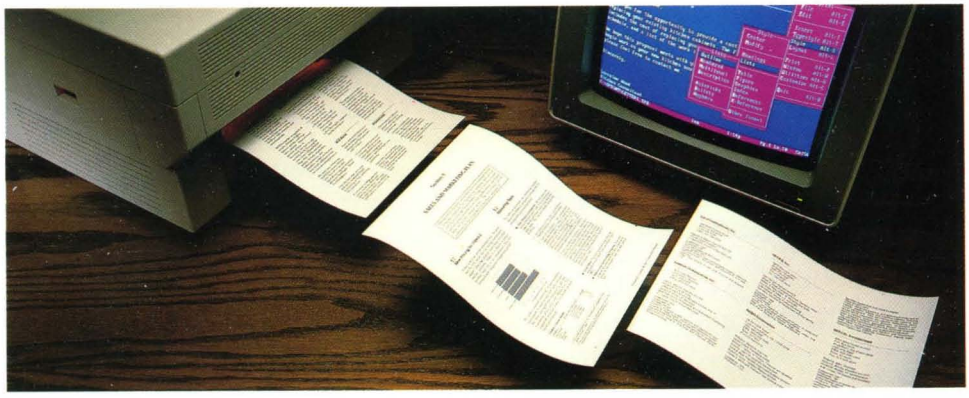
# SPRINT®

THE PROFESSIONAL WORD PROCESSOR

## REFERENCE GUIDE



BORLAND
INTERNATIONAL

# SPRINT®

The Professional Word Processor

## Borland's No-Nonsense License Statement!

This software is protected by both United States copyright law and international treaty provisions. Therefore, you must treat this software *just like a book*, with the following single exception. Borland International authorizes you to make archival copies of the software for the sole purpose of backing-up our software and protecting your investment from loss.

By saying, "just like a book," Borland means, for example, that this software may be used by any number of people and may be freely moved from one computer location to another, so long as there is **no possibility** of it being used at one location while it's being used at another. Just like a book that can't be read by two different people in two different places at the same time, neither can the software be used by two different people in two different places at the same time. (Unless, of course, Borland's copyright has been violated).

## LIMITED WARRANTY

With respect to the physical diskette and physical documentation enclosed herein, Borland International, Inc. ("Borland") warrants the same to be free of defects in materials and workmanship for a period of 60 days from the date of purchase. In the event of notification within the warranty period of defects in material or workmanship, Borland will replace the defective diskette or documentation. **If you need to return a product, call the Borland Customer Service Department to obtain a return authorization number.** The remedy for breach of this warranty shall be limited to replacement and shall not encompass any other damages, including but not limited to loss of profit, and special, incidental, consequential, or other similar claims.

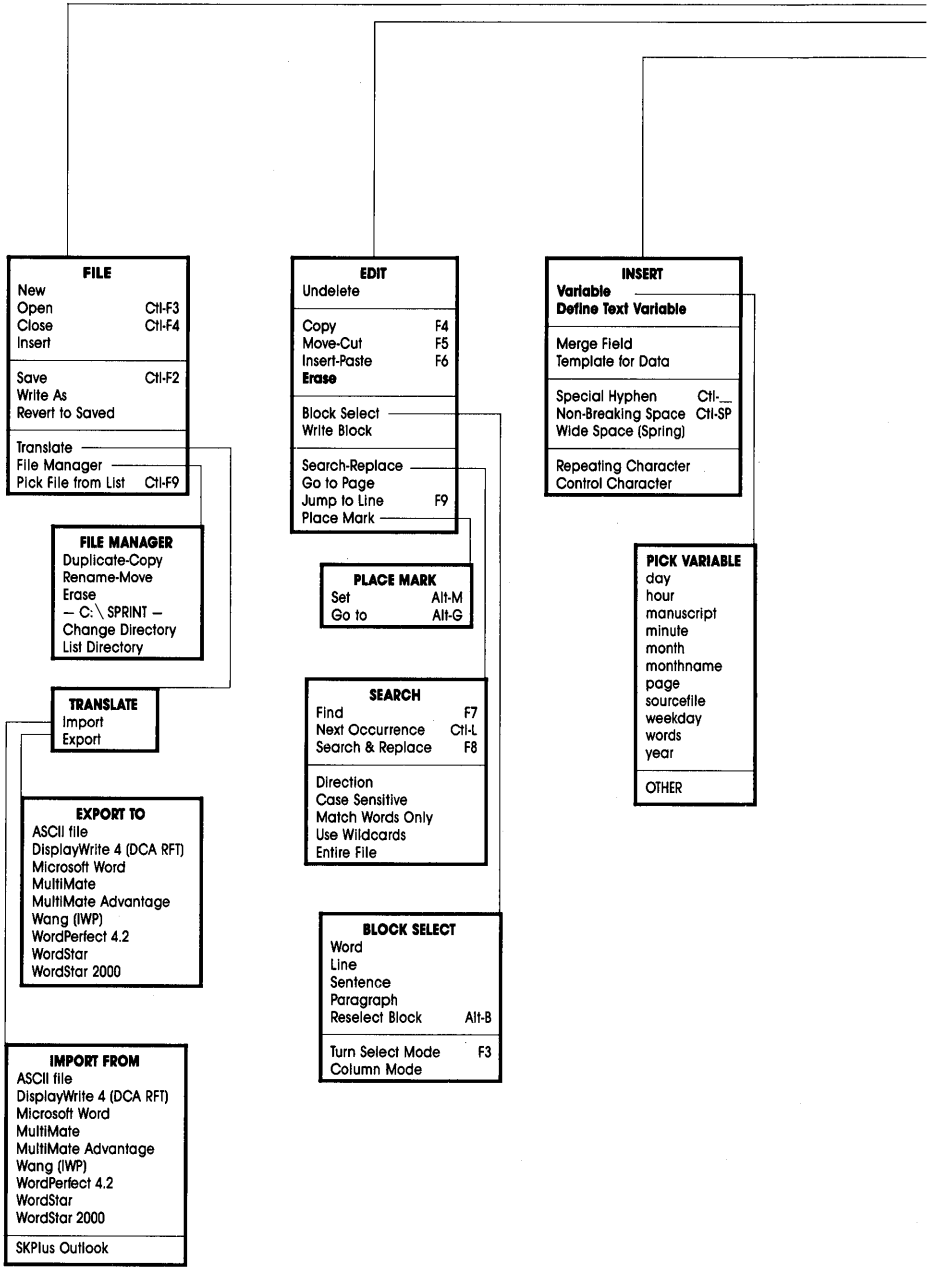
Borland International, Inc. specifically **disclaims** all other warranties, expressed or implied, including but not limited to implied warranties of merchantability and fitness for a particular purpose with respect to defects in the diskette and documentation, and the program license granted herein in particular, and without limiting operation of the program license with respect to any particular application, use, or purpose. **In no event shall Borland be liable for any loss of profit or any other commercial damage, including but not limited to special, incidental, consequential or other damages.**

## GOVERNING LAW

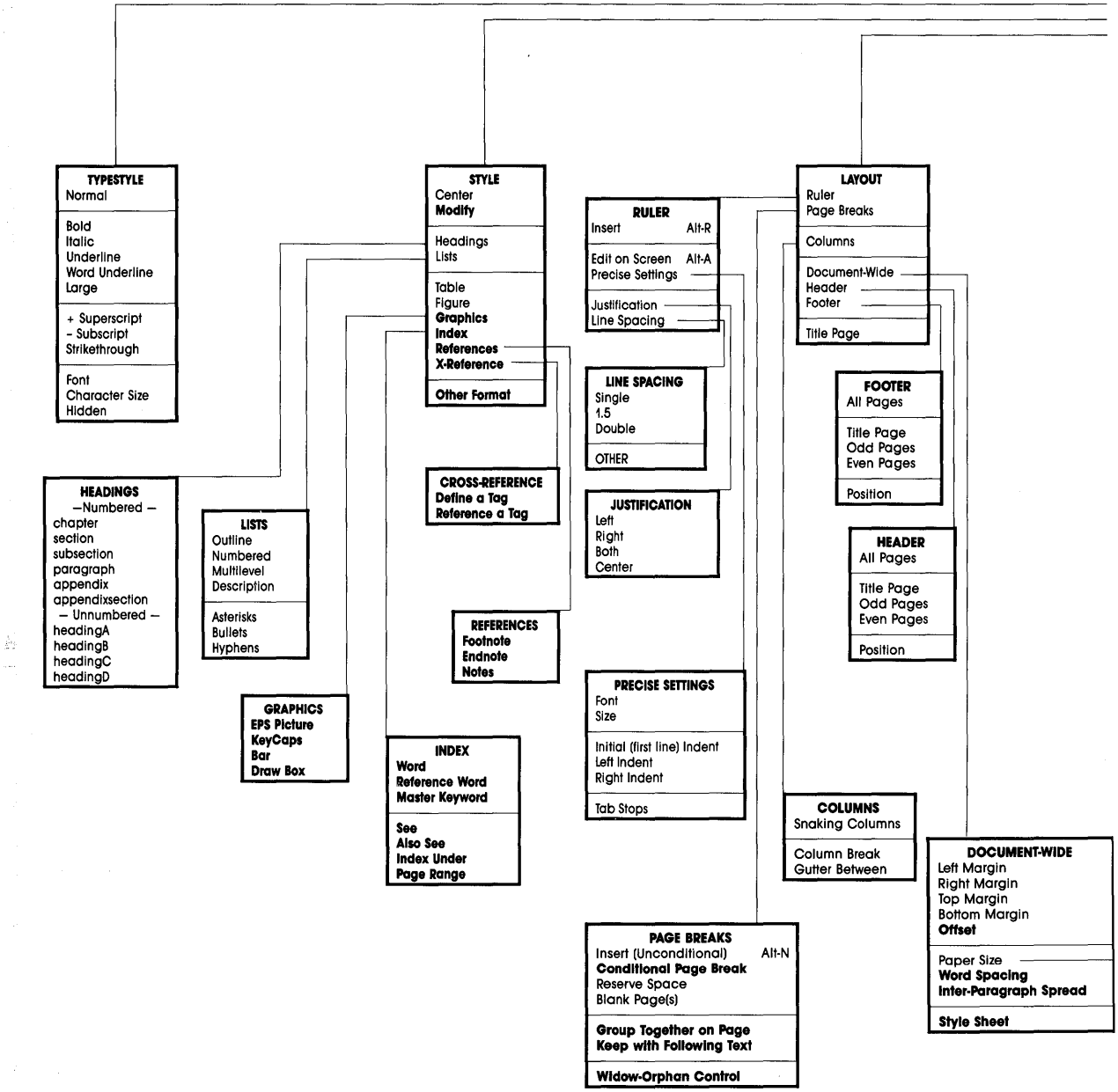
This statement shall be construed, interpreted, and governed by the laws of the state of California.

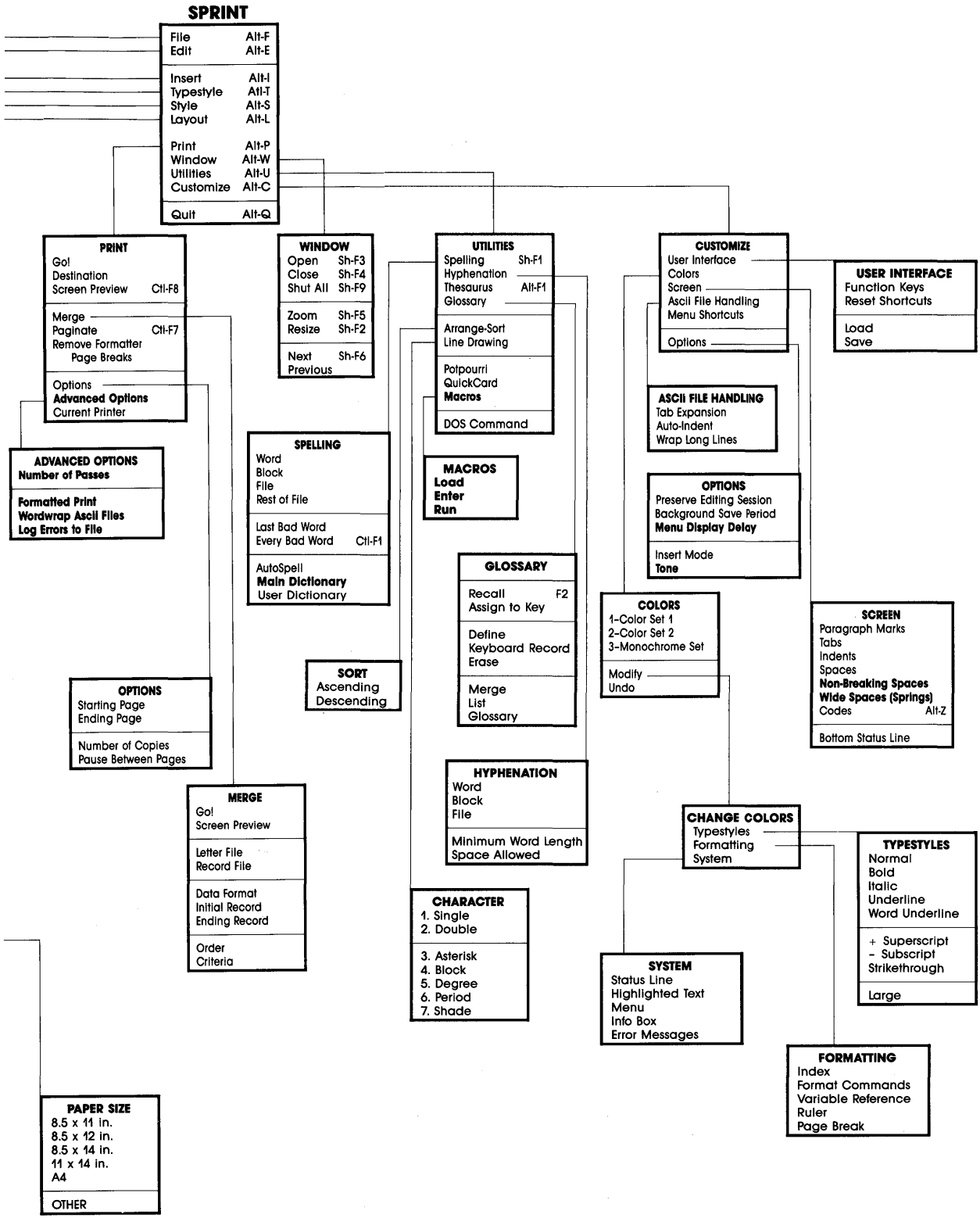
First Edition
Printed in U.S.A.
9 8 7 6 5 4 3 2

**NOTE:** Boldface items are available in Advanced (SPADV.UI) user interface only.



# SPRINT MENU TREE (with Keyboard Shortcuts)




SPRINT
File Alt-F
Edit Alt-E
Insert Alt-I
Typestyle Alt-T
Style Alt-S
Layout Alt-L
Print Alt-P
Window Alt-W
Utilities Alt-U
Customize Alt-C
Quit Alt-Q
PRINT
Go!
Destination
Screen Preview Ctl-F8
Merge
Paginate Ctl-F7
Remove Formatter
Page Breaks
Options
Advanced Options
Current Printer
ADVANCED OPTIONS
Number of Passes
Formatted Print
Wordwrap Ascii Files
Log Errors to File
OPTIONS
Starting Page
Ending Page
Number of Copies
Pause Between Pages
MERGE
Go!
Screen Preview
Letter File
Record File
Data Format
Initial Record
Ending Record
Order
Criteria
PAPER SIZE
8.5 x 11 in.
8.5 x 12 in.
8.5 x 14 in.
11 x 14 in.
A4
OTHER
WINDOW
Open Sh-F3
Close Sh-F4
Shut All Sh-F9
Zoom Sh-F5
Resize Sh-F2
Next Sh-F6
Previous
SPELLING
Word
Block
File
Rest of File
Last Bad Word
Every Bad Word Ctl-F1
AutoSpell
Main Dictionary
User Dictionary
SORT
Ascending
Descending
UTILITIES
Spelling Sh-F1
Hyphenation
Thesaurus Alt-F1
Glossary
Arrange-Sort
Line Drawing
Potpourri
QuickCard
Macros
DOS Command
MACROS
Load
Enter
Run
GLOSSARY
Recall F2
Assign to Key
Define
Keyboard Record
Erase
Merge
List
Glossary
HYPHENATION
Word
Block
File
Minimum Word Length
Space Allowed
CHARACTER
1. Single
2. Double
3. Asterisk
4. Block
5. Degree
6. Period
7. Shade
CUSTOMIZE
User Interface
Colors
Screen
Ascii File Handling
Menu Shortcuts
Options
USER INTERFACE
Function Keys
Reset Shortcuts
Load
Save
ASCII FILE HANDLING
Tab Expansion
Auto-Indent
Wrap Long Lines
OPTIONS
Preserve Editing Session
Background Save Period
Menu Display Delay
Insert Mode
Tone
COLORS
1-Color Set 1
2-Color Set 2
3-Monochrome Set
Modify
Undo
SCREEN
Paragraph Marks
Tabs
Indents
Spaces
Non-Breaking Spaces
Wide Spaces (Springs)
Codes Alt-Z
Bottom Status Line
CHANGE COLORS
Typestyles
Formatting
System
TYPESTYLES
Normal
Bold
Italic
Underline
Word Underline
+ Superscript
- Subscript
Strikethrough
Large
SYSTEM
Status Line
Highlighted Text
Menu
Info Box
Error Messages
FORMATTING
Index
Format Commands
Variable Reference
Ruler
Page Break

# SPRINT® The Professional Word Processor

# Reference Guide

Copyright ©1988
All rights reserved

Borland International
4585 Scotts Valley Drive
P.O. Box 660001
Scotts Valley, CA 95066-0001

All Borland products are trademarks or registered trademarks of Borland International, Inc. or Borland/Analytica, Inc. Other brand and product names are trademarks or registered trademarks of their respective holders. Copyright ©1988 Borland International.

Printed in the U.S.A.

10 9 8 7 6 5 4 3 2

This manual was produced with
Sprint:® The Professional Word Processor

# Table of Contents

**Introduction** 1
What's in This Manual ..... 1
Typographic Conventions ..... 2
The Distribution Disks ..... 2
Hardware and Software Requirements ..... 3
Borland's No-Nonsense License Statement ..... 3
How to Contact Borland ..... 4

**Chapter 1 The Sprint Menu Encyclopedia** 5
Overview ..... 5
Encyclopedia Functional Index ..... 5
Getting Around ..... 5
File Maintenance ..... 6
Editing ..... 6
Customization ..... 6
Creating Large Documents ..... 6
Basic Formatting ..... 7
Special Formats ..... 7
Printing ..... 8
Miscellaneous Concepts ..... 8
How the Encyclopedia Is Organized ..... 8

Advanced Options ..... 10
Appendix ..... 12
AppendixSection ..... 13
Arrange-Sort ..... 14
ASCII Files (General Information) ..... 15
Auto-Indent ..... 18
AutoSpell ..... 19
Background Save Period ..... 20
Blank Page(s) ..... 21
Blank Space (Horizontal) ..... 22
Block Select Menu ..... 23
Bulleted Lists ..... 26
Canceling ..... 29
Caption ..... 30
Centering Text ..... 31
Chapter ..... 33

Clipboard ..... 34
Close (File) ..... 35
Codes ..... 37
Colors ..... 38
Columns ..... 39
Comments ..... 42
Conditional Page Break ..... 43
Control Characters (General Information) ..... 44
Copy ..... 47
Current Printer ..... 48
Customize Menu ..... 49
Define Text Variable ..... 50
Delete a Block ..... 51
Deleting (General Information) ..... 52
Description ..... 54
Dimensions ..... 56
Document-Wide Menu ..... 59
DOS Command ..... 62
Edit Menu ..... 64
Endnote ..... 66
Erase ..... 68
Error Messages and Warning Messages ..... 69
Figure ..... 71
File Manager Menu ..... 73
File Menu ..... 75
Font ..... 77
Font Size ..... 79
Footer Menu ..... 81
Footnote ..... 84
Formats (General Information) ..... 87
Function Keys ..... 88
Glossary ..... 91
Graphics ..... 95
Group Together on Page ..... 97
Hard and Soft Returns ..... 98
Header Menu ..... 100
HeadingA ..... 103
HeadingB ..... 104
HeadingC ..... 105
HeadingD ..... 106
Headings Menu ..... 107
Help ..... 108
Hyphenation Menu ..... 108
Indenting ..... 110

Index ..... 112
Insert (File) ..... 118
Insert Menu ..... 118
Insert Mode ..... 120
Insert-Paste ..... 120
Insert (Unconditional) Page Break ..... 121
Justification ..... 123
Keep with Following Text ..... 125
Layout Menu ..... 126
Line Drawing ..... 128
Line Spacing ..... 131
Lists ..... 133
Macros Menu ..... 136
Margins (General Information) ..... 137
Menu Shortcuts ..... 141
Merge ..... 142
Modifying Formats ..... 145
Mouse Commands ..... 155
Move-Cut ..... 157
Multilevel ..... 158
Non-Breaking Space ..... 160
Numbered ..... 162
Open (File) ..... 165
Options Menu (Customize) ..... 167
Other Format ..... 168
Outline ..... 169
Page Breaks ..... 171
Page Numbers (General Information) ..... 172
Paginate ..... 176
Paragraph ..... 177
Paragraphs (General Information) ..... 179
Pick from List ..... 181
Place Mark ..... 181
Potpourri ..... 183
Precise Settings Menu ..... 185
Preserve Editing Session ..... 186
Print Menu ..... 188
QuickCard ..... 191
Quit ..... 191
References Menu ..... 193
Remove Formatter Page Breaks ..... 193
Repeating Character ..... 194
Reserve Space ..... 195
Revert to Saved ..... 197

Ruler ..... 198
Save ..... 205
Screen Menu ..... 206
Scrolling ..... 210
Searching ..... 211
Section ..... 217
Special Hyphen ..... 221
Spelling Menu ..... 222
Status Line ..... 225
Style Menu ..... 226
Subsection ..... 228
Tab Expansion ..... 229
Table ..... 230
Tabs (General Information) ..... 233
Tab Stops ..... 235
Tags ..... 235
TCapt and TCaption ..... 238
Thesaurus ..... 240
Title Page ..... 241
Tone ..... 241
Translate ..... 242
Typestyle Menu ..... 244
Undelete ..... 247
User Interface Menu ..... 248
Utilities Menu ..... 252
Variables (General Information) ..... 252
Wide Space (Spring) ..... 257
Window ..... 259
WordStar (User Interface) ..... 263
Wordwrapping ..... 264
Write As ..... 264
Write Block ..... 265

