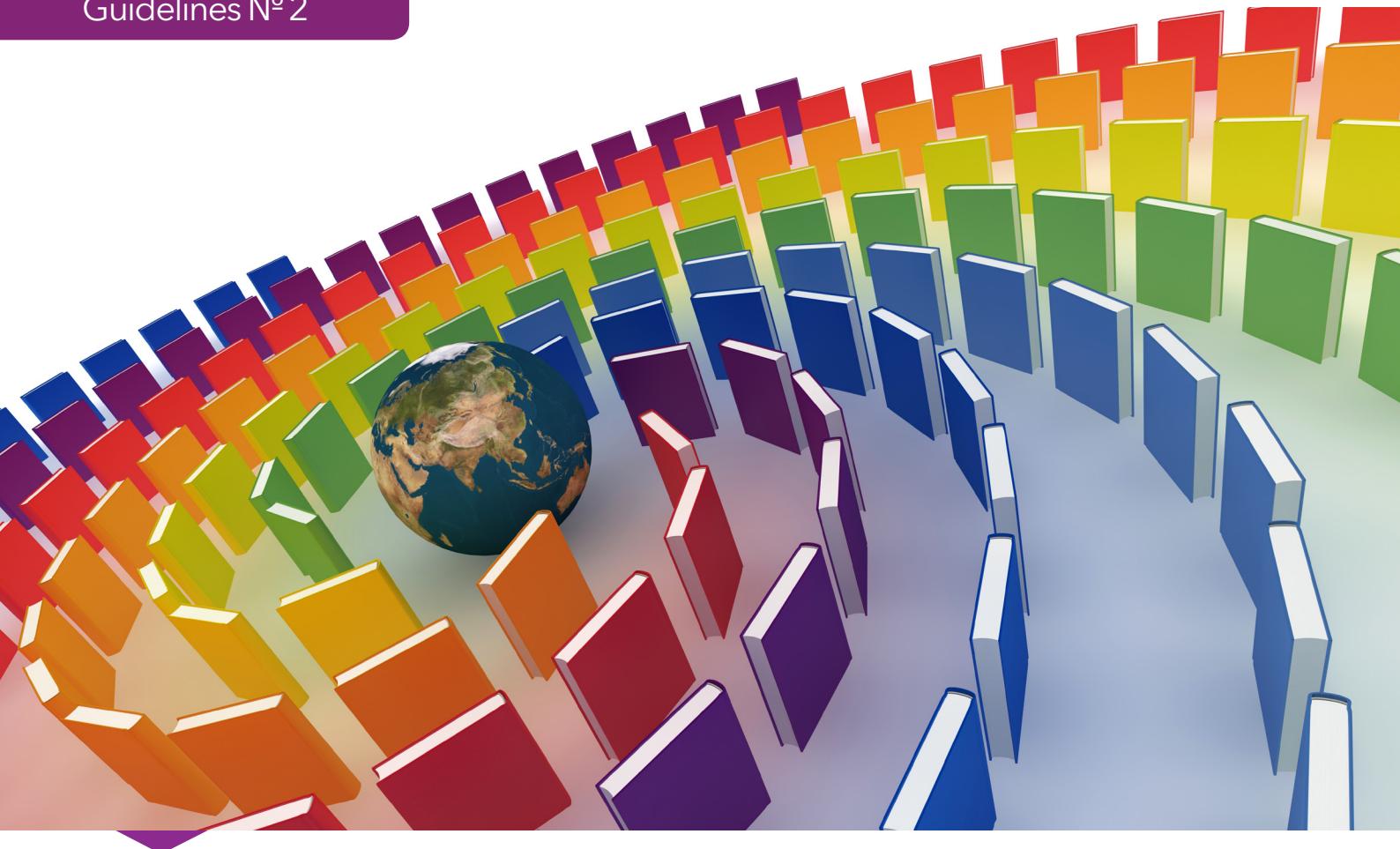


Guidelines Nº2



## SAS® Press Style Guide



---

## Writing with Readers' Needs in Mind

As you begin writing your manuscript, and later when you revise it, this chapter will help you ensure that your readers remain engaged and undistracted as they move through your material. If you apply these guidelines consistently, your reader will not only better comprehend your information while reading, but will also better retain it after reading. In addition, your consistent adherence to these guidelines will prevent you from having to complete time-consuming revisions, ensuring that your book is delivered into your readers' hands by or before your scheduled publication date.

