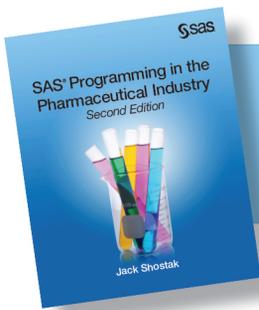


SAS[®] Programming in the Pharmaceutical Industry

Second Edition



Jack Shostak



From SAS[®] *Programming in the Pharmaceutical Industry, Second Edition*. Full book available for purchase [here](#).

Contents

List of Programs	xi
About This Book	xv
About The Author	xix
Acknowledgements	xxi
Chapter 1 Environment and Guiding Principles	1
The Statistical Programmer’s Working Environment	2
Pharmaceutical Industry Vocabulary	2
Statistical Programmer Work Description	2
The Drug/Device Development Process	3
Industry Regulations and Standards	4
Your Clinical Trial Colleagues	8
Guiding Principles for the Statistical Programmer	10
Understand the Clinical Study	10
Program a Task Once and Reuse Your Code Everywhere	11
Clinical Trial Data Are Dirty	13
Use SAS Macros Judiciously	15
A Good Programmer Is a Good Student	17
Strive to Make Your Programming Readable	17
Chapter 2 Preparing and Classifying Clinical Trial Data	19
Preparing Clinical Trial Data	20
“Clean” the Data If They Are Needed for Analysis	20
Categorize Data If Necessary	21
Avoid Hardcoding Data	24
Classifying Clinical Trial Data	26
Demographics and Trial-Specific Baseline Data	27

Concomitant or Prior Medication Data	27
Medical History Data	28
Investigational Therapy Drug Log	29
Laboratory Data	30
Adverse Event Data	31
Endpoint/Event Assessment Data	34
Clinical Endpoint Committee (CEC) Data	35
Study Termination Data	36
Treatment Randomization Data	36
Quality-of-Life Data	38
Chapter 3 Importing Data	39
Importing Relational Databases and Clinical Data Management Systems	40
SAS/ACCESS SQL Pass-Through Facility	40
SAS/ACCESS LIBNAME Statement	41
Importing ASCII Text	41
PROC IMPORT and the Import Wizard	42
SAS DATA Step	48
SAS Enterprise Guide	49
Importing Microsoft Office Files	52
LIBNAME Statement	53
Import Wizard and PROC IMPORT	55
SAS/ACCESS SQL Pass-Through Facility	58
SAS Enterprise Guide	59
Importing XML	62
XML LIBNAME Engine	63
SAS XML Mapper	67
Importing CDISC Model Content Files	68
Importing CDISC SAS Transport Format Files	69
Importing define.xml	69
Importing CDISC ODM Files	70
Chapter 4 Transforming Data and Creating Analysis Data Sets	71
Key Concepts for Creating Analysis Data Sets	72
Defining Variables Once	72
Defining Study Populations	72

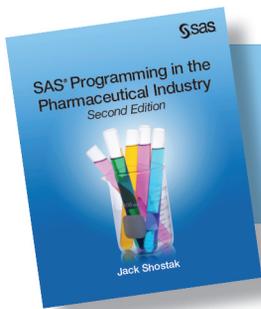
Defining Baseline Observations	73
Last Observation Carried Forward (LOCF)	73
Defining Study Day	78
Windowing Data.....	78
Transposing Data	82
Categorical Data and Why Zero and Missing Results Differ Greatly	90
Performing Many-to-Many Comparisons/Joins	93
Using Medical Dictionaries.....	95
Other Tricks and Traps in Data Manipulation.....	99
Common Analysis Data Sets.....	105
Subject Level Analysis Data Set.....	105
Change-from-Baseline Data Set	105
Time-to-Event Data Set.....	108
Chapter 5 Creating Tables and Listings	113
Creating Tables	114
General Approach to Creating Tables	114
A Typical Clinical Trial Table	114
Using PROC TABULATE to Create Clinical Trial Tables.....	116
Using PROC REPORT to Create Clinical Trial Tables.....	118
Creating Typical Continuous/Categorical Summary Tables.....	122
Creating Adverse Event Summaries	130
Creating Concomitant or Prior Medication Tables	140
Creating a Laboratory Shift Table.....	145
Creating Kaplan-Meier Survival Estimates Tables.....	152
Creating Listings	159
Output Appearance Options and Issues.....	164
Creating ASCII Text Output	164
Creating Rich Text Format (RTF) Output.....	165
Creating Portable Document Format (PDF) Files.....	167
“Page X of N” Pagination Solutions.....	168
Footnote Indicating SAS Program and Date.....	170
ODS Report Writing Interface.....	170
The Power of ODS STYLE.....	170
SAS Macro-Based Reporting Systems	172
Chapter 6 Creating Clinical Trial Graphs.....	173

Common Clinical Trial Graphs	174
Scatter Plot.....	174
Line Plot.....	174
Bar Chart	175
Box Plot.....	176
Forest Plot	176
Kaplan-Meier Survival Estimates Plot	177
SAS Tools for Creating Clinical Trial Graphs	178
Sample Graphs	179
Creating a Scatter Plot.....	179
Creating a Line Plot	182
Creating a Bar Chart.....	185
Creating a Box Plot.....	189
Creating a Forest Plot	193
Creating a Kaplan-Meier Survival Estimates Plot	198
Using SAS Graphics Assistants	212
Graph-N-Go	212
SAS Enterprise Guide	213
ODS Graphics Designer	213
ODS Graphics Editor	213
When You Should Use SAS Graphics.....	214
Chapter 7 Performing Common Analyses and Obtaining Statistics	215
Obtaining Descriptive Statistics	216
Using PROC FREQ to Export Descriptive Statistics	216
Using PROC UNIVARIATE to Export Descriptive Statistics	217
Obtaining Inferential Statistics from Categorical Data Analysis.....	218
Performing a 2x2 Test for Association	218
Performing an NxP Test for Association	219
Performing a Stratified NxP Test for Association	220
Performing Logistic Regression	221
Obtaining Inferential Statistics from Continuous Data Analysis	221
Performing a One-Sample Test of the Mean	221
Performing a Two-Sample Test of the Means	223
Performing an N-Sample Test of the Means	224
Obtaining Time-to-Event Analysis Statistics.....	225

Obtaining Correlation Coefficients.....	226
General Approach to Obtaining Statistics	226
Chapter 8 Exporting Data	229
Exporting Data to the FDA	229
Using the SAS XPORT Transport Format.....	230
Creating ODM XML and define.xml	231
Exporting Data Not Destined for the FDA.....	232
Exporting Data with PROC CPORT.....	232
Exporting ASCII Text	233
Exporting Data to Microsoft Office Files	240
Exporting Other Proprietary Data Formats	243
Encryption and File Transport Options	244
Chapter 9 The Future of SAS Programming in Clinical Trials.....	245
Changes in the Business Environment	245
Changes in Technology	246
Changes in Regulations.....	246
Changes in Standards	247
Use of SAS Software in the Clinical Trial Industry.....	247
Chapter 10 Further Resources.....	249
Regulatory Resources	250
SAS Programming Validation	250
FDA Resources	250
Standards and Industry Organizations	251
SAS Help.....	252
Google Search.....	252
lexjansen.com	252
SAS-L	252
SAS Technical Support	252
SAS Users Groups	253
SAS Manuals and Online Documentation	253
SAS Press	253
SAS Focus Areas	253
Third-Party SAS Web Pages	254
Useful Technical Skills.....	254

Scripting.....	254
Version Control Software.....	254
VBScript/JavaScript for Applications	254
Systems Development Methodology.....	254
Modeling Tools.....	255
Markup Languages.....	255
File Transport and Data Encryption Technologies.....	255
Other Applications Development Languages	255
Qualifying for and Obtaining a Job.....	256
Glossary	257
Index	273

From *SAS® Programming in the Pharmaceutical Industry, Second Edition* by Jack Shostak.
Copyright © 2014, SAS Institute Inc., Cary, North Carolina, USA. ALL RIGHTS RESERVED.



From SAS® *Programming in the Pharmaceutical Industry, Second Edition*. Full book available for purchase [here](#).

Chapter 2 Preparing and Classifying Clinical Trial Data

Preparing Clinical Trial Data	20
“Clean” the Data If They Are Needed for Analysis	20
Categorize Data If Necessary	21
Avoid Hardcoding Data	24
Classifying Clinical Trial Data	26
Demographics and Trial-Specific Baseline Data	27
Concomitant or Prior Medication Data	27
Medical History Data	28
Investigational Therapy Drug Log	29
Laboratory Data	30
Adverse Event Data	31
Endpoint/Event Assessment Data	34
Clinical Endpoint Committee (CEC) Data	35
Study Termination Data	36
Treatment Randomization Data	36
Quality-of-Life Data	38

This chapter describes the key clinical data preparation issues and the different classes of clinical data that are found in clinical trials. Each class of data brings with it a different set of challenges and special handling issues. Sample case report form (CRF) pages are provided. These pages are loosely based on the Clinical Data Interchange Standards Consortium’s (CDISC) Clinical Data Acquisition Standards Harmonization (CDASH) data collection standard. They are provided with each type of data to aid you in visualizing what the data in the CDISC Study Data Tabulation Model (SDTM) standard would look like. The key data preparation issues presented are concepts that apply universally across the various classes of clinical trial data.

Preparing Clinical Trial Data

Clinical trial data come to the statistical programmer in two basic forms: numeric variables and character string (text) variables. With this in mind, there are two considerations for all numeric and text variables. All data should be cleaned if they are needed for analyses, and any data entered as *free-text variables* should be coded or categorized if they are needed for analyses. Generally speaking, it is much more preferable if the data is coded either inherently by data collection design or later by clinical data management before it ever is sent to a statistical programmer.