**Chapter 2 Miscellaneous Formats** 267
How to Enter Non-Menu Formats ..... 267
Address ..... 269
Begin and End (Formats) ..... 270
Case ..... 271
Char ..... 276
Closing ..... 278
Column ..... 279
Display ..... 281
Escape ..... 282
Example ..... 285
HaveSpace ..... 286

Hsp ..... 287
Include ..... 288
Incr ..... 290
Justify ..... 291
Kern ..... 292
Label ..... 293
MakeTOC ..... 296
Message ..... 296
Modify ..... 297
NeedSpace ..... 299
NoteChapter ..... 300
NoteSection ..... 300
O and Ovp ..... 301
Place ..... 302
Quotation ..... 303
Scale ..... 304
Set ..... 304
StringInput ..... 308
Style ..... 309
Tab ..... 311
TabDivide ..... 311
TagString ..... 312
TCT/NoTCT ..... 313
Template ..... 315
Text ..... 320
Timestamp ..... 321
Title ..... 322
Undent ..... 322
Verbatim ..... 324
Word ..... 325
! ..... 326
< ..... 327

**Chapter 3 @-Sign Commands** 329
How to Enter @ Commands ..... 329
Delimiters ..... 331
Nesting @ Sign Commands ..... 331
Nesting Delimiters ..... 332
When to Use @-Sign Commands ..... 334
@-Signs vs. Menus ..... 334
@-Command Tables ..... 336
@-Sign Syntax ..... 340
A Note to FinalWord Users ..... 342

**Chapter 4 Running Sprint from DOS** 343
Running Sprint ..... 343
Printing from DOS ..... 344
Command-Line Options ..... 345
–# (Number of Copies) ..... 346
–dname{=value} (Defining Variables) ..... 346
–f (Fill Paragraphs) ..... 346
–f=name (Use Alternative Font) ..... 347
–l (Create .LOG File) ..... 347
–o{=name} (Send Output to a File) ..... 348
–p (Print Unformatted) ..... 348
–p# (Number of Passes) ..... 348
–p=name (Using Another Printer) ..... 349
–page=# (Prints a Single Page) ..... 350
–pause (Pause between Pages) ..... 350
–start=# (Starting Page #) ..... 351
–stop=# (Ending Page #) ..... 351
–s{=name} (Use Alternative Screen) ..... 351
–t=# (Set Tab Stops) ..... 352
–v (Preview File) ..... 352
Merging Files from DOS ..... 352

**Appendix A** Formatter Error Messages 355

**Appendix B** ASCII Character Set 367

**Index** 370

# List of Figures

Figure 1.1: Sprint's Line-Drawing Characters .....130
Figure 1.2: Sprint's Special Screen Characters .....209
Figure 4.1: Sprint SPFMT Command-Line Options .....345
Figure 4.2: SprintMerge Command-Line Options .....354

# List of Tables

Table 1.1: Control Codes (Standard User Interface) ..... 46
Table 1.2: Formatting Dimensions ..... 58
Table 1.3: Function Key Assignments (Standard User Interface) ..... 90
Table 1.4: The SprintMerge Program Commands ..... 144
Table 1.5: Parameters Used Only with Formats Affecting Regions ..... 148
Table 1.6: Parameters Used Only with Style Commands ..... 148
Table 1.7: Parameters Used Anywhere ..... 148
Table 1.8: Format Parameters (Complete List) ..... 149
Table 1.9: Potpourri Menu Commands ..... 183
Table 1.10: Typestyle Commands ..... 246
Table 1.11: Built-in Formatting Variables ..... 253
Table 1.12: Variables Defined in STANDARD.FMT ..... 256
Table 3.1: @-Commands and Menu Equivalents ..... 336
Table 3.2: Menu Commands and @-Command Equivalents ..... 339
Table 3.3: Parameters and Menu Equivalents ..... 340

# I N T R O D U C T I O N

This book, the companion volume to the Sprint *User's Guide* and *Advanced User's Guide,* is a complete reference manual for Sprint: The Professional Word Processor.

You'll want to refer to this book for complete, no-nonsense information about Sprint. The bulk of this manual is made up of the Sprint "Menu Encyclopedia"—an alphabetical listing of all Sprint menu commands and concepts.

To become familiar with the program, you'll probably first want to work through the *Getting Started* section of the *User's Guide.* That volume contains tutorials and how-to information about the Sprint editor and formatter. The *Advanced User's Guide* discusses editor macros, large-document formatting, and customizing commands.

## What's in This Manual

This *Reference Guide* contains the following chapters and appendixes:

Chapter 1, "The Sprint Menu Encyclopedia," comprises the bulk of the book and contains an alphabetical list of all the Sprint menu commands as well as general information entries on Sprint concepts.

Chapter 2, "Miscellaneous Formats," is a separate alphabetical listing of all the commands accessible through the **Style/Other Format** menu command.

Chapter 3, "Using @-Commands," explains why and when you would use the @-sign commands in Sprint. The chapter contains a list and descriptions of commands you would use exclusively when modifying format (.FMT) files. This chapter also contains a list of @-sign equivalents for menu commands.

Chapter 4, "Using Sprint from DOS," explains how you can use the Sprint programs from the DOS command line.

Appendix A, "Error Messages," lists and explains all Sprint error messages.

Appendix B, "ASCII Character Set," is a table of the standard ASCII characters with decimal equivalents.

# Typographic Conventions

All typefaces and styles used in this manual were produced by Sprint, and output on a PostScript printer. Their uses are as follows:

`Monospace type` — This typeface represents text as it appears on the screen and anything you must type. For example, "At the Sprint prompt, type `2` for the number of columns desired."

**`Bold Monospace`** — This typeface represents highlighted text that Sprint inserts on the screen, for example, "**`BEGIN FNOTE`**`This is a footnote.`**`END FNOTE`**."

*Italic* — Italic type is used to introduce a new term or to emphasize a word. We also use italic to represent variables and parameters, as opposed to commands. For example, "Use the *Spread* parameter to modify the distance between paragraphs."

*Keycaps* — This special typeface indicates a key on your keyboard. It is often used when describing a particular key you should press, for example, "Press *Esc* to cancel a menu."

# The Distribution Disks

Your Sprint disks contain many files. Some of the more important ones are described here. More information about the files you need to run Sprint can be found in the *User's Guide*. Also be sure to look at the README file on your distribution disk, which contains last-minute information about your Sprint disks and files.

- SP.EXE, the main Sprint program.
- SP-SETUP.COM is the Sprint installation program that you use to select your printer, computer screen, and user interface.
- SP.SWP, the so-called swap file, is the file that holds your current work session and provides a backup in case of system creashed or power failure.
- SP.SPM, Borland's default user interface.
- STANDARD.FMT, is the default style sheet, that is, a text file that contains a number of formatter commands. You can modify or copy any of the commands in this file, but always keep an uncorrupted backup copy.
- SPRECOVE.COM is the program Sprint uses to recover a swap file.
- Files with the .SPM extension are macro files that you can load into Sprint. What the macro file does can vary widely. Sprint has an extensive macro programming language built in.
- Files with the .SPL extension are printer drivers that Sprint needs to work with different printers.

# Hardware and Software Requirements

Sprint runs on the IBM PC family of computers, including the XT and AT, along with true IBM compatibles.

Sprint requires

- DOS 2.0 or higher
- at least 256K of RAM

The Sprint program is in the file called SP.EXE. It is not copy-protected, so you can easily transfer it to a hard disk or RAM disk. You should, however, read Borland's No-Nonsense License Agreement (which follows) for an explanation of your responsibilities with respect to copying Sprint. Also be sure to read your warranty card, and then sign it and mail it to Borland.

# Borland's No-Nonsense License Statement

This software is protected by both United States Copyright Law and International Treaty provisions. Therefore, you must treat this software *just like a book* with the following single exception: Borland International authorizes you to make archival copies of Sprint for the sole purpose of backing up your software and protecting your investment from loss.

By saying, "just like a book," Borland means, for example, that this software may be used by any number of people and may be freely moved from one computer location to another so long as there is *no possibility* of its being used at one location while it's being used at another. Just like a book that can't be read by two different people in two different places at the same time, neither can the software be used by two different people in two different places at the same time. (Unless, of course, Borland's copyright has been violated.)

# How to Contact Borland

The best way to contact Borland is to log on to Borland's Forum on CompuServe: Type `GO BORAPP` from the main CompuServe menu and select "Enter Business Products Forum" from the Borland main menu. Leave your questions or comments there for the support staff to process.

If you prefer, write a letter detailing your comments and send it to:

Technical Support Department
Borland International
P.O. Box 660001, 4585 Scotts Valley Dr.
Scotts Valley, CA
95066-0001, USA

You can also telephone our Technical Support department (at 408-438-5300). Please have the following information handy before you call:

- version number and serial number
- computer make and model number
- operating system and version number
- the user interface you're using

C H A P T E R

# 1

# The Sprint Menu Encyclopedia

## Overview

This chapter contains a comprehensive, alphabetical listing of Sprint concepts, menus, and commands. Commands not found in the menus are in Chapter 2; @-sign commands are in Chapter 3.

## Encyclopedia Functional Index

Even though the entries in this chapter are in alphabetical order, we also include a "functional index," which presents almost every entry in the encyclopedia organized by category. Using this list, you can easily scan the possible entries for a particular topic or task, and then look them up alphabetically.

And if you don't find what you're looking for, don't forget to check the index.

### *Getting Around*

Canceling
DOS Command
Go to Page
Help
Jump to Line
Menu Shortcuts
Mouse Commands
Quit
Scrolling
Status Line
Window

## *File Maintenance*

ASCII Files
Background Save Period
Canceling
File Menu
  Close a File
  File Manager
  Insert
  Open
  Pick from List
  Revert to Saved
  Save
  Translate
  Write As
Preserve Editing Session

## *Editing*

Case
Clipboard
Deleting
Edit Menu
  Copy
  Go to Page
  Jump to Line
  Place Marker
  Searching
Modifying Formats
Insert Menu
  Control Character
  Special Hyphen
Spelling Menu
  AutoSpell
  Hyphenation
  Thesaurus

**Block Commands**
  Block Select Menu
  Deleting
  Erase
  Insert-Paste
  Move-Cut
  Undelete
  Write Block

## *Customization*

Customize Menu
  Colors
  Menu Shortcuts
  Options Menu
    Background Save Period
    Insert Mode
    Preserve Editing Session
    Tone
  Screen
    Codes
    Non-Breaking Spaces
    Wide Spaces (Springs)
  User Interface
    Function Keys
    Reset Shortcuts
    WordPerfect User Interface
    WordStar User Interface
Macros
Utilities Menu
  Glossary
  Potpourri
  QuickCard

## *Creating Large Documents*

Case
Conditional Page Break
Document-Wide Menu
Empty Page(s)
Footer
Header

Headings Menu
  Appendix
  AppendixSection
  Chapter
  HeadingA
  HeadingB
  HeadingC
  HeadingD
  Paragraph
  Section
  Subsection
Insert (Unconditional) Page Break
Other Format
Page Breaks
Page Numbers
Paginate
Style Menu
Title Page

**Text References**
  Define Text Variable
  Index
  Index Under
  References Menu
    Endnote
    Footnote
  Tags

## *Basic Formatting*

Auto-Indent
Centering Text
Formats (General Information)
Group Together on Page
Hard and Soft Returns
Hyphenation Menu
Indenting
Justification
Keep with Following Text
Layout Menu
  Footer
  Header
  Page Breaks
  Ruler
    Line Spacing
    Precise Settings Menu
  Title Page
Margins (General Information)
Paragraphs (General Information)
Tabs (General Information)
Tab Stops
Typestyle
Wordwrapping

## *Special Formats*

Begin and End (Formats)
Blank Space (Horizontal)
Caption
Columns
  Snaking Columns
Conditional Page Break
Description
Dimensions
Figure
Font
Font Size
Large
Lists
  Bulleted
  Multilevel
  Numbered
  Outline
Other Format
Place
Repeat Character
Reserve Space
Table
TCapt and TCaption
Title Page
Font Size

*Printing*

Conditional Page Break
Error Messages and
    Warning Messages
Graphics
Line Drawing
Nonprinting Text
Print Menu
  Advanced Options
  Current Printer
  Merge
  Paginate
  Remove Formatter Page
    Breaks

*Miscellaneous Concepts*

ASCII Files
Clipboard
Control Characters
Deleting
Formats
Merge
Mouse Commands
Status Line
Variables

# How the Encyclopedia Is Organized

All entries in this menu encyclopedia follow the same general format, sketched out as follows:

# Command or Menu Name

Keystrokes

This section lists the keystrokes required to enter a command. Since there is often more than one way to enter a command, we list all methods in this section. You will see one or more of these entries here:

- A list of the menu options required to perform the command. (All menus listed in this reference guide are for the standard advanced user interface as defined in the file SPADV.UI.)
- A list of the *shortcuts* (control or function keys) that also perform the command. (All shortcuts listed in this menu encyclopedia are for the normal Sprint user interfaces (as defined in the files SPTUTOR.UI, SPBASIC.UI, and SPADV.UI). If you've selected an alternative user interface, such as Sprint's WordStar or WordPerfect, these keystrokes may not accomplish the same function.)

| | |
|---|---|
| Function | A brief explanation of what the entry does, followed by a more detailed explanation of the entry. If the item listed is a menu, this section also lists the commands available from the menu and briefly explains the function of each command. |
| How To | Examples and explanatory text about how to perform the command or function. Examples often include the printed results. |
| Tips | A catch-all that explains various other ways to use the command, technical notes, or things to consider when using this command to format your text. If the command is defined in STANDARD.FMT, this fact is mentioned here. (Any file defined in STANDARD.FMT is modifiable by editing its definition in that file.) |
| See Also | A cross-reference to other entries in this chapter or in other chapters. |

This chapter serves as a reference guide for all but the most advanced commands and provides specific information about Sprint functions, menus, and concepts. If you need more background information, such as how to move the cursor or how to choose a command from the menus, refer to the Sprint *User's Guide.*

The "Menu Encyclopedia" is arranged alphabetically. To save space, the explanations of many entries have been grouped. Every menu command has an alphabetical entry, however, which either goes into detail about the command or refers you to the entry that does.

If you don't find the command you're looking for, be sure to check the functional index on page 5, and don't forget the index at the end of this manual, which will lead you to any Sprint term that does not have its own section. (For example, individual parameter or variable names are not in the alphabetical section but are grouped under the entries for Modifying Formats or Variables, respectively.)

# Advanced Options

Keystrokes *Alt-P* (or *F10*, Print), Advanced Options

Function Displays a menu of "special-case" print options.

The Advanced Options menu displays the following list of options:

**Number of Passes**
This command tells the formatter how many "passes" through your document it should complete before printing your document. The default value is Auto, which means the formatter tries to correctly judge how many passes to make before printing. (For example, if it detects a tag in your file, it will automatically make two passes instead of one.) The formatter performs the following functions during each pass:

**1 Pass** The formatter prints the file while looking for errors. If it finds any errors, it stops printing. Cross-references (tags) are printed with three question marks (???) when printing with 1 Pass.

**2 Passes** The formatter formats the text of your pages, looking for errors and recording the values of all tags for later substitution. If it finds no errors, it prints the document.

**3 Passes** The formatter takes a third "cleanup" pass to make sure all cross-references are right and that no filled-in cross-references caused page breaks to change.

**Formatted Print**
This command determines whether Sprint interprets your formatting commands before printing it. If Formatted Print is Yes (the default), the formatter interprets your formatting commands before printing. If Formatted Print is No, Sprint prints your formatting commands literally instead of interpreting them. (Many menu commands print with control codes like ^B and ^N when Formatted Print is set to No.) You should use

this command when printing program listings, for example.

**Wordwrap ASCII Files**
When this option is set to Yes, and you print an ASCII file, the formatter ignores single hard returns and formats the text using the default margin settings. Two hard returns in a row define a paragraph ending. The default setting is No, which means that lines in ASCII files end wherever the formatter sees a hard return character. Use this command only when you want to print an ASCII file that you want formatted.

**Log Errors to File**
When this option is set to Yes, and you format a file with formatting errors, the formatter writes all error messages to a file on disk. The formatter gives this file the same name as the file being formatted but adds a .LOG extension to distinguish between the two files. The default setting is No, which means the formatter displays error messages onscreen but does not write them to a file during formatting.

For detailed information on printing and a list of all Sprint print options, refer to the "Printing" chapter in the *User's Guide*. (Typing `SPFMT` at the DOS command line also gives you a list of print options.)

How To

To select one of these advanced options, choose **A**dvanced Options from the **P**rint menu.

**Note to two-floppy system users:** Sprint may have to prompt you to insert the correct Program Disk into Drive A in order to print a file.

Tips

You can print the resulting log file, but you must first choose **P**rint/**A**dvanced Options and toggle **F**ormatted Print to No. Then choose **G**o from the **P**rint menu.

See Also

ASCII Files, Error Messages and Warning Messages

# All Pages

Refer to the Footer Menu or Header Menu entry for details.

# Also See

Refer to the Index entry for details.

# Appendix

Keystrokes — *Alt-S* (or *F10*, Style), **H**eadings, **A**ppendix

Function — Starts an appendix and makes an entry for it in the table of contents.

The **A**ppendix command begins an appendix on a new, odd-numbered page (inserting a blank page if necessary) and prints a large, centered, boldfaced, lettered appendix heading. This heading is preceded by six blank lines and followed by three blank lines. Sprint also enters this heading into the table of contents.

**A**ppendix is similar to **C**hapter, except that the boilerplate text is *Appendix*, and the "numbering" system is letters (starting with *A*). **C**hapter produces numbered chapters (beginning with 1).

You will find this command defined in the Sprint file STANDARD.FMT.

How To — You can choose the **A**ppendix command and then type the text of the appendix heading, or type the appendix heading, select it, and then choose **A**ppendix from the **H**eadings menu.

If you choose **A**ppendix before typing your text, Sprint prompts you to enter the title. In either case, Sprint displays the appendix title as highlighted text and inserts the command word APPENDIX. For example,

```
APPENDIX Further Reading
```

See Also — Appendix, Chapter, Headings, Modifying Formats, Page Numbers, and Chapter 2 (Template)

# AppendixSection

| | |
|---|---|
| Keystrokes | *Alt-S* (or *F10*, Style), **H**eadings, **A**ppendixSection |
| Function | Numbers and prints the title of a section within an appendix and enters the title in the table of contents.<br><br>AppendixSection prints the specified title of a section within an appendix, numbers the section, and creates an entry in the table of contents. The appearance of the title is similar to the one produced by Section, but AppendixSection numbers are preceded by the letter of the appendix, such as A.1 and A.2.<br><br>You will find this command defined in the Sprint file STANDARD.FMT. |
| How To | Choose the **A**ppendixSection command from the **H**eadings menu and type the section title. You can also type the section title, select it, and then choose **A**ppendixSection from the **H**eadings menu. Sprint highlights the command and title onscreen. |
| Tips | There isn't a variable named *AppendixSection*. Instead, Sprint reuses the variable *Section* to number appendix sections. |
| See Also | Appendix, Section |

# AppendixTitle (Variable)

Refer to the Variables entry for details.

# Appendix (Variable)

Refer to the Variables entry for details.

# Arrange-Sort

Keystrokes — *Alt-U* (or *F10*, **Utilities**), **Arrange-Sort**

Function — Lets you sort a selected block of text in alphabetical order.

How To — Select the text you want sorted, then choose this command. Sprint prompts for you to choose **A**scending (*A* to *Z*) or **D**escending (*Z* to *A*) order. Sprint redisplays your text in the new order.

When sorting paragraphs of text, Sprint treats each paragraph as if it were one (long) line and sorts them by their initial characters.

Note that the order used is actually according to ASCII numbers (see Appendix B). This means that uppercase letters precede lowercase ones, and some symbols (like % and &) come before numbers, while other (like > and =) come after them. Because of these anomalies inherent in an ASCII sort, you may have to fine-tune the order by hand.

**Note to two-floppy system users:** Sprint may have to prompt you to insert the correct Program Disk into Drive A in order to sort files with **A**rrange-Sort.

Tips — You can sort one column of a table independently from the others by selecting the column of text using the **C**olumn Mode from the **B**lock Select menu. Then press *F3* and choose **A**rrange-Sort as usual.

Remember, you must select all the text that you want sorted. If you want a column of words in alphabetical order, you must select all the words, not just their first characters.

See Also — Block Select Menu

# Ascending Order

Refer to the Arrange-Sort entry for details.

# ASCII File Handling

Refer to the ASCII Files entry for details.

# ASCII Files (General Information)

Keystrokes

*Alt-F* (or *F10*, File), Translate, Import, ASCII File

*Alt-F* (or *F10*, File), Translate, Export, ASCII File

*Alt-P* (or *F10*, Print), Advanced Options, Wordwrap ASCII Files

*Alt-C* (or *F10*, Customize), ASCII File Handling

Function

Commands for creating, editing, and importing and exporting ASCII files.

**Import/ASCII File**
Reads an ASCII file on disk, converts it to Sprint format, and displays the converted file on screen.

**Export/ASCII File**
Writes a Sprint file to a file in pure ASCII format. The new file has the same name as the original but has the extension .ASC added to it.

**Wordwrap ASCII Files**
Sets a wrap margin in an ASCII file (one without a ruler line).

**ASCII File Handling**
Displays the following menu to let you make settings for your ASCII files:

| | |
|---|---|
| Tab Expansion | Lets you set the number of characters to move the cursor for each press of the *Tab* key. Default setting is 5. |
| Auto-Indent | If set to On, Sprint automatically indents however many spaces and tabs the preceding line is indented whenever you press *Enter*. Default is Off. |
| Wrap Long Lines | If set, lets you determine where Sprint wordwraps your line (default |

is 65). When set to 0, this onscreen wordwrap is turned off.

If you've formatted a paragraph with the **W**rap Long Lines command and later edit the paragraphs so that the wordwrap is no longer accurate, you can "refresh" the screen appearance by pressing *Ctrl-B*. Sprint reformats the text.

Pure ASCII files are text files that contain universally understood ASCII characters only; that is, files that contain standard *printing* characters and these control codes:

| | |
|---|---|
| ^I | ASCII tab character |
| ^J | Line feed (hard return) |
| ^L | Form feed |
| ^M | Carriage return |

These control codes are universally understood by virtually every computer. This is the advantage of pure ASCII.

ASCII files are typically used for program files, files you want to be read or edited with another word processor, or files you prepare for electronic mail applications. Under "normal" circumstances, you don't create pure ASCII files when you use Sprint. Sprint's ruler line, menu options, and commands insert nonstandard control codes in your file (see Table 1.1 on page 46 for a complete list). As long as you have these control codes in your file, you don't have a *pure* ASCII file.

Nonetheless, it's important to understand that Sprint files are still considered ASCII (albeit nonpure) because its files are composed of ASCII characters from 0 to 128. Any word processor or editor can open and read a Sprint file without translation. The only confusion will be when that word processor tries to interpret Sprint's use of certain control codes (for example, the ^K that makes up a Sprint ruler).

