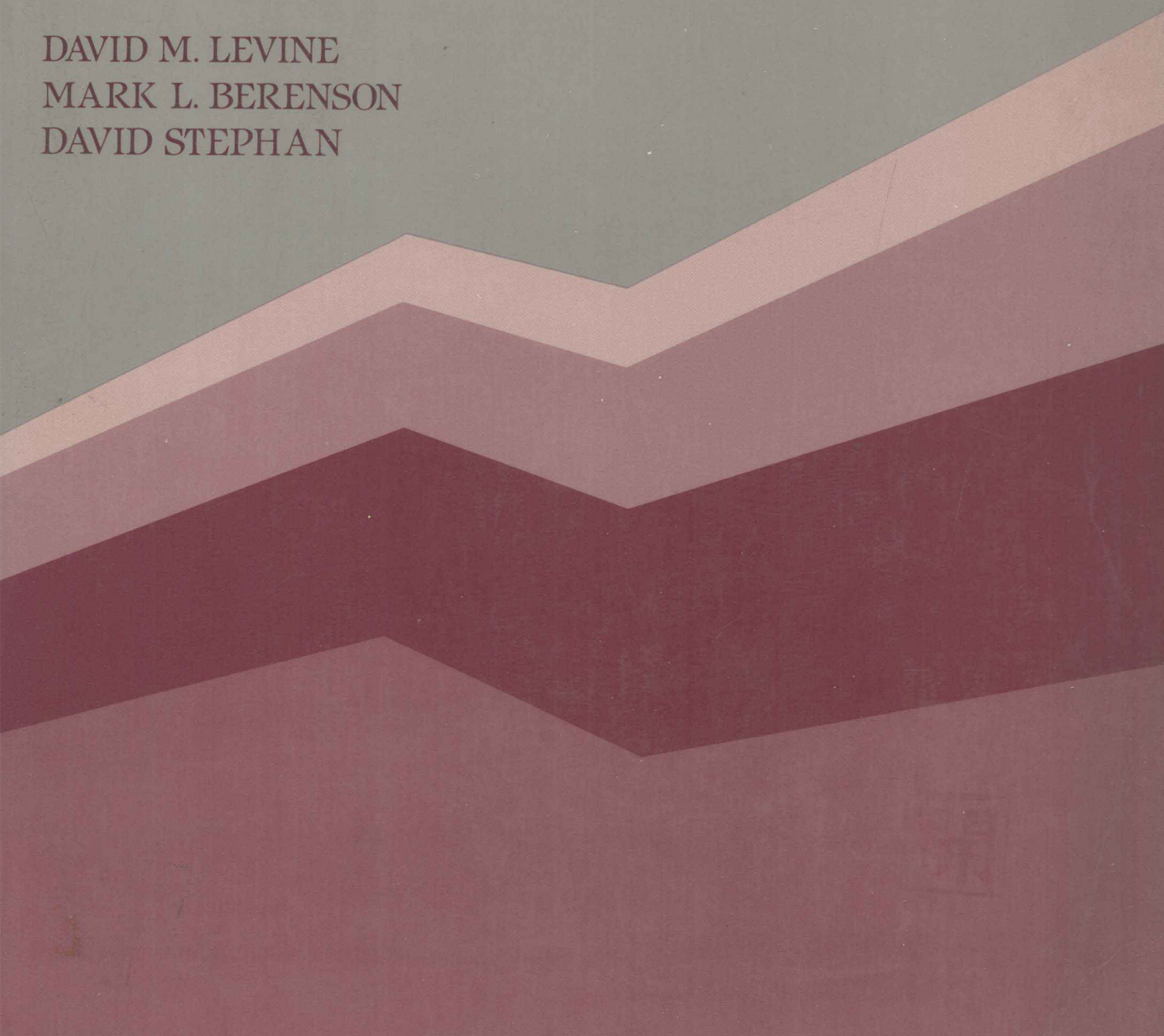
USING THE STATISTICAL ANALYSIS SYSTEM (SAS) WITH BASIC BUSINESS STATISTICS

DAVID M. LEVINE
MARK L. BERENSON
DAVID STEPHAN



USING THE STATISTICAL ANALYSIS SYSTEM (SAS) WITH BASIC BUSINESS STATISTICS

DAVID M. LEVINE
MARK L. BERENSON
DAVID STEPHAN
Baruch College
City University of New York