---

## Consulting Additional Resources for More Information

SAS Press style is based on *The Chicago Manual of Style*, 16th edition, which you may consult for style and usage guidance beyond the scope of this chapter: <http://www.chicagomanualofstyle.org/16/contents.html>.

For correct spelling, see *Merriam-Webster's Collegiate Dictionary*, 11th edition. Wherever *Merriam-Webster* gives variant spellings for entries, use the first spelling listed. See the convenient online version: <http://www.merriam-webster.com/>

For style points that govern books that target a physical sciences or social sciences audience, but that are not covered in this document or in *Chicago*, consult *The Publication Manual of the American Psychological Association*, 6th edition: <http://www.apastyle.org/>.

For style points that govern books that target programmers, but that are not covered in this document or in *Chicago*, consult the *Microsoft Manual of Style*, 4th edition: <http://www.microsoft.com/learning/en-us/book.aspx?ID=15053>.

For guidelines on writing for your international readers, see John R. Kohl's *Global English Style Guide: Writing Clear, Translatable Documentation for a Global Market* (2008), a free copy of which SAS Press sends to all of our authors.

---

## Engaging Your Readers by Addressing Them Directly

SAS Press books personally engage and directly instruct the reader; therefore, remember to keep your sentences consistently in the *imperative mood*, which means simply as direct "commands." With imperative mood, use the *second-person point of view*, which simply means that you address the reader directly, as "you." In sentences that convey the imperative mood, the "you" is ordinarily implied rather than actually stated.

Imperative mood and second-person point-of-view mean that you will write predominantly in the present tense. Whenever possible, avoid past tense and future tense.

### Example 1: Incorrect Mood, Point of View, and Tense

*The user will end PROC steps with a RUN statement.*

### Example 2: Correct Mood, Point of View, and Tense

*End your PROC steps with a RUN statement.*

---

## Engaging Your Readers with Active Voice

Build your sentences around strong (“active”) verbs and specific nouns. Doing so will reduce your reliance on adjectives, adverbs, and modifying phrases, all of which dilute your impact.

When reviewing your work, circle all forms of the verb *to be* (*is, are, was, were has been, had been*); then try to replace 90% of them with more specific verbs. Often you will have to completely recast the sentence and eliminate modifiers after forcing the verb to convey most of your meaning.

Writing with active verbs instead of passive constructions naturally results in subject-verb-object syntax. This syntax aids in translation and is easier and less convoluted for all readers to comprehend.

### **Example 1: Passive Voice**

Economic incentive programs *have been designed* by EPA to reduce emission of air pollutants.

### **Example 2: Active Voice**

EPA *designed* incentive programs to reduce emissions of air pollutants.

---

## Helping Your Readers Comprehend Your Information

Although, conventionally, user guidebooks adopt a friendly, informal writing style, in at least two respects—diction and sentence construction—avoiding excessive informality will ensure that you don’t lose any of your readers.

---

### **Avoid Confusing Phrasing**

Some kinds of phrases frequently confuse readers whose first language is not English. They can also “confuse” translation software and therefore add tremendously to the costs of book production. To avoid confusing any readers, use precise denotative wording, even if such phrasing seems formal and abstract. In choosing your words, guard against the following:

- figurative language (metaphors)
- colloquial and idiomatic expressions
- slang
- humor (such as puns and “inside” jokes)
- jargon and excessive acronyms or initialisms

### **Examples: Inappropriate Diction**

- Going down this *road* leads to *big headaches*.
- This programming *trick* is especially *neat*.
- Follow this *rule of thumb*.

### **Examples: Appropriate Diction**

- Using this technique causes more work.
- This programming technique is especially effective.
- Follow this rule.

---

### **Vary Sentence Length and Complexity**

Although short sentences sound informal, exclusive use of them impedes reading comprehension across an entire document. Cognitive testing has repeatedly shown that superior reading comprehension and retention depends on sentence *variety*:

- A succession of sentences that are short produces choppiness.
- A succession of sentences that are lengthy and complex becomes difficult to follow.

- A succession of sentences that are all the same length is monotonous.

Instead of checking only the average number of *words* per sentence, monitor the number of *ideas* per sentence. A sentence that is 25 words long and has one main idea supported by a subordinate clause is often more easily comprehended than a sentence comprising two equally important ideas (two independent clauses).