“Clean” the Data If They Are Needed for Analysis

If data will be summarized or analyzed as part of the protocol-defined statistical analysis, they should be cleaned first. “Cleaned” in this context means that erroneous data that have been entered into a variable are repaired before data analysis. Under the direction of the statistics group and based on the needs of the statistical analysis plan, the data management group is responsible for cleaning the clinical data.

Before the statistical programmer receives data that are ready for analysis, the clinical data management group cleans the data. This is done through a query process, which is built into the clinical data management system. The clinical data management query process usually looks like this:

1. A programmatic or manual investigation of the data finds an errant data point.
2. A “query” or data clarification form (DCF) for that data point is sent to the clinical site.
3. The clinical site responds to the query. If the data is collected via an electronic data capture system, the site may fix the data issue.
4. If the clinical site does not fix the data issue themselves, then the clinical data management group updates the database or CRF based on the response from the clinical site.

Depending on the size and complexity of the clinical trial, queries sent to sites can easily number in the thousands. Because the cost of reconciling these queries quickly rises, it is important to be judicious when creating them. It is worth noting that electronic data capture (EDC) systems may reduce the number of queries needed, because the entry screens are often programmed so that errant data cannot be entered. It is also worth noting that if the clinical data is placed into the CDISC SDTM format, there can be a large number of automatic data queries generated because standard queries and cross data type queries are easy to generate from the SDTM data model.

In order to reduce unnecessary data queries, the statistics group should be consulted early in the clinical database development process to identify variables that are critical for data analysis. Optimally, the statistical analysis plan would already be written by the time of database development so that the queries could be designed based on the critical variables indicated in the analysis plan. However, at the database development stage, usually only the clinical protocol exists to guide the statistics and clinical data management departments in developing the query or data management plan.

How clean the data must be depends on the importance of the data. Critical analysis variables must be clean, so this is where the site and data management groups should focus their resources. If the data are “dirty” at the time of statistical analysis, many inefficient and costly workarounds may need to be applied in the statistical programming, and the quality of the data analysis could suffer. However, if a variable is not important to the statistical analysis, then it is better to save the expense of cleaning that variable.

Categorize Data If Necessary

Clinical trial data come in two basic forms: numeric variables and text variables. Numeric variables are easy for the statistical programmer to handle. Numbers can be analyzed with SAS in a continuous or categorical fashion without much effort. If a numeric variable needs categorization, it is easy enough to categorize the data within SAS. For example, if you had to classify patient age, a simple DATA step such as the following might serve well.

Program 2.1 Categorizing Numeric Data

```
data adsl;
  set adsl;

  if . < age <= 18 then
    agegrln = 1;
  else if 18 < age <= 60 then
    agegrln = 2;
  else if 60 < age then
    agegrln = 3;
run;
```

The problem for the statistical programmer in categorizing data comes from text variables or, more specifically, free-text variables. A “free-text” variable is one that may contain any characters and is typically limited only in length. As an example, let’s say you need to summarize the adverse events for a set of patients in a trial. The following SAS code shows the data and a quick summarization of the adverse events.

Program 2.2 Summarizing Free-Text Adverse Event Data

```
data AE;
  input USUBJID $ 1-7 AETERM $ 9-41;
  datalines;
  100-101 HEDACHE
  100-105 HEADACHE
  100-110 MYOCARDIAL INFARCTION
  200-004 MI
  300-023 BROKEN LEG
  400-010 HIVES
  500-001 LIGHTHEADEDNESS/FACIAL LACERATION
  ;
run;
```

22 SAS Programming in the Pharmaceutical Industry, Second Edition

```
options nodate nonumber missing = ' ';
ods escapechar='#';
ods pdf style=htmlblue file='program2.2.pdf';

proc freq
  data = ae;
  tables aeterm;
run;

ods pdf close;
```

Program 2.2 yields the following output.

The SAS System				
The FREQ Procedure				
AETERM	Frequency	Percent	Cumulative Frequency	Cumulative Percent
01 HEDACHE	1	14.29	1	14.29
01 LIGHTHEADEDNESS/FACIAL LACERAT	1	14.29	2	28.57
04 MI	1	14.29	3	42.86
05 HEADACHE	1	14.29	4	57.14
10 HIVES	1	14.29	5	71.43
10 MYOCARDIAL INFARCTION	1	14.29	6	85.71
23 BROKEN LEG	1	14.29	7	100.00

There are three problems with this adverse events summary. First, “HEADACHE” and “HEDACHE” are counted as separate events even though it is clear that the latter is simply a misspelling of the former. Second, “MI” and “MYOCARDIAL INFARCTION” are considered as separate events even though the former is simply an abbreviation of the latter. Finally, “LIGHTHEADEDNESS/FACIAL LACERATION” refers to perhaps related but different adverse events that need to be counted separately. All three of these problems exist because the data were entered in free-text fashion and summarized from the free-text variable AETERM.

There is only one good solution to handling free-text variables that are needed for statistical analysis. The free-text variables need to be coded by clinical data management in the clinical database. If the adverse events were coded with a dictionary, such as *MedDRA*, which will be explored further in Chapter 4, the previous example might look like Program 2.3.

Program 2.3 Summarizing Coded Adverse Event Data

```

data ae;
label USUBJID = "Unique Subject Identifier"
      AEPTCD = "Preferred Term Code"
      AETERM = "Reported Term for the Adverse Event"
      AEDECOD = "Dictionary-Derived Term";

input USUBJID $ 1-7 AEPTCD $ 9-16
      AETERM $ 18-38 AEDECOD $ 40-60;

datalines;
100-101 10019211 HEDACHE HEADACHE
100-105 10019211 HEADACHE HEADACHE
100-110 10028596 MYOCARDIAL INFARCTION MYOCARDIAL INFARCTION
200-004 10028596 MI MYOCARDIAL INFARCTION
300-023 10061599 BROKEN LEG LOWER LIMB FRACTURE
400-010 10046735 HIVES URTICARIA
500-001 10013573 LIGHTHEADEDNESS DIZZINESS
500-001 10058818 FACIAL LACERATION SKIN LACERATION
;
run;

options nodate nonumber missing = ' ';
ods escapechar='#';
ods pdf style=htmlblue file='program2.3.pdf';

proc freq
  data = ae;
  tables aeterm_aedecod;
run;

ods pdf close;

```

Program 2.3 yields the following output.

The SAS System				
The FREQ Procedure				
Dictionary-Derived Term				
AEDECOD	Frequency	Percent	Cumulative Frequency	Cumulative Percent
DIZZINESS	1	12.50	1	12.50
HEADACHE	2	25.00	3	37.50
LOWER LIMB FRACTURE	1	12.50	4	50.00
MYOCARDIAL INFARCTION	2	25.00	6	75.00
SKIN LACERATION	1	12.50	7	87.50
URTICARIA	1	12.50	8	100.00

You can see the benefit of coding the adverse events in the resulting summary. The headaches and myocardial infarctions are grouped appropriately, and splitting lightheadedness and facial laceration into separate events leads to those data being summarized separately as well.

However, there are some alternative, albeit poor, solutions to the free-text variable problem. One option is to *hardcode* events so that they are categorized properly. We will discuss hardcoding further in the next section, but it is generally a practice to be avoided as much as possible. Another option is to use a SAS DATA step string function such as SOUNDSEX, INDEX, INDEXW, or SUBSTR to try to categorize data in like groups. This approach is very risky, because you cannot be guaranteed to capture all free-text data and categorize them the same way with these text-scanning tools. If the free-text data are unimportant, then such tools can be used. However, if the data are unimportant, then they probably should not be analyzed anyway and at best should be presented in some type of data listing.

Avoid Hardcoding Data

Sometimes even after clinical data management make a good attempt at cleaning and coding the data, you may find that the data still contain some undesired or discrepant values. Perhaps a variable was left uncoded, or perhaps there is a serious adverse event known to have occurred that has not yet been entered in the clinical database. When this happens, the statistical programmer may result to hardcoding. Hardcoding is explicitly stating the value of a symbolic object or variable in a program. An example of hardcoding follows.

Program 2.4 A Hardcoding Example

```

data endstudy;
  set endstudy;

  if subjid = "101-1002" then
    discterm = "Death";
run;

```

In this example, it is known from non-database sources that at study termination, subject 101-1002 died. That information is hardcoded into the program and overrides the information coming from the clinical data management system. Here are two reasons why hardcoding is a bad practice:

- Hardcoding overrides the database controls in a clinical data management system. With hardcoding, there is no clear audit trail of data change, and CFR 21 – Part 11 controls might be considered compromised.
- Data often change in a trial over time, and the hardcode that is written today may not be valid in the future. Unfortunately, a hardcode may be forgotten and left in the SAS program, and that can lead to an incorrect database change.

Many organizations expressly forbid hardcoding in their SAS programming standard operating procedures, while others allow the practice. Occasionally, there may be a justifiable reason for hardcoding. For instance, there may be an upcoming *data safety and monitoring board (DSMB)* or *independent data monitoring committee (IDMC)* meeting where the clinical trial must be monitored for safety information using the best available data. If there is a critical adverse event that the statistical staff is aware of but it cannot be entered in the clinical data management system in time, then perhaps that would justify hardcoding. However, it is better to avoid hardcoding at all costs and instead correct data in the clinical data management system. If hardcoding must be done, then an approach like the following might be used.