PRENTICE-HALL, INC., ENGLEWOOD CLIFFS, NEW JERSEY 07632

Editorial/production supervision: Suzette Walker
Cover adaptation: Debra Lynne Watson
Manufacturing buyer: Ed O'Dougherty

Printed in the United States of America

10 9 8 7 6 5 4 3 2 1

ISBN: 0-13-058701-X

Prentice-Hall International, Inc., London
Prentice-Hall of Australia Pty. Limited, Sydney
Editora Prentice-Hall do Brasil, Ltda., Rio de Janeiro
Prentice-Hall Canada Inc., Toronto
Prentice-Hall of India Private Limited, New Delhi
Prentice-Hall of Japan, Inc., Tokyo
Prentice-Hall of Southeast Asia Pte. Ltd., Singapore
Whitehall Books Limited, Wellington, New Zealand

PREFACE

Today's increasing use of computers in business decision-making has created the demand for reliable, ready-to-use programming or software. One aspect of this demand has been the development of generalized sets of programs capable of executing statistical analyses on many different computer systems. These so-called statistical packages allow relatively unskilled users to benefit from using computers as a tool in statistical problem-solving. The standardized nature of these packages insures that the results of the computerized analyses will be reliable and that the time spent on implementating the statistical programs will be minimized.

Statistical packages form an important part of modern decision support systems. However, several potential problems exist with the use of such packages. Users unskilled in statistics may instruct the computer to perform an inappropriate or invalid analysis, or they may be unable to properly interpret the results of the requested analysis. Furthermore, although statistical packages are relatively easy to learn (certainly easier than having to write a series of, say, FORTRAN programs), users still must first become familar with the command languages of these packages in order to perform any analysis.

These problems are reflected in statistics courses that incorporate the use of statistical packages. Unfortunately, no one textbook at this level can provide a student with the necessary instruction to gain mastery of both statistical concepts as well as statistical packages, if only because of the many different packages that are currently in use. At best an elementary statistics textbook, such as Berenson and Levine's Basic Business Statistics: Concepts and Applications (Prentice-Hall, 1983), can display representative samples of outputs from several different and widely-used packages. To provide full instructions in the use of a specific statistical package requires the publication of a separate manual.

With this in mind, the authors of Basic Business Statistics, along with David Stephan, have written Using SAS with Basic Business Statistics for students in courses that use SAS Institute's Statistical Analysis System (SAS). Instructors will find that the sample programs presented in this manual use data from the results of the marketing survey described in the Berenson and Levine textbook (and briefly presented here in Chapter One). As written, all the sample programs access this data from the special survey data base tape available free to institutions that adopt the Basic Business Statistics text* Of course, these programs can be easily modified by users wishing to substitute other data. (See Sections 1.5 and 1.6 for the changes necessary).

This manual assumes no previous training in the use of computers or SAS. Chapter 1 presents some necessary basic computer concepts and definitions along with the commands necessary to describe data in a SAS program. Chapter 2 explains how data can be manipulated and modified in a SAS program. Chapter 3 summarizes some elements in common to all SAS PROCedures. Thereafter, each subsequent chapter is keyed to a particular statistical topic or methodology.

---

* For further information contact your local Prentice-Hall representative or write to: Director of College Marketing, Quantitative Methods in Business, Prentice-Hall Inc., Englewood Cliffs, New Jersey 07632.

Each of these chapters contains:

1) a brief review of the statistical topic or methodology being used.

2) a discussion of the specific SAS statements needed to invoke the appropriate statistical analysis.

3) an interpretation of the output generated by the SAS program.

Although the authors suggest that the chapters be read in order, any chapter can be referenced after a reading of the initial three chapters. For more advanced users, an appendix explaining the use of SAS data sets and the creation of random samples is presented.

We would like to thank the SAS Institute for permission to use SAS output and syntax in this manual. We are also particularly grateful to Ann Festa and Ruth Meyer for their careful typing.

David M. Levine
Mark L. Berenson
David Stephan

Contents

CHAPTER 1

Introduction to the Statistical Analysis System (SAS)

1.1 Before Using SAS

Before we can begin using the Statistical Analysis System (SAS), there are certain terms and concepts with which we should be familar. Batch processing is the method of grouping multiple computer jobs for submission to a computer system at the same time. As each of these jobs is executing, there exists no means by which we can modify instructions or data. We also cannot review the output until the job has finished execution. In contrast, interactive processing is the method by which we can enter modifications to instructions or data as the job is executing and review output as it is generated.