As you write, strive to do the following:

- Keep the number of *and*-linked independent clauses to a minimum: Independent clauses that are conjoined by *and* are not only difficult to comprehend, but also rhetorically weak.
- To emphasize an idea, cast it as a short sentence.
- Where you have two ideas per sentence, try relegating one to a subordinate clause. Subordination might lengthen individual sentences, but it reduces overall wordiness (word count). In addition, subordination of one idea to another creates a foreground-background relationship between ideas, which supports reading comprehension.
- Avoid excessive subordination (subordinate clauses) in a single sentence; in technical user guides, one subordinate idea per sentence, as well as 25 words per sentence, is usually plenty.

### **Use a Clarifying Noun after *This*, *That*, *These*, and *Those***

The pronouns *this*, *that*, *these*, and *those* often obliquely refer back to, or substitute for, previous nouns or whole ideas, called *antecedents*. To ensure clarity and to keep your reader moving forward without pause or backtracking, always immediately follow these pronouns with a noun.

### **Example: Pronouns with Unclear Antecedents**

Don't assume that, because *this* works from time to time, you don't need the ODS CLOSE statement. *This* is a common error.

### **Example: Effective Use of Demonstrative Pronouns**

Don't assume that, because *this technique* works from time to time, you don't need the ODS CLOSE statement. *This assumption* is a common error.

### **Write with a Global Audience in Mind**

Because many of your readers will not have English as their native language, adopt a global English style to help them comprehend your material. Before writing and again before revising, review the copy of John R. Kohl's *Global English Style Guide: Writing Clear, Translatable Documentation for a Global Market* (2008) that SAS Press has provided to you.

Following are the ten most crucial global English rules for you to apply to your book, some of which appear elsewhere in these *Instructions for Authors* as helpful to all audiences. Consult Kohl's book for full explanations and examples of these guidelines:

- Use short sentences (but with some sentence variety).
- Use complete sentences to introduce vertical lists. Don't interrupt sentences with programming code.
- Untangle long noun phrases (don't have numerous nouns functioning as adjectives before a noun).
- Expand *—ed* verbs by inserting *that is* before the verb.
- Always revise *—ing* verbs that follow nouns by inserting *that is* before the verb.
- Use the word *that* liberally to improve readability and to eliminate ambiguities.
- Choose simple, precise words that have a limited, denotative range of meanings.
- Standardize your terminology and phrasing; avoid using different words to convey the same meaning.
- Don't use slang, idioms, abbreviations, colloquialisms, or figurative language.
- Clarify which parts of a sentence are being joined by *and* or by *or*.

---

### Define Your Acronyms and Initialisms

If you use acronyms or initialisms in your book, spell out each one at its first appearance in each chapter, like so: analysis of variance (ANOVA).

---

## Using Capitalization Judiciously and Consistently

In conformance with Chicago style, SAS Press prefers a “downstyle,” which means that you should lowercase words except where a compelling reason exists for capitalization. Capital letters cause readers to pause; conversely, lowercasing words keeps the reader moving through your material.

---

### Running Text

Observe the following style points in running text:

- Lowercase job or role titles, like so: senior software developer.
- Capitalize department and division names, like so: the Publications Division.
- Lowercase common nouns, such as generic computing-related names, like so:
- access control list (ACL)
- application programming interface (API)
- central processing unit (CPU)

Capitalize proper nouns, such as proprietary names or names of very specific computer-related elements and processes, like so:

- Common Gateway Interface (CGI)
- Common Object Request Broker Architecture (CORBA)
- Lightweight Directory Access Protocol (LDAP)

In instructional running text that points the reader to elements of a specific user interface or window, match the case to the one used in the relevant interface, window, tab, field, check box, list box, or other such element. To set off any of the resulting lowercased element names from running text, make them boldface. The case of these terms in general running text will not necessarily match the case (or boldface) you use in these instructional passages.