Program 2.5 An Improved Hardcoding Example

```

data endstudy;
  set endstudy;

  **** HARDCODE APPROVED BY DR. NAME AT SPONSOR ON 02/02/2012;
  if subjid = "101-1002" and "&sysdate" <= "01MAY2012"d then
  do;
    discterm = "Death";
    put "Subject " subjid "hardcoded to termination reason"
      discterm;
  end;
run;

```

Note that this program uses SAS code comment text to indicate that hardcoding is being used and with whose approval. Requiring a keyword such as “HARDCODE” in the comment facilitates searches for hardcodes later. Also, note that a PUT statement is provided to the SAS log, verifying during program execution that hardcoding has been used. The hardcode in Program 2.5 has an

expiration date. For example, if you know that you have an upcoming IDMC date next year, you can program the hardcodes to expire in the month that precedes the IDMC meeting.

In summary, for data to be useful in clinical trial analyses, they need to be quantifiable. The data must be either a continuous measure or a categorical value. Free text poses a problem for analysis, and, if it is a valuable variable for the statistical analyses, it really must be coded. Finally, hardcoding should be used only when absolutely necessary, because it is inherently problematic. Organizations that do allow hardcoding should document in their standard operating procedures (SOPs) that it is an approved business practice and how it is to be used.

Classifying Clinical Trial Data

There are different ways to classify clinical trial data. As mentioned earlier, data can be classified by their physical nature into discrete chunks or as a more continuous measurable quantity. In clinical trials, there are other important contextual ways of grouping data as well. For instance, clinical trials are primarily focused on determining two things about a drug, biologic, or device: Is it efficacious, and is it safe? The data that help to answer these questions are broadly classified as *efficacy data* and *safety data*, respectively.

The Clinical Data Interchange Standards Consortium (CDISC) and its Submission Data Standards group have provided another way to broadly categorize clinical trial data. They have categorized data into *interventions class*, *events class*, *findings class*, and other special-purpose “*domains*” such as demographics. Interventions are the drug administration and surgical procedures that the patient receives during the course of the trial. Events are the unplanned clinical occurrences that the patient experiences over the course of the trial. Findings capture the planned examinations of the patient over the course of the trial. The demographics of a patient are that person’s essential baseline characteristics.

The following sample CRF forms have been made to align with the CDISC CDASH standard.

Demographics and Trial-Specific Baseline Data

Here is a typical demographics CRF:

Protocol Name _____	Subject: ____ - ____	Subject Initials: ____
DEMOGRAPHICS:		
Birth Date: ____ / ____ / ____ (Day/Month/Year)		
Sex: <input type="checkbox"/> Male <input type="checkbox"/> Female		
Race: <input type="checkbox"/> Caucasian <input type="checkbox"/> Black <input type="checkbox"/> Asian <input type="checkbox"/> Other		
Ethnicity: <input type="checkbox"/> Hispanic <input type="checkbox"/> Non-Hispanic		

Trial-specific patient characteristics may be included with the demographics data as well. Height, weight, smoking status, and sometimes vital signs are common additions. These measures are collected because they may be relevant to the therapeutic intervention and could be used to stratify the statistical analysis. Demographic and other baseline characteristics are used to define patient groupings, or *strata*, for subpopulation analyses, or they may be used as *covariates* during *inferential analyses*. Demographic and baseline characteristics are also commonly used to show that the therapeutic treatments under study have comparable populations at baseline. Demographics data fall into the special purpose demographics SDTM domain and play a part in efficacy and safety analyses, because either may be stratified by demographics and baseline characteristics. Other baseline subject characteristics would get stored in the subject characteristics SDTM domain or in the appropriate SDTM domain (e.g., blood pressure in vital signs).

Concomitant or Prior Medication Data

Concomitant medications and *prior medications* are collected in one of two forms: a list-type free-text format where the medications get coded later by data management, or a pre-categorized data format. Here is the free-text CRF format:

Protocol Name _____	Subject: ____ - ____	Visit _____			
Concomitant Medications:					
	Medication or Therapy	Dose	Start Date	End Date	Indication
1	_____	_____	__/__/__	__/__/__	_____
2	_____	_____	__/__/__	__/__/__	_____
3	_____	_____	__/__/__	__/__/__	_____
4	_____	_____	__/__/__	__/__/__	_____
	etc.				

Here is the pre-categorized per protocol CRF format:

Protocol Name		Subject: ___ - ___		Visit	
Concomitant Medications:					
Medication or Therapy	Did the subject take?		Start Date	End Date	Indication
	Yes	No			
ACE Inhibitor	—	—	__/__/__	__/__/__	_____
Anticonvulsant	—	—	__/__/__	__/__/__	_____
Beta Blocker	—	—	__/__/__	__/__/__	_____
Psychoactive Medication	—	—	__/__/__	__/__/__	_____
etc.					

The free-text CRF format is useful in that it allows for an explicit description of the medication taken, whereas the pre-categorized format omits that detail. However, the free-text list format necessitates additional coding with a coding dictionary such as *WHOdrug* in order to be useful for analyses. The pre-categorized format has the benefit of capturing only the medications of concern for the given protocol and therapy and eliminates the cost of additional coding.

An essential detail for the statistical programmer to watch for in prior or concomitant medications data is whether or not the start and end dates are important for analyses. Unfortunately, it is often the case that the importance of the timing of prior or concomitant medications is not determined until after much of the data have been entered or even after the database is closed to entry. For instance, it may be decided later that a specific concomitant medication has to be watched carefully for interaction with a medication used in the study. If insufficient attention was placed on the quality of the medication start and end dates, then determining whether there is overlap with study medication is difficult if not impossible.

Concomitant or prior medications may be used in either safety or efficacy analyses. The presence of specific medications may be used as covariates for inferential analyses. Also, medications are often summarized to show that the therapies under study come from medically comparable populations. Medications may be used to determine protocol compliance and to help define a protocol-compliant study population. Concomitant medications may be examined to determine whether they interact with study therapy or whether they can explain the presence of certain adverse events. From a CDISC SDTM perspective, concomitant medications are considered an intervention.

Medical History Data

Like concomitant medication data, patient *medical history* data are collected in one of two forms: a list-type free-text format where the histories get coded, or a pre-categorized data format. Here is the free-text CRF format:

Protocol Name		Subject: ___ - ___	Visit
Medical History:			
	Medical History Term	Start Date	
1	_____	_/_/_	
2	_____	_/_/_	
3	_____	_/_/_	
4	_____	_/_/_	
	etc.		

Here is the pre-categorized medical history CRF format:

Protocol Name		Subject: ___ - ___	Visit
Medical History:			
Medical History Term	Does the Subject Have?		Start Date
	Yes	No	
Diabetes			
Stroke			
Hypertension			
Neurological Disorders			
etc.			

Again, the free-text CRF format is useful in that it allows for explicit description of the historical condition, whereas the pre-categorized CRF format omits that detail. However, the free-text list format necessitates coding with a coding dictionary such as MedDRA in order to be useful for analyses. The pre-categorized format is useful here, because only medical history relevant to the investigational therapy can be captured and the cost of additional coding of the history data is eliminated entirely.

Medical history data may be used in either safety or efficacy analyses. The presence of historical medical conditions may be used as covariates for inferential analyses. Also, medical histories are typically summarized to show that the therapies under study come from study populations with comparable disease histories. Medical histories may be used to determine protocol compliance and to help define a protocol-compliant study population. Medical history is considered a finding from a CDISC SDTM perspective.

Investigational Therapy Drug Log

Drug logs, or drug exposure data, capture the investigational drug dosing times. Here is a sample drug log CRF form:

Protocol Name _____		Subject: ___-___-___	
Study Drug Dosing:			
Dose #	Start Date	Start Time (24-hour clock)	Dose (mg)
1	__/__/__	__:__	_____
2	__/__/__	__:__	_____
3	__/__/__	__:__	_____
4	__/__/__	__:__	_____
etc.			

The investigational therapy drug log can be a source of problems for the statistical programmer. Here again, dates and times of dosing may be critical for effective use of this data. Missing dosing records, start times, or stop times can seriously hinder the quality of the reporting of dosing data. It is important to look at the analysis plan to determine if the dosing data are important to analysis. If they are important, then data management should clean the data to ensure the quality of the medication start and stop times.

Drug log or exposure data are used in many ways for both efficacy and safety analyses. As a safety issue, the drug record is often used in conjunction with adverse events to determine whether adverse events were treatment-emergent. In other words, did the patient have an adverse event that might have been caused by the investigational therapy? Also, drug log data may be used for safety analysis purposes to watch for abnormal laboratory values or other clinical events after dosing. Finally, drug log data are useful for determining protocol violations and can be used to determine treatment compliance. The drug log data are categorized as an intervention from a CDISC SDTM perspective.

Associated with drug log or drug exposure data is another type of data called drug accountability. This data captures the disposition of the study drug. It is not concerned with whether a patient was exposed to the drug but where the drug went. Drug accountability tracks data such as how many pills a patient was sent home with and how many they returned. It can be used to calculate protocol dosing compliance and is categorized as a finding from a CDISC SDTM perspective. Because the data is so interrelated, it is not uncommon to find data collection forms merge or integrate information from drug exposure and drug accountability.