SAS Institute, Inc., markets both batch and interactive versions of SAS. All versions use the same syntax for SAS statements and produce similar results. Differences exist, however, in the manner by which we request to use, or access, SAS and any data previously stored in the computer system. These differences are not a function of the various SAS versions, per se, but rather they reflect differences between the various batch and interactive master control programs under which all SAS jobs execute.

In interactive systems, accessing SAS may be as easy as entering the word "SAS". Unfortunately, due to variations from one system to another, no one set of words or commands can be listed for general use.

The same, however, is not true for batch systems. In all batch systems job control language (JCL) statements are used to access SAS. Because SAS currently executes only on IBM or IBM plug-compatible computers that use similar control programs, the four JCL statements listed below may be similar if not identical to ones every batch user of SAS will need.

```
//jobname JOB (account-number) REGION=240K
//label EXEC SAS
//SYSIN DD *
(SAS program statements)
/*
```

The first JCL statement identifies the physical start of our job. The second statement requests access to SAS. The third and fourth statements serve as "bookends" between which all SAS program statements will be placed.

How are these JCL (and SAS program) statements entered into a computer system? Traditionally, we conceive of these statements being keypunched and the resulting job deck of punched cards being placed in a card reader for submission. However, in most installations punched cards are being phased out and on-line video display terminals are replacing keypunches and card readers. While these terminals eliminate the need for physically punched cards, in many cases they preserve the concept of a punched card by displaying lines of data up to 80 characters in length. Accordingly, throughout this manual, the term card is used to mean any

(up to) 80 character unit of information and not necessarily a physically punched card.[1]

1.2 Before Using SAS Statistical Analysis

Originally developed at North Carolina State University, the Statistical Analysis System is a software system designed for data analysis. SAS is not merely a statistical package, but a report writer and file maintenance package as well. This manual, however, concentrates only on the elements necessary to write SAS programs to perform statistical analyses. Students interested in learning about the other capabilities of SAS will find those features are discussed in detail in SAS (1979).

Due to its multipurpose nature, SAS is perhaps more difficult to master than MINITAB (another package discussed in Basic Business Statistics). However, SAS is relatively easy to learn because it develops patterned SAS language statements after statements found in a procedure oriented language such as PL/I.

```
COMMENT FIGURE ONE A 'DO NOTHING PROGRAM';
DATA NOTHING;
PROC PRINT DATA = NOTHING;
```

Figure 1.1 A Do Nothing Program Using SAS

Figure 1.1 represents just about the simplest SAS program one could write. Although unproductive in that it performs no statistical analysis, the program illustrates features common to all SAS programs.

The program consists of three statements. Each statement is terminated by a semi-colon and contains words or symbols separated by at least one blank. Although not required by SAS, these statements have been typed on separate cards to improve the readability of the program. For a a similar reason, a comment statement has been included to explain the purpose of the program. This first statement is only for program documentation and is not processed by SAS. The program's second and third statements illustrate the fundamental rule of SAS statistical programming:

> A SAS program consists of one or more DATA steps
> and one or more PROC (PROCEDURE) steps.

In Figure 1.1 there is only one of each type of step, and each step consists of a single statement. The DATA statement asks SAS to create a SAS data set and to give it the symbolic name NOTHING. The PROC PRINT statement then requests SAS to print the contents of this data set. As such, this program illustrates in principle the operation of SAS:

---

1. The reader should note that using this size is not a prerequisite for accessing SAS. SAS statements can be coded in units that are up to 133 characters in length. Limiting each line to a shorter length, though, makes for more easily read output.

The DATA steps convert the data to be analyzed into a special format, called the SAS data set, and the PROC steps act upon these special data sets.

Will this program execute? One might guess that "nothing will happen." In fact, this SAS program generated a listing containing the program's three statements and the actions taken by SAS in response to them (see Figure 1.2).

```
S T A T I S T I C A L   A N A L Y S I S   S Y S T E M
NOTE: THE JOB DMLBB239 HAS BEEN RUN UNDER RELEASE 79.5 OF SAS
      AT THE CITY UNIVERSITY OF NEW YORK (00121).
NOTE: SAS OPTIONS SPECIFIED ARE:
      NOCENTER NODATE NONUMBER LS=64


1           COMMENT FIGURE 1.2;
2           DATA NOTHING;

NOTE: DATA SET WORK.NOTHING HAS 1 OBSERVATIONS AND 0 VARIABLES.
4766 OBS/TRK
NOTE: THE DATA STATEMENT USED 0.11 SECONDS AND 164K.

3           PROC PRINT DATA=NOTHING;

NOTE: NO VARIABLES IN DATA SET
      OR NO VARIABLES TO BE ANALYZED.
NOTE: THE PROCEDURE PRINT USED 0.29 SECONDS AND 160K
      AND PRINTED PAGE 1.

NOTE: SAS USED 164K MEMORY.
NOTE: SAS INSTITUTE INC.
      SAS CIRCLE
      BOX 8000
      CARY, N.C. 27511
```