*The only time you need to translate from or to Sprint-ASCII format to pure ASCII is when you need to strip or add hard returns at the end of every line.*

How To

There are three ways to create a pure ASCII file using Sprint:

1. You can create and edit your file normally, using Sprint rulers and menus, and then choose the **Export** command from the **File/Translate** menu. Sprint automatically creates an ASCII version of your file in which it strips all control codes (except those listed in the previous "Function" section), replacing all soft newline characters with hard return characters (^J).
2. You can delete the ruler line in your Sprint file and press *Enter* whenever you want to start a new line. Avoid using menu commands that insert control characters (you should use @-sign commands instead).
3. You can delete the ruler line in your Sprint file, choose **Wrap Long Lines** from the **Customize/ASCII File Handling** menu, and specify the column at which you want Sprint to wrap your text. When your line extends beyond the column number you specified, the editor automatically inserts a hard return character (^J), and begins the next line of text.

   This is similar to having a ruler line in your file and entering a right bracket (]) on it to specify the right indent. The difference is that **Wrap Long Lines** inserts a hard return, while the right indent on the ruler inserts a soft return (^_) at the end of each line (unless you press *Enter*).

**Note to two-floppy system users:** Sprint may have to prompt you to insert the correct Program Disk into Drive A in order to use this command.

When creating pure ASCII files, use either formatter tabs (press the *Tab* key) or the **Tab Expansion** command on the **Customize/ASCII File Handling** menu to produce columns of text. *Do not use editor tabs* (that is, those set on the ruler line or with the **Precise Settings** commands on the **Layout/Ruler** menu), since these tabs require a ruler line in your file.

Pure ASCII files shouldn't contain ruler lines; the control codes comprising ruler lines can often corrupt a program file or a file that you're preparing for electronic

mail. If you don't want to manually enter hard returns, set the **Wrap Long Lines** command. Then you can create short, easy-to-view lines and have Sprint wrap the lines automatically, without inserting a ruler line. Your file remains a pure ASCII file.

Before printing an ASCII file, remember to choose **Wordwrap ASCII Files** on the **Print/Advanced Options** menu.

Tips

If you want to create a pure ASCII file, and also want to take advantage of Sprint's formatting capabilities, use the @-sign method of entering formatting commands. @-sign commands don't insert control codes in your file; they are comprised of ASCII characters only and have no effect on your text until you print or preview your file.

See Also

Auto-Indent, File Menu, Hard and Soft Returns, Translate, and Chapter 3

# Assign to Key

Refer to the Glossary entry for details.

# Asterisks

Refer to the Bulleted Lists entry for details.

# Auto-Indent

Keystrokes

*Alt-C* (or *F10,* Customize), **ASCII File Handling, Auto-Indent**

Function

Causes the first-line indent of the next line to match that of the preceding line.

This command is used in pure ASCII files, and is useful for writing programs in which you want the indentation of one or more lines to match the indentation of the preceding line. Sprint does this by inserting a combi-

nation of tabs and spaces to move the cursor to match the line above the current line.

How To

Set **Auto-Indent** to **On** from the **ASCII File Handling** menu. Once you've selected **Auto-Indent** (**On**), you can begin typing at the desired column. Whenever you press *Enter*, the cursor automatically moves to the column at which you began the preceding line. When you want to change the indent value manually, move the cursor forward or backward to the desired column. Thereafter, whenever you press *Enter*, lines will match this new indent value.

**Note to two-floppy system users:** Sprint may have to prompt you to insert the correct Program Disk into Drive A in order to use this command.

See Also

ASCII Files, Indenting, Tabs

# AutoSpell

Keystrokes

*Alt-U* (or *F10*, **U**tilities), **S**pelling, **A**utoSpell

*Shift-F1*, **A**utoSpell

Function

Toggles the automatic spelling checker on and off.

When this command is set to **On**, Sprint loads the spelling program if necessary, and then automatically checks each typed word against the words listed in its dictionary. If you type a word that's either misspelled or not in Sprint's dictionary, you'll hear a short warning beep. (If you don't like **A**utoSpell's warning tone, you can change the beep with the **C**ustomize/**O**ptions/**T**one command.) You can correct the word immediately by backspacing and retyping or wait until later.

When **A**utoSpell is **On**, Sprint records all bad (misspelled) words so that you can find and correct them with the **E**very Bad Word and **L**ast Bad Word commands. If **A**utospell is set to **Off**, Sprint will not check your spelling accuracy as you type (but you can still use the **S**pelling commands **W**ord, **B**lock, **F**ile, and **R**est of File to check your text).

How To

Choose AutoSpell from the Utilities/Spelling menu. This command is a *toggle*, so, if it was set to Off, choosing the command changes it to On. If AutoSpell is already on, choose AutoSpell to turn it off.

Tips

AutoSpell checks words when you press the *Spacebar* or *Enter*, which in normal typing indicates the end of a word. When you're editing and retype a word, but don't press the *Spacebar* or *Enter*, AutoSpell won't check your word.

If you're typing text or program code that contains technical words or terms not usually found in a dictionary, it's probably a good idea to turn off Sprint's spell-checking facility.

**Warning:** Don't use AutoSpell if you're not using a hard disk. (You won't have enough disk space with most floppy disks.)

See Also

Customize Menu, Spelling Menu, Thesaurus, Tone

# Background Save Period

Keystrokes

*Alt-C* (or *F10*, Customize), Options, Background Save Period

Function

Determines how often Sprint updates the contents of its backup file on disk.

The default setting is 3 seconds, which means whenever you stop typing for 3 seconds (or more), Sprint updates the contents of its swap file on disk. This activity is called *swapping* and is a great crash-recovery tool.

Even if you seldom explicitly choose Save from the File menu, Sprint tirelessly saves your entire work session (including as many files as you have open) to the special swap file. This saving is done in the background, virtually invisible to you.

How To

Choosing this command displays the following prompt:

```
Background save period (in seconds): 3
```

Enter the maximum amount of "quiet" time (time during which there is no keyboard activity) required

before Sprint updates the swap file. For example, enter `10` to tell Sprint to wait until you stop typing for 10 seconds before updating the swap file. Valid delay values are from 1 to 60 seconds; entering 0 prevents all swap file updates.

**Note to two-floppy system users:** Sprint may have to prompt you to insert the correct Program Disk into Drive A in order to use this command.

Tips

Backing up your files can briefly interrupt the screen display, so it might seem that Sprint takes a little extra time to display the keystrokes you enter. To avoid this, delay the backup period by increasing the **B**ackground Save Period value. Remember, though, **the smaller the delay period, the better protection you have against crashes!**

Remember also, if you enter the value of 0, you will have no backup file to rely on if your system crashes.

See Also

Open (File), Options Menu, Preserve Editing Session, Save

# Bar

Refer to the Graphics entry for details.

# Begin (Format)

Refer to the Begin and End (Formats) entry in Chapter 2 for details.

# Blank Page(s)

Keystrokes

*Alt-L* (or *F10*, **L**ayout), **P**age Breaks, **B**lank Page(s)

Function

Creates one or more blank pages after printing the current page.

The **B**lank Page(s) command (displayed onscreen as BlankPage) creates *n* blank pages with printed headings, footings, and page numbers (that is, BlankPage 2 creates

two blank pages in succession; BlankPage 1 (default) inserts a single blank page). This command does not cause the current page to end immediately; the formatter creates the blank page(s) *after* it fills the current page. (If you want the formatter to insert the blank pages immediately, you can follow the command with a **P**age Breaks/**I**nsert (Unconditional) command.)

How To

Choose **B**lank Page(s) from the **L**ayout/**P**age Breaks menu. Sprint displays the prompt:

```
Number of blank pages to insert: 1
```

After you type in the number of blank pages you need and press *Enter*, Sprint inserts the command BlankPage *n*, where *n* is the number you entered at the prompt.

Tips

This command is useful when you want to leave room for full-page figures, diagrams, or tables. Since the blank pages contain page numbers, headers, and footers, your document looks more polished than if you paste-up figures on a totally blank sheet of paper and inserted these. You can, however, insert blank pages if necessary.

If you want completely blank pages (no page numbers, headers, or footers), use the **B**lank Page(s) command and physically insert the blank piece(s) of paper after printing your document, replacing the blank page(s) supplied by Sprint.

See Also

Page Breaks

# Blank Space (Horizontal)

Keystrokes

*Alt-I* (or *F10*, **I**nsert), **N**on-Breaking Space

*Ctrl-Spacebar*

*Alt-I* (or *F10*, **I**nsert), **W**ide Space (Spring)

Function

The first two commands insert a single, non-breaking, fixed-width blank space.

Sprint provides a number of commands that let you adjust the amount of horizontal blank space (typically, blank space between words). You can use some of these commands to leave room to insert special characters that

your printer can't print, or you can use others to help format non-breaking text or columns in a table. For a complete description of a particular command, refer to its entry in this menu encyclopedia.

The Non-Breaking Space command tells Sprint to insert a single blank space at the current cursor position and keep the words on either side of the command together on a line.

Wide Space (Spring) inserts enough blank space to align text at the right margin or next tab stop.

See Also — Non-Breaking Space, Wide Space, and Chapter 2 (Hsp, Tab)

# Blank Space (Vertical)

Refer to the Reserve Space entry for details.

# Block (Hyphenate)

Refer to the Hyphenation Menu entry for details.

# Block (Spelling)

Refer to the Spelling Menu entry for details.

# Block Select Menu

Keystrokes — *Alt-E* (or *F10*, Edit), **B**lock Select

*F3* (toggles selection mode)

*Ctrl-KB*, then *Ctrl-KK* (begins and ends block selection)

*Ctrl-KK* (displays End Region menu)

Function — Displays a menu of commands for selecting a block.

A *block* is an area of text that you want to affect. Once you select a block, you can delete it from your file, move

it to a different place in your file, make a copy of it elsewhere, write it to a new or existing file, or change the typeface or format of all text within the block. When you select text, the word "Sel" appears in the status line.

Selecting a block means identifying an area of text (anything from a single character to an entire file) so that you can easily change the appearance or location of the text. Rather than retype three paragraphs of boilerplate text, for example, you can select these paragraphs as a block, and then copy the block to the desired location. If you decide that a paragraph would look better if it appeared in italic rather than plain text, you can mark the block and then choose the **I**talic typestyle command.

When you choose the **B**lock Select menu, Sprint lists a variety of options for you to use:

**Word**
Selects the word the cursor is on or immediately after.

**Line**
Selects the line the cursor is on.

**Sentence**
Selects the entire sentence the cursor is on. Sprint looks for a normal sentence terminator (like a period or question mark) followed by a space or hard return in order to determine a sentence.

**Paragraph**
Selects the entire paragraph the cursor is on. Sprint looks for a hard return in determining where a paragraph starts and stops.

**Reselect Block** *Alt-B*
Instantly marks the block of text that was last selected.

**Turn Select Mode** *F3*
Toggles selection mode. As you type or use the arrow keys, text will be selected. When you've ended the selection, you can choose any command that affects a block of text. The shortcut for this command is *F3*.

**Column Mode** *Ctrl-KN*
Toggles "column mode," which lets you select text in *columns* instead of *rows*. When you're in column mode,

"Col" appears in the status line. Use the cursor keys to highlight the column of text.

If you are used to WordStar-like commands, use the shortcuts *Ctrl-KB* and *Ctrl-KK* to select text, and *Ctrl-KH* to "hide" it.

The End Region menu appears when you choose *Ctrl-KK* and affects your selected block in one of three ways: **C**opy, **M**ove-Cut, or **C**ancel Selection. Choosing **X** unmarks your selected block.

If you're using a mouse, the left button toggles text selecting (same as *F3*), and double-clicking the left button selects the word the cursor is in.

Tips

Before you start using these Block commands, you should be familiar with the concept of Sprint's *Clipboard.*

The Clipboard is an in-memory "holding tank" for text that you mark and then move, delete, or copy. For instance, when you mark a block and enter the **M**ove-Cut command, Sprint removes the block from the screen and places it on the Clipboard. Once you move the cursor to where you want this text to appear, you choose the **I**nsert-Paste command, which copies the Clipboard text to the current cursor position.

A block remains on the Clipboard until you replace it with new text. This means you can paste numerous copies of the Clipboard text, but once you enter a command that puts new text on the Clipboard (like **M**ove-Cut or **C**opy), you overwrite what's currently there. Therefore, it's important to immediately paste text that you moved or copied to the Clipboard.

When you're selecting text, you can press any key to have the selection automatically extended to the next occurrence of that character. This means, for example, you can press *Spacebar* to advance the selection one word at a time (that is, the selection jumps to the next space character every time you press *Spacebar*).

See Also

Clipboard, Deleting, Insert-Paste, Write As

# Bold

Refer to the Typestyle Menu entry for details.

# Bottom Margin

Refer to the Document-Wide Menu entry for details.

# Bottom Status Line

Refer to the Screen entry for details.

# Bulleted Lists

Keystrokes

*Alt-S* (*F10*, Style), Lists, **A**sterisks

*Alt-S* (*F10*, Style), Lists, **B**ullets

*Alt-S* (*F10*, Style), Lists, **H**yphens

Function

Places a "bullet" character before each new paragraph in an unnumbered list. The shape of these bullets depends on which list command you choose, what kind of printer you're using, and how many levels of "nesting" there are. On non-PostScript printers, for example, an **A**sterisk list item is preceded by a regular asterisk (*); a **B**ullets list has lowercase *o*'s (o); and a **H**yphens list has regular hyphens (-). Second-level (nested) lists shift the symbol (**A**sterisks has *o*'s, **B**ullets has hyphens, and **H**yphens has asterisks).

On PostScript printers, however, the "bullet" characters are a bit fancier. In this case, **A**sterisks uses a diamond character (♦) and **B**ullets uses a filled in *o* (•). **H**yphens still uses a hyphen on PostScript printers.

This command formats lists by separating each item (or each paragraph) with a symbol. As defined in the STANDARD.FMT file, Sprint indents the symbols two characters from the left indent; the text of the list begins five spaces from the left indent. You must leave a blank

line before each item. Otherwise, Sprint considers the next paragraph to be part of the preceding item.

How To

If you want a list with hyphens, choose **Lists** from the **Style** menu and then choose **Hyphens**. Sprint automatically inserts the Begin and End commands if your text was selected before you chose the command. Otherwise, Sprint displays the prompt:

```
Press (B) for Begin command, (E) for End command,
or ESC to cancel:
```

Type the text of the list, making sure there's a blank line between each paragraph, and choose the command to end the format (type E). Sprint needs the blank lines to determine where one paragraph ends and the other begins. For example,

```
BEGIN HYPHENS
1/2 cup milk

3/4 cup cocoa powder

1/2 cup butter

1/2 cup sugar

TabBlend these four ingredients together until
creamy.

1 egg, beaten

1/2 tsp. vanilla

TabAdd these two ingredients to the mixture, and
stir well. Place the batter in a greased oblong pan
and bake until done.
END HYPHENS
```

If you want to align a paragraph with the text of the preceding paragraph, but don't want a dash in front of it, press *Tab* before beginning the paragraph (as shown in the preceding example).

The printed result looks like this:

– 1/2 cup milk

– 3/4 cup cocoa powder

– 1/2 cup butter

– 1/2 cup sugar

Blend these four ingredients together until creamy.

– 1 egg, beaten

– 1/2 tsp. vanilla

Add these two ingredients to the mixture, and stir well. Place the batter in a greased oblong pan and bake until done.

If you don't want blank lines in the printout, modify the bulleted list format and add the parameter *Spread 0*. For example,

```
BEGIN HYPHENS Spread 0
```

Follow these same steps for the **B**ullets and **A**sterisks commands. The same list using bullets (with the *Spread 0* parameter) prints out like this:

• 1/2 cup milk
• 3/4 cup cocoa powder
• 1/2 cup butter
• 1/2 cup sugar
Blend these four ingredients together until creamy.

• 1 egg, beaten
• 1/2 tsp. vanilla
Add these two ingredients to the mixture, and stir well. Place the batter in a greased oblong pan and bake until done.

Tips

You can nest these formats (insert a format within another format); the next two levels use the other two symbols, and then the cycle starts over. You can nest these commands to any depth, so long as there is room between the indents.

We modified the Hyphen definition for the Sprint manuals to print boxes instead of dashes. If you're using

an Apple LaserWriter Plus (or any PostScript printer with the Dingbats font) and want to achieve the same effect, open a copy of POSTSCR.TCT, go to the last line and replace

```
@symbol(@char(168))
```

with

```
@dingbat(@char(110))
```

See Also Description, Lists, Multilevel, Numbered

# Bullets

Refer to the Bulleted Lists entry for details.

# Canceling

Keystrokes *Alt-F* (or *F10*, **F**ile), **R**evert to Saved

*Esc*

*Shift-Esc*

*Ctrl-U*

*Ctrl-C* (for SprintMerge and formatting only)

Function Aborts, cancels, or stops Sprint's current activity or menu display.

Use the **R**evert to Saved command to cancel (discard) all changes you've made to a file since the last time you explicitly chose a save command.

Use *Esc* to remove the last-displayed menu or to cancel a prompt on the status line.

Use *Shift-Esc* or *Ctrl-U* to remove *all* displayed menus in one fell swoop.

Use *Ctrl-U* to remove menus (just like *Esc*) and to abort Sprint activities like searching.

*Ctrl-C* cancels the current operation in the SprintMerge program and also aborts a formatting pass, prior to printing a document.

See Also — Deleting, Merge

# Caption

Keystrokes — *Alt-S* (or *F10*, Style), **F**igure

Function — Numbers and creates a caption (title) for the current figure.

This command works in conjunction with the **F**igure command. In fact, when you choose **F**igure from the **S**tyle menu, Sprint prompts you for the caption text.

When the formatter finds a Caption command in your file, it determines the current value of *Figure* and then automatically centers and prints the word *Figure* in a small font, followed by the correct figure number and its caption.

You will find this command defined in the Sprint file STANDARD.FMT.

How To — Choose the **F**igure command from the **S**tyle menu. At the `Caption:` prompt, type in your figure caption (title) and press *Enter*. Sprint displays the following onscreen:

```
BEGIN FIGURE
CAPTION Block Diagram
END FIGURE
```

If you're going to paste-in the figure, you'll need to add the formatter command, Reserve *dimension* after the Begin Figure command; choose **L**ayout/**P**age Breaks/**R**eserve Space. If you're going to type in the figure, the text goes here instead. Your caption prints after the figure drawing.

If you want the caption to print before the figure, the Caption command should immediately follow the Begin Figure command. For example,

```
BEGIN FIGURE
CAPTION Block Diagram
RESERVE 180 points
END FIGURE
```

In this case, Sprint prints the figure caption before the figure drawing.

If you want a caption with your figure but *don't* want it in the List of Figures, you should use the FCapt command. Choose Style/Other Format and enter `FCapt`, followed by the caption text. When prompted, press *C* to tell Sprint that this command does not affect a region of text.

Tips

If your file is set up so that your page numbers include a chapter or section number (for example, page 3-3, 4-1, and so on), you can specify that your figure numbers also follow this format (Figure 1-1, Figure 1-2, Figure 2-1, and so on). See the Parent entry in Appendix C ("Style Sheet Commands") of the *Advanced User's Guide* for details.

You can use the Define a Tag command after the Caption command to cross-reference the figure in text. The Define a Tag command must follow the Caption command and equal figure, or Sprint won't print the correct figure number in your text.

Note that the Style/Table command also prompts for a caption but inserts a TCaption command onscreen instead of the Caption command.

See Also

Figure, Page Numbers, Reserve Space, Tags, TCapt and TCaption

# Case Sensitive

Refer to the Searching entry for details.

# Centering Text

Keystrokes

*Alt-S* (or *F10*, Style), Center

*Alt-L* (or *F10,* Layout), **R**uler, **E**dit on Screen, and type C (or just press *Alt-A* or *Alt-R* and type C)

*Alt-L* (or *F10,* Layout), **T**itle Page

*Alt-I* (or *F10,* **I**nsert), **W**ide Space (Spring)

*Alt-U* (or *F10,* **U**tilties), **P**otpourri, CenterTab

*Ctrl-OC* (centers line)

Function

Centers text.

If no text is selected, the **C**enter command (or *Ctrl-OC*) centers the line of text the cursor is on between the left and right margins by putting wide spaces (^F characters) at each end. If a block is selected, Sprint surrounds the block with the onscreen commands BEGIN CENTER and END CENTER. This command centers each line of text both onscreen and when you print.

C, when typed on a ruler, centers all text governed by this ruler.

The **T**itle Page command centers all the text on a page *vertically* around the specified point (the default is .5 pages). Page headers and footers are *not* centered; they print as they normally would within the top and bottom margins of the page.

This command is useful for such things as positioning titles on the first page of a document. The Title Page entry in this menu encyclopedia explains this command in greater detail.

You use the **W**ide Space command to center text between margins by inserting wide spaces at each end.

You can use the **P**otpourri/CenterTab command to center text between two tab stops.

How To

Choose **S**tyle/**C**enter Line (or *Ctrl-OC*) wherever you have one line of text to be centered.

To center many paragraphs, insert (*Alt-R*) or edit (*Alt-A*) a ruler, and type a C in it. All lines after the ruler will be centered (both onscreen and in printing) until a subsequent ruler changes the setting.

Tips — Don't try to center a line in the middle of a word-wrapped paragraph. The center commands work best when the line to be centered ends with a hard return.

See Also — Justification, Ruler, Wide Space

# Center

Refer to the Centering Text entry for details.

# Change Directory

Refer to the File Manager Menu entry for details.

# Chapter

Keystrokes — *Alt-S* (or *F10*, Style), **H**eadings, **C**hapter

Function — Starts a numbered chapter on a new page, and enters the title in the table of contents.

This command specifies the beginning of a new chapter. Sprint automatically begins a new page on an odd-numbered page (adding a blank page if necessary), leaves six blank lines at the top, and prints a large, bold, centered, and numbered heading for the chapter. It also makes an entry for the chapter in the table of contents. Chapter entries in the table of contents are double-spaced.

You will find this command defined in the Sprint file STANDARD.FMT.

How To — If you haven't yet typed the chapter title, choose **C**hapter from the **S**tyle/**H**eadings menu. At the prompt, type the title of your chapter, and press *Enter*.

If you've already typed your chapter title, move the cursor to the line containing the title, and then choose **C**hapter. Either way, Sprint highlights the text onscreen, like so:

```
CHAPTER Executive Summary
```

See Also Appendix, Headings Menu, HeadingB, Section

# Chapter (Variable)

Refer to the Variables entry for details.

# ChapterTitle (Variable)

Refer to the Variables entry for details.

# Character Size

Refer to the Font Size and Typestyle Menu entries for details.