Lowercase names of laws, theories, models, statistical procedures, diseases, effects, variables, factors, and hypotheses, like so:

- parallel distributed processing model
- theory of evolution
- Heisenberg’s uncertainty principle

---

### Titles and Headings

For all titles and headings, including figure and table captions, apply *Chicago* headline style as follows:

- Capitalize the first and final word, regardless of all other rules.
- Capitalize all major words (nouns, pronouns, verbs, adverbs, and adjectives).
- Lowercase minor words (conjunctions, prepositions, and the articles *a*, *an*, and *the*).
- Capitalize the second element of a hyphenated compound, such as *SAS-Based* and *Follow Up*. For details about capitalizing code, see “How to Correctly Style Your Code” on Page 22.

---

## Constructing and Styling the Elements of Your Book

Here, you will learn the structural elements of your book and how to style them, as well as important standards for displayed elements such as program output and original figures.

---

## Front Matter

The front matter includes all the book parts that help introduce the book and its navigation to your reader.

### ***Table of Contents***

Although creating a table of contents during your outline is very helpful, our production department will generate your final table of contents and your individual chapters' tables of contents.

### ***About This Book***

Please supply this front matter content by filling in the “About This Book” template and returning it to your developmental editor with your draft manuscript. This front matter normally covers all information that would otherwise appear in a preface.

### ***About the Author***

About the time that you submit your full draft manuscript, your editor will begin working with you on a biographical blurb that will appear in the front matter, on the book cover, and on your Author web page.

### ***Preface (Optional)***

If you like, explain why you wrote the book and what readers might like to know about the content before diving into your first chapter. Note that that prefaces have been largely replaced with “About This Book,” so our preference is that you fit prefatorily content in under the headings offered by the About This Book template, which your editor will supply you.

### ***Acknowledgments (Optional)***

If you like, write an acknowledgments page or a dedication.

---

## Chapters

As you write, the most crucial question to constantly ask yourself is, “Why does the reader care about this?” Consider including some scenarios in which a user would use the particular method or concept you’re explaining: for example, open your chapter with a “hook” (a question, dilemma, or problem) and then using your chapter to solve that problem. Keep reminding the reader why your content is useful and valuable to them, and make your thought process transparent and easy to follow. Additionally, **be concise** – do not make your reader do extra work to find and extract the information they need from what you’ve written.

Ask yourself the following questions:

- Did I contextualize why this information or example is important for my reader to know?
- Do I give my readers enough of an introduction to this concept to fully understand it?
- Have I included too much background information? Can I cut any of it?
- Is the content I’ve included necessary and sufficient to explain this concept?

When I finished explaining one concept, did I provide a logical transition into the next topic?

Separate your material into chapters of approximately equal scope and length, guarding against both overlong and skimpy chapters. Once you have defined your chapters, sequence them logically, so that each chapter topic forms the foundation for the next chapter topic. Create chapter titles sufficiently clear and specific that the reader can discern the logic of your topic development from the table of contents alone.

### **Example: Ineffective Chapter Title**

Chapter 1: Errors

### **Example: Better Chapter Title**

Chapter 1: Understanding the Types of Errors in SAS Programs

---

## Sections and Subsections

After you have defined your chapters, consider how you might further divide each chapter into sections and subsections.

Headings and subheadings guide your reader through your chapter. They provide the reader information about the overall organization, make the material scannable (which is very useful to a busy reader), and serve as transitions. Each chapter should begin with a heading. Also, if you would like to include a level-2 heading, it will need to directly follow the level-1 heading. There should be no “floating text” between different levels of headings. Please limit yourself to three levels of headings.

As you outline and draft, make your headings meaningful and descriptive. They should clearly indicate the content of each section so that the reader—or prospective book buyer—can discern the logic of your development from the table of contents alone. For example:

### **Example: Ineffective Heading**

The LIBREF Statement

### **Example: Much Better Heading**

Using the LIBREF Statement to Assign a Permanent Library

In general, insert a heading or subheading every few paragraphs. Don’t go more than a page without inserting a heading. As you draft define subsections and draft subheadings, keep in mind that each section must contain at least two subsections or none.

In addition to striving to make each of your titles and headings as specific as possible, make all titles and headings that occupy the same “level” grammatically parallel with one another.