Figure 1.2 Job Listing for the DO NOTHING Program

SAS always generates a listing called the SAS log for every SAS job run. The SAS log given in Figure 1.2 includes several messages preceded by the word NOTE. Every time SAS attempts to execute a DATA or PROC statement, SAS prints a NOTE explaining the action or actions taken. In this case, SAS "notes" that there are no variables to be analyzed and that there is nothing to print out.

Sometimes a SAS log includes messages that are preceded by the word ERROR. Any time such a message occurs on the SAS log, something unexpected has occurred. The "unexpected" might include a typographical error, invalid data, or an improper or incorrect usage of SAS language statements.

As in the case with any language or package that performs error-checking, SAS will sometimes incorrectly diagnose the source of the error. However, SAS will always properly indicate (by underlining) the place where the unexpected occurred. By closely examining the statement containing the underlined word or phrase and the statement that immediately precedes it, almost every error can be diagnosed by the student.[2]

1.3 Getting Started Using SAS

The "do nothing" program of the previous section could do no productive work because it did not contain SAS statements to describe a set of data. To help explain the choices that must be made before this description can be given, consider the set of data shown in Table 1.1

Table 1.1

Statistics Exam Scores for Ten Students

	Scores		
Last Name	I	II	III
Bovary	82	65	74
Brown	64	57	65
Caufield	73	80	65
Garp	84	75	66
Holmes	94	96	99
Karenina	90	96	87
Marner	80	73	92
Prynne	67	84	90
Sawyer	88	94	80
Zhivago	78	86	80

Table 1.1 displays the scores from three exams of a group of students enrolled in an "imaginery statistics course." Before SAS can analyze this data, the data must be entered into the computer system. Although SAS allows this data to be entered in a variety of machine readable forms, it will, by default, assume that the data is in the form of letters, numbers, and special characters as per Table 1.1.[3] This assumption will be used by all programs presented in this manual.

Specifying the form of the data is not enough, however, for SAS also needs to know the ordering of the data. Will the names of all the students be entered first, followed by all the scores for exams one, two, and three, respectively? Or will all the data associated with the

---

2. Occassionally, the "unexpected" is a problem beyond the control of SAS to identify and/or correct, such as a program that runs out of memory. Such problems may cause the operating system to abnormally end (ABEND) the SAS job. If an ABEND occurs, some output, including a partial SAS log may still be produced. However, the SAS log for the ABENDed SAS job will never contain the SAS institute name and address--all other SAS jobs always contain this information at the end of the log.
3. Advanced SAS users will note that this is an oversimplification of the data conversion abilities of SAS. Full details of the data conversion process are given in Chapter 8 of the SAS Users Guide.

first student (his name and three test scores) be entered contained on the data for the second student second, and so on until the must be spec SAS offers alternatives, but the simplest method of orderin SAS permits latter, that is, placing all the data concerning each student (or, more generally, about each sample point) together. To this unit of information, SAS gives the name observation; readers familiar with computer programming will recognize it as another name for a data record. Although observations may be continued over a series of cards, for simplicity, only sets of data with observations that can be contained on one card ("unit record" data sets) are discussed in this manual.

Having specified the form and ordering of the data, one additional thing, the storage medium to be used for the data also must be specified before SAS can access a set of data. SAS distinguishes between sets of data that have been previously stored in the computer system on a magnetic tape or disk and sets of data that have not. SAS permits these latter sets to be included in the body of a SAS program. This so called "instream" data method is explained first.