# Clipboard

Keystrokes

*Alt-E* (or *F10,* Edit), **Undelete** (returns the contents of the Clipboard to where it was cut from)

*Alt-E* (or *F10,* Edit), **Copy** (copies the selected text to the Clipboard)

*Alt-E* (or *F10,* Edit), **Move-Cut** (removes the selected text and places it in the Clipboard)

*Alt-E* (or *F10,* Edit), **Insert-Paste** (inserts the contents of the Clipboard at the current cursor position)

*F4* (same as **Copy**)

*F5* or *Del* (same as **Move-Cut**)

*F6* (same as **Insert-Paste**)

*Ctrl-U* (when no menu is displayed, same as **Insert-Paste**)

*Ctrl-T* (removes the word to the right of the cursor and places it in the Clipboard)

*Ctrl-Y* (removes the current line and places it in the Clipboard)

*Ctrl-QY* (removes the current line starting from cursor position and places it in the Clipboard)

Function

The Clipboard is a Sprint "holding tank" for text that you want to move or copy elsewhere (even to other Sprint files). The Clipboard is actually a memory buffer that Sprint uses to keep selected text until you need it.

Ordinarily, the Clipboard holds only one selection at a time. If you cut a paragraph, move the cursor, and then cut another paragraph, only the *second* paragraph will be in the Clipboard ready for pasting. If, however, you don't move the cursor between successive cuts (using, for example, *Ctrl-Y* to delete a bunch of lines one after another), Sprint assembles all your cuts into the Clipboard, appending each successive cut after the last one.

Note that the Erase command on the Edit menu also deletes selected text but *does not* put that text into the Clipboard. Use Erase, therefore, when you want to quickly cut a block without altering the contents of the Clipboard.

See Also

Copy, Deleting, Edit Menu, Undelete

# Close (File)

Keystrokes

*Alt-F* (or *F10*, File), Close

*Ctrl-F4*

*Ctrl-KQ* (closes a file)

Function

Closes the current file.

This command closes the current file and removes it from the swap file (SP.SWP, the backup file created each time you start Sprint); it does not, however, save the file to disk. Once you are done working on a file, you should save it to disk and then close it.

A file remains open and available for editing until you close the file. Even if you open other files, Sprint keeps all files open (up to 24 of them) and available (even if there is a system crash) until explicity closed. Because

each open file requires space in the swap file, it's a good idea to close files as you finish working on them.

When you choose the **Close** command, Sprint checks to see if the file has been modified since it was last saved. If it has been modified, Sprint asks you if you want to save the file before closing it. You then have the option to save the file before Sprint closes it. If it has not been modified since the last save, Sprint closes the file and removes its contents from the swap file.

If the **Preserve Editing Session** (in the **Customize/Options** menu) is set to No, Sprint closes all documents in the swap file when you exit the program. If the command is set to Yes, Sprint keeps a backup file so that the files will be open the next time you use Sprint.

If an open file has been modified since the last save, Sprint will, before exiting, display the unsaved file and ask if you want to exit without saving. You then have the option to save the file before Sprint closes it.

How To

To close a file, choose **Close** from the **File** menu. If you've modified the file since the last time you printed or saved it to disk, Sprint displays the prompt:

```
The file C:\DIRECTORY\FILENAME.EXT has not been saved;
save it (Y,N,ESC)?
```

If you do not want to save the changes you've made, type `N`. If you want to save the changes, type `Y`. If the file you closed isn't the only open file, Sprint switches to the previous open file.

Once you close the last open file in the swap file, Sprint displays a ruler line at the top of the screen, and the status line shows the file name "Unnamed." You can continue creating new files with Sprint, or you can exit to DOS.

Tips

You do not need to close a file before quitting Sprint when the **Preserve Editing Session** is set to Yes, but to avoid having different versions of the same file, be sure to save it to disk.

See Also

Background Save Period, File Menu, Save

# Close (Window)

Refer to the Window entry for details.

# Codes

Keystrokes

*Alt-C* (or *F10*, Customize), Screen, Codes

*Alt-Z* (toggles control codes)

Function

Shows or hides the control codes that make up Sprint commands.

This command is a toggle that instructs the editor to either display or hide the control codes used to make up Sprint commands. Some commands consist solely of control characters; for example, *Tab* is really a *Ctrl-I* character (displayed as ^I), and a ruler line is really a set of control codes that define the left and right indents, tab settings, and so on (for example, ^KR 65, T 5). Other commands contain command text *and* control codes; for example, the Begin Description command that is used throughout this chapter looks like this when Codes is set to On:

```
^OBEGIN DESCRIPTION^N
```

In this example, the ^O instructs the formatter to begin the command, and the ^N ends the command line.

When you set the Codes command to show (On), the editor won't wrap your paragraphs within the margins on the ruler line, nor will it reformat them. Once you set Codes to Off again, the editor automatically resumes its wrapping and reformatting functions.

For a list of Sprint control codes, see Table 1.1 on page 46.

How To

To display the control codes in Sprint commands, choose Customize/Screen. Sprint displays the current setting of the command. Off means you don't want Sprint to display control codes; On means you want to see the control codes. All of the commands on the Screen menu are *toggles*.

**Note to two-floppy system users:** Sprint may have to prompt you to insert the correct Program Disk into Drive A in order to use this command.

Tips

Set the Codes command to On when you want to delete a control code. That way, you can see the code you want to delete. When Codes is set to Off, the editor automatically moves the cursor beyond the control codes in text, so you don't accidentally delete them.

See Also

Control Characters, Insert Menu

# Colors

Keystrokes

*Alt-C* (or *F10*, Customize), Colors

Function

Displays the Color Set menu.

This command displays the Color Set menu, which lets you change how Sprint displays on your screen. You can select from two color sets or a monochrome set. You can also customize your own color set or undo any changes you have made.

How To

To change any of the color settings, choose Customize, Colors. Move the cursor to the desired color set or command and press *Enter*.

You can also choose Modify from the Color Set menu. When you do, you'll see a choice of Typestyle, Formatting, and System. Choosing one of these gives you a new menu of screen elements in that category. Choosing the screen element, in turn, displays a full palette of colors and attributes to choose from. Use your arrow keys and *PgUp* and *PgDn* keys to select a color, then press *Enter*. Pressing *Esc* goes back to the previous menu to let you change the color of another screen element.

**Note to two-floppy system users:** Sprint may have to prompt you to insert the correct Program Disk into Drive A in order to use the Customize/Colors command.

Tips

If all your changes end up looking like a garish crazy quilt, you can choose Undo and return to the colors you started with.

If your computer uses the Hercules InColor card or the Hercules Graphics Card Plus, the colors palette will display many different fonts to choose from in addition to colors and attributes.

**Note:** Some memory-resident programs might cause certain custom colors to blink. If this happens, reselect the blinking item in the Colors menu and set it to a color from the first set of colors. (If you're using SideKick Plus, choose Services/Setup/Exit Conditions, and set Allow Blink Attribute to Yes.)

# Column Break

Refer to the Columns entry for details.

# Column Mode

Refer to the Block Select Menu entry for details.

# Columns

Keystrokes

*Alt-L* (or *F10*, Layout), Multiple Columns

*Alt-S* (or *F10*, Style), Other Format, and type `FormatName Columns = "n"`

Function

Formats the text in *n* columns (a maximum of 6).

The Columns menu lists three commands for creating columns in your text: Snaking Columns, Column Break, and Gutter Between. These columns are called "snaking" columns because the text in one automatically flows into the next one as needed (flowing from bottom to top to bottom like a snake's winding motion). Snaking columns are the type you're used to seeing in newsletters and newspapers.

When you choose the Snaking Columns command from the Layout/Columns menu, Sprint automatically inserts an onscreen command, matching the numbers of

columns you want. For example, if you choose three columns, Sprint inserts the command COLUMNS3. Here's how it might look onscreen:

```
Words that rhyme with sponge, almost:

BEGIN COLUMNS3
expunge
lunge
grunge
flange
splotch
scrunch
munch
bunch
lunch
END COLUMNS3
```

Tips

You can also create snaking columns by modifying formats with the special parameter called *Columns*. You can do this by choosing **Style/Modify** and then entering the parameter.

For example, within a Bullets format, choose **Style/Modify** and type `columns=2`. Press *Enter*, and your modification appears onscreen. For example,

```
BEGIN BULLETS, COLUMNS=2
one

two

three

four

five

six
END BULLETS
```

prints like this:

- one
- two
- three
- four
- five
- six

If you want multiple columns, but don't plan on using a lot of special features (like dashes or numbers before paragraphs), use the Text command and modify it to include the *Columns* parameter. For example,

```
BEGIN TEXT, COLUMNS = 2
Sprint is quite good at creating columns. It has the
ability to combine two-column output on the same page
as one-column output. Footnotes also appear at the
bottom of the current column.BEGIN FNOTE This is an
example of a footnote.END FNOTE Sprint can also do
three, four, five, and even six columns.
END TEXT
```

results in:

Sprint is quite good at creating columns. It has the ability to combine two-column output on the same page as one-column output. Footnotes also appear at the bottom of the current column.[1] Sprint can also do three, four, five, and even six columns.

1. This is an example of a footnote.

How To

Select the text you want to format into columns. Choose Layout/Columns/Snaking Columns. You see a prompt reading

```
Number of columns: 1
```

Type in a number from 1 to 6 and press *Enter*. Sprint then prompts:

```
Space between columns (ENTER for .5 in):
```

Enter a dimension or just press *Enter*. If you had already selected the text, Sprint inserts an onscreen command like COLUMNS2.

If you want to change the gutter dimension, choose Columns/Gutter Between. Sprint prompts you for the Columns commands you want to modify. After choosing, you can enter a new figure for the gutter.

Tips

You can force the start of a new column by using the Columns/Column Break command.

If you want to create a format to make parallel (not snaking) columns, you have to use the command called

Column (note there's no *s*). See the Column entry in Chapter 2.

See Also — Modifying Formats, and Chapter 2 (Column; Text)

# Comments

Keystrokes — *Alt-T* (or *F10*, **Typestyle**), **Hidden**

*; single line of hidden text*

Function — Specifies text that appears onscreen but does *not* print.

These commands let you insert text in your file that you *don't want to print*. For example, you might want to temporarily omit sections of your document or insert some instructions in a file that someone else is going to edit for you.

How To

***One-Line Comments***

Type a semicolon (;) at the left margin and then type the text of your comment. When you print your file, Sprint will skip all individual lines that begin with a semicolon.

To insert a comment at the end of a line, press *Tab*, type a semicolon, and then type the text of your comment.

This technique of embedding commands works only if you have set the *Commands* parameter to Yes at the top of your file or at the end of the .FMT file you're using. To do this, you would choose **Style/Other Format** at the top of your document and enter `Style Comments Yes`. At the end of your .FMT file, you would enter

```
@Style[Comments Yes]
```

***Multi-Line Comments***

Mark the text as a block, then choose **Typestyle/Hidden**. Sprint inserts a Begin and End Comment command onscreen. For example,

```
BEGIN COMMENT
 * * * * * * * * * * * * * * * * * * * * * * * * * * *

The following paragraph reflects version A. When we
go to version B, this paragraph should be deleted.
See John for update schedules and planned
enhancements.

 * * * * * * * * * * * * * * * * * * * * * * * * * * *
END COMMENT
```

The asterisks aren't required, but draw attention to the commented text.

Tips

By default, the first line of the STANDARD.FMT file specifies *@Style(comments yes)*, and the last line says *@Style(comments no)*, which means that comments are observed in that file, but not subsequently. This means that if your file contains single-line comments (those that begin with a semicolon), the formatter will print these lines. If you want to suppress printing single-line comments, delete the @Style command at the end of the STANDARD.FMT file or add a Style command at the top of your document.

If you place hidden text on a separate line (not within a line of printing text), Sprint removes the extra line during printing.

**Note:** You cannot nest Comment commands with other Comment commands.

See Also

Modifying Formats, Typestyle Menu, and Chapter 2 (Style)

# Conditional Page Break

Keystrokes

*Alt-L* (or Layout), Page Breaks, Conditional Page Break

Function

Causes a page break to occur where normally it would be prevented.

By default, page breaks are not allowed to isolate the first or last line of a paragraph or to separate section titles, headings, and so on from the text to which they belong. The Conditional Page Break command can

override this function. Also, Conditional Page Break will override any surrounding Group Together on Page commands or grouped formats. Note, if Conditional Page Break appears in the middle of a page, it has no effect.

How To — Press *Alt-L*, and choose Page Breaks/Conditional Page Break. Enter the Conditional Page Break command wherever you want to allow a page break that would normally be prevented. Sprint inserts the word PGBREAK to represent the conditional page break.

See Also — Layout, Page Breaks

# Control Characters (General Information)

Keystrokes — *Alt-C* (or *F10*, Customize), Screen, Codes

*Alt-I* (or *F10*, Insert), Control Character

*Alt-Z* (displays the hidden control codes in your file)

*Ctrl-Q* (inserts control character in Search menu prompts only)

Function — Lets you display, insert, or search for control characters.

*Viewing Control Characters*
The Codes command (or *Alt-Z*) instructs the editor to display the control codes it normally hides from view. Instead of a ruler line, for example, you'll see the control codes that comprise the ruler. Where you've pressed the *Tab* key, you'll see ^I.

**Note to two-floppy system users:** Sprint may have to prompt you to insert the correct Program Disk into Drive A in order to use the Customize/Screen/Codes command.

*Inserting Control Characters*
To enter control characters, go to the Insert menu and choose Control Character. Sprint prompts for the character. Type it in, and Sprint inserts it at the current cursor position. You cannot simply press *Ctrl-F* to insert that control character into your document. That's because Sprint tries to *interpret* a control character entered in this way (to Sprint, *Ctrl-F* means to move the

cursor one word to the right). (**Note:** You will not see the control code unless **C**odes is set to On. Also, the control character you insert in this manner will affect the text following it.)

**Warning:** Don't insert a *Ctrl-Z* into your file because it is used as an end-of-file mark by many programs!

*Searching for Control Characters*

To search for control characters, you can use the special keystroke *Ctrl-Q*. For example, let's say that you created a 20-page proposal and put your company name in **bold** type throughout the text. After printing, your boss decides the company name should be in *italics* instead. You could use the **F**ind command for each occurrence of your company name, select the name as a block, choose **I**talic from the **T**ypestyle menu, and then edit out the control codes from the **B**old command—but what a job! Here's a far better approach:

1. Assuming you're at the top of your file, choose **S**earch & Replace from the **S**earch-Replace menu. Sprint displays the prompt

   `Search for:`

2. Press *Ctrl-Q* and then enter *Ctrl-B*, followed by your company name. For example,

   `Search for:` *Ctrl-Q Ctrl-B* `Borland`

   When you press *Ctrl-Q*, nothing happens, but when you type *Ctrl-B*, Sprint displays the code ^B in the status line.

3. When Sprint prompts for the replacement string, enter *Ctrl-Q*, followed by *Ctrl-E* (the control code that represents italics), followed by your company name. For example,

   `Replace with:` *Ctrl-Q Ctrl-E* `Borland`

   Sprint begins its search for the first occurrence and displays the usual Replace prompts: **Y**es, **N**o, **A**nd the rest.

You can use this example as a guide to search for all nonprinting characters, such as ^I (Tab) and ^J (hard return), as well as the control codes Sprint inserts as part

of a formatting command, such as ^U (start underline) and ^F (wide space).

Table 1.1 lists all the Sprint-defined control codes and their functions.

Table 1.1: Control Codes (Standard User Interface)

| Onscreen | Function |
|---|---|
| ^A | Begins large type |
| ^B | Begins bold type |
| ^D | Places the current word in the index |
| ^E | Begins italic type |
| ^F | Inserts wide space |
| ^G | Indents a region |
| ^H | Backspace |
| ^I | Inserts tab (go to next tab) |
| ^J | Hard return (paragraph mark) |
| ^K | Starts a ruler line |
| ^L | Starts a hard or soft page break (not used mid-line) |
| ^M | Carriage return |
| ^N | Ends most-recent open delimiter |
| ^O | Begins a named formatting command |
| ^Q | Begins subscript |
| ^R | Begins Roman |
| ^S | Begins superscript |
| ^T | Begins typewriter font |
| ^U | Begins underline |
| ^V | Begins a Variable command |
| ^W | Begins word underline type |
| ^X | Begins strikethrough |
| ^\ | Inserts non-breaking space |
| ^] | Hyphen line break (never displayed, always results in a hyphen) |
| ^^ | Discretionary (soft) hyphen |
| ^_ | Soft return (never displayed, always results in a newline) |

**Note:** The caret (^) stands for the control character.

See Also

Codes, Hard and Soft Returns, Searching, Typestyle Menu

# Copy

Keystrokes

*Alt-E* (or *F10*, Edit), Copy

*F4*

*Ctrl-KC*

*Ctrl-KK*, and type C

Function

Copies a marked block.

This command works on a marked (highlighted) block of text (one that you have selected using the **B**lock Select menu). When you copy a block to Sprint's Clipboard, you are instructing Sprint to leave the block in its present location and to also copy it to Sprint's Clipboard. You can then move to another area in your file, or another file, and *paste* this block into the desired location.

**A word of caution:** Sprint's Clipboard can ordinarily hold only one copied block at a time. This means that when you copy a block of text to the Clipboard, you should quickly paste the block into the desired location. If you don't and then enter another command to move, copy, or delete text, Sprint will replace your previous block. If this happens, you'll have to mark and copy your block again.

How To

To copy a block to Sprint's Clipboard, you must first select the block you want to copy (press *F3* and use the cursor keys). After you have done this, press *F4*, or choose Edit/Copy. The text remains in its present location, but is also now in the Clipboard. Move the cursor to the place you want to paste the copied text, and press *F6*. (You could also press *Alt-E* and choose Insert-Paste.) Sprint pastes the block in the current location. This block remains in the Clipboard, too, so you can paste it elsewhere if you like. It stays in the Clipboard until you move or copy another block of text.

See Also

Block Select Menu, Clipboard, Deleting

# Criteria

Refer to the Merge entry for details.

# Current Printer

Keystrokes — *Alt-P* (or *F10*, Print), Current Printer

Function — Selects an alternate printer when formatting.

This command tells Sprint to format a file for output on a printer other than the default printer. Depending on the number of printers defined when you installed Sprint, this command may display a menu of alternate printers from which you can select the desired printer for output.

Before you can use this command, however, you must have already selected the printer with the SP-SETUP installation program. When you select the printer, SP-SETUP creates a file called *printername*.SPP, which it needs to properly format your file for output on the desired printer.

How To — To specify an alternate printer, choose **Current Printer** from the **Print** menu. Sprint displays the alternate printers you've installed for Sprint use. Once you select a printer, you can continue to set print options or choose **Go** to begin formatting.

**Note to two-floppy system users:** Sprint may have to prompt you to insert the correct Program Disk into Drive A in order to print a file.

See Also — Print Menu

# Custom Set

Refer to the Colors entry for details.

# Customize Menu

Keystrokes — *Alt-C* (or *F10*, Customize)

Function — Displays the Customize menu.

This command gives you access to a list of commands and submenus that allow you to make changes to how Sprint looks and acts.

**User Interface**
Lists the alternative user interfaces menu. There are three Sprint UIs (called SPTUTOR, SPBASIC, and SPADV), as well as several alternative UIs that offer keystrokes compatible with different word processors.

**Note to 360K two-floppy system users:** To change a user interface, you need to rerun the SP-SETUP program and choose a new UI. If your computer has disk drives with larger capacity than 360K, you should have had room for several UI files when you ran SP-SETUP.

**Colors**
Displays a menu of commands that affect how Sprint displays screen components on either a color or monochrome monitor and lets you undo any changes.

**Screen**
Displays a menu of toggle commands that affect how Sprint shows certain special characters (paragraph marks, tabs, and so on).

**ASCII File Handling**
Lets you make certain settings when you are creating or formatting ASCII files.

**Menu Shortcuts**
Lets you toggle the shortcut display on Sprint's menus; Yes shows the shortcut keystrokes, No hides the keys.

**Options**
Allows you to set certain swap file and menu delay settings, to toggle insert and overwrite modes, and to change the sound of Sprint's warning tone.

See Also — ASCII Files, Colors, Function Keys, Menu Shortcuts, Options Menu, Screen

# Data Format

Refer to the Merge entry for details.

# Day (Variable)

Refer to the Variables entry for details.

# Define a Tag

Refer to the Tags entry for details.

# Define (Glossary)

Refer to the Glossary entry for details.

# Define Text Variable

Keystrokes — *Alt-I* (or *F10*, Insert), **D**efine Text Variable

Function — Assigns a user-defined variable to a text string.

This command makes a variable equal to a text string. A *string* is any sequence of characters; it can contain text, as well as formatting commands, and is always enclosed within quotation marks. For example,

```
Product="Sprint: The Professional Word Processor"
```

In this example, *Product* is the variable, and the text in quotes is the string you've assigned to the variable. Strings are often used to represent text that you may later want to change. For example, let's say you're creating a user's manual for a product that's still in development and know that the name of the product will change. When the product name changes, you only have to change the string assignment (a one-time change!) to reflect the new product name. Your cross-

references will automatically reflect the correct product name. (See the Variables entry for more information.)

How To

To assign a variable to a string, choose **Insert/Define Text Variable**. Sprint prompts for a name to give the string. Enter the name you want to assign. For example,

```
Name to give the variable: Product
```

Press *Enter*. Spring prompts you for the string itself. Type it in.

```
Enter the text: Raspberry
```

Press *Enter*. Spring inserts the string assignment into your document.

When you want to refer to the product name in your file, don't type the name; instead, use the **Insert/Variable** command, choose **Other**, and type `Product`. At the Pick Template menu, choose **NONE**. For example,

```
STRING Product="Sprint"
This manual explains the features and functions of
the PRODUCT system. You can use PRODUCT for all types
of documents--simple memos to complex, multi-volume
books.
```

The result:

This manual explains the features and functions of the Sprint system. You can use Sprint for all types of documents—simple memos to complex, multi-volume books.

Tips

To insert a quotation mark (") in the definition, you have to use any delimiter except quotation marks. For example,

```
STRING Product=<Quimbley's "Quotable" Thesaurus>
```

makes *Product* equal to *Quimbley's "Quotable" Thesaurus.*

See Also

Chapter, Footer Menu, Header Menu, Variables

# Delete a Block

Keystrokes

*Alt-E* (or *F10*, Edit), Move-Cut

*Alt-E* (or *F10*, Edit), Erase

*Del*

*F5*

*Ctrl-KK*, and then type D

*Ctrl-KY*

Function — Deletes a marked block.

These commands delete a marked block of text. They all place it on Sprint's Clipboard, except for Erase. Use Erase with caution; once the text is erased, it cannot be retrieved.

How To — To delete a block of text, select the block first. If you want to move it, choose Move-Cut (or press *F5* or *Del*). If you're sure you want to erase it, choose Erase.