Laboratory Data

Laboratory data may consist of many different collections of tests, such as ECG laboratory tests, microbiologic laboratory tests, and other therapeutic-indication-specific clinical lab tests. However, laboratory data traditionally consist of results from urinalysis, hematology, and blood chemistry tests. Traditional laboratory data can come from what are called local laboratories, which are labs at the clinical site, or from central laboratories where the clinical sites send their samples for centralized analysis. Often when the laboratory data come from a central laboratory, there is no CRF page for the data, and they are loaded into the clinical data management system directly from an electronic file. Local laboratory data may be represented with a CRF page such as this:

Protocol Name		Subject: ___ - ___		Visit
Laboratory Data:				
Hematology				
Test Name	Collection Date	Collection Time (24-hour clock)	Result	Units
Platelets	__/__/__	__:__	_____	_____
Hemoglobin	__/__/__	__:__	_____	_____
Hematocrit	__/__/__	__:__	_____	_____
etc.				
Chemistry				
Test Name	Collection Date	Collection Time (24-hour clock)	Result	Units
Serum Creatinine	__/__/__	__:__	_____	_____
Total Cholesterol	__/__/__	__:__	_____	_____
Basophils	__/__/__	__:__	_____	_____
etc.				

Laboratory data can pose a challenge to the statistical programmer in many ways. Simply obtaining the data can sometimes be difficult. Occasionally you have to work with a specialized local laboratory, and sometimes just getting the data to the statistics group in a usable format can be hard if CDISC CDASH and SDTM standards are not used. For example, the local laboratory staff may have used Microsoft Excel for data entry, and when they entered the data they entered rows within the columnar data with inconsistent formats, making machine readability of the resulting data file difficult. Another common issue is found within the “units” variable shown above. If local labs were used, it is likely that the lab units will have to be converted to a common unit for each laboratory test. Finally, laboratory values often need to be flagged as outside the normal range or perhaps outside the “clinical concern”/“panic range,” where the latter is just a more extreme version of the former. Sometimes, the local or central laboratory flags these records, but it is not uncommon for the statistical programmer to have to make these assignments as well.

Laboratory data are most often associated with safety analyses, but they may play a part in efficacy analyses as well, especially if the laboratory data are part of the clinical endpoint definition. From a CDISC SDTM perspective, laboratory data are a finding, because they are a planned assessment. The CDISC SDTM has a number of specialized laboratory-like data domains besides LB for laboratory data. These domains that are very laboratory-like include EG for ECG data, VS for vital signs data, MB for microbiology, and PC and PP for pharmacokinetic data.

Adverse Event Data

In the FDA’s “Guidance for Industry E6 Good Clinical Practice: Consolidated Guidance,” an adverse event is defined as follows:

Any untoward medical occurrence in a patient or clinical investigation subject administered a pharmaceutical product and that does not necessarily have a causal relationship with this treatment. An AE can therefore be any unfavorable and unintended sign (including an abnormal laboratory finding), symptom, or disease temporally associated with the use of a medicinal (investigational) product, whether or not related to the medicinal (investigational) product.

The adverse event form is fairly standard across clinical trials. The form consists of a list of events for which data are entered as free text and are later coded with a dictionary such as MedDRA and some associated event attribute variables. In just about any clinical trial, an adverse event form similar to the following sample will be found.

Protocol Name		Subject: ___ - ___ - ___						
Adverse Events:								
	Adverse Event	Start Date	End Date	Ongoing?	Severity	Action Taken with Study Treatment	Relationship to Study Treatment	Serious?
1		___/___/___	___/___/___	—	<input type="checkbox"/> Mild <input type="checkbox"/> Moderate <input type="checkbox"/> Severe	<input type="checkbox"/> Dose not changed <input type="checkbox"/> Dose reduced <input type="checkbox"/> Drug interrupted <input type="checkbox"/> Drug withdrawn	<input type="checkbox"/> Yes <input type="checkbox"/> No	—
2		___/___/___	___/___/___	—	<input type="checkbox"/> Mild <input type="checkbox"/> Moderate <input type="checkbox"/> Severe	<input type="checkbox"/> Dose not changed <input type="checkbox"/> Dose reduced <input type="checkbox"/> Drug interrupted <input type="checkbox"/> Drug withdrawn	<input type="checkbox"/> Yes <input type="checkbox"/> No	—
3		___/___/___	___/___/___	—	<input type="checkbox"/> Mild <input type="checkbox"/> Moderate <input type="checkbox"/> Severe	<input type="checkbox"/> Dose not changed <input type="checkbox"/> Dose reduced <input type="checkbox"/> Drug interrupted <input type="checkbox"/> Drug withdrawn	<input type="checkbox"/> Yes <input type="checkbox"/> No	—
	etc.							

The adverse event form is a cornerstone of patient safety monitoring, and as such it contains very important data. There are several data issues for the statistical programmer to be concerned about here.

Treatment-Emergent Signs and Symptoms

In guidance document ICH E3, “Structure and Content of Clinical Study Reports,” the FDA defines *treatment-emergent signs and symptoms (TESS)* as “events not seen at baseline and events that worsened even if present at baseline.” As simple as that may sound, it can sometimes be quite difficult to implement in programming. The important data variables that come into play are dosing record dates and times, adverse event start and stop times, and adverse event severity. All of these data variables need to be completed accurately for TESS to be calculated properly.

Serious Adverse Event Reconciliation

Just as there is an adverse event form, there is usually a *serious adverse event (SAE)* form. Note here that “serious” as defined by the FDA is different from “severe” on the adverse event form. A patient can have a “severe” headache that may not be considered “serious.” The ICH guideline (also in ICH E3) entitled “Clinical Safety Data Management: Definitions and Standards for Expedited Reporting” defines serious adverse events as follows:

A serious adverse event (experience) or reaction is any untoward medical occurrence that at any dose: results in death, is life-threatening, requires inpatient hospitalization or prolongation of existing hospitalization, results in persistent or significant disability/incapacity, or is a congenital anomaly/birth defect.

Historically, a separate CRF is used to capture serious adverse events, because those often must be reported to the FDA within 24 hours. Often, this means that the serious adverse events CRF data and the regular trial CRF adverse events are collected in different data tables, if not entirely different software systems. Pharmaceutical companies often want to reconcile the two databases to ensure that all serious adverse events appear in the regular-trial CRF adverse events database and that any event in the serious adverse events database is flagged properly as serious in the regular CRF adverse events database.

The problem is that the regular-trial adverse events database and the serious adverse events database do not join well if at all programmatically. You can attempt to join or merge the two databases by event start date and coded term, and that will join many regular-trial adverse events to the serious events. However, this is far from foolproof, because of mismatches in adverse event start dates and because the adverse events may have been coded slightly differently in the two systems. The best way to link the serious adverse events and adverse events databases is to have the clinical data management system create a linking variable key for you. In lieu of that, the only way to reliably link the two data sources is manually.

The good news is that with modern electronic data capture systems and the upcoming absorption of electronic health care data into clinical trials databases, the problem of reconciling adverse events to serious adverse event data will be fixed. Many electronic data capture systems now collect the serious and regular adverse event data in the same electronic form, which makes integration of the data unnecessary.

Concomitant Medication Reconciliation

Additional concomitant medication may be given in response to an adverse event, and especially with serious adverse events. Often you want to know precisely which medication was taken, but because that information may not be well captured on the adverse event form, there needs to be a linkage with the concomitant medications form. Once again, this is not something that can reliably be done with a program unless the clinical data management system creates a linking variable key behind the adverse event and concomitant medications forms. Some data management systems do this and, again, with electronic data capture, this is becoming more prevalent.

Laboratory Data Reconciliation

The adverse event for a patient may indicate a medical condition such as hypercholesterimia, so there may be a request to ensure that there are elevated cholesterol laboratory data that can verify such a claim. You can sometimes make this kind of verification with programming if you know precisely which lab tests are involved and what level indicates a probable adverse event.

In the end, because of the importance of the data, it is imperative that the entire adverse event form data are cleaned. Reconciling the adverse event data with other clinical data in the clinical data management system can be very difficult if the data management system does not provide variable keys for linking such data. Adverse event data fall into the safety area of statistical analyses and are considered an event from a CDISC SDTM perspective.

Endpoint/Event Assessment Data

Endpoint or event assessments typically capture what the clinical trial was designed to study. For example, if a clinical trial were studying an anti-epilepsy medication, then the event form would likely collect seizure information. The endpoint or event assessment form is designed to collect data after the investigational drug or device intervention so that these data can be statistically compared to data from the patient's state before the drug or device intervention. Endpoint or event collection pages vary widely because of the broad range of ways to measure clinical disease, but here is a simplified sample endpoint collection page:

Protocol Name	Subject: ___ - ___	Visit
Endpoint Assessment:		
Visit Date: __/__/____ (Day/Month/Year)		
Did the patient have an event of interest? <input type="checkbox"/> Yes <input type="checkbox"/> No		
If yes, what day did the event occur on? __/__/____ (Day/Month/Year)		

In this form, “event” would be replaced by some clinical finding such as “myocardial infarction,” “stroke,” “seizure,” or the like. This example form is extremely simplified, because there are usually a number of associated event qualifying data variables captured as well. The event/endpoint page data must be clean, because it likely captures the primary efficacy data for the clinical trial.

The problem with endpoint data usually occurs when they need to be reconciled against data that are collected by the *clinical endpoint committee (CEC)*, which we discuss next. The endpoint/event data are almost always used for efficacy analyses but may be used for safety analyses as well. From a CDISC SDTM perspective, the endpoint/assessment is often considered a finding, because it is a planned examination, but it could also be considered an unplanned event.