1.4 The DATA Step for Instream Data

Figure 1.3 shows a SAS program that lists a set of instream data corresponding to Table 1.1. Note that in comparison to the "do nothing" program, this program contains two additional DATA step statements, the INPUT statement and the CARDS statement.[4]

```
DATA SCORES;
  INPUT LASTNAME$ 1-10 EXAM1 12-13
  EXAM2 15-16 EXAM3 18-19;
CARDS;
BOVARY␣␣␣␣␣82␣65␣74
BROWN␣␣␣␣␣64␣57␣65
CAUFIELD␣␣␣73␣80␣65
GARP␣␣␣␣␣84␣75␣66
HOLMES␣␣␣␣94␣96␣99
KARENINA␣␣␣90␣96␣87
MARNER␣␣␣␣80␣73␣92
PYRNNE␣␣␣␣67␣84␣90
SAWYER␣␣␣␣88␣94␣80
ZHIVAGO␣␣␣␣78␣86␣80
PROC PRINT;
  TITLE CLASS ROSTER AND SCORES;
```

Figure 1.3 SAS Program for the data of Table 1.1

In all SAS programs the DATA statement is the first part of the DATA step. The format for this statement is

```
DATA data-set-name;
```

4. A ␣, representing a blank space, has been included on the data cards for emphasis.

where data-set-name may be any sequence of up to eight alphanumeric characters that begins with a letter. In Figure 1.3, the data-set-name is SCORES. We note that a blank space must be provided between DATA and the data-set-name.

The second part of the DATA step, the INPUT statement, describes the ordering or layout of each observation (data record). Among its many similar formats is

```
INPUT name-of-variable1 column(s) name-of-variable2 column(s)
                ...   name-of-variablei column(s);
```

where the name-of-variable may be any sequence of up to eight alphanumeric characters that begins with a letter and column(s) is the column number or range of columns that contains the value for the variable. A range of columns is represented by a hyphen. For example, 1-10 indicates that the value for a variable is contained in the first ten columns on a card. (Optionally, the column numbers may be omitted if each data value is separated by one or more blank characters). For a variable that takes on an alphanumeric value, a dollar sign ($) must immediately follow the name-of-variable as is the case with the variable LASTNAME in Figure 1.3.

The third part of the DATA step, the CARDS statement, informs SAS that the instream data immediately follows. SAS will interpret each subsequent card as a data card until it encounters a card that contains a semi-colon. Although, as in Figure 1.3, this semi-colon may be part of a PROC step statement, a card containing only a semi-colon (a "null" SAS statement) is allowed and may be preferred as the means to signal the end of the set of data.

The fourth SAS statement, PROC PRINT, marks the beginning of the program's PROC step. This statement will be discussed later.

1.5 The DATA step for Previously Stored Sets of Data

Sometimes many different analyses will be run on the same set of data. In such cases, it may be convenient to permanently store the set of data on either a magnetic tape or disk device. Then, each time the set of data is to be used, it does not have to be physically reentered into the computer system.

Figure 1.4 shows the SAS program of Figure 1.3, altered to accept from a tape or disk the now permanently stored set of exam scores data.

```
DATA SCORES; INFILE EXAMS;
  INPUT LASTNAME$ 1-10 EXAM1 12-13
  EXAM2 15-16 EXAM3 18-19;
PROC PRINT;
  TITLE CLASS ROSTER AND SCORES;
```

Figure 1.4 SAS Program for the data of Table 1.1 Assuming Previous Storage

Note that the CARDS statement and the instream data have been eliminated from the earlier program and, in their place, an INFILE statement has been inserted after the DATA statement. The format of this new statement is

```
INFILE ddname;
```

where ddname (short for data definition name) is a value also found on a system control statement used to define input and output data sets. The nature of this system control statement will vary depending on the version of SAS being used. The system control statements for three common versions of SAS are given in Table 1.2.

Table 1.2

Control Statement Used with the INFILE statement

Version of SAS	Statement
batch	DD JCL statement placed directly after the EXEC SAS statement.
interactive-TSO	ALLOCate statement issued before requesting set of data.
interactive-CMS	FILEDEFinition statement issued before requesting set of data

1.6 Documenting Variable Names

Often, the (up to) eight characters that SAS allows for naming variables is insufficient to fully describe the name of a variable. In other cases where eight characters seems sufficient to name a variable (for example, EXAM1,) the exact meaning may be obscure to all but the writer of the SAS program. For both these reasons and to provide variable legends in the output generated by various SAS procedures, a DATA step may contain a LABEL statement.

The format of this statement is

```
LABEL variable-name1 = label for variable-name1
      variable-name2 = label for variable-name2...
      variable-namei = label for variable-namei ;
```

where a label for a variable-name is any combination of up to 40 printable characters in length, subject to these two conditions:

1) if the label contains equal signs or semi-colons then the entire label must be enclosed in single quotation marks.