See Also — Clipboard, Deleting, Edit Menu, Erase

# Deleting (General Information)

Keystrokes — *Alt-E* (or *F10*, Edit), Move-Cut, or
*Ctrl-KY*, or
*Ctrl-KK*, then type D,
or *F5* (deletes the selected block and copies it to the Clipboard)

*Del* (deletes the character at the cursor position or the selected text block)

*Ctrl-G* (deletes the character at the cursor position)

*Alt-E* (or *F10*, Edit), Erase (deletes the selected block without copying it to the Clipboard)

*Backspace* or *Ctrl-Backspace* (deletes the character or word, respectively, to the left of the cursor)

*Ctrl-H* (same as *Backspace*)

*Ctrl-T* (deletes the current word)

*Ctrl-QT* (deletes from the cursor to specified character)

*Ctrl-Y* (deletes the current line)

*Ctrl-QY* (deletes from the cursor to the end of the line)

*Ctrl-QDel* (deletes from the cursor to the beginning of the line)

*Ctrl-U* or *F6* (pastes text that's been deleted)

*Alt-E* (or *F10,* Edit), **Undelete** (pastes deleted text back where it started from, regardless of current cursor position)

*Alt-U* (or *F10,* **Utilities**), **Potpourri** (contains a variety of macros to delete text)

DeleteLine
DeleteLineBeg
DeleteLineEnd
DeleteParagraph
DeleteSentence
DeleteToChar
DeleteWord
DeleteWordLeft

*Alt-U* (or *F10,* **Utilities**), **Glossary**, **Erase** (deletes a Glossary entry)

*Alt-C* (or *F10,* **Customize**), **User Interface**, **Reset Shortcuts** (deletes all currently defined macro definitions)

*Alt-F* (or *F10,* **File**), **File Manager**, **Erase** or *Ctrl-KJ* (performs the DOS delete command (DEL or ERASE) and then returns to Sprint)

Function

Deletes text or an entry (see above).

Sprint's editing commands let you delete the ruler line, control characters, and text in a variety of ways.

Tips

You can delete a ruler line by moving to the ruler and pressing *Ctrl-Y.* If you delete the ruler, you can enter lines up to 32,000 characters in length; the editor will automatically scroll the display to the left so that you can still view the lines as they extend beyond the right edge of the screen.

If you want to delete a control character, such as those used to turn typestyles on and off, you need to first display the hidden codes. Press *Alt-Z.* Now move the cursor to the control character you want to delete and then press *Del.* When you're done, you can hide the control codes by pressing *Alt-Z* again.

The commands and shortcuts that copy text to the Clipboard listed at the beginning of this section do not permanently erase text from your file. For example, if you delete a word by pressing *Ctrl-T* or delete a line with *Ctrl-Y*, you can paste the text at the current cursor location by pressing *Ctrl-U* or *F6*, or by choosing **Edit/Insert-Paste**.

Sprint continues to append text to the Clipboard as long as you don't move the cursor between commands. If you move the cursor and enter a delete command, only the text affected by the last delete command will be kept on the Clipboard.

You can also "undelete" by choosing **Edit/Undelete**. This pastes the text you cut *back in its original location regardless of the cursor's current position.*

**Note:** The **Erase** command on the **Edit** menu does not copy your text to Sprint's Clipboard. Only use this command if you're absolutely sure you don't want to recover your text. Once you use this command, your text is gone for good.

The **Potpourri** utility contains several macros to delete lines, paragraphs, sentences, characters, and words, as well as deleting text to the beginning or to the end of a line. You can easily assign any of these commands to a keyboard shortcut of your choice.

See Also — Block Select Menu, Canceling, Potpourri

# Descending Order

Refer to the Arrange-Sort entry for details.

# Description

Keystrokes — *Alt-S* (or *F10*, **Style**), **Lists**, **Description**

Function — Creates a two-column format with headings on the left and descriptive paragraphs on the right.

This format prints a list of items or titles into two columns: a *subject* column on the left, and a paragraph of descriptive text on the right. Each subject (item) prints in boldface type against the left margin, and the text describing the subject prints in a block of plain text on the right. You use a *Tab* to separate the subject text and its descriptive paragraph.

The menu encyclopedia entries in this chapter are examples of a Description command at work.

You will find this command defined in the Sprint file STANDARD.FMT.

How To

Choose **Description** from the **Style/Lists** menu. Enter each subject and its description as a single paragraph; press *Tab* to separate the subject from the descriptive text. You must leave a blank line between the end of each description and the next subject. If you want to continue describing the same item in a new paragraph, begin the paragraph with a *Tab* character.

For example:

```
BEGIN DESCRIPTION
Apple    TabA red fruit that is very useful in making
apple pies. Apples have a light crunchy texture.

Orange   TabA round orange fruit that is used for
making juice, punch, and fruit cocktail. The best
varieties for eating are Navel and Valencia.

TabBy the way, "orange" is said to be the only
English word that cannot be rhymed.
END DESCRIPTION
```

Prints like this:

**Apple** A red fruit that is very useful in making apple pies. Apples have a light crunchy texture.

**Orange** A round orange fruit that is used for making juice, punch, and fruit cocktail. The best varieties for eating are Navel and Valencia.

By the way, "orange" is said to be the only English word that cannot be rhymed.

Tips

You must insert a blank line between paragraphs, so the formatter knows where one paragraph ends and the next begins. If you don't want this blank line to print, you can modify the Description format by adding the parameter *Spread 0*.

Likewise, if you want to change the amount of whitespace between the two columns, you can modify the format by giving it a new *Indent* parameter.

**Note:** Do not insert a ruler inside a Description format.

See Also

Lists, Modifying Formats

# Destination

Refer to the Print Menu entry for details.

# Dimensions

Function

Specifies distance in formatting commands.

Dimensions are a required part of many formatting *parameters*. For example, if you include the *LeftIndent* parameter in a format command, you must also include a *dimension* (such as characters, picas, or inches) to specify where you want the formatter to set the left indent. Similarily, several menu commands require that you enter a dimension. For instance, when you choose **Layout/Columns/Snaking Columns**, Sprint prompts you for the space between columns. You can enter any

horizontal dimension Table 1.2 in this entry (like 2 picas, 30 characters, 10.35 cm, and so on).

Unlike commands that affect ruler lines, commands that accept dimensions let you specify distance in measures other than column number.

**Note:** If you don't specify a dimension as part of a parameter, Sprint will automatically use *characters* for horizontal measures and *lines* for vertical measures. For example, *spacing* 2 tells Sprint to use the default *lines* and double-space the text.

Parameters are generally used to modify a defined format. For example, if you want to double-space text in the Display format, you could modify the Display format by adding the parameter *spacing* 2. (For a complete list of format parameters, refer to the Modifying Formats entry.)

You often use dimensions when you modify commands. (See the Modifying Formats entry for details.)

Table 1.2 lists all valid dimensions (and their legal spellings) and briefly explains each one.

Table 1.2: Formatting Dimensions

| | |
|---|---|
| char, chars, character, characters | The width of a typical character (normally the same as an "en" space). Since fonts can be different sizes, this measurement varies from font to font. Also, this measurement can only be used to indicate horizontal distances. |
| cm | Centimeters. |
| em, ems | Horizontally, the printer unit that is equal to the width of a lowercase *m* (the widest character in a proportionally spaced font). The "em" space varies from font to font.<br><br>Vertically, an "em" is the same distance as a line. |
| en, ens | The width of a 0 (zero) in the current font. |
| in, inch, inches | Inches. |
| line, lines | Vertically, this is the height of a single-spaced line (usually equal to the point size of the current font). Horizontally, this is the distance between the left and right margin. Also, for the *Spacing* and *Spread* parameters, entering the dimension in lines indicates that it is relative to the surrounding text's point size (rather than an absolute dimension). |
| mm | Millimeters. |
| page | The height of the paper, which is usually 11 inches. This dimension specifies vertical distance only. |
| pica, picas | Equal to 12 points, or approximately 1/6 of one inch. |
| point, points pt, pts, p | Equivalent to 1/72 of an inch. |
| u, unit, units | Derived from the printer definition, *u*'s represent the units that measure the minimal horizontal and vertical movement of the print head on the printer. This is useful for special effects, but is a printer-dependent dimension. Horizontal and vertical *u*'s can be different sizes. |

See Also

Blank Space, Formats, Indenting, Line Spacing, Margins, Tabs

# Direction

See the Searching entry for details.

# DisplayWrite 4 (DCA RFT)

Refer to the Translate entry for details.

# Document-Wide Menu

Keystrokes *Alt-L* (or *F10,* Layout), Document-Wide

Function Displays a menu of commands that affect the overall layout of your text.

Choosing this menu gives you access to these commands:

**Left Margin**
Choosing Left Margin from the Document-Wide menu lets you change the left margin for your entire document. The new margin will not be obvious until you print. (The default left margin is 1 inch).

**Right Margin**
Choosing Right Margin from the Document-Wide menu lets you change the right margin for your entire document. The new margin will not be obvious until you print. (The default right margin is 1 inch.)

**Top Margin**
Choosing Top Margin from the Document-Wide menu lets you change the top margin for your entire document. The new margin will not be obvious until you print. (The default top margin is 1 inch.)

**Bottom Margin**
Choosing Bottom Margin from the Document-Wide menu lets you change the bottom margin for your entire

document. The new margin will not be obvious until you print. (The default bottom margin is 1 inch.)

**Offset**
Choosing **O**ffset from the **D**ocument-Wide menu lets you set a so-called binding margin for your entire document. This margin adds the desired amount of space to the *inner* margin (alternating left and right margins) to facilitate binding. The new margins will not be obvious until you print.

**Paper Size**
Lets you select the size paper you will be printing on. The default setting is 8.5 × 11 inches. You can also choose **O**ther and type in the exact width and length.

**Word Spacing**
Sets the maximum extra space the formatter can insert between words when justifying lines. Sprint stretches the space between words first in order to justify a line; if this stretching has reached its maximum, and the line is still not justified, Sprint spreads out the letters of individual words.

**Inter-Paragraph Spread**
Specifies the distance between paragraphs by defining the depth of single blank lines in your file. For example, a setting of 12 points means the blank line between paragraphs will be formatted as 12 points of vertical space.

When you select this command, Sprint displays the following prompt:

```
Spacing to use between paragraphs:
```

You can use any valid vertical dimension; the default is *lines*. Setting this to 0 removes blank lines (you must still insert them in your file, however).

**Style Sheet**
Uses *filename*.FMT instead of STANDARD.FMT (the default file) when formatting a document. By default, the formatter reads the STANDARD.FMT file to retrieve definitions assigned to formatting commands in your file. The **S**tyle Sheet command lets you specify a different file (one that you create).

Also note that there are a handful of commands in Sprint that are *only* used when editing and creating style sheets or when creating new formatting commands. These commands are not covered here but are explained instead in the *Advanced User's Guide*. Refer to the appendix titled "Style Sheet Commands" for details. For your information, the commands to be found there are as follows:

| | |
|---|---|
| AtEnd | Parent |
| Define | ReadEPS |
| Error | Reset |
| Eval | VUnits |
| HUnits | Warn |
| If | @* |
| IfDef | @, |
| IfOdd | @; |
| Macro | @' |
| MakeOdd | @` |
| Merge_Init_ | @~ |
| PageInit | |

For instructions on creating your own .FMT file, refer to the "Custom Document Design" chapter in the *Advanced User's Guide*.

How To

Choosing **P**aper Size displays another menu of choices. The other **D**ocument-Wide commands prompt you for a number or a file name.

Tips

Sprint inserts all these commands as parameters to the Style command, which it places at the top of your file. The **S**tyle Sheet command appears as a Format command, which goes on the very first line of your file. These commands must be before any text, ruler, or other commands in your file.

Setting **W**ord Spacing to 1.5 or 2 can improve output but also can slow down some printers. Set it to a large number like 10,000 to disable the setting.

See Also

Margins, Paragraphs, Ruler, and Chapter 2 (Style)

# Double Line

Refer to the Line Drawing entry for details.

# DOS Command

Keystrokes

*Alt-U* (or *F10,* Utilities), DOS Command

*Ctrl-KF*

Function

Temporarily exits to DOS, where you can enter any DOS command.

This command is helpful when you want to execute (run) a program (typically a DOS program) without permanently exiting Sprint. This means you can enter the command from the Sprint editor, run the program, and then return to Sprint when the program is finished. For example, you can execute the DOS PRINT command from the Sprint editor; once DOS sends the specified file to the printer, it prompts to press any key to return to the file from which you entered the command after you press any key.

If you want to call a program in a directory that's not on the system search path, you can include the directory name as part of your file name.

Sprint always remembers the last DOS command you entered, so choosing this command usually displays a command on the prompt line already. You can press *Enter* to run this command again, or you can type over it.

If there is no command in response to the DOS command prompt, when you press *Enter,* Sprint temporarily "exits" to the DOS prompt. Type `Exit` at the DOS prompt, and you'll return to Sprint in the file you were editing when you pressed *Enter.*

This exiting to the DOS "shell" environment is useful if you have to perform several DOS commands in a row.

**Note to two-floppy system users:** Sprint may have to prompt you to insert the correct Program Disk into Drive A in order to choose Utilities/DOS Command.

A few words of caution:

- Do *not* load any RAM-resident programs (such as SideKick or SuperKey) with this command; these types of programs consume a permanent block of memory and will split your available Sprint memory in half. If you want to load a program such as SideKick, you should run it *before* you start Sprint, and you'll avoid this memory allocation problem. You can use the program at any time, once it's already loaded in Sprint, of course.
- Avoid running any program (like SPRECOVE.COM) that will modify the Sprint backup file (SP.SWP), since the editor is still in memory when you enter the **DOS Command**.
- For those of you using Sprint with a floppy-disk-only system, do *not* take out the disk containing the backup (swap) file or, if you have to, be sure you put it back in the drive *before* returning to Sprint.
- Do not delete the SP.SWP file—Sprint needs that file.

How To

When you choose **DOS Command**, Sprint displays the following prompt in the status line:

```
DOS command:
```

Enter the command you want to execute. For example,

```
DOS command: COPY *.RPT B:
```

Sprint copies all files with the .RPT extension from the current directory to Drive B and then returns to the file you were working on when you selected the **DOS Command**. If it doesn't find any file, Sprint displays the usual DOS messages `File not found` and `Press any key to continue.`

See Also

File Manager Menu

# Draw Box

Refer to the Graphics entry for details.

# Duplicate-Copy

Refer to the File Manager Menu entry for details.

# Edit Menu

Keystrokes

*Alt-E* (or *F10*, Edit)

*F9* (jumps to a specified line)

*F4*, *Ctrl-KC*, or *Ctrl-KK* and type C (copies text)

*F5*, *Del*, *Ctrl-KY*, or *Ctrl-KK* and type D (moves text)

*F6*, *Ctrl-U*, or *Ctrl-KV* (pastes text)

*F7* or *Ctrl-QF* (finds text)

*Alt-G* (goes to mark)

*Alt-M* (sets mark)

*Ctrl-KW* (writes block to file)

*Ctrl-QA* (searches for and replaces text)

Function

Displays the Edit menu, which lists Sprint's primary editing functions:

**Undelete**
Pastes the Clipboard text back where it came from, regardless of the current cursor position.

**Copy** *F4*, *Ctrl-KC*
Leaves the block in its current location and puts a copy of the marked text in Sprint's Clipboard. To retrieve this text, press *F6* (or *Ins*) or choose Insert-Paste.

**Move-Cut** *F5*, *Del*, or *Ctrl-KY*
Removes the block from its current location and puts it in Sprint's Clipboard. To retrieve this text, press *F6* or choose Insert-Paste.

**Insert-Paste** *F6*, *Ctrl-U*, or *Ctrl-KV*
Inserts the block that's currently in the Clipboard at the current cursor position.

**Erase**
Erases the block from its current location and does *not* put it in the Clipboard. Use with caution; you cannot retrieve a block of text once it's been erased.

**Block Select**
Displays a list of commands that let you move, copy, erase, and paste a marked block of text. You can also write a marked block to a file on disk.

**Write Block** *Ctrl-KW*
Leaves the block in its current location and also places it in another file. Sprint prompts for the name of the file to receive the block. The selected text remains highlighted, so you can delete it or move it if necessary.

**Search-Replace** *F8, Ctrl-QA*
Displays the commands that let you search for and replace text. Sprint first prompts for the string that you want to find and then prompts for the replacement string. When Sprint finds the text, it displays a Replace this? menu, which allows you to specify **Yes**, **No**, or **And** the Rest.

**Go to Page**
Moves the cursor to a specific page number. You must have paginated your file at least once before choosing this command.

**Jump to Line** *F9*
Moves the cursor to a specific line number. You don't have to paginate your file before choosing this command.

**Place Mark** *Alt-M* (set) or *Alt-G* (go)
Displays two commands that allow you to set marks in your text and use those marks to move quickly around your file.

See Also — Block Select Menu, Deleting, Paginate, Place Mark, Searching

# Edit on Screen

Refer to the Ruler entry for details.

# EMACS

Refer to the User Interface Menu entry for details.

# End (Format)

Refer to the Begin and End (Formats) entry in Chapter 2 for details.

# Ending Page

Refer to the Print Menu entry for details.

# Ending Record

Refer to the Merge entry for details.

# Endnote

Keystrokes — *Alt-S* (or *F10*, Style), **R**eferences, Endnote

Function — Creates a numbered note at the *end* of the document.

This command is similar to Footnote, except that it places the numbered text at the end of your document rather than at the bottom of the page. When you format your document, Sprint numbers the endnote and places it on the special Endnotes page, which appears just before the Index and which it titles with a bold, large heading **Notes**. Sprint numbers endnotes consecutively; the first endnote in your file is numbered 1, the second is numbered 2, and so on.

You can, however, use the formatter command Set to make Sprint renumber endnotes (for example, at the beginning of a chapter). You can also use the commands called NoteChapter and NoteSection to create sections within your endnote listings, referencing chapter and section, respectively.

If you want your endnotes to appear somewhere other than the end of the document, you can use the format command Place to indicate where you want Sprint to print your notes.

You will find this command defined in the Sprint file STANDARD.FMT.

How To

The **Endnote** command works on a marked block of text. To create an endnote, type the text of your note, select the text, and choose **Style/References/Endnote**.

Sprint inserts the onscreen BEGIN and END ENOTE commands wherever you choose an **Endnote** command.

If you want the Notes page to include the chapter or section numbers, use the format commands NoteChapter and NoteSection. On the new line following your **Chapter** and **Section** commands, choose **Style/Other Format** and enter NoteChapter or NoteSection. (Press *C* to insert it as a command.) When the formatter prints your Notes page, it will include the chapter number and title, followed by the endnotes contained in this chapter. If you've entered the NoteSection command, you'll also see the number and title of each section, followed by the endnotes contained in each section.

One other use of the variable *Footnote* is to assign the same endnote number to two different passages. If both passages relate to the same topic, and a single endnote applies to both, make your first endnote reference, using the Endnote command. For the second reference, choose **+ Superscript** from the **Typestyle** menu, then choose **Insert/Variable/Other**, and type `Footnote`. Finally, choose **None** from the **Pick Template** menu. This procedure references the *Footnote* variable directly and prints it in superscript typestyle.

Use the Place command to tell the formatter where to print all of your endnotes.

Tips

If you want the notes page to be titled "Bibliography" instead of "Notes," modify the *Notes* definition in your .FMT file.

Don't mix endnotes with footnotes in the same document. Sprint numbers both types of notes using a single series of numbers. As a result, your numbering will switch from the bottom of the page, to the end, and back again.

See Also — Footnote and Chapter 2 (NoteChapter, NoteSection, Place)

# Enter (Macro)

Refer to the Macros entry for details.

# Entire File

Refer to the Searching entry for details.

# EPS Picture

Refer to the Graphics entry for details.

# Erase

Keystrokes — *Alt-E* (or *F10*, Edit), Erase

Function — Irretrievably deletes a marked block.

The Erase command deletes a marked block of text *without* putting it in the Clipboard. This means the block cannot be pasted with *Ctrl-U* or *F6*.

How To — To permanently delete a block of text, select the block you want to erase, and choose **Edit/Erase**. If you want to *move* rather than erase the block, press *F5* or choose **Move-Cut**. (Refer to the Move-Cut entry for details.)

Tips — If you think you might later use the block to be erased, write it to a file. Select the text, and choose **Edit/Write Block**. This is the only way to save text that is erased.

See Also — Deleting, Edit Menu

# Erase (Glossary)

Refer to the Glossary entry for details.

# Erase (File)

Refer to the File Manager Menu entry for details.

# Error Messages and Warning Messages

Keystrokes *Alt-P* (or *F10*, Print), Advanced Options, Log Errors to File

Function Writes to a file any error or warning messages that Sprint displays while formatting your document.

Choosing Log Errors to File from the Print/Advanced Options menu, toggles the command. If set to Yes, Sprint displays the error and warning messages onscreen and also places them a file with the same name, but with the .LOG extension. If set to No, Sprint briefly displays the messages onscreen, but does *not* write the messages to a file.

Logging messages to a file facilitates fixing your errors, since you can easily go back to the .LOG file to note the line numbers where your errors occur, and then fix your document so it prints correctly.

Sprint will not actually print your document if errors are encountered during its formatting pass. A typical error message might read

```
Error: Begin Numbered on line 20 missing End
```

This message tells you that you started a Numbered format on line 20 but you forgot to end it.

If you can't find the error in your document file on the exact line number given in Sprint's message, start searching backward for the specific error.

Refer to Appendix A for a list of error messages.

A "warning" (as opposed to an error) during a formatting pass is just that—an alert message that something *might* be awry at the line number given. Sprint prints the file even if it displays one or more warning messages (as long as there are no error messages as well, of course). You might find that parts of your document don't print as you wanted when Sprint warns you of a possible problem.

A typical warning might read

```
Warning: Undefined tag 'insideout' used in
C:\SPRINT\DVTEST.SPR line 34
```

This message tells you that a tag was referenced on line 34 that Sprint never found a definition of.

Note that such undefined-tag warnings occur at the end of the formatting pass because Sprint does not know if a tag is undefined until it has gone through the entire document. (If the tag called *insideout* was defined in the very last line of the document, there would have been no warning.) In most of these cases, two line numbers appear: the first number refers to the last formatted line, and the second is the actual line number where the undefined tag appears. A warning message of this sort happens, for example, when the tag you're referring to appears in another chapter or section that is not currently being formatted for print. Sprint replaces these undefined tags with three question marks in the printout, for example, *"see Chapter ???"* or *"refer to page ???."*

Tips

An error message referring to an "unknown command" is usually the result of

- entering a command from the Style/Other Format prompt, and forgetting to include a space between the command name and the first word following the command (for example, "Error: Unknown command 'BEGINSTYLE'")
- entering an @-sign as text and forgetting to double the @-sign (like so: @@), or using an @-sign immediately preceding text that is not a recognizable Sprint command (for example, "Error: Unknown command @compounded")

You can use the **Paginate** command for an easy way to spot formatting errors in your file. If Sprint finds any formatting errors when paginating, it inserts a bold line and an error message at the location of each error. This makes it easier to spot and fix errors in your document before printing.

