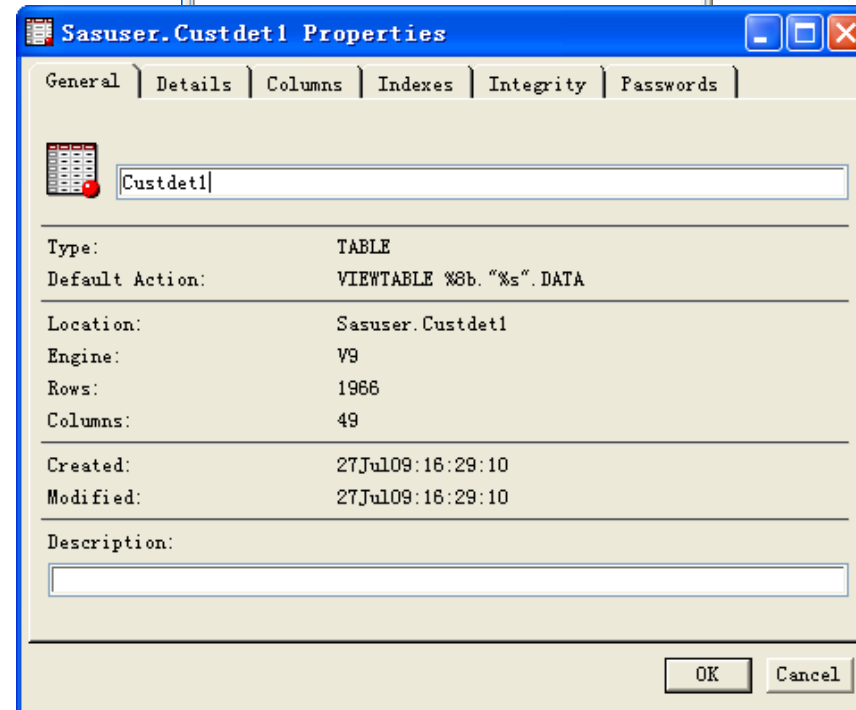


Submit SAS program

- ▶ Issue the submit command
- ▶ Click on
- ▶ Select Run Submit on the toolbar

Explore SAS Libraries and Files

- Select  tab on the SAS window bar to open the Explorer window
- You can expand and collapse directories, and drill-down and open specific files.
- You can view the properties of each file by right-click the file and choose properties.





SAS Data Libraries

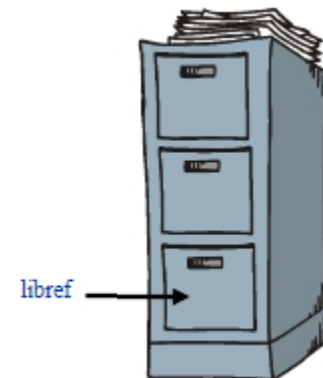
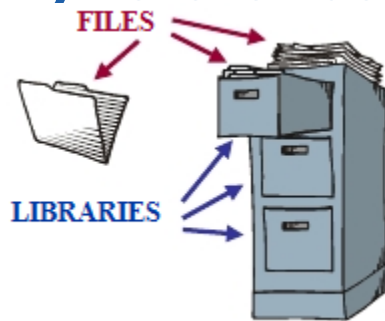


Objectives

- ▶ Explain the concept of a SAS data library
- ▶ State the difference between a permanent library and a temporary library
- ▶ Use the CONTENTS procedure to investigate a SAS data library

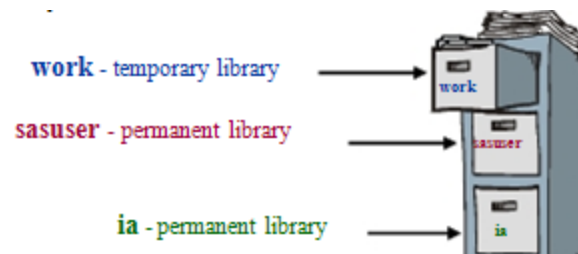
SAS Data Libraries

- A SAS data library is a collection of SAS files that are recognized as a unit by SAS
- You can think 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
- Assigning a libref: regardless of which OS you use, you identify SAS data libraries by assigning each a library reference name (libref)



SAS Data Libraries

- When you invoke SAS, you automatically have access a temporary and a permanent SAS data library
- You can create and access your own permanent libraries
- The Work library and its SAS data files are deleted after your SAS session ends
- SAS data sets in permanent libraries are saved after your SAS session ends

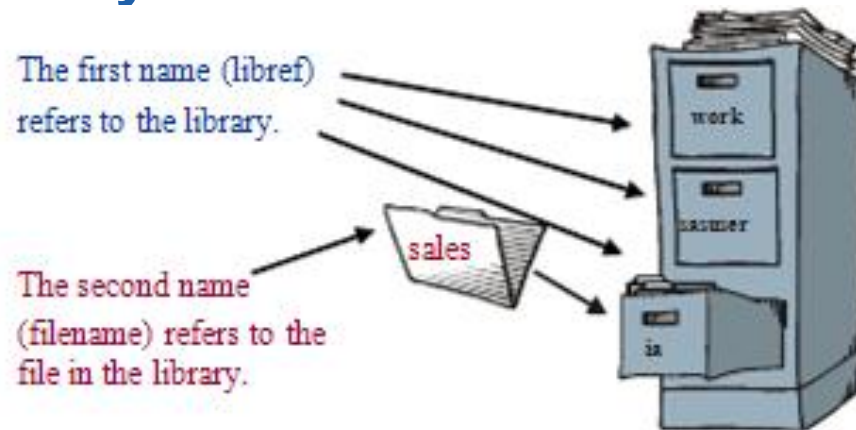


Assigning a Libref

- You can use the **LIBNAME** statement to assign a libref to a SAS data library
- General form of the **LIBNAME** statement:
 - ▶ **LIBNAME** libref 'SAS-data-library' <options>;
- Rules for naming a libref:
 - ▶ Must be 8 characters or less
 - ▶ Must begin with a letter or underscore
 - ▶ Remaining characters are letters, numbers, or underscores

Two-level SAS Filenames

- Every SAS file has a two-level name:
 - ▶ Libref.filename
- The libref work can be omitted when you refer to a file in the work library.



- You can use `PROC CONTENTS DATA=libref._all_nods;RUN;` to list all SAS files.



Q&A