2) if the label contains any single quotation marks then each of these single quotation marks must be followed by an additional single quotation mark.

As such, all of the following are valid examples of variable labels:

```
THIS IS A FORTY-CHARACTER VARIABLE LABEL
'THIS 37-CHARACTER LABEL CONTAINS AN ='
LABEL CONTAINS 3 SINGLE QUOTE MARKS''''''
```

Although not required by SAS, LABEL statements enhance the documentation of a SAS program and, therefore, in the opinion of the authors, they should always be used.

The LABEL statement for the imaginary scores set of data discussed previously might be

```
LABEL LASTNAME = STUDENT'S LASTNAME
      EXAM1 = SCORE ON FIRST EXAM
      EXAM2 = SCORE ON MIDTERM EXAM
      EXAM3 = SCORE ON FINAL EXAM;
```

Note that in this LABEL statement, each variable-name and label has been placed on a separate card. This, too, is not required by SAS, but such placement increases the readability of each variable label. Although the variables in this LABEL statement have been arranged in the order of appearance in the INPUT statement, they may be arranged in any order. Indeed, for very large lists of variables, an arrangement in alphabetical order may be preferred.

1.7 The DATA Step for the Marketing Survey

In Basic Business Statistics, sample data from a marketing survey of a population containing 1250 students is used to help explain and illustrate various statistical procedures. The questionnaire is displayed in Figure 1.5. In a similar manner, the same survey sample (of 60 respondents) will be used throughout this manual to help explain and illustrate the SAS statements necessary to perform the desired analysis.

As a result, all SAS programs developed in subsequent chapters will share the same initial DATA step. This DATA step is given in Figure 1.6. The INPUT statement of this step names 27 variables that corresponds to the survey's 27 questions plus an identification number. As written, the DATA step uses a set of data previously stored in the computer system. However, the step can be easily modified for instream data by eliminating the INFILE statements and adding a CARDS statement as discussed in Section 1.5.[5]

---

5. Those users wishing to access their copies of the Basic Business Statistics data base tape can use this DATA step unmodified. In batch systems, the necessary additional job control statement would be
//SURVEY DD DSN = BBS2ED.SAMPLE,LABEL = (n,SL),DISP = SHR
where n represents a file number as explained in Appendix C.

Using the same initial DATA step is not a requirement of SAS. For example, if one wanted to run an analysis examining only the sex and major of the student respondents then the INPUT statement

```
INPUT SEX 6 MAJOR 10;
```

and the LABEL statement

```
LABEL SEX = SEX OF INDIVIDUAL
      MAJOR = MAJOR AREA OF STUDY;
```

could be substituted for their lengthier counterparts in Figure 1.6. However, keeping the same initial DATA step reduces time spent preparing a SAS program. This method also has another benefit as will be discussed in Appendix B.

Codes	STUDENT QUESTIONNAIRE
1 2 3 4	
6	1. What is your sex? Male [1] Female [2]
8	2. What is your current registered class designation? Freshman [1] Sophomore [2] Junior [3] Senior [4]
10	3. What is your major area of study? Accountancy [1] Economics or Finance [2] Marketing/Management [3] Statistics/Computers [4] Other [5]
12 13 14	4. What is your college grade-point index? ___ . ___
16 17	5. What is your age (as of your last birthday)? _____
19 20 21 22	6. What would you expect your starting annual salary to be if you were to seek employment immediately after obtaining your baccalaureate? $ _____
24	7. Upon graduation, which one of the following types of organizations would you most prefer to work? Large corporation [1] Small privately owned company [2] Nonprofit or government agency [3]
26	8. What is your current employment status? Full-time [1] Part-time [2] Unemployed [3]
28 29 30	9. If you were to purchase a birthday gift for a good friend or relative, up to how much money would you consider spending? $_____
32	10. If you were given a gift of $10,000, how would you primarily consider using it this year? Place in savings bank [1] Invest in "money market" fund [2] Invest in stocks [3] Invest in bonds [4] Spend for goods and services [5]
34	11. How satisfied are you with the type of items carried at your college bookstore? (circle choice) Extremely unsatisfied 1 2 3 4 5 6 7 Extremely satisfied
36 37	12. To how many magazines do you currently subscribe? _____
39	13. Do you currently own a 35mm single-reflex-lens camera? Yes [1] No [2]
41	14. Do you currently own a stereo system? Yes [1] No [2]

Figure 1.5 Questionnaire.