See Also — Advanced Options, Print Menu, Chapter 3, Appendix A

# Even Pages

Refer to the Footer Menu or Header Menu entry for details.

# Every Bad Word

Refer to the Spelling Menu entry for details.

# Export

Refer to the Translate entry for details.

# Figure

Keystrokes — *Alt-S* (or *F10*, Style), Figure

Function — Creates a format for figures.

This command works similar to the **Group Together on Page** format. It keeps all text, blank lines, and the figure caption (if you enter one) together on a page. If there isn't room for the figure block and caption on the current page, Sprint starts the Figure format at the top of the next page.

The Caption command and any tags set for referencing the figure number should appear within the Figure format. The Caption command assigns the figure number and lets you title your figure. The caption is centered horizontally and prints in a small font.

You will find this command defined in the Sprint file STANDARD.FMT.

How To

Let's say you want to leave room for a figure to be pasted in. Choose Style/Figure. Sprint prompts for a caption; if you do not want a caption under the figure, just press *Enter*. When you've finished, Sprint inserts the Begin and End commands into the file. Create the necessary blank space by using the Reserve Space command (choose Layout/Page Breaks/Reserve Space). This enters the formatter command RESERVE.

If you want to cross-reference this figure, choose the Define a Tag command from the X-Reference menu and create a tag, making it equal to the variable named *Figure*. Choose the Reference a Tag command and type the tag name when you want to reference your figure. For example,

```
BEGIN FIGURE
RESERVE 20 picas
CAPTION Alfred E. Newman at the Beach
TAG beach=figure
END FIGURE

As shown in Figure beach, you can meet interesting
people at the beach.
```

The sentence prints like this:

> As shown in Figure 1, you can meet interesting people at the beach.

Tips

**Note:** The Figure command by itself does not create an entry in the table of contents. If you want a List of Figures to print as part of the table of contents, be sure to include a Caption command within each of your Figure formats.

If you want a caption with your figure but *don't* want it in the List of Figures, you should use the FCapt command. Choose Style/Other Format and enter `TCapt`, followed by the caption text. When prompted, press *C* to tell Sprint that this command does not affect a region of text.

If you have a PostScript printer, you can have a graphic figure print in the reserved space by choosing Style/

Graphics/EPS Picture. The graphic must be saved in encapsulated PostScript (EPS) format. The screen shots in the Sprint manuals were printed in this way. You could also draw a box around the figure using the **Draw Box** command (from the **Graphics** menu), which should go within the **Figure** command.

Be careful not to make the area defined in the Figure format *more* than one page long! If you do, you'll find your blank space or text of the figure on one page, and the caption at the top of the next page. Also, you might want to add a blank line or some sort of divider (such as a solid line created with the **Insert/Repeating Character** or, if your printer supports PostScript, with the **Bar** commands) to separate the figure visually from the text above or below it.

Be sure you insert a tag for a figure on the line *after* the caption. If you don't, Sprint will print an incorrect reference.

See Also — Caption, Graphics, Repeating Character, Reserve Space, Tags

# Figure (Variable)

Refer to the Variables entry for details.

# File (Hyphenate)

Refer to the Hyphenation Menu entry for details.

# File Manager Menu

Keystrokes — *Alt-F* (or *F10*, File), **File Manager**

*Ctrl-KE* (renames file)

*Ctrl-KJ* (erases a file)

*Ctrl-KL* (changes current directory)

Function

Allows you access to DOS file commands without leaving Sprint.

The File Manager menu lets you perform common disk operations from within Sprint. These include the following:

**Duplicate-Copy**
Lets you make a copy of a file to another directory or under another name. It is equivalent to the DOS command COPY.

**Rename-Move**
Lets you give a new name to a file; it is equivalent to the DOS command REN with one exception: You can use it to move a file from one directory to another. In other words, it's equivalent to these *two* DOS commands:

```
COPY MEMO.SPR C:\STUFF
DEL MEMO.SPR
```

**Erase**
Lets you delete a file on disk; it is equivalent to the DOS command DEL or ERASE.

**Change Directory**
Lets you insert a new default path name that Sprint uses to save to and read from. The command is the same as the CHDIR command in DOS.

**List Directory**
Shows you all the files in your current directory that meet your specifications. Like the DOS command DIR, this command uses the wildcards * and ? to select which files to show. If you want to see *all* files, enter *.* at the prompt. If you just press *Enter* at the prompt, Sprint displays all files with the .SPR extension. If there are more files than can be shown on a specific screen, you can use your cursor keys to see more of the list. You can also use *PgUp* or *PgDn* to see another screen's worth of files, or *Home* and *End* to take you to the top or bottom of the list.

Selecting a file from the list makes it the "current" file, ready to be used by the Duplicate-Copy, Rename-Move, or Erase commands.

How To

Most of the commands closely follow their DOS equivalents and so should present no problems. You can also perform all these commands from the "DOS" shell to Sprint (*F10*/**U**tilities/**D**OS Command). The **F**ile Manager commands are usually more convenient, however.

Tips

You can use DOS wildcards as part of a file name in any of these commands. Refer to your DOS documentation if you need help, or refer to Appendix A in the *User's Guide* for an introduction to DOS commands.

If you have a three-button mouse, you can use it as a shortcut for moving from one open file to another. Clicking the middle button in an area to the right of the onscreen margin (that is, where there is no text) affects the current file. (If you are working in a file that has no ruler, the non-text area is anywhere to the right of a hard return paragraph mark.) Clicking in the upper quarter of the non-text zone switches to the previous file. Clicking in the lower quarter of the non-text zone switches to the next file. Clicking in the middle part of the non-text zone calls the Pick from List file menu.

See Also

File Menu

# File Menu

Keystrokes

*Alt-F* (or *F10*, **F**ile)

*Ctrl-F2* (saves a file)

*Ctrl-F3* (opens a file)

*Ctrl-F4* (closes a file)

*Ctrl-F9* (displays a list of open files)

Function

Displays the **F**ile menu.

The **F**ile menu lists the commands that allow you to access and manipulate the files stored on your disk. The commands are

**New**
Allows you to create a new file. Pressing *Enter* without entering a name results in Sprint's opening a new file with "Unnamed" displayed in the status line.

**Open** *Ctrl-F3*
Opens a new or accesses an existing file.

**Close** *Ctrl-F4*
Closes an open file.

**Insert**
Prompts for the name of the file you wish to insert or read into the file that is currently open.

**Save** *Ctrl-F2*
Saves an open file to disk.

**Write As**
Writes the current file to a different file name on disk. Sprint prompts for the name of the file to which you want to write this file. The original file remains unchanged on disk.

**Revert to Saved**
Reverts to last saved version of the file (ignores changes made since last save).

**Translate**
Displays a list of file-translation commands. Refer to the Translate entry for more information.

**File Manager**
Lists a number of file-related DOS functions. Refer to the File Manager entry for more information.

**Pick from List** *Ctrl-F9*
Lists the currently open files and lets you select the file you want to access.

**Note:** Sprint's default file extension is .SPR. If you open a new file and give it a name with no extension, Sprint automatically tacks on the .SPR. If you want to create a file with no extension at all, you have to enter a name that ends with a period.

Similarly, when opening an existing file, if you don't specify a file extension, Sprint looks for a file by that name. If it can't find one, it looks for a file by that name with the .SPR extension.

You can use normal DOS wildcards (* and ?) at any appropriate prompt when dealing with file-related commands in Sprint.

See Also File Manager Menu, Translate

# File (Spelling Menu)

Refer to the Spelling Menu entry for details.

# Find

Refer to the Searching entry for details.

# FirstPage (Variable)

Refer to the Variables entry for details.

# Font

Keystrokes *Alt-L* (or *F10*, **Layout**), **Ruler**, **Precise Settings**, **Font**

*Alt-T* (or *F10*, **Typestyle**), **Font**

Function Specifies a particular font for printing.

The **Typestyle/Font** command affects blocks of selected text. The **Layout/Ruler/Precise Settings/Font** command affects all text affected by the current ruler.

The **Font** command displays a list of fonts available for your selected printer and tells you the characteristics (italic, bold, underlined, plain) of the character at the cursor. Choosing this command from either menu displays a list of fonts supported by your currently selected printer. If you select a block of text, you can specify the font you want to use to print the text in this block.

If you printer does not support the specified font, Sprint displays a warning message when it formats your file, but prints your file in the default font anyway.

When you installed your printer, Sprint prompted for the default font (if your printer is capable of printing different fonts). This means that you only need to use the Font commands if you want your file, or part of your file, to print in a font other than the default.

How To

*Selecting a Font for the Entire Document*

When you installed Sprint for your printer using the SP-SETUP program, you created a default font to use when printing. Ordinarily, your entire document prints in this font. You can, however, change the font a particular document prints in.

If you have only one ruler line in your document, you can change the font of your whole document by choosing **L**ayout/**R**uler/**P**recise Settings/**F**ont and then selecting the font you want. If you have more than one ruler line in your document, you can also specify a font for the entire document using the formatter command Style and the *Font* parameter. For example, at the top of your file choose **S**tyle/**O**ther Format, and enter the following command:

```
Style Font = Times Enter
```

Then press *C* to complete the command. This tells the formatter to print the file in a Times font.

*Changing Fonts within a Document*

The easiest way to specify a font for a particular area of your file is to set off the text with rulers and choose **L**ayout/**R**uler/**P**recise Settings/**F**ont.

If you only want to affect a small amount of text (like a word or a single line), you can select the text as a block and choose **T**ypestyle/**F**ont. When you choose this command, Sprint inserts the onscreen command FONT followed by the font name in front of the selected text and inserts the ENDF command to mark the end font.

For example, if you want to change the font of a couple of words in a sentence, your screen would look like this:

```
The restaurant sign read: FONT ChanceryPlease wait to
be satedENDF.
```

This prints as

The restaurant sign read: *Please wait to be sated.*

***Using Your Main Font within an Alternate-Font Format***
Some formats, such as Example, automatically cause Sprint to change to a fixed-width or typewriter-like font. You can still use your main or *default* font within a format like Example by modifying the format (choose **Style/Modify** and enter the font parameter).

Tips

You can modify an existing format and add the parameter *Font=FontName* to change the font used to print the text within the format.

If your computer uses the Hercules InColor card or the Hercules Graphics Card Plus, your *onscreen* fonts can be made to be quite varied. Choose **Customize/Colors/Modify/Typestyle** to choose from a palette of fonts and font styles.

See Also

Colors, Font Size, Modifying Formats, Typestyle Menu

# FontName (Variable)

Refer to the Variables entry for details.

# Font Size

Keystrokes

*Alt-T* (or *F10*, **Typestyle**), **Large**

*Alt-T* (or *F10*, **Typestyle**), **Character Size**

*Alt-L* (or *F10*, **Layout**), **Ruler**, **Precise Settings**, **Size**

Function

Prints characters in a different font size.

The **Large** command prints characters in a large font (if your printer has that capability) and in bold type so that you can emphasize important text, like warnings and cautions. The STANDARD.FMT file includes this com-

mand in many of its titling commands (Chapter, HeadingA, and so on).

You will find this command defined in the Sprint file STANDARD.FMT.

How To

To print text in large, bold type, mark the text and choose **Large** from the **Typestyle** menu. The editor highlights the selected text. When you print, the type will come out big and bold—if your printer is up to that. For example,

```
This is B-I-G Type!
```

Prints like this:

# This is B-I-G Type!

The **Typestyle/Character Size** command lets you specify a particular size for any amount of selected text. You can use a variety of dimensions to express this size (see list on page 58), and you must be using a printer that can scale fonts (like a PostScript printer). When you choose this command, Sprint inserts the onscreen command SIZE followed by the character size in front of the selected text and inserts the ENDS command onscreen to mark the end of the size change.

For example, if you're normal font size is 10 points, but you want to print a few words in 12-point type for emphasis, your screen might look like this:

```
Remember, this is due by SIZE 12 POINTSnoon
tomorrow!!ENDS
```

The sentence prints with the last two words 2 points larger than the rest of the sentence.

The **Precise Settings/Size** command lets you specify a font size for all the text governed by the current ruler.

You can also specify a size smaller than the default font size if your printer is capable of scaling fonts (as all PostScript printers are).

On most non-PostScript printers, however, you can explicitly use a special font called "Small" that Sprint uses by default in certain instances (as in footnotes). You

can use the Small font by selecting the text and choosing Style/Other Format, entering `Small`, and then pressing *R*.

See Also Font, Precise Settings Menu, Typestyle Menu

# Footer Menu

Keystrokes *Alt-L* (or *F10*, Layout), Footer

Function Displays a menu of commands to create page footers.

The Footer menu options are:

**All Pages**
Inserts into your document the command for a footer that appears on all pages *except the first*. The onscreen command name for this type of footer is FOOTER.

**Title Page**
Inserts a command at the top of your document to print a footer on the first page only. The onscreen command name for this type of footer is FOOTERT.

**Odd Pages**
Inserts a command at the current cursor position to create a footer on all odd pages (except page 1, which is handled with the Title Page command). The onscreen command name for this type of footer is FOOTERO.

**Even Pages**
Inserts a command at the current cursor position to create a footer on all even pages. The onscreen command name for this type of footer is FOOTERE.

**Position**
Lets you determine the exact place the footer will start printing.

Sprint has a default footer line, which centers the current page number in the footing line at the bottom of the page. If you choose any of the Footer commands, you override this default footer, but you can still include the page number as part of your footer text.

Most of the Footer commands create so-called *running footers*—text that automatically appears in the bottom

margin of some or all pages of your formatted document. You can insert a **Footer** command anywhere in your document, and the text of the footer will appear on all pages following the command.

A footer can contain one or more lines of text, can have embedded formatting commands or rulers, and can have alternating formats for left and right pages (as this *Reference Guide* has).

How To

To create a running page footer, press *Alt-L* and choose **Footer**. Then choose the particular type of footer you want.

If you choose conflicting footers (for example, if you choose **Odd Pages** and then choose **All Pages**), Sprint carries out the last-entered footer command.

After you make your selection, Sprint displays the Begin and End commands and places your cursor between them. Type the footing text after the Begin command. Your footing can be as many lines as you wish.

If you want text to print at the *right* margin, choose **Insert/Wide Space (Spring)** before you type the text.

If you want to set-up alternating footers so that one footer line appears on odd-numbered pages and another, different footer line appears on even-numbered pages, you can choose two **Footer** commands. For example, choose **Odd Pages** from the **Footer** menu and type `Confidential`. Then choose **Even Pages** from the **Footer** menu and type `Annual Report`. Your screen will look like this:

```
BEGIN FOOTERO
Confidential
END FOOTERO

BEGIN FOOTERE
Annual Report
END FOOTERE
```

Sprint will print "Confidential" on every odd-numbered page of your text and "Annual Report" on every even-numbered page.

When you want to eliminate page footers, choose the **Footer** command but don't enter any text. This tells Sprint to leave your footer lines blank.

If you want the footer to print text *and* a page number, you need to use the **Insert/Variable** command and choose the variable *Page*. This tells the formatter to determine the current page number when printing the footer and insert this page number wherever specified in your **Footer** command.

For example, if you want to print the text "Annual Report, 1988" in the bottom left part of your footer and the page number in the bottom right of your footer, choose the **Footer** command and type `Annual Report, 1988`. Choose **Insert/Wide Space (Spring)** and then choose **Insert/Variable**. Choose the variable *Page*. When your document prints, the formatter will automatically replace the variable *Page* with the correct page number.

You can use any of Sprint's other variables as part of your footer text. For example, if you want the footer to include the current chapter number, choose **Insert/Variable/Other** from the menus. When Sprint prompts for the variable to reference, type `Chapter`. When your document prints, the formatter will print the current chapter number as part of your footer text. If you also want the word "Chapter" to print, type `Chapter` followed by a space, and then choose the **Insert/Variable** command.

If you want your footer text to print in bold type, or any other typeface listed on the **Typestyle** menu, mark your footer as a block and then choose the desired **Typestyle** command.

You can also insert a ruler within the **Footer** command and format the text using the usual ruler editing techniques.

The **Footer** menu also lets you select a specific position for your footer. Choose **Position** and Sprint prompts for the spacing below the last line of text where the footer should go.

If you want a page footer that affects only the first page of your file, you should choose **Title Page** from the **Footer** menu.

If you want a footer that affects *all* pages (including the first), choose **Title Page** and then choose **All Pages** and enter the same text in both.

To print a footer that reads "page *x* of *y*" (that is, one that tells you what page you're on in relation to the *total* number of pages), you can do this:

1. On the last line of your document, insert a tag using the **Define a Tag** command:

   ```
   TAG lastpage
   ```

2. In the footer, reference this tag as well as the value of the *Page* variable:

   ```
   Page page of PAGEREF lastpage
   ```

When Sprint prints your document, the footer will read "Page 1 of 30", "Page 2 of 30", and so on.

See Also — Header Menu, Page Numbers, Tags, Variables

# Footnote

Keystrokes — *Alt-S* (or *F10*, **Style**), **References**, **Footnote**

*Alt-S* (or *F10*, **Style**), **References**, **Note**

Function — Prints a footnote at the *bottom* of the current page.

The **Footnote** command inserts a numbered footnote; the **Note** command inserts an unnumbered one that uses asterisks instead.

When Sprint formats and prints your file, it replaces the **Footnote** command with a small, raised number (if your printer has this capability), and then places the numbered footnote text at the bottom of the current page. If your printer doesn't support a small font, Sprint prints the number in plain text; if your printer can't perform vertical microspacing, Sprint places the number one-half line above the text to be referenced. If it can't do that, it prints the footnote number on its own line.

If your footnote text is quite lengthy, the formatter will automatically extend the footnote to the next page.

The Notes command operates just like the Footnote command, except that the notes use an asterisk (*) as the reference marker instead of a number. If you use two Notes commands on the same page, the second note will be referenced with two asterisks.

How To

There are two ways to invoke the Footnote command:

- You can type the text of your footnote, select it, and then choose the Footnote command. Sprint inserts a BEGIN and END FNOTE command around the selected text.
- You can choose the Footnote command and then type the text of your footnote. When you choose the command, Sprint inserts the BEGIN and END FNOTE commands and places the cursor between them, ready for you to type the text.

Either way, your footnote text will appear in the middle of your normal file text, like this:

```
As a result of the XYZ affair,BEGIN FNOTEThis is the
sample footnote.END FNOTE and because of other
things...
```

This may look a little strange when you're editing your file, but when you print, the formatter automatically replaces the onscreen FNOTE command with a number (beginning with 1 and incremented with each Footnote command) and places the number and text of your footnote at the bottom of the page. For example, the previous Footnote command produces the following when you print:

As a result of the XYZ affair,[2] and because of other things...

Look at the bottom of this page for the actual footnote.

Tips

Resetting a footnote is useful when you are printing a partial document. For instance, if you're printing only the second half of a document, you might want you footnotes to start with 23. To do this, type the following

2. This is the sample footnote.

formatter command (first choose **Style/Other Format**) before the next **Footnote** command:

```
SET Footnote 22
```

Note that you'd type "22" not "23." **Footnote** always adds 1 to the *Footnote* variable to get the new footnote value.

If you want to renumber your footnotes with each chapter (that is, restart footnote numbering with the number 1 whenever you start a new chapter), type the formatter command

```
SET Footnote 0
```

at the beginning of each chapter. If you include this formatter command in a **Header** command, footnotes will be numbered from 1 on each page. (You can also modify the macro for @Chapter in the .FMT file if you want Sprint to do this automatically.)

One other use of this variable is to assign the same footnote number to two different passages. If both passages relate to the same topic and a single footnote applies to both, make your first footnote reference using the **Footnote** command. For the second reference, choose **+ Superscript** from the **Typestyle** menu, then choose **Insert/Variable/Other**, and type in `Footnote`. (Choose **None** for no template at the next menu.) This procedure references the *Footnote* variable directly and prints it in superscript typestyle.

The **Notes** command operates just like the **Footnote** command, except that the notes are referenced with asterisks instead of numbers (one asterisk for the first one, two for the second, and so forth). Sprint inserts the SNOTE command onscreen (for "star note") wherever you place a note.

See Also — Endnote, Variables, Page Numbers, and Chapter 2 (Set)

# Footnote (Variable)

Refer to the Variables entry for details.

# Format (Editing)

Refer to the Modifying Formats entry for details.

# Format Parameters

See the list at the end of the Modifying Formats entry on page 149.

# Formats (General Information)

Keystrokes — *Alt-S* (or *F10*, Style)

Function — Affects the look of characters, words, lines, paragraphs, and marked blocks of text when you print.

Formats are the most common and most powerful type of Sprint commands. Almost every command from the **S**tyle menu is a format command. Moreover, many commands from other menus are format commands as well (especially **T**ypestyle and **L**ayout). What distinguishes these commands is that their effects are not seen until you print your document. Instead, format commands insert special control characters (and often special command words within the control characters) in your document. The control characters signal how the text should be formatted when you print.

In addition to the many format commands available from the menus, there are dozens more that are not (the menus would have become too large and confusing if we had included all of them). You can still input them from the menu structure, however, using the catch-all command **O**ther Format (on the **S**tyle menu).

Note that some formats act on blocks of your document's text (for instance, **D**escription, **N**umbered, and **F**ootnote), and others do not (for instance, **R**eserved Space, **C**olumn Break). Common sense will tell you which are which. If the command affects a block of text (be it one character or pages of text), you should mark the text as a block before choosing the command. If you

forget to mark the text, Sprint will let you specify whether this is the beginning or the end of the block (region). Every format command affecting a block must have an explicit begin and end statement.

Any format that has a Begin and End can be modified with certain parameters. The best way to do this is to choose the **Style/Modify** command. The cursor jumps to the last-entered format, and Sprint prompts you to choose to edit that format or the one before it. In either case, you can then enter any applicable parameter (as discussed in the Modifying Formats entry) by typing it in. Press *Enter* to return to your last cursor position.

The Style command in the .FMT file you're using, or a Style command entered at the top of a file, specifies a general, document-wide format. For example, the Style command can specify margins in a dimension other than column number (like, *Offset 3 picas, LineLength 28 picas*), define character size and line spacing, and so on. When you want to temporarily deviate from this general format, you can choose a format to create the look you want. For example, choose the **Numbered** format when you want the formatter to automatically number the steps of a procedure. Once you end a format, Sprint resumes formatting the text as before.

Refer to the individual entries for the format commands for further information.