Clinical Endpoint Committee (CEC) Data

It is often the case that the endpoint/event form captures data that are not entirely objective because they contain some level of clinical judgment. For instance, when precisely is a cold cured, was an event truly a myocardial infarction, or did any given event truly occur? The clinical site investigator may decide, using his or her clinical judgment, that a given event occurred, but often it is necessary to have an independent assessment of that event by another physician. This independent review helps to ensure that events are reported in a consistent way across multiple clinical sites for a clinical trial. Usually what happens is that a condition on the regular case report form “triggers” the release of a CEC form to be sent to the CEC. The CEC then takes the CEC form and verifies whether or not an actual event occurred based on the data available in the patient’s clinical records at the given site. A sample CEC form follows:

Protocol Name	Subject: ___ - ___	Visit
Endpoint Assessment:		
Did the patient have the event of interest? <input type="checkbox"/> Yes <input type="checkbox"/> No		
If yes, on what day did the event occur? ___ / ___ / ____ (Day/Month/Year)		
Other supportive data fields go here to verify that the event happened.		
Reviewer signature: _____ ___ / ___ / ____ (Day/Month/Year)		

In this CEC form, “event” would be replaced by some clinical finding such as “myocardial infarction,” “stroke,” “seizure,” or the like. Once again, this form is extremely simplified, and there are usually a number of associated data variables captured that help to support the existence of the event.

The biggest problem for the statistical programmer when using CEC data is reconciling these data against the regular CRF endpoint/event data. This can be a difficult task, especially when you consider that a patient may have more than one event on a given day. Fortunately, because the endpoint/event data are so critical to a clinical trial, the quality of the reconciliation from the CEC form to the CRF form is not often relegated to some form of fuzzy data join. Usually there will be a definitive linkage via a key mapping data set that links the CEC event data to the CRF event data. However, if that key data set does not exist, then the statistical programmer must prepare for some difficult programming. It is also worth noting that the data from the adverse event forms, laboratory forms, and other forms, as well as a specific “event” form, may in fact trigger clinical events. This may add to the complexity of the reconciliation programming.

The clinical endpoint committee data are almost always used for efficacy analyses, but they may also be used for safety analyses. From a CDISC SDTM perspective, the endpoint/assessment is considered a finding, as it is a planned examination.

Study Termination Data

The study termination form collects patient exit information from the clinical trial. Here is a sample study termination form:

Protocol Name	Subject: ___ - ___
Study Termination:	
Did the patient complete the study? Yes <input type="checkbox"/> No <input type="checkbox"/>	
If not, please indicate why:	
___ Adverse event	
___ Study medication unsatisfactory	
___ Subject withdrew consent to participate	
___ Protocol violation	
___ Death ___ / ___ / ___ (Day/Month/Year)	
___ Lost to follow-up	
Last day of study medication: ___ / ___ / ___ (Day/Month/Year)	
Investigator signature: ___ / ___ / ___ (Day/Month/Year)	

The study termination form data may be used for efficacy or safety analysis purposes. With regard to safety, if patients discontinue a study medication earlier than patients on standard therapy or placebo, then that is important to know. For efficacy analyses, patients who withdraw due to a lack of efficacy or adverse event may be precluded from being considered a treatment responder or success. Also, often the study termination date is used as a censor date in time-to-event analyses for therapy efficacy. Study termination forms play a key role in patient disposition summaries found at the start of a clinical study report. From a CDISC SDTM perspective, the study termination form is a finding.

Treatment Randomization Data

The *randomization* of a patient to a given therapy is the cornerstone of a randomized clinical trial. You may find these data in more than one place. They are often found within some form of *Interactive Voice Response System (IVRS)*, but they may also be found in an electronic file that contains the treatment assignments or on the CRF itself. If randomization data are found on the CRF, they usually consist only of the date of randomization for treatment-blinded trials. IVRS data are often found outside the confines of the clinical data management system and usually consist of the following three types of data tables.

Randomization Scheme Data Set

The *randomization scheme* assigns a therapy randomly across a study population based on various stratification factors such as site, *blocking factor*, and perhaps subject demographics. There is no actual patient assignment information in this data table. Here is an example of a randomization scheme with a blocking factor size of four and a *treatment ratio* of 2:2:

Index	Site	Block	Treatment
1	101	1	Study Medication
2	101	1	Placebo
3	101	1	Study Medication
4	101	1	Placebo
5	101	2	Placebo
6	101	2	Placebo
7	101	2	Study Medication
8	101	2	Study Medication

Notice that treatment is randomly assigned within the given blocks and that there are two placebos and two study medications in each block. Also notice the “index” variable. The order of the randomization scheme is critical to the usefulness of the scheme, because that is the order in which patients are assigned treatment. If the order of the scheme is altered in any way, then the scheme is damaged.

Drug Kit List Data Set

The *drug kit list* is simply a list that shows which drug container/kit label goes with which study medication. It might look something like this:

Kit Number	Treatment
10000001	Study Medication
10000002	Study Medication
10000003	Study Medication
10000004	Study Medication
10000005	Study Medication

Drug Assignment Data Set

The *drug assignment data set* indicates which patient got which drug. It might look something like this:

Site	Subject	Treatment
101	0001	Study Medication
101	0002	Study Medication
101	0003	Placebo
101	0004	Placebo

Note that the drug assignment data may not exactly match the order in the randomization scheme, because different patients pass screening procedures and are eligible for randomization at different times. Sometimes there are errors in treatment assignment, due to drug kits being misallocated or lost, that lead to a discrepancy between the drug assignment and the randomization scheme.

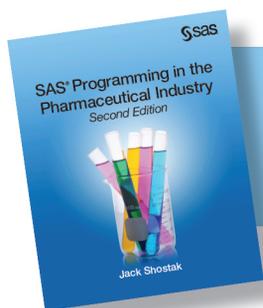
Other data sets may be found within the IVRS system that prove useful to the statistical programmer as well. Often the IVRS collects several baseline patient characteristics that are used in the stratification of the randomization scheme and subsequent assignment of study therapy. Finally, the preceding examples show in detail what the treatment variable is, in the “treatment” column. It is more often the case that the treatment variable is coded, such as “A” or “B” or “C.” It is of paramount importance that you know with absolute certainty how the treatment code can be properly interpreted.

The randomization data are used in both efficacy and safety analyses, because they are typically the key stratification variable for the trial. The randomization data allow you to answer the question of whether patients who are getting the study therapy fare better than the alternative. The CDISC SDTM allocated that actual treatment assignment information to the special demographics domain. The study therapy kit number would go in the CDISC SDTM DA domain.

Quality-of-Life Data

Sometimes you may also see *quality-of-life (QOL) data* collected for your clinical trial. Quality-of-life data are collected to measure the overall physical and mental well-being of a patient. These data are usually collected with a multiple-question patient questionnaire and may be summed up in an aggregate patient score for analysis. Some commonly used quality-of-life questionnaires are the SF-36 and SF-12 Health Survey, but there are quite a few disease-specific QOL questionnaires available to clinical researchers. Quality-of-life data are often a subset of a type of data called patient-reported outcomes. They are patient-reported outcomes, because many times the patient reports them directly into a data collection tool, such as a website, themselves. From a CDISC SDTM standpoint, questionnaire data is classified as a finding.

From *SAS® Programming in the Pharmaceutical Industry, Second Edition* by Jack Shostak.
Copyright © 2014, SAS Institute Inc., Cary, North Carolina, USA. ALL RIGHTS RESERVED.



From SAS® *Programming in the Pharmaceutical Industry, Second Edition*. Full book available for purchase [here](#).

Index

A

Access files

See Microsoft Office files

ACDM (Association for Clinical Data Management) 250, 251

ADaM (Analysis Data Model) 5, 73, 83

Adobe PDF files 167–168

adverse events

 about 31–32

 concomitant medication 33

 creating summaries 22, 130–139

 laboratory data 34

 serious adverse events (SAE) 33

 treatment-emergent signs and symptoms (TESS) 32

American Statistical Association 251

analyses of clinical data

 correlation coefficients 226

 descriptive statistics 216–218

 inferential statistics 218–225

 statistical analysis plan (SAP) 11

 time-to-event analysis statistics 225

Analysis Data Model (ADaM) 5, 73, 83

Analysis Data Model and ADaM Implementation Guide 105, 108

analysis data sets

 about 71–72

 categorical data 90–93

 change-from-baseline 105–108

 common types of 105–111

 creating 72–104

 data manipulation 99–104

 defining baseline observations 73

 defining study day 78

 defining study populations 72–73

 defining variables 72

 last observation carried forward (LOCF) 73–77

 many-to-many comparisons/joins 93–95

 medical dictionaries 95–98

 missing results 90–93

 time-to-event 108–111

 transposing data 82–89

 windowing data 78–82

 zero results 90–93

Analysis of Clinical Trials Using SAS: A Practical Guide (Dmitrienko) 215

analysis of variance (ANOVA) 224

analysis plan 11

Anatomical-Therapeutic-Chemical (ATC) classification 98

angle brackets (<>) 11

annotated CRF 11

ANOVA (analysis of variance) 224

APIs (application program interfaces) 9

approval process (FDA) 3–4

ARRAY statements 84

ASCII text

 about 41–42

 creating output 164–165

 exporting 233–239

 IMPORT procedure 42–47

 Import Wizard 42–47

 SAS DATA step 48–49

 SAS Enterprise Guide 49–52

ascii2pdf script 168

association, tests for 218–221

Association for Clinical Data Management (ACDM) 250, 251

ATC (Anatomical-Therapeutic-Chemical) classification 98

autocall facility 13

AVAL variable 108

B

bar charts 175, 185–189

baseline 10

baseline observations 73

BAT files 254

- bias 4
- binomial variables 218
- biostatistics 8
- blinding 4
- BODYTITLE option 166
- box plots 176, 189–192
- BOXPLOT procedure 178
- Burlew, Michele
 - SAS Macro Programming Made Easy* 16
- business environment 245
- BY processing 16, 84
- BY statement 16, 214