### **Example 1: Faulty Parallelism**

The following Level-1 headings contain a noun phrase, then a clause beginning with *-ing* verb, and then a complete imperative sentence. This lack of parallelism impairs reading comprehension and retention:

- SAS Programming Language
- Reading the SAS Log
- View the Properties of Data Sets

### **Example 2: Effective Parallelism**

The following Level-1 headings are parallel: Each heading is a clause beginning with a gerund (an *-ing* verb form) and ending with an object; this attention to parallelism aids reading comprehension and retention:

- Understanding the SAS Programming Language
- Reading the SAS Log
- Viewing the Properties of Data Sets

---

## Displayed Elements: Figures, Tables, Programs, and Output

The four types of displayed elements that you may include in your book are defined and exemplified in Table 1.1. In the caption heading, the first numeral represents the chapter number, and the second is the ordinal number for that type of displayed element. For example, the example figure caption heading is for the second figure in chapter 1.

**Table 2.1: Required Headings for Displayed Elements**

<b>Category</b>	<b>Caption Heading Style</b>	<b>Content</b>
Program	<b>Program 1.2: Descriptive Title</b>	Text copied and pasted from your programming code and styled from the template as ACode.
Output	<b>Output 1.2: Descriptive Title</b>	Screen captures of your ODS output (see Chapter 3)
Figure	<b>Figure 1.2: Descriptive Title</b>	Your original line drawings, graphs, flowcharts, and any screen captures that are <i>not</i> ODS output
Table	<b>Table 1.2: Descriptive Title</b>	Data or brief text that is rendered in original tabular format and that is <i>not</i> from ODS output
Log	<b>SAS Log 1.2: Descriptive Title</b>	Text copied and pasted from your SAS log and styled from the template as ACode.

Number your displayed elements by type, not across types. For example, if you create and insert a table after Figure 1.2, and it is the first table in the chapter, then its label will be Table 1.1.

Remember that we require you to deliver a separate folder that contains your original high-resolution images and screenshots. Although you will also insert your images where you want them to appear in the chapter files themselves, this separate graphics folder ensures that our production department can efficiently complete a quality check of all of your images.

Name each graphic in the folder individually, with the naming convention “**surname\_TW#\_label#.tif**,” where *label* is how you intend to label the element in the book. The number should match the caption number in your chapter manuscript. For example, “Smith\_12345\_Figure2.1.tif” would be the first figure in chapter 2 (Figure 2.1), and “Smith\_12345\_Output2.1.tif” would be the first output in chapter 2 (Output 2.1).

## Caption Headings

Next, in the Word file, use a caption heading for your graphic. So, immediately before the displayed element, type one of these labels and add a title that is sufficiently specific that the reader will understand at a glance exactly what information the element covers. Style the entire caption like so: **Figure 1.2** SAS AppDev Studio Opening Window.

## Reference to Elements in Running Text

In running text, mention each program, output, figure, and table by number in the paragraph immediately preceding the display. Choose for your reference location the first sentence that begins to refer to content in the displayed element. Either refer to the displayed element by number as part of a sentence, or indicate it in parentheses at the end of a sentence that explains or refers to it, like so: (Figure 1.1).

## Original Figures

Consider creating a figure, such as a line graph or a flowchart, wherever the reader needs to see a trend or a process across time. Quantifiable comparisons lend themselves to pie charts or bar charts.

Figure images must appear clear and must not be stretched, scaled up, pixelated, or blurry. If you need graphics assistance to finalize your figures, then contact your developmental editor. By the time you submit your final manuscript, all figures should be final.

If revisions are necessary, a graphics specialist will create soft-copy drafts (PDF) of any revised figures, which will be routed back to you for editing and review.

Remember that figures, when well designed and captioned, should be virtually self-contained. Ideally, your running text will not repeat what the figure conveys, and the figure will not merely summarize what is in the text. To ensure that the figure is self-contained, devise a unique, specific title for it.

When creating figures observe the following:

- Your hardcopy book will be printed in black and white, and your ebook will be in color. Submit color figures with this in mind.
- Medium-sized figures (preferred) should be in the range of 4 to 5 inches wide (288–360 points). Use the software application’s ruler or set the size of the layout page accordingly.
- For wide or complex figures, you may set the width to the maximum, 5.85 inches (421 points).
- Use 11 point Arial font for the text.
- Use 100% solid colors if possible. If you are using gradients or tints (a percentage of a solid color) within a figure, use one no lower than 12.5%.
- Draw all shapes and lines with a 1-point stroke weight.
- Placing an outside frame/border around a figure is not necessary; however, if you do use a border, please be consistent and do so throughout all figures within the book (1 point weight for the border lines is preferred).