See Also — Modifying Formats, Style Menu

# Formatted Print

Refer to the Advanced Options entry for details.

# Function Keys

Keystrokes — *Alt-C* (or *F10*, **Customize**), **User Interface**, **Function Keys**

Function — Provides shortcuts to editing, file, and window-related functions.

The meaning assigned to each of your keyboard's function keys depends on the user interface you've selected. Table 1.3 lists the function key assignments for the standard user interface (as defined in SPTUTOR, SPBASIC, or SPADV).

(**Note:** Function keys are labeled *F1* through *F10* on IBM PC-compatible keyboards. If you have function keys labeled 11 or higher, they will display the main Sprint menu. The Function Keys command in Sprint includes these traditional function keys as well as *Ctrl* and *Alt* shortcuts.)

Table 1.3: Function Key Assignments (Standard User Interface)

| Key | Function |
|---|---|
| ***Edit-Related Functions:*** | |
| *F1* | Help |
| *F2* | Glossary recall |
| *F3* | Select mode On/Off |
| *F4* | Copy |
| *F5* | Cut |
| *F6* | Paste |
| *F7* | Find |
| *F8* | Search and replace |
| *F9* | Jump to line |
| *F10* | Main menu |
| ***File-Related Functions:*** | |
| *Ctrl-F1* | Correct every bad word |
| *Ctrl-F2* | Save file |
| *Ctrl-F3* | Open file |
| *Ctrl-F4* | Close file |
| *Ctrl-F5* | Previous file |
| *Ctrl-F6* | Next file |
| *Ctrl-F7* | Paginate file |
| *Ctrl-F8* | Print onscreen |
| *Ctrl-F9* | Pick file |
| *Ctrl-F10* | Main menu |
| ***Window-Related Functions:*** | |
| *Shift-F1* | Spell menu |
| *Shift-F2* | Resize window |
| *Shift-F3* | Open window |
| *Shift-F4* | Close window |
| *Shift-F5* | Zoom window |
| *Shift-F6* | Next window |
| *Shift-F7* | Scroll all up one line |
| *Shift-F8* | Scroll all down one line |
| *Shift-F9* | Close all windows |
| *Shift-F10* | Main menu |
| ***Miscellaneous Functions:*** | |
| *Alt-F1* | Thesaurus |
| *Alt-F10* | Main menu |
| *Shift-Tab* | Indent character |

How To

You can change a function key definition by choosing the **Customize/User Interface/Function Keys** command. When you choose this command, you'll see an information box explaining what keys you can change (the function keys, any modified version of them, such as *Alt-*

*F3* or *Shift-Alt-F4*, or any *Alt-* or *Ctrl*-version of letter keys or numeric keypad keys).

Sprint prompts you for the key to be defined. Press, for example, *Alt-F3*. Sprint then prompts you for the key that currently performs this command. Press, for example, *Ctrl-PgDn*. Sprint lets you keep redefining function keys until you answer No to its `Re-assign another?` prompt. When you return to Sprint editing, you'll find that pressing *Alt-F3* performs the job that *Ctrl-PgDn* does—it moves your cursor to the end of the file.

If your keyboard has *F11* and *F12* keys, they will function like *F10*, to display the main menu.

**Note to two-floppy system users:** Sprint may have to prompt you to insert the correct Program Disk into Drive A in order to use this command.

Tips

You can get a handy list of all the function key meanings by choosing the **Utilities/QuickCard** command, which creates a text file of all the shortcuts for you.

You can also have Sprint generate a visual map of function key definitions by pressing *F1* to display the Help menu and then pressing *F1* again.

To delete the function key reassignments, choose **Customize/Reset Shortcuts**.

See Also

Menu Shortcuts, QuickCard, User Interface Menu

# Glossary

Keystrokes

*Alt-U* (or *F10*, **Utilities**), **Glossary**

*F2* (recalls Glossary entry)

Function

Displays the **Glossary** menu.

Sprint's **Glossary** menu lets you assign a string of characters, a useful macro, or a set of recorded keystrokes to an abbreviation, and it allows you to define boilerplate lines or words to an abbreviation as well. When you choose **Recall** (*F2*) and type the abbreviation, Sprint expands the abbreviation to its full text.

The menu commands are as follows:

**Recall** *F2*
Lets you expand saved glossary abbreviations. Sprint first tries to expand automatically from the glossary file by looking up the word to the left of the cursor. If that word is not in the list of abbreviations, Sprint prompts you for the name of the glossary item to insert. If you can't remember the name, press *Enter* for a list to choose from.

**Assign to Key**
Lets you assign to a keystroke the boilerplate text that you've already given a glossary abbreviation.

**Define**
Sprint prompts for the glossary abbreviation. If you have text selected, that text is saved to the glossary name. If no text is selected, the contents of the Clipboard is used.

**Keyboard Record**
Lets you automatically capture any series of keystrokes. When you choose this command, Sprint records all subsequent keystrokes (including typing, choosing menu commands, and cursoring) until you press *Esc*. (Pressing *Esc* when a menu is showing removes the menu but does not stop the recording.) Sprint then prompts for a name to give the recorded set of keystrokes. (You can press *Esc* if you don't want to save it at all.)

**Erase**
Lets you delete any glossary entry. Sprint displays a list of glossary abbreviations if you can't remember the name you want.

**Merge**
Combines two glossaries. If Sprint finds the same abbreviation in both glossaries, it prompts you for which entry to discard.

**List**
Creates a document with the name you assign and a .TXT extension, which lists the contents of the glossary in a table. The entry name is in the left column, the

description in the right column. If the entry is a macro or a recording, this is noted.

**Glossary**
Lets you choose what glossary file to use (default is STANDARD.SPG). If you enter a file name that does not exist, Sprint lets you create a new glossary file with that name.

There are two times Sprint saves a glossary file to disk. If you change glossaries, Sprint will notify you if the current glossary has not been saved and gives you an opportunity to save your changes. Similarly, when you quit Sprint, you are asked whether you want to save a modified or new glossary file before exiting.

Tips

If your glossary file begins with a percent sign (%), Sprint assumes the text following it is a list of *macro* commands to be executed and not text to be inserted.

If your glossary entry is a list of macros, you can include a special descriptor, which will be used if you choose **Macros/List**. Start the last line of the glossary entry with a semicolon (;) followed by the description you want to appear in the List file.

When you recall a glossary entry, Sprint automatically determines whether the entry is straight text, a macro file, or a keyboard recording, and carries it out appropriately.

If you want to have a set of ruler settings (and any other settings) come up as the default every time you open a new Sprint document, you can do this by following these steps:

1. Modify the ruler in any way (including any setting from the **Precise Settings** menu).
2. Enter any text that you want to be automatically present (for example, the To:, From:, and Re: of a memo).
3. Select all the text and the ruler (press *F3* and use the arrow keys).
4. Define a glossary entry by choosing **Define** from the **Glossary** menu. When prompted, call the entry AUTOEXEC. (Sprint will save the file as AUTOEXEC.SPG.)

Now, every time you create a new document, Sprint prompts you with:

```
Use your default settings?
```

If you type `Y` for yes, Sprint inserts your customized ruler and text (if any) at the top of the new document. If you type `N` for no, Sprint opens the new file with the regular ruler line at the top.

# Go (Merge)

Refer to the Merge entry for details.

# Go (Print)

Refer to the Print Menu entry for details.

# Go to (Mark)

Refer to the Place Mark entry for details.

# Go to Page

Refer to the Edit Menu entry for details.

# Graphics

Keystrokes

*Alt-S* (or *F10*, Style), Graphics

Function

Displays a menu of commands for drawing lines and printing graphics on PostScript printers.

When you choose this command, Sprint displays a menu with four choices:

**EPS Picture**
Inserts a graphic file that's been saved in encapsulated PostScript (EPS) format. Sprint prompts for the file name and the (optional) height and width. You should only provide the height and width if you want to *scale* the graphic to that size. When you choose this command, Sprint inserts the onscreen command EPS followed by the file name and the optional scaling factors.

**KeyCaps**
Simulates a computer "key cap" by boxing the selected text (or the word following the cursor) and printing it in boldface Helvetica. When you choose this command, Sprint inserts the onscreen command KEY.

**Bar**
Prints a horizontal line from margin to margin in a specified size. When you choose this command, Sprint inserts the onscreen command BAR.

**Draw Box**
Prints a box around selected paragraphs or whitespace. The box is drawn in 1-point lines and extends from the current left margin to the current right margin. When you choose this command, Sprint inserts the onscreen command BOX.

How To

Typically, you use the EPS Picture command in conjunction with the Figure command, like this:

```
BEGIN FIGURE
EPS "C:\QUATTRO\GRAPH.EPS"
CAPTION Projected Sales
END FIGURE
```

When Sprint prints this document, it automatically looks for the file called GRAPH.EPS in the directory called

QUATTRO and formats its PostScript code to print the graphic. It then prints the caption and ends the figure.

The **Bar** command prompts for a height. You can use any vertical dimension (like points, picas, inches, cm, lines, and so on).

Tips

If you plan to use the **KeyCaps** command frequently in your document, you should increase the line spacing to avoid lines with key caps having more space around them then lines without key caps.

You can insert formatted lists or any other format commands within the Begin and End Box commands. You can also nest one box within another. You cannot, however, place boxes side-by-side.

If you're using Borland's Quattro spreadsheet program, here are the steps you would take to save your graphs in EPS format and import them to Sprint:

- After you have graphed your data in Quattro, choose **Graph/Customize/Grids** and set **Frame Graph** to Yes to generate a graph with a box around it. (If you don't want a box around it, leave the command set to No.)
- Next, set the size of the graph by choosing **Graph/Print/Layout** and then setting dimensions for the **Left Edge**, **Top Edge**, **Height**, and **Width**. (The **Width** setting shouldn't be greater than the line length you're using in your Sprint document.)

  Note that the numbers you assign in Quattro affect the *size* of the graph, not its eventual placement on your Sprint page.
- While you are in the **Graph/Print/Layout** menu, choose **Orientation** and set it to **Portrait** (Vertical).
- You are now ready to save your Quattro graph in EPS format. Choose **Graph/Print/Write EPS-PIC** and then choose **EPS File**. Give your file a name (you don't need to add the .EPS extension; Quattro does that for you.)
- Leave Quattro and open the Sprint document you want to import the graph to.
- At the spot you want the graph inserted in your document, choose **Style/Graphics/EPS Picture** and enter

the file name (and path name, if necessary) of the EPS file you're importing.

- If you already set the size of the graph when you were in the Quattro Layout menu, you should not have scale the graph in Sprint. If you decide to do so, however, answer Yes to the Sprint prompt and enter the desired height and width.

  Sprint inserts the onscreen EPS command to signify that your Quattro graph will be inserted in your Sprint document at that location when you print the file to a PostScript printer.

Remember, the commands in this menu will *only* work if you are printing to a PostScript-compatible printer or typesetter.

If you don't have a PostScript printer, you can still use the Line Drawing command to draw simple lines and boxes.

See Also — Line Drawing, Chapter 2 (Escape)

# Group Together on Page

Keystrokes — *Alt-L* (or *F10*, Layout), Page Breaks, Group Together on Page

Function — Prevents a page or column break.

This command keeps marked text together on a page (or in a column if you're working with multiple columns). If there isn't enough room for the grouped text on the current page or column, Sprint moves all the text to the beginning of the next page or column. Some formats, such as Closing, Example, Figure, and Quotation, *automatically* group the text enclosed within the format.

How To — Select the text you want to keep together, press *Alt-L*, and choose Page Breaks/Group Together on Page. Sprint automatically places the Begin and End Group commands around your text. For example,

```
BEGIN GROUP
Here are a bunch of lines that we really want to keep
together.

Even blank lines are kept together on the same page.

Note that a long group may result in a large blank
space on the previous page.
END GROUP
```

These grouped lines of text, when printed, will always appear together on the same page.

If you try to group more text than can be printed on a single page, Sprint displays a warning message during formatting. Go back to your file and break up the group.

See Also Page Breaks

# Gutter Between

Refer to the Columns entry for details.

# Hard and Soft Returns

Keystrokes *Enter* (single hard return character; pressed twice, creates a blank line and begins a new paragraph)

Function Return characters can be either *hard* or *soft*. A hard return instructs the formatter to end the current line and begin printing text on the following line. Whenever you press *Enter*, you insert a hard return character at the current position. If the formatter sees two hard returns in a row, it inserts a blank line.

You can have Sprint display hard returns onscreen by setting Customize/Screen/Paragraph Marks to On.

**Note to two-floppy system users:** Sprint may have to prompt you to insert the correct Program Disk into Drive A in order to use this command.

The editor, on the other hand, automatically inserts a soft return when it wordwraps your document. For example, if you set the right indent at column 65, the

editor will place as many words as will fit on the line without going beyond column 65. After printing the last word that will fit on the line, the editor inserts a soft return, and begins printing on the next line. This means that while typing running text, you don't press *Enter* unless you want to end a paragraph.

How To

Press *Enter* twice to insert a blank line in your text. Press *Enter* once to start a new paragraph without a blank line in between.

Several formats (like Display and Example) do not recognize soft returns. Text within these formats prints until the formatter sees a hard return character, at which point the formatter begins a new line. If the formatter doesn't find a hard return within this type of format, it tries to print all lines ending with soft returns as one long line.

***Technical Notes***

Sprint recognizes four different characters as line endings (don't read this unless you're really curious):

^J A ^J (linefeed) character ends a paragraph, and we refer to this as a *hard return* character. When Sprint formats your file, the formatter converts the ^J to a line feed/carriage return combination.

^M A ^M carriage return also terminates a paragraph but causes the formatter to print the next line on top of the current line (unless, of course, the ^M is immediately followed by another ^M character, in which case both the editor and formatter just throw it away on input).

^_ A *SpaceNL* (stored as ^_, ASCII 31, one less than a space) replaces a space character to indicate where a line is wrapped. This is called a *soft return*. Although this should be logically equivalent to a space, storing it as a different code makes detection of where to wrap lines much easier. This is non-ASCII, but notice that you won't get any of these if you don't have any ruler lines. The formatter, when it reads this code, converts it immediately into a space but uses it to count line numbers for error messages.

^] A *HyphenNL* (stored as ^], code 29, one less than the discretionary hyphen, ^^) replaces a hyphen code to indicate a line wrap at a hyphen. It will also display on the screen as a dash if it is preceded by a letter or number. The formatter converts this code immediately on input into a discretionary hyphen.

Tips

Lines in pure ASCII files all end with hard returns. This means that when the formatter formats the file, it will produce the same line endings as those in the file, regardless of the right indent set on the ruler line. To override this function and force the formatter to wordwrap the lines in an ASCII file, choose **Print/Advanced Options/Wordwrap ASCII Files** and toggle the value of this command to Yes. If you do this before printing the file, the formatter replaces the hard return characters with soft returns and wordwraps lines between the left and right indent settings.

See Also

ASCII Files, Wordwrapping, and the *Advanced User's Guide*, Appendix C, "Style Sheet Commands"
(* [Asterisk])

# Header Menu

Keystrokes

*Alt-L* (or *F10,* **Layout**), **Header**

Function

Displays a menu of commands for page headers.

The **Header** menu options follow:

**All Pages**
Inserts into your document the command for a header that appears on all pages *except the first*. The onscreen command name for this type of header is HEADER.

**Title Page**
Inserts a command at the top of your document to print a header on the first page only. The onscreen command name for this type of header is HEADERT.

**Odd Pages**
Inserts a command at the current cursor position to create a header on all odd pages (except page 1, which is

handled with the **Title Page** command). The onscreen command name for this type of header is HEADERO.

**Even Pages**
Inserts a command at the current cursor position to create a header on all even pages. The onscreen command name for this type of header is HEADERE.

**Position**
Lets you determine the exact place the header will start printing.

Most of the **Header** commands create so-called *running headers*—headings that automatically appear at the top of some or every page of your formatted document. You can insert any **Header** command except **Title Page** anywhere in your document, and the text of the header will appear on some or all pages following the command (depending on the command you chose from the **Header** menu). A heading can contain one or more lines of text and can have alternating formats for left and right pages.

How To

To create a running page heading, press *Alt-L*, and choose **Header**. Then choose the type of header you want.

After you make your selection, Sprint displays the BEGIN and END commands and places your cursor between them. Type the heading text after the BEGIN command. Your heading can be as many lines as you wish.

For example,

```
BEGIN HEADERO
First Draft
END HEADERO
```

This header will print the text "First Draft" at the top of every odd-numbered page. The text will begin printing at the left margin.

If you want text to print at the *right* margin, choose **Insert/Wide Space** (Spring) before you type the text. For example, the following header has a wide space (^F) inserted before the date:

```
BEGIN HEADER
First Draft                                    11/16/87
END HEADER
```

As with footers, if you want to set-up alternating headings so that one header line appears on odd-numbered pages and another, different header line appears on even-numbered pages, you can choose two **H**eader commands. For example, choose **O**dd Pages from the **H**eader menu results in

```
BEGIN HEADERO
Confidential
END HEADERO
```

Then choosing **E**ven Pages and typing a different heading results in

```
BEGIN HEADERE
First Draft
END HEADERE
```

Sprint will print "Confidential" on every odd-numbered page and "First Draft" on every even-numbered page.

When you want to eliminate page headers, choose the **H**eader command but don't enter any text. This tells Sprint to leave your header lines blank.

If you want your header text to print in bold type, or any other typeface listed on the **T**ypestyle menu, choose the desired **T**ypestyle command before typing the text of your heading.

The **H**eader menu also lets you select a specific position for your header. Choose **P**osition, and Sprint prompts for the spacing above the top margin of the text, where the header should go.

If you want a header to print on the first page only, choose **T**itle Page from the **H**eader menu.

If you want a header that affects *all* pages (including the first, choose **T**itle Page and then choose **A**ll Pages and enter the same text in both.

Since headers are substantially the same as footers (except for their placement, of course), refer to the entry on Footers for more information on Headers.

See Also — Footer Menu

# HeadingA

Keystrokes — *Alt-S* (or *F10*, Style), **H**eadings, **H**eadingA

Function — Creates an unnumbered heading.

The **H**eadingA command prints a large, bold, unnumbered, centered heading preceded by six blank lines and followed by two blank lines. If placed at the top of a page, this command produces a heading identical to the one produced by the **C**hapter command, except that **C**hapter is numbered, and **H**eadingA is not.

**H**eadingA does not automatically create an entry in the table of contents. If you want to use this command to create unnumbered chapter or section titles, but want the titles to print in the table of contents, insert the MakeTOC command in your file. This command tells the formatter you want to create a table of contents, which will include the text and page number of your **H**eadingA commands.

You will find this command defined in the Sprint file STANDARD.FMT.

How To — Type the text of your heading on a line by itself. With you cursor on that line, choose **H**eadingA from the **H**eadings menu. For example,

```
HEADINGA Epistemology
Epistemology is the branch of philosophy concerned
with elucidating the nature of truth and knowledge
(as opposed to mere belief). It's a very difficult
subject to discuss, in part due to everyone already
believing they know everything.
```

Which prints as:

## Epistemology

Epistemology is the branch of philosophy concerned with elucidating the nature of truth and knowledge (as opposed to mere belief). It's a very difficult subject to discuss, in part due to everyone already believing he knows everything.

Tips

Unlike the **H**eadingB command, **H**eadingA spans the full page width, even if you are printing in two columns. If you use the **H**eadingA command with the **S**naking Columns command, be sure to enter **H**eadingA *before* the column command.

See Also

Chapter, HeadingB, Section, and Chapter 2

# HeadingB

Keystrokes

*Alt-S* (or *F10,* **S**tyle), **H**eadings, **H**eadingB

Function

Creates a heading.

The **H**eadingB command prints a large, bold, *un*numbered, centered head, preceded by four blank lines and followed by two blank lines. This command can be used to make unnumbered sections in your document or good-looking headings in reports, memos, and the like.

You will find this command defined in the Sprint file STANDARD.FMT.

How To

There are two ways to invoke the **H**eadingB command. Type the text of your heading on a line by itself. Put your cursor on that line, press *Alt-S,* and then choose **H**eadings/**H**eadingB. Sprint captures the whole line and formats it as a heading.

```
HEADINGB Speak Esperanto Like a Native
```

You can also choose the **H**eadingB command and, at Sprint's prompt, type in the title of your heading. Either way, Sprint highlights your heading to set it apart from the text that follows.

Tips

We modified the HeadingB definition for this manual so it produces a solid line below the heading. If you want to achieve the same result, add the following line to the end of the HeadingA macro in a renamed copy of STANDARD.FMT (search for `@macro(HeadingA)`:

```
BeforeExit @*@ux(@>)@*
```

If you are printing in multiple columns, the **H**eadingB command left-justifies your heading text over the current column. If you want to create a heading that's centered over all columns, use the **H**eadingA command. If you're printing in only one column, **H**eadingB and **H**eadingA have the same effect on your text, except for the number of blank lines preceding your column. (See the HeadingA entry for more information.)

Since this heading *isn't* numbered, you can't use the **D**efine a Tag or **R**eference a Tag commands to cross-reference it.

**H**eadingB does not automatically create an entry in the table of contents. If you want to use this command to create unnumbered chapter/section titles, but want the titles to print in the table of contents, insert the MakeTOC command near the top of your file. This command tells the formatter you want to create a table of contents, which will include the text of your *un*-numbered headings commands.

See Also

Chapter, HeadingA, Section, and Chapter 2 (MakeTOC)

# HeadingC

Keystrokes

*Alt-S* (or *F10*, Style), **H**eadings, **H**eadingC

Function

Creates an unnumbered subheading.

The **H**eadingC command prints a large, bold, unnumbered heading, flush with the left margin. **H**eadingC text is preceded and followed by a single blank line. If your printer does not have a large, bold typeface, the formatter will do whatever it can—wide spacing, and so on—to make the subhead stand out.

You will find this command defined in the Sprint file STANDARD.FMT.

How To

Use the **H**eadingC command when you want to create an emphasized section title, but don't want to number the heading text. Type the text of your subheading and select **H**eadings/**H**eadingC from the **S**tyle menu. For example,

```
HEADINGC Murphy's Law
Anything that can go wrong, will!
```

Prints as:

## Murphy's Law

Anything that can go wrong, wilt!

Tips

If you want your subheadings to be numbered, use the **S**ection or **S**ubsection commands. If you don't want subheadings numbered, but do want them to appear in the table of contents, be sure to include the MakeTOC command near the top of the file. See this entry in Chapter 2 for details.

See Also

Headings Menu, HeadingA, HeadingB, Section, Subsection, and Chapter 2 (MakeTOC)

# HeadingD

Keystrokes

*Alt-S* (or *F10*, **S**tyle), **H**eadings, **H**eadingD

Function

Prints an unnumbered paragraph heading.

The **H**eadingD command prints an unnumbered title in boldface type, flush with the left margin, but makes no entry in the table of contents. **H**eadingD sectioning is the lowest level of sectioning in the STANDARD.FMT file.