- C**
- C# programming language 255–256
- Carpenter, Art
 - Carpenter's Complete Guide to the SAS Macro Language* 16
 - Carpenter's Complete Guide to the SAS Macro Language* (Carpenter) 16
- case report form (CRF) 8, 19
- categorical data 90–93, 218–221
- Categorical Data Analysis Using the SAS System* (Stokes) 218
- categorizing data 21–24
- CDASH (Clinical Data Acquisition Standards Harmonization) model 5, 19
- CDER Common Data Standards Issues Document 7
- CDER Data Standards Program 8
- CDISC (Clinical Data Interchange Standards Consortium) 5, 19, 26, 68–69, 251
- CDISC ADaM Basic Data Structure for Time-to-Event Analyses* 109
- CDISC procedure 68–69, 70, 231–232
- CEC (clinical endpoint committee) 34–35
- Censor variable 108
- certifications 256
- CFR 21 - Part 11 law 6, 246
- change-from-baseline data set 105–108
- change-from-baseline scatter plots 174
- Class tab (SAS XML Mapper) 68
- classifying clinical trial data 19, 26–38
- cleaning data 20–21
- clinical data
 - classifying 19, 26–38
 - management systems 40–41
 - preparing 19, 20–26
- Clinical Data Acquisition Standards Harmonization (CDASH) model 5, 19
- clinical data analysis
 - correlation coefficients 226
 - descriptive statistics 216–218
 - inferential statistics 218–225
 - statistical analysis plan (SAP) 11
 - time-to-event analysis statistics 225
- Clinical Data Interchange Standards Consortium (CDISC) 5, 19, 26, 68–69, 251
- clinical endpoint committee (CEC) 34–35
- Clinical Standards Toolkit 70, 231–232
- clinical studies 10–11
- clinical study report (CSR) 2
- clinical trial graphs
 - about 173–174
 - bar charts 175, 185–189
 - box plots 176, 189–192
 - common types of 174–178
 - forest plots 176–177, 193–197
 - Kaplan-Meier Survival Estimates plots 177–178, 198–212
 - line plots 174–175, 182–185
 - samples of 179–212
 - SAS tools for creating 178–179
 - scatter plots 174, 179–182
- clinical trial tables
 - See* tables
- clinical trials
 - colleagues 8–10
 - future of SAS programming in 245–247
 - study designs 4
 - using SAS software in 247
- CMH (Cochran-Mantel-Haenszel) test 220
- CNSR variable 108
- Cochran-Mantel-Haenszel (CMH) test 220

- code reuse 11–13
 - Cody, Ron
 - SAS Functions by Example, Second Edition* 104
 - colleagues, of statistical programmers 8–10
 - comma-delimited files
 - See ASCII text
 - Common Statistical Methods for Clinical Research with SAS Examples* (Walker and Shostak) 215
 - COMPUTE block 121, 162
 - "Computer Systems Validation in Clinical Research: A Practical Guide" (ACDM) 250
 - concomitant medication data 27–28, 33, 140–145
 - Condition tab (SAS XML Mapper) 68
 - conditional logic 14–15
 - CONTENTS procedure 65–66, 68
 - continuous data
 - inferential statistics 221–225
 - summary tables 122–130
 - contract research organization (CRO) 2
 - CORR procedure 226
 - correlation coefficients 226
 - covariates 27
 - Cox proportional hazards model 196
 - CPORT procedure 231, 232–233
 - CRF (case report form)
 - about 8
 - concomitant medications 27–28
 - demographics 27
 - drug logs 29–30
 - laboratory data 30–31
 - medical history data 28–29
 - prior medications 27–28
 - samples 26–38
 - trial-specific baseline data 27
 - CRO (contract research organization) 2
 - crossover trials 4
 - CSR (clinical study report) 2
 - CSS variable 217
 - CSV files
 - See ASCII text
 - CV variable 217
 - CYCLEATTRS option, SGPLOT procedure 189
- D**
- data
 - See also clinical data
 - See also exporting data
 - See also importing data
 - categorizing 21–24
 - cleaning 20–21
 - continuous 122–130, 221–225
 - efficacy 26
 - encryption options 255
 - hardcoding 24–26
 - managing 8–9
 - manipulating 99–104
 - missing 90–93
 - normalization of 82–89
 - transposing 82–89
 - windowing 78–82
 - data analysis
 - correlation coefficients 226
 - descriptive statistics 216–218
 - inferential statistics 218–225
 - statistical analysis plan (SAP) 11
 - time-to-event analysis statistics 225
 - data safety and monitoring board (DSMB) 25
 - data sets 13–14, 100–103
 - DATA steps
 - about 48–49
 - redefining variables within 100–103
 - transposing data with 86–89
 - DATA_NULL 170
 - dBASE database 243
 - DBSASLABEL option 58
 - DDE (Dynamic Data Exchange) 61
 - define.xml 5, 69, 231–232
 - Delwiche, Lora
 - The Little SAS Book: A Primer* 17

demographic data 27
 dependent variables 83
 DESCENDING option 221
 descriptive statistics 216–218
 device approval process 4
 dictionaries, referencing 95–98
 digital signatures 6, 246
 Dmitrienko, Alex
 *Analysis of Clinical Trials Using SAS: A
 Practical Guide* 215
 domains 26
 double-blind trials 4
 drug approval process 3–4
 drug assignment data sets 37–38
 Drug Information Association 251
 drug kit lists 37
 drug logs CRF 29–30
 DSMB (data safety and monitoring board) 25
 Dynamic Data Exchange (DDE) 61

E

"E3 Structure and Content of Clinical Study
 Reports" (FDA) 6, 32, 250
 "E6 Good Clinical Practice: Consolidated
 Guidance" (FDA) 6–7, 250
 "E9 Statistical Principles for Clinical Trials"
 (FDA) 6, 250
 eCTD (Electronic Common Technical
 Document) 7
 EDC (electronic data capture) 8, 20
 efficacy data 26
 electronic signatures 6, 246
 electronic submission to FDA 229–232
 encryption options 244, 255
 endpoint assessments 34
 Enterprise Guide
 about 213
 exporting data to Microsoft Office files with
 242–243
 exporting data with 238–239
 importing data 49–52, 59–62
 Enumeration tab (SAS XML Mapper) 68

equivalence trial 4
 event assessments 34
 events class 26
 Excel files
 See Microsoft Office files
 EXPORT procedure
 exporting ASCII text with 233–237
 exporting data to Microsoft Office files with
 240–242
 Export Wizard
 exporting ASCII text with 233–237
 exporting data to Microsoft Office files with
 240–242
 exporting data
 about 229–230
 ASCII text 233–239
 with CPORT procedure 232–233
 creating define.xml 231–232
 creating ODM XML 231–232
 descriptive statistics with FREQ procedure
 216
 descriptive statistics with UNIVARIATE
 procedure 217–218
 encryption options 244
 file transport options 244
 to Microsoft Office files 240–243
 proprietary data formats 243–244
 using SAS XPORT transpose format
 230–231
 Extreme Programming (XP) 255

F

failure estimate plots 201–205
 FDA (Food and Drug Administration)
 about 3
 exporting data to 229–232
 regulation and guidance 5–6
 resources 250–251
 File Transfer Protocol (FTP) 255
 file transport options 244, 255
 findings class 26
 Fisher's exact test 219

floating-point comparisons 104
 Food and Drug Administration
 See FDA (Food and Drug Administration)
 FOOTNOTE statement 166
 footnotes 170
 forest plots 176–177, 193–197
 FORMAT procedure 195
 Format tab (SAS XML Mapper) 67
 free-text variables 21–22
 See also ASCII text
 FREQ procedure
 clinical trial graphs 189
 Cochran-Mantel-Haenszel tests in 220
 for data sets 84, 92
 exporting descriptive statistics with 216
 output for 24
 tables and listings 146
 FTP (File Transfer Protocol) 255
 future of SAS programming 245–247

G

GCHART procedure 178
 GCPs (Good Clinical Practices) 6–7
 "General Principles of Software Validation:
 Final Guidance for Industry and FDA
 Staff" 250
 GETNAMES option 58
 GLM procedure 224–225
 Good Clinical Practices (GCPs) 6–7
 Google Search 252
 GPLOT procedure 178
 graphics assistants 212–214
 GRAPHICS statement 182
 Graphics Template Language (GTL) 212
 Graph-N-Go facility 212
 GROUP= option 182, 185, 192
 GROUP=TRTP statement 182
 GTL (Graphics Template Language) 212
 "Guidance for Clinical Trial Sponsors:
 Establishment and Operation of
 Clinical Trial Data Monitoring
 Committees" (FDA) 251

"Guidance for Industry: Computerized Systems
 Used in Clinical Investigations" 250
 "Guidance for Industry: Providing Regulatory
 Submissions in Electronic Format -
 General Considerations" 164
*Guidance for Industry Providing Regulatory
 Submissions in Electronic Format:
 Submissions Under Section 745(a) of
 the Federal Food, Drug, and
 Cosmetic Act* 251
*Guidance for Industry Providing Regulatory
 Submissions in Electronic Format-
 Standardized Study Data* 251
 "Guidance on Electronic Standardized Study
 Data" 229
 "Guidance on Electronic Submission of
 Applications" 229
 Gupta, Sunil
 *Quick Results with the Output Delivery
 System* 165

H

hardcoding data 24–26
 Haworth, Lauren E.
 *Output Delivery System: The Basics and
 Beyond* 165
 hazard ratios 196
 HIPAA regulations 246
 HTML database 243
 HTML output 182
 HTML/XHTML/XML specifications 255

I

ICH (International Conference on
 Harmonization) 5, 251
 ID statement 86–88
 IDMC (independent data monitoring committee)
 report 25, 113
 IEEE (Institute of Electrical and Electronics
 Engineers) 254–255