## Original Tables

If you have a large volume of interrelated information or data that you need to summarize, consider creating a table. Tables work best if the tabular matter consists of numerical results or brief text. Avoid tables with lengthy cell entries, and if you have less than three columns of information, then choose to create a bullet list instead of a table. Ensure that each table you plan contains at least two columns in addition to the *stub*, which is the leftmost column that lists the row headings.

Traditionally, table row headings name independent variables, whereas the column headings name the test or dependent variables. Even if you are not tabulating scientific data, consider that readers naturally will pay less attention to the stub and more attention to the column headings, reading them from left to right. Therefore, think about capturing the new or most important information in the column headings, and think about the logical or rhetorically effective sequencing of them from left to right. One overruling consideration, however, is the need to avoid mixing units in the same column. For example, do not include percentages and currency in the same column.

Do not duplicate words across title, column headings and stub; place any duplicated word in question only at the highest level to which it can be logically assigned.

In column headings and cells, use single words, short phrases, numbers, or symbols. In addition check that column headings are grammatically parallel with each other and that row headings are grammatically parallel. For example, don’t use nouns in some headings and verbal clauses in others.

Remember that tables, when well designed and captioned, should be virtually self-contained. Ideally, your running text will not merely repeat what is evident on the face of the table, and the table will not merely summarize what is already in running text. To ensure that the table is self-contained, devise a unique, specific title for it.

Consult your developmental editor with any questions as you are designing original tables.

## Referring to Your Code: Comments, Shading, and Callouts

You might find that you need refer back to specific steps or elements of a displayed program for subsequent discussion in running text. Do not refer to sections of code by color, for your book will be printed in black and white. Choose one of the three permitted styles, and use only it throughout your book. Parts of displayed program code can be highlighted for later commentary by means of shading, bolding, or callout numbers, all of which can be done with the template. Choose one of the following three callout formats to use in a consistent way throughout your book:

- **Option 1.** Comment on parts of your program as blocks, using /\*...\*/.
- **Option 2.** You may use boldface or gray shading for sections of code that you want to highlight or discuss after the graphic display.
- **Option 3.** You may use numeral callout symbols to mark lines of programming code for subsequent discussion in a vertical list, as in the following example:

```
%updateStatus ①
  (statusDataSet = &statusDataSet,
   msg = Request Beginning Execution
  )

data _null_;
  x = sleep(&refresh); ②
run;
```

① The updateStatus macro adds an observation to a SAS data set containing an update on the program's status. The name of the data set is made available as a macro variable. The macro should be called, with an appropriate value for the Msg parameter, at key checkpoints in the program. This call to the macro generates the observation in the status data set shown in Figures 19.4 and 19.5 (the first row is inserted by the sample framework that spawns the independent SAS session).

② Use the Sleep function to simulate a long-running process.

## Capitalization Standards for SAS Programming Language

In running text that includes SAS programming language, adhere to the following casing styles:

- Use all-caps for language elements, arguments, and values that are literals, like so: the NAME= option in the MATCH statement.
- Capitalize user-supplied values that consist of only one word, like so: the *Salary* column and the *Fitness* data set.
- Capitalize user-supplied values that consist of conjoined words, like so: the *MyData* table and the *Sasuser.Houses* data set.
- Capitalize the names of the Sasuser, Sashelp, and Work libraries.
- If you must begin a sentence with the name of an argument, parameter, or user-supplied value, capitalize the name, like so: *Array-name* must be a SAS name.
- If a sentence begins with a lowercase <charvar-value>, or any other term that users must enter in lowercase exactly as shown, revise the sentence to avoid capitalizing the first letter of the word. When such words are referred to as words, set them in boldface, like so: The values **equipment** and **other** appear on the first page.

In displayed programming code, adhere to these same styles, generally lowercasing everything except user-supplied values and using all-caps for literals. For multiword language element names in displayed code, you might need to add capitalization to make some elements readable. The following display, for example, includes the following:

- traditional one-word names
- multiword names that are easy to read in either all caps or lowercase
  - multiword names that would be difficult to read if case weren't changed

```

filename XmlInp 'an_ODM_to_import.xml';

proc cdisc model=odm
read=XmlInp
formatactive=yes
formatnoreplace=no
;
odm ODMVersion="1.2"
ODMMMaximumOIDLength=16
ODMMMinimumKeyset=no
;
clinicaldata out=Current.MyAe
sasdatasetname="AE"
;
run;

```

- Do not use Tab key to indent code; indent by using either spaces or the “indentcode” style provided in your style template.
- Be consistent with indentures, spacing, and use of special fonts (such as bold or italic).
- Do not use line numbers.