You will find this command defined in the Sprint file STANDARD.FMT. For information on modifying commands, refer to the Modifying Formats entry in this menu encyclopedia chapter.

How To

To create a new, unnumbered paragraph title, type the title of your paragraph on a separate line, press *Alt-S*, and

then choose **H**eadings/**H**eadingD. A paragraph heading command might look like this:

```
HEADINGD The Presocratics
The earliest Greek philosophers included such
notables as Heraclitus of Ephesus, who sagely said
you can't step into the same river twice. Not to be
outdone, Cratylus later declared you can't step into
the same river even once.
```

Results in:

### The Presocratics

The earliest Greek philosophers included such notables as Heraclitus of Ephesus, who sagely said you can't step into the same river twice. Not to be outdone, Cratylus later declared you can't step into the same river even once.

See Also — Chapter, HeadingB, HeadingC, Paragraph, Section, Subsection

## Headings Menu

Keystrokes — *Alt-S* (or *F10*, Style), **H**eadings

Function — Displays the **H**eadings menu.

The **H**eading menu lists several format commands to create section heads in your documents. The *Numbered* choices include

**C**hapter
**S**ection
**S**ubsection
**P**aragraph
**A**ppendix
**A**ppendixSection

The *Unnumbered* choices follow:

**H**eadingA
**H**eadingB
**H**eadingC
**H**eadingD

Refer to the individual entries for these commands for more information.

# Help

Keystrokes *F1*

*Ctrl-J*

Function Accesses Sprint's Help utility.

You can press *F1* or *Ctrl-J* to view a function key template or to get context-sensitive help on menu commands. You can also get help information on various topics drawn from a list.

**Note to two-floppy system users:** Sprint may have to prompt you to insert the correct disk into Drive A in order to access the Help menu.

# Hidden

Refer to the Comments entry for details.

# Hour (Variable)

Refer to the Variables entry for details.

# Hyphenation Menu

Keystrokes *Alt-U* (or *F10*, **Utilities**), **Hyphenation**

Function Displays the **Hyphenation** menu.

This command displays the **Hyphenation** menu, which lists the commands used to place discretionary hyphens in the text of a file.

**Note to two-floppy system users:** Sprint may have to prompt you to insert the correct disk into Drive A in order to use the **Utilities**/**Hyphenation** commands.

The formatter will only break a word where it sees a soft (discretionary) hyphen, and only if it needs to do so to fill out the line as much as possible. If it doesn't need to break a word, the formatter ignores the soft hyphen.

Hyphenation commands let you conditionally hyphenate a word, block, or an entire file. You can also specify the minimum number of characters required before a word may be hyphenated, and the widest space allowed on a line to justify that line.

The **H**yphenation menu list the following commands:

**Word**
Conditionally hyphenates the current word.

**Block**
Conditionally hyphenates the selected block of text.

**File**
Conditionally hyphenates the current document.

**Minimum Word Length**
Lets you change the length of the shortest word to be hyphenated (default is 8). For example, entering 12 means that only words containing 12 or more characters should be conditionally hyphenated.

**Space Allowable**
Lets you change the number of spaces that Sprint can add in order to justify a line of text before trying to justify the text by hyphenating words (default is 4). For example, if you set this to 3, any time Sprint needs to add 4 or more spaces to justify a line, it would attempt to hyphenate the first word on the following line to get it to take up some of the extra space on the preceding line.

**Note:** Sprint chooses only certain words to hyphenate when you choose these hyphenation commands. The words it chooses are those that Sprint considers need breaking up so that a syllable or two of them can go up to the previous line to help optimally fill out the line. For this reason, hyphenation is really only useful with fixed-width fonts, since Sprint cannot predict where a line will break with a proportional-width font. (Remember, it's the formatter that figures this out, not the editor.)

How To — To display the Hyphenation menu, press *Alt-U* and choose **Hyphenation**.

See Also — Special Hyphen

# Hyphens

Refer to the Bulleted Lists entry for details.

# Import

Refer to the Translate entry for details.

# Indenting

Keystrokes

*Alt-L* (or *F10*, **Layout**), **Ruler**, **Precise Settings**, **Initial (First Line) Indent** (indents the first line)

*Alt-L* (or *F10*, **Layout**), **Ruler**, **Edit on Screen** (or *Alt-A*), and type `I` (typed at the desired column on a ruler line, indents paragraphs governed by this ruler)

*Alt-L* (or *F10*, **Layout**), **Document-Wide**, **Offset** and type `n dimension` (offsets all text from the *inner* (that is, binding) margin by the specified dimension)

*Alt-S* (or *F10*, **Style**), **Modify**, and type `indent n dimension` (modifies the format so that it indents paragraphs *n* distance from the current left margin)

*Ctrl-OG* (indents region to next tab)

*Shift-Tab* (indents region to next tab)

*Alt-U* (or *F10*, **Utilities**), **Potpourrri**, **RegionIndent** (indents the current region to the nearest tab stop)

*Alt-U* (or *F10*, **Utilities**), **Potpourrri**, **RegionOutdent** (outdents the current region)

Function — Sprint provides several commands to indent or outdent:

- the first line of each paragraph
- from the current left margin, all text *within a format*

- from the left margin, all text *governed by a ruler*

Depending on what kinds of measurements you're dealing with, there are two ways to indent the first line of a paragraph. If you're using the onscreen ruler (which measures in characters and works well with fixed-width fonts), you'll want to choose Edit on Screen from the Layout/Ruler menu and then move the cursor to where you want the indent and type I. This kind of indent affects the text (onscreen, as well as at print time) governed by this ruler.

If, however, you're using the precise ruler measurements (which work better with proportionally spaced fonts), you'll want to set the first-line indent by choosing Layout/Ruler/Precise Settings/Initial (First Line) Indent. This kind of indent affects the text governed by this ruler only at print time.

The Initial (First Line) Indent command specifies how far the first line of each paragraph should be indented (or outdented) from the left margin. It is functionally the same as typing an *I* on the current ruler line. The first line of each paragraph following this ruler will be indented by the number of columns you select, until Sprint finds another ruler line with a different Initial Indent setting. If you do not have a ruler line in your file, selecting this command will create one at the top.

How To

When you want to indent (or outdent) a block of text that's affected by a format, you can modify the format by adding an *indent* parameter. Choose Style/Modify.

You can "outdent" text (cause text to appear to the left of the margin), by making the indent value *less than* the current left indent setting. Sprint accepts both positive and negative indent values.

If you want to indent just one paragraph, press *Shift-Tab* or *Ctrl-OG*. When you press *Shift-Tab* or *Ctrl-OG*, Sprint automatically pushes the paragraph to the next tab stop. A small right-pointing arrow appears if you have Customize/Screen/Indents set to On. Successive *Shift-Tab* or *Ctrl-OG*'s move to the next tab stop.

Another way to indent text governed by a ruler is to choose Layout/Ruler/Precise Settings/Left Indent or

Right Indent. When prompted, type in the amount of indent from the margin (in any valid dimension). When Sprint formats your document, the text governed by the ruler prints out suitably indented.

Tips

The Description format (on the Style/Lists menu) does outdenting automatically, so you don't need to reset the left margin or paragraph indent.

Note that there is sometimes a fine line between margins and indenting. (When you type [ on a ruler line, are you setting a new left margin or setting an indent from the margin? Technically, it's the latter, but common lingo prefers the former.) Be sure to refer to the Margins entry for related information.

Don't use a negative indent in a ruler or errors will result.

See Also

Margins, Precise Settings Menu, Ruler

# Indents

Refer to the Screen entry for details.

# Index

Keystrokes

*Alt-S* (or *F10*, Style), Index

Function

Allows you to index a word or words within text.

Using the commands in the Index menu, Sprint automatically creates and prints an index at the end of your document.

**Word**
Prints the current word or selected block in the index. The formatter will also print the word in its present location. If Screen/Codes is set to On, the selection or word in your text begins with ^D and ends with ^N.

If you're using a mouse, you can also use the Mouse menu as a shortcut to the Word index command.

**Reference Word**
Prints the current word in the index only; the formatter *will not* print the word in its present location. You can also select one or more words in your file and then choose this command. The formatter will insert all marked text in the index. Sprint inserts the onscreen command IXREF where you chose the command.

For example, this line in the first page of your text:

```
Nixon resigned in disgrace.IXREF presidents, Nixon,
resignation of
```

results in only these words appearing when page 1 prints:

```
Nixon resigned in disgrace.
```

But the index will show this entry:

presidents
  Nixon
    resignation of  1

**Master Keyword**
Marks the word and prints it in regular type in the index, but prints the page number on which it appears in bold type at the index entry. The formatter *will not* print the word in its current location. Sprint inserts the onscreen command IXMASTER where you chose the command.

**See**
Allows you to direct the reader to the section of the index where information on the subject can be found when the selected word (or subject) is not an index entry. The selected text will not print in its present location, and the index entry will not have page numbers. Sprint inserts the onscreen command IXSEE where you chose the command.

For example, this line in the second page of your text:

```
Nixon barely escaped impeachment.IXSEE impeachment,
Nixon resignation
```

results in page 2 including these words only:

```
Nixon barely escaped impeachment.
```

But the index includes this entry:

impeachment *See* Nixon resignation

Note that the text you enter must be in quote marks if a comma appears in the string. For example,

```
IXSEE impeachment, "Nixon, resignation of"
```

**Also See**
Allows you to direct the reader to another section of the index where more information on the subject can be found. The selected text will not print in its present location, and the index entry will not have page numbers. Sprint inserts the onscreen command IXSEEALSO where you chose the command. If the *See also* reference has a comma in it, you must enter the text in quote marks.

**Index Under**
Lets you index the selected text under a different heading (in a different alphabetical section). For example, you might want to index "1984" under the *N*s. Select 1984, choose **I**ndex Under, and type `Nineteen` when prompted. Sprint inserts the onscreen command IXREF UNDER where you chose the command. The indexed text does not print at its present location.

**Page Range**
Lets you print a range of pages in your index. You have to first define a tag where the range begins. Next, at the place in the text where the range *ends*, type the index entry (and select the text if the entry is more than one word) and choose **P**age Range. Sprint prompts you for the tag name you created earlier and then captures the selected text or the word the cursor is on as the index entry item.

Sprint prints the word your cursor is on (or the selected text) in the index with a range of pages from where the tag appears to where the index command appears. Sprint inserts the onscreen command IXRANGE where you chose the **P**age Range command.

To use a multileveled entry (one that has commas in it) with the **P**age Range command, the entry must first be typed in your text enclosed in quotation marks and then

selected. Then you choose the **P**age Range command. The onscreen results would be like this:

```
IXRANGE tag1, "computers, types of"
```

Sprint's index format is defined in the STANDARD.FMT file. As defined in this file, the index prints in two columns. The title *Index* is centered at the top of the page and prints in large, bold letters between the two columns. Nonalphabetic characters print first, followed by words beginning with letters. Before the formatter begins printing words that begin with a different character, it prints the new character on a separate line in bold type. For an example, see the index at the end of this manual.

How To

Select the word or words you want indexed and then choose **I**ndex.

To use the **P**age Range command, you should define a unique tag name at the spot you want the page range to begin. (Use *Alt-S*/**X**-Reference/**D**efine a Tag.) Then, at the spot you want the page range to end, enter the **P**age Range command. Sprint prompts for the tag name you previously assigned and then "groups" the selected text (or the word your cursor is at if no text is selected) as the index item.

For example, in an extended passage on how to make Mexican-style sushi that runs from pages 4 to 6, you could enter the tag *Mexsush* at the start of the passage (on page 4) and the following command at the end (on page 6):

```
IXRANGE Mexsush, sushi making
```

The result in the printed index is

S

sushi making 4-6

The **I**ndex Under command tells the formatter where to place text in the index. By default, the formatter lists text alphabetically and groups numbers and non-alpha ASCII characters together at the beginning of the index. The **I**ndex Under command lets you override this default placement of text in the index.

For example, let's say you want to index the number 22, but you don't want the number filed at the beginning of the index; you want the formatter to place the number where the word *twenty-two* would appear. Move the cursor to the text 22, choose **Index/Index Under**. At Sprint's prompt

```
Index under:
```

type `twenty-two` and press *Enter*. Sprint displays the command like this:

```
.IXREF UNDER twenty-two 22
```

This tells Sprint to index 22 under *twenty-two*. When the formatter prints the document, it will print the number 22 in its current location and in the index, as part of the *T* listings, where the word *twenty-two* would normally appear.

When using the **See** or **Also See** commands, you must enclose the terms in quote marks if they are more than one word. For example, to generate this *See* reference:

death and dying *See* thanatology

you first type `"death and dying"` (the quotes are mandatory), select the text, and then choose **Index/See**. At the prompt, enter `"thanatology"` (these quotes are optional).

Tips

You can create multilevel index entries by separating the main entry and its subentry with a comma. For example,

```
.IXREF Tires, rubber
```

and then on the next page,

```
.IXREF Tires, steel-belted
```

The index entries will look like this when printed:

Tires
- rubber 2
- steel-belted 3

Use the **Master Keyword** command to draw a reader's attention to the index entry. Such bold references could indicate an important or lengthy discussion.

**Note:** The **Word** command uses the typestyle of the word as the typestyle to use in the index. This means that, if the word is in italics in your document, it will appear in italics in the index, too. If you want to avoid this, use the **Reference Word** command *inside* the typestyle format. (You may have to show the control codes with *Alt-Z* to make sure you have it right.)

To ensure that the page numbers in your index will be as accurate as possible when using the **Reference Word** command, always try to place the indexed word *next to* the word it refers to.

If you want an index entry to contain a comma without having it force a new sublevel, you should enclose the entry with the Word command (choose **Style/Other Format**, and then type `Word` followed by the entry). This forces Sprint to format the selected text as if it were one word.

For example, if you wanted to index "Allen, Woody" as one entry, you would first define it as a "word" with the Word command, and then index it as a **Reference Word**. Refer to the index of this manual to see how it prints.

See Also Mouse Commands, Tags, and Chapter 2 (Word)

# Index Under

Refer to the Index entry for details.

# Initial (First Line) Indent

Refer to the Precise Settings Menu entry for details.

# Initial Record

Refer to the Merge entry for details.

# Insert (File)

Keystrokes

*Alt-F* (or *F10,* File), **Insert**

*Ctrl-KR*

Function

Inserts a file into the current file at the current cursor location.

The **File/Insert** command allows you to insert the contents of another file into your current file at the current cursor position. The text of the file you're inserting remains in its own file; Sprint *copies* the contents to your current file.

How To

When you choose this command, Sprint displays the following prompt in the status line:

```
Name of file to insert:
```

Once you enter the name of the file to be inserted, Sprint reads the file from disk and then inserts the entire contents of the file at the current cursor position. The file you inserted remains unchanged.

Tips

Sprint sees the file you just inserted as a *block* that is marked, copied into your current file, and then *unselected.* If you realize that you inserted the wrong file, put it in the wrong place, or want to set a typestyle for the inserted file, you can choose the **Reselect Block** command from the **Block Select** menu. You can then choose **Edit/Erase** if you inserted the wrong file, or **Edit/Move-Cut** if you inserted it in the wrong place, or choose the desired **Typestyle** command.

See Also

Block Select Menu, File Menu

# Insert Menu

Keystrokes

*Alt-I* (or *F10,* **Insert**)

*Ctrl-_* (inserts soft hyphen)

*Ctrl-Spacebar* (inserts a non-breaking space character)

Function

Displays the **Insert** menu.

The commands on the Insert menu let you insert variables and special characters.

**Variable**
Displays a list of variables and tells Sprint to insert the chosen variable at the current cursor position. You can choose one of the predefined variables, or you can choose Other.

**Define Text Variable**
Prompts you for the name of a text variable to create, and then for the text that the variable stands for.

**Merge Field**
Prompts you for a field variable to reference. You type in the field name, such as *FirstName*, and the SprintMerge program will substitute the value of *FirstName* record by record in the form letter. See the Merge entry for more information.

**Template for Data**
Inserts the commands @Template{ } and @Data. You type the fields within the braces. Your records must follow the format set in the Template and Data commands. See the Merge entry for more information.

**Special Hyphen** *Ctrl-_*
Inserts a special (soft) hyphen at possible breaking points in a word. The hyphen won't print unless the word needs to be broken at that point.

**Non-Breaking Space** *Ctrl-Spacebar*
Inserts a single blank space at the current cursor position that keeps the words on either side of the command together on a line.

**Wide Space (Spring)**
Inserts a special space character that enlarges as much as possible, pushing text following it to the right margin or next tab stop.

**Repeating Character**
Repeats a specified character to the right margin or next tab stop.

**Control Character**
Inserts a control character of your choosing at the current cursor position.

Refer to the individual entries for more information on these commands.

How To — To reach the Insert menu, press *F10* and choose Insert (or just type *Alt-I*).

See Also — Control Characters, Define Text Variable, Merge, Non-Breaking Space, Repeating Character, Special Hyphen, Wide Space, Variables, and the *User's Guide*, (Part 3, "SprintMerge")

# Insert Mode

Keystrokes — *Alt-C* (or *F10*, Customize), Options, Insert Mode

*Ctrl-V* (toggles Insert and Overwrite modes)

*Ins* (toggles Insert and Overwrite modes)

Function — Turns Insert mode from Insert to Overwrite.

The Insert Mode command tells Sprint what to do with text that you type. When Sprint is in Insert mode, it moves existing text to the right to make room for the new text. The status line displays "Ins." When Sprint is in Overwrite mode, Sprint does *not* move existing text; any text that you enter will write over (replace) existing text. The status line displays "Ovr" to indicate Overwrite mode.

**Note to two-floppy system users:** Sprint may have to prompt you to insert the correct Program Disk into Drive A in order to use this command.

**Note:** When in Overwrite mode, the *Backspace* key does not delete the character to the left of the cursor. This makes correcting typos easier. Also, pressing *Tab* in Overwrite mode moves the cursor to the next tab stop, but if the cursor is at the end of the line, pressing *Tab* inserts a tab character.

# Insert-Paste

Keystrokes — *Alt-E* (or *F10*, Edit), Insert-Paste

| | |
|---|---|
| | *F6* |
| | *Ctrl-U* |
| | *Ctrl-KV* |
| Function | Inserts text from the Clipboard. |
| | These commands take the contents of Sprint's Clipboard and *paste* it at the current cursor position. |
| Tips | Because Sprint leaves the text on the Clipboard, you can continue pasting this text elsewhere. The text remains on the Clipboard until you enter a command to move, delete, or copy another block of text. |
| See Also | Clipboard, Deleting |

# Insert (Ruler)

Refer to the Ruler entry for details.

# Insert (Unconditional) Page Break

| | |
|---|---|
| Keystrokes | *Alt-L* (or *F10,* Layout), Page Breaks, Insert (Unconditional) |
| | *Alt-N* |
| Function | Inserts an unconditional page break. |
| | This command specifies where you want the formatter to break (end) the current page and begin a new page. Normally when you print your file, the formatter creates its own page breaks, based on your page length and the top, bottom, header, and footer margins. If you want to override one of the formatter's page breaks (which you can see when you view or print the file), you can insert the Insert (Unconditional) Page Break command. |
| How To | To override the formatter's default pagination, move the cursor to the line or paragraph that you want to start on a new page. Press *Alt-N.* Sprint displays a solid underline showing you where it will break the page during printing. If you display Sprint's hidden control codes, you'll see the page break represented as ^L. |

Tips

The **Print/Paginate** command (or *Ctrl-F7*) shows you where the formatter will automatically break the pages of your file. Before inserting a page break, you might want to choose the **Paginate** command, so you can see where the formatter's default page breaks will occur. Then, if you don't like a particular page break, you can override it with this **Page Breaks** command. Choose **Paginate** again to see how your inserted page break affects the pages following the command. When you're satisfied with the page endings, choose **Remove Formatter Page Breaks**. Sprint removes the underlines it inserted when you chose **Paginate** but *does not* remove any of your inserted **Page Breaks** lines.

After you have paginated, the status line for that document shows the current page, the current line in that page, and the current column number.

If you want to remove a page break line created with the **Insert (Unconditional)** command, use one of Sprint's delete commands (for example, *Ctrl-Y* or *Del*).

If you choose **Insert (Unconditional)** when the cursor is in the middle of a line, Sprint inserts the command at the beginning of the line. If you choose **Insert (Unconditional)** in the middle of a paragraph, Sprint inserts the command at the beginning of the paragraph.

To prevent an *unwanted* page break, use the **Group Together on Page** command.

For a rundown of commands related to page breaks, refer to the Page Breaks entry.

See Also

Page Breaks

# Inter-Paragraph Spread

Refer to the Paragraphs entry for details.

# Italic

Refer to the Typestyle Menu entry for details.

# Jump to Line

Refer to the Edit Menu entry for details.

# Justification

Keystrokes

*Alt-L* (or *F10*, Layout), **Ruler**, **Justification**

*Alt-L* (or *F10*, Layout), **Ruler**, Edit on Screen

*Alt-R* (edit ruler onscreen)

*Alt-S* (or *F10*, **Style**, **Modify**, and type `Justify yes` (or `Justify no`)

*Ctrl-OC* (centers each line independently between the left and right margins)

*Ctrl-OJ* (inserts a ruler above the cursor line and toggles between justified and left-justified)

Function

Specifies text alignment.

The **Justification** command displays a menu that lists the various ways you can justify text affected by the current ruler:

**Left** Aligns text at the left margin only. The right margin will appear *ragged*, as if you typed your text on a typewriter. Choosing **Left** is the same as typing `L` on the ruler line.

**Right** Aligns text at the right margin only. The left margin will appear *ragged*. Choosing **Right** is the same as typing `R` on the ruler line.

**Both** Aligns text at the left *and* right margins. Choosing **Both** is the same as typing `J` on the ruler line.

**Center** Centers text between the left and right margins. Choosing **Center** is the same as typing `C` on the ruler line.

For the record, left justification looks like this:

You can insert an *L* anywhere on the ruler line, or change the Justification setting to Left, if you want text aligned at the left margin and "ragged" (not justified) at the right margin.

Right justification looks like this:

You can insert an *R* anywhere on the ruler line, or change the Justification setting to Right, if you want text aligned at the right margin and "ragged" (not justified) at the left margin.

Full (both) justification looks like this:

You can insert an *J* anywhere on the ruler line, or change the Justification setting to Both, if you want text aligned at the left and right margins.

Finally, centered justification results in this:

You can insert a *C* anywhere on the ruler line, or change the Justification setting to Center, if you want text "ragged" (not justified) at the left and right margins.

The *Justify yes* or *Justify no* parameters can be added to a format command to specify how text *within the format* should be aligned. Once you end the format, the formatter aligns text as specified *outside* the format.

How To

To define the alignment of your text, choose Layout