- IF-THEN/ELSE logic 14–15
Implementing CDISC Using SAS: An End-to-End Guide 69, 72, 89, 232
- IMPORT procedure 42–47, 55–58
- Import Wizard 42–47, 55–58
- importing data
 about 39
 ASCII text 41–52
 CDISC Model content files 68–69
 CDISC ODM files 70
 CDISC SAS transport format files 69
 clinical data management systems 40–41
 define.xml 69
 Microsoft Office files 52–62
 relational databases 40–41
 SAS/ACCESS LIBNAME statement 41
 SAS/ACCESS SQL pass-through facility 40–41
 XML 62–68
- %INCLUDE macro statement 12
- independent data monitoring committee (IDMC)
 report 25, 113
- independent variables 83
- INDEX function 24
- INDEXW function 24
- industry regulations and standards 4–8, 251
- inferential analyses 27
- inferential statistics
 obtaining from categorical data analysis 218–221
 obtaining from continuous data analysis 221–225
- INFILE statement 48–49
- information technology (IT) 9
- INPUT statement 48–49
- Instant ODS: Style Templates for the SAS Output Delivery System* (Johnson) 166
- Institute of Electrical and Electronics Engineers (IEEE) 254–255
- intent-to-treat population 73
- Interactive Voice Response System (IVRS) 36
- International Conference on Harmonization (ICH) 5, 251
- interquartile range 189
- interventions class 26
- Investigational New Drug (IND) application 3, 7
- investigator 10
- IT (information technology) 9
- IVRS (Interactive Voice Response System) 36
- J**
- Jansen, Lex 252
- Java/JavaScript 254, 255–256
- JMP database 243
- jobs, qualifying for and obtaining 256
- Johnson, Bernadette
Instant ODS: Style Templates for the SAS Output Delivery System 166
- joins 93–95
- K**
- Kaplan-Meier Survival Estimates tables 152–159, 177–178, 198–212
- KEYLEGEND statement 185
- Kruskal-Wallis test 224–225
- KURTOSIS variable 217
- L**
- laboratory data 30–31, 34, 145–152
- Laboratory Data Model (LAB) 5
- last observation carried forward (LOCF) 73–77
- lexjansen.com 252
- LIBNAME statement 12, 53–55
- LIFETEST procedure
 clinical trial graphs 201, 209
 creating Kaplan-Meier Survival Estimates plot using 210–212
 creating Survival Estimates plot directly from 209–212
 for data sets 109
 time-to-event analysis statistics 225
- line plots 174–175, 182–185

LINEPARM statement 182
 LINESIZE output option 164, 166
 LinkedIn 256
 listings, creating 159–164
The Little SAS Book: A Primer (Delwiche and Slaughter) 17
 LOCF (last observation carried forward) 73–77
 LOGISTIC procedure 84, 195, 196–197, 221
 logistic regression analysis 176, 221
 log-rank test 225
 Lotus 1-2-3 database 243

M

macro-based reporting systems 172
 macros 15–16
 %MAKECOD macro 15
 Mantel-Haenszel test 219–220
 many-to-many comparisons/joins 93–95
 MARKERATTRS statement 196
 markup languages 255
 matrix management structure 9
 Matthews, Carol
 Validating Clinical Trial Data Reporting with SAS 6, 250
 MAX variable 217
 MEAN variable 217
 MEANS procedure 218
 MedDRA (Medical Dictionary for Regulatory Activities) 95–97
 MEDIAN variable 218
 medical devices, approving 4
 medical dictionaries 95–98
 Medical Dictionary for Regulatory Activities (MedDRA) 95–97
 medical history data CRF 28–29
 medical writing 10
 MERGE statement 102
 MERGE-BY statement 93
 Microsoft Developer Network 255
 Microsoft Office files
 about 52–53
 Excel pivot-point year 100

exporting data to 240–243
 footnotes in Windows files 170
 IMPORT procedure 55–58
 Import Wizard 55–58
 LIBNAME statement 53–55
 reading Access files with LIBNAME statement 54–55
 reading Access files with SQL pass-through facility 59
 reading Excel files with IMPORT procedure 57–58
 reading Excel files with LIBNAME statement 53–54
 reading Excel files with SQL pass-through facility 58–59
 SAS Enterprise Guide 59–62
 SAS/ACCESS SQL pass-through facility 58–59
 MIN variable 217
 MISSING option
 REPORT procedure 161–162
 TABLES statement 216
 missing results 90–93
 MIXED option 58
 MODE option 217
 modeling tools 255
 multi-center trials 4

N

N variable 217
 NMISS variable 217
 NOBS variable 217
 non-inferiority trials 4
 nonparametric tests 219
 normalization of data 82–89
 NPARIWAY procedure 224–225
 N-sample test of the means 224–225
 NxP tests 219–220

O

Object Linking and Embedding (OLE) 61
 ODBC (Open Database Connectivity) 61

- ODM (Operational Data Model) 5, 70
 - ODM XML
 - See XML files
 - ODS destination 129, 170, 182
 - ODS Graphics Designer 179, 213
 - ODS Graphics Editor 213–214
 - ODS HTML statement 182
 - ODS LISTING statement 213
 - ODS OUTPUT statement 223–224, 226–227
 - ODS Report Writing Interface 170
 - ODS RTF statement
 - BODYTITLE option 166
 - sending output to 165–166
 - ODS STYLE 170–171
 - ODS TRACE 226–227
 - OLE (Object Linking and Embedding) 61
 - ON clause 95
 - one-sample t tests 221–223
 - one-way analysis of variance 224–225
 - online documentation 253
 - Open Database Connectivity (ODBC) 61
 - OpenCDISC Validator 232
 - Operational Data Model (ODM) 5, 70
 - Oracle database, getting data from 40–41
 - ORDER= option 221
 - OTHERWISE clause 15
 - OUT= option 144, 216
 - OUTPCT option 216
 - Output Delivery System: The Basics and Beyond* (Haworth) 165
 - OUTPUT statement 217
- P**
- P1 variable 217
 - P5 variable 217
 - P10 variable 217
 - P90 variable 218
 - P95 variable 218
 - P99 variable 218
 - page counter 168–169
 - PANELBY statement 189
 - Paradox database 243
 - parallel trials 4
 - parametric tests 219
 - parent-child data problem 13–14
 - Pass-Through Facility 40–41, 58–59
 - patient disposition tables 122
 - patient listings, creating 159–164
 - patient medical history data 28–29, 139
 - PDF (portable document format) files 167–168
 - PDUFA (Prescription Drug User Fee Act) V 246
 - Pearson chi-square tests 219
 - per-protocol populations 73
 - Pharmaceutical Users Software Exchange (PhUSE) 251
 - PharmaSUG user group 253
 - phases 3
 - PHREG procedure 109, 196, 225
 - PhUSE (Pharmaceutical Users Software Exchange) 251
 - pivot point 99–100
 - portable document format (PDF) files 167–168
 - pre-clinical studies 3
 - predictor variables 221
 - preparing clinical trial data 19, 20–26
 - Prescription Drug User Fee Act (PDUFA) V 246
 - PRINT procedure 53–54, 65–66, 159
 - prior medication data 27–28, 140–145
 - PROC TEMPLATE Made Easy: A Guide for SAS Users* (Smith) 171
 - programming
 - readability of 17
 - tasks 11–13
 - project management 9
 - Properties tab (SAS XML Mapper) 67
 - protocols 10
 - PSI (Statisticians in the Pharmaceutical Industry) 251
 - PUT statement 25, 131
 - p-value 222–223

Q

Q1 variable 218
 Q3 variable 218
 QA (quality assurance) 10
 QOL (quality-of-life) data 38
 QRANGE variable 218
 query process 20
Quick Results with the Output Delivery System
 (Gupta) 165

R

randomization 4, 36–38
 RANGE variable 217
 RDBMS (relational database management system) 9
 readability of code 17
 redefining data set variables 100–103
 REFLINE statement 185, 196
 regulations
 changes in 246
 industry 4–8
 regulatory resources 250–251
 relational database management system
 (RDBMS) 9
 relational databases 40–41
 REPORT procedure 159, 166
 creating clinical trial tables with 118–122
 creating listings with 160–164
 MISSING option 161–162
 resources
 jobs 256
 regulatory 250–251
 SAS help 252–254
 standards and industry organizations 251
 technical skills 254–256
 reusing code 11–13
 ROUND function 104
 RTF (rich text format) 165, 255
 %RUN macro 15

S

SAE (serious adverse events) 33
 safety data 26
 safety populations 73
 SAP (statistical analysis plan) 11
SAS 9.4 Output Delivery System: User's Guide
 170
 SAS certifications 256
 SAS Enterprise Guide
 about 213
 exporting data to Microsoft Office files with
 242–243
 exporting data with 238–239
 importing data 49–52, 59–62
 SAS Focus Areas 253–254
SAS Functions by Example, Second Edition
 (Cody) 104
 SAS Global Forum (SGF) 253
SAS Graph Template Language: Reference 179
SAS Graph Template Language: User's Guide
 179
 SAS Graphics, when to use 214
 SAS Help 252–254
SAS Macro Programming Made Easy (Burlew)
 16
SAS ODS Graphics Designer: User's Guide 179
SAS ODS Graphics: Procedures Guide 179
 SAS Press 253
 SAS programming, future of 245–247
 SAS technical support 252–253
 SAS XML Mapper 62, 67–68
 SAS XPORT transport format 230–231
 SAS/ACCESS LIBNAME statement 41
 SAS/ACCESS SQL pass-through facility 40–41,
 58–59
 SAS-L mailing list 252
 SCANTEXT option 58
 SCANTIME option 58
 scatter plots 174, 179–182
 SCATTER statement 182, 196, 209
 scripting 254
 SDLC (system development life cycle) 172,
 254–255