## Lists

If you need to present a series of any kind, consider introducing a vertical list. Cognitive testing has found that vertical lists, as opposed to horizontal lists run in with text paragraphs, increase reading comprehension and retention. If your list is a series of steps that must be completed in order, use a numbered vertical list; otherwise, use a bulleted list. Use a vertical list under the following circumstances:

- you have at least three items to list
- the items are not suited to tabular presentation (the table would contain only the stub and one other column)
- the items are—or can be edited to be—grammatically parallel (all gerunds, all clauses, or all imperative sentences, and so forth)

## Formatting and Style

**Be sure to apply the numbered list or bulleted list formatting style *only from the Styles pane that contains all styles in the book template that your developmental editor has supplied you*.** Do not enter bullets or numbers from the Paragraph menu on the HOME tab of the Microsoft Word Ribbon.

As you write your lists, observe the following points:

- Introduce your list with a grammatically complete sentence that expresses a complete thought.
- If your introductory sentence explicitly points to the list that follows, then incorporate the phrase *as follows* or *the following* to ensure that you have a complete sentence.
- Punctuate the introductory sentence with a colon (:).
- Make all list items grammatically parallel: Use either all complete sentences, or all incomplete sentences (dependent clauses or phrases).
- When listing words, phrases, or clauses rather than sentences, lowercase the initial letter of each list item and omit all end punctuation.

**Example 1: Bullets as Grammatically Parallel Phrases****In this chapter you will learn how to do the following:**

- use the SAS programming language
- read the SAS log
- view the properties of data sets

**Example 2: Bullets as Complete Sentences****From this chapter you will draw several conclusions:**

- In assessing personal risk, human beings tend to ignore probability. This tendency crosses cultures.
- Households update risk perceptions systematically in response to new information.
- Human beings tend to offer others statistically sound risk assessments.

**Equations and Other Mathematical Expressions**

As you create displayed equations and any in-line mathematical expressions, remember that SAS Press copyeditors will not be editing your equations. Consequently, you will need to double-check for both content accuracy and consistent styling all of your math expressions. For consistency, observe the following:

- To construct your mathematical expressions, use the Equation editor in Word.
- If the Equation editor alters the line spacing in running text (this problem will be obvious), then one remedy is simply to display the expression on its own line, as you would any full displayed equation.
- If the Equation editor alters the line spacing in running text, then an even better remedy is simply to change the display in the Equation editor from “Professional” to “Linear”:
 
$$(1 + x)^n = 1 + \frac{nx}{1!} + \frac{n(n-1)x^2}{2!} + \dots$$

[PROFESSIONAL]

$$(1 + x)^n = 1 + nx/1! + (n(n - 1) x^2)/2! + \dots$$

[LINEAR]
- Center displayed equations by applying the style AEquationCentered from the template.
- Do not number a displayed equation unless the reader must be referred to it by number in later text. Apply the style AEquationWithParenNumbers to any equations that must be numbered.
- Italicize all Latin single-character variables; otherwise, set variables roman (nonitalic).
- Set all Greek symbols roman (nonitalic).
- Omit all punctuation from displayed equations, but do add a conjunction, or other verbiage, in running text after each equation, on the next text line, to complete sentences properly.
- Don’t stack one displayed equation directly after another unless you have set them up as a vertical list, with a complete introductory sentence and a colon.
- Style significance, correlation coefficients, and any other value that cannot equal unity (1) without a leading zero, like so:  $p = .5$ . Otherwise, use the leading zero.
- Express significance as a  $p$  value, like so:  $p > .1$ .
- For additional style points governing statistics, see *The Chicago Manual of Style*, 16th edition, section 12.57. Above all, maintain consistency in symbols, abbreviations, and style adopted.

**Hyperlinks**

When you need to refer to an outside link, use URLs and the “AHyperlink” style in the template. Please avoid linking to content that will not be likely to persist on the web: i.e., instead of social media, messages, forums, or discussion boards, choose SAS websites, government websites, corporate, etc. If you have a question about a source, please ask your developmental editor about procuring “go.sas.com” links for potentially unstable web material.

---

## Notes

Use notes sparingly to provide the reader extra information that would be disruptive or glaringly tangential if included in running text. Do not use notes to give source references, unless the note begins with content related to or commenting on the source, in which case use attribution and in-text citation and include the full documentation in your References list.

Do not use footnotes; instead, use numbered endnotes. Either place your endnotes at ends of their respective chapters, or run all notes as a single list titled *Notes* in the back matter of the book.

---

## References

---

### In-Text Citations

If you quote, paraphrase, or summarize content from another work—even if it is your own work—clearly attribute the information to its source at the point in the text where you cite from it. Do so by adding an introductory clause that signals the beginning of the borrowed material, like so: “According to Lora D. Delwiche and Susan J. Slaughter in *The Little SAS Book: A Primer*. . . .” Signal the end of the borrowed material with an in-text parenthetical citation, like so: (Delwiche and Slaughter 2012, pp. 26–28).

To send your readers to works other than your book, use attribution and style the in-text reference like so: “For details about the PRINT procedure, see the *Base SAS Procedures Guide* (Smith 2008, 29–72).” If the work has no listed author or is a SAS portal, include only page or other location numbers in the parenthetical citation.

---

### End-of-Book References List

References that are cited in text must be fully documented at the end of your book in an alphabetically ordered list titled *References*. It is solely your responsibility to provide complete and accurate references.

Following are examples conforming to *The Chicago Manual of Style*, 16th edition, for various publication types that commonly appear in a SAS author’s References list. If a book has only one edition, has no translator or editor on the title page, and is a single volume, simply omit these elements from the reference. For more examples, please follow The Chicago Manual of Style’s citation guide:  
[http://www.chicagomanualofstyle.org/tools\\_citationguide.html](http://www.chicagomanualofstyle.org/tools_citationguide.html)

### Example: Book

Aster, Rick, John Smith, and Rhena Seidman. 1992. *Professional SAS® User Interfaces*. Edited by Ray Martin. 3rd ed. 2 vols. Blue Ridge Summit, PA: Windcrest.

SAS Institute Inc. 1989. *SAS/FSP Software: Usage and Reference, Version 6*. 2nd ed. Cary, NC: SAS Institute Inc.

### Example: Book Chapter or Published Conference Proceedings

Barrow, John. 1992. “Overview of the SAS Interface.” In *Solutions for Statistical Inquiries*, edited by Lee Wolfson, 307–11. New York: Macmillan.

Batkhan, Leonid. 2006. “Document-Driven Techniques for Building SAS® Data Warehouses.” In *Proceedings of the Thirty-first Annual SAS® Users Group International Conference*. Cary, NC: SAS Institute Inc. <http://www2.sas.com/proceedings/sugi31/033-31.pdf>.

### Example: Periodical

Cleveland, W. S., and R. McGill. 1984. “Graphical Perception: Theory, Experimentation, and Application to the Development of Graphical Methods.” *Journal of the American Statistical Association* 79 (2):531–54.

Foerst, Jenny Jennings. 2013. “How to Best the Beast Within.” A Review of *The Book of Barely Imagined Beings: A 21st Century Bestiary* (book), by Casper Henderson. *American Scientist* 101 (November–December): 522–23, <http://www.americanscientist.org/bookshelf/pub/how-to-best-the-beast-within>.

### **Example: Web Page**

SAS Institute Inc. 2002–2006. SAS OnlineDoc® 9.1.3. SAS Knowledge Base / Product Documentation. <http://support.sas.com/onlinedoc/913/docMainpage.jsp>.

---

### **Cross-References**

If a chapter-length document is organized well, then cross-references to other parts of the same chapter are ordinarily unnecessary. The reader can also use the chapter table of contents or index to locate topics. If you refer your reader to another part of the same chapter or to another chapter in your book, avoid using “above” or “below” to send the reader backtracking or skipping ahead for the content you have in mind. Instead, style your cross-references as follows:

- Style cross-references to a section within the same chapter like so: For more details, see the section “Data Set Options by Category.”
  - Style cross-references to another chapter in your book like so: For further information, see Chapter 4, “Data Set Options by Category.”
- 

### **Index**

SAS Press will arrange for the indexing of your book once your book is in the galley proof stage.