- SDTM (Study Data Tabulation Model) 5, 19, 83
 Secure File Transport Protocol (SFTP) 244, 255
 SELECT statement 15, 59, 95
 SERIES statement 185
 serious adverse events (SAE) 33
 SET statement 102
 SF-36/SF-12 Health Survey 38
 SFTP (Secure File Transfer Protocol) 244, 255
 SGANNO= option 179
 SGF (SAS Global Forum) 253
 SG PANEL procedure 178, 186–189
 SGPLOT procedure
 clinical response line plot using 183–185
 clinical trial graphs 178, 179–182, 209–210, 214
 creating a Kaplan-Meier Survival Estimates Plot using 198–201
 creating box plot using 190–192
 creating forest plot using 193–196
 creating Kaplan-Meier Failure Estimates Plot using 202–205
 creating Kaplan-Meier Survival Estimates plot using 206–209
 CYCLEATTRS option 189
 SGRENDER procedure 213
 SGSCATTER procedure 178
 shift, laboratory data 145–152
 Shilling, Brian
 Validating Clinical Trial Data Reporting with SAS 6, 250
 Shostak, Jack
 Common Statistical Methods for Clinical Research with SAS Examples 215
 sign test 223
 signatures, electronic 6, 246
 single-blind trials 4
 site management 8
 site-based trials 4
 SKEWNESS variable 217
 Slaughter, Susan
 The Little SAS Book: A Primer 17
 Smith, Kevin
 PROC TEMPLATE Made Easy: A Guide for SAS Users 171
 Society for Clinical Data Management 251
 software, using in clinical trial industry 247
 software development life-cycle model (SDLC) 172
 SOUNDEX function 24
 Spearman correlation coefficient 226
 SPSS database 243
 SQL (structured query language) 40
 SQL procedure 76, 93–95
 standards
 changes in 246–247
 industry 4–8
 Stata database 243
 statistical analysis plan (SAP) 11
 Statistical Graphics Procedures by Example: Effective Graphs Using SAS 179, 206
 statistical programmers
 being a good student 17
 drug/device development process 3–10
 guiding principles for 10–17
 work description 2
 statistically significant association 218
 Statisticians in the Pharmaceutical Industry (PSI) 251
 statistics, obtaining 226–227
 STD variable 217
 STDMEAN variable 217
 Stokes, Maura
 Categorical Data Analysis Using the SAS System 218
 strata 27
 stratified NxP test 220
 structured query language (SQL) 40
 "Study Data Standards for Submissions to CDER" 7, 251
 Study Data Tabulation Model (SDTM) 5, 19, 83
 Study Data Technical Conformance Guide: Technical Specifications Document 230, 251
 study day variables 78

study populations, defining 72–73
 study termination form 36
 STYLE= option 165, 168, 170
 STYLEATTRS statement 189
 subject level analysis data set 105
 submission of electronic files to FDA 229–232
 subsetting data sets 13–14
 SUBSTR function 24
 SUM variable 217
 SUMMARY procedure 218
 SUMWGT variable 217
 superiority trials 4
 survival plots, creating with number at risk
 205–209
 system development life cycle (SDLC) 172,
 254–255
 systems development methodology 254–255

T

table shell 130
 tables
 about 114–115
 clinical trial table 114–122
 creating 114–159
 creating adverse event summaries 130–139
 creating concomitant or prior medication
 tables 140–145
 creating Kaplan-Meier Survival Estimates
 tables 152–159
 creating laboratory shift tables 145–152
 creating with REPORT procedure 118–122
 creating with TABULATE procedure
 116–118
 typical continuous/categorical summary
 tables 122–130
 TABLES statement 216, 220
 TABULATE procedure 84, 116–118, 218
 tasks, programming 11–13
 technical skills 254–256
 technology, changes in 245–246
 TEMPLATE procedure 182

TESS (treatment-emergent signs and symptoms)
 32
 tests for unequal variances 223
 TEXTSIZE option 58
 third-party SAS Web pages 254
 time-to-event analysis statistics 108–111, 177,
 225
 Time-to-Event variable 108
 TITLE statement 166
 tools, for creating clinical trial graphs 178–179
 transport format files 69
 TRANSPOSE procedure 84–89
 transposing data 82–89
 treatment-emergent signs and symptoms (TESS)
 32
 trial-specific baseline data CRF 27
 triple-blind trials 4
 TTEST procedure 223–224
t-tests 222–223
 "21 CFR-Part 11 Electronic Records: Electronic
 Signatures-Final Rule" 250
 2x2 test 218–219
 two-sample *t*-tests 223–224
 two-sided tests 221

U

UML (Unified Modeling Language) 255
 unequal variances, tests for 223
 UNIVARIATE procedure 217–218, 221
 UNIX environment 170
 UPDATE statement 102
 USEDDATE option 58
 users groups 253
 USS variable 217

V

*Validating Clinical Trial Data Reporting with
 SAS (Shilling and Matthews)* 6, 250
 variables, defining 72
 VBARPARAM statement 189
 VBOX statement 192

284 *Index*

VBScript programming 254
version control software 254

W

W3C 255
Walker, Glenn
 *Common Statistical Methods for Clinical
 Research with SAS Examples* 215
WHERE clause 14
WHO Drug Dictionary 95
WHO Drug Dictionary Enhanced (WHO DDE)
 97–98
Wilcoxon rank sum test 224
Wilcoxon signed rank test 223
windowing data 78–82

X

XAXIS statement 196
XML files
 about 62–63, 231–232
 SAS XML Mapper 62, 67–68
 XML LIBNAME engine 62, 63–66,
 231–232
XML LIBNAME engine 62, 63–66, 231–232
XML Mapper 62, 67–68
XMLMap Settings tab (SAS XML Mapper) 68
XP (Extreme Programming) 255
XPORT transport format 230–231

Y

YEARCUTOFF option 99–100

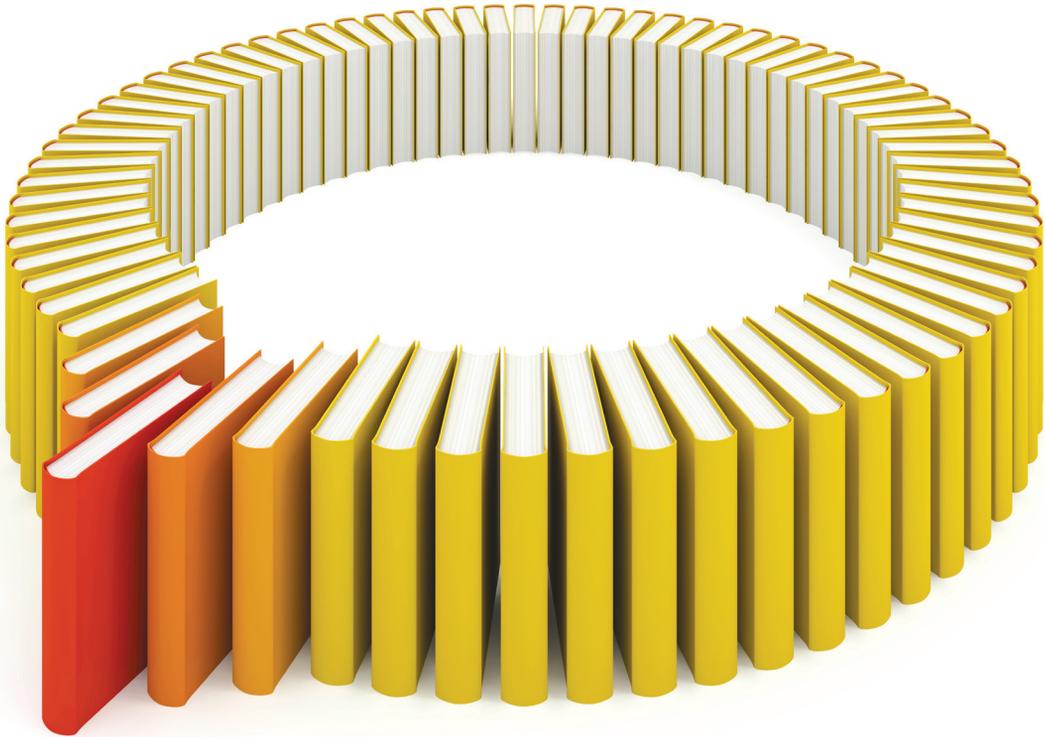
Z

zero results 90–93

About The Author



Jack Shostak, Associate Director of Statistics, manages a group of statistical programmers at the Duke Clinical Research Institute. A SAS user since 1985, he is the author of *SAS Programming in the Pharmaceutical Industry*, and coauthor of *Common Statistical Methods for Clinical Research with SAS Examples, Third Edition*, as well as *Implementing CDISC Using SAS: An End-to-End Guide*. Shostak has published papers for the Pharmaceutical SAS Users Group (PharmaSUG) and the NorthEast SAS Users Group (NESUG), and he contributed a chapter, "Reporting and SAS Tool Selection," in the book *Reporting from the Field*. He is active in the Clinical Data Interchange Standards Consortium (CDISC) community, contributing to the development of Analysis Data Model (ADaM), and he serves as an ADaM trainer for CDISC.



Gain Greater Insight into Your SAS[®] Software with SAS Books.

Discover all that you need on your journey to knowledge and empowerment.

 support.sas.com/bookstore
for additional books and resources.


THE POWER TO KNOW.®

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration. Other brand and product names are trademarks of their respective companies. © 2013 SAS Institute Inc. All rights reserved. S107969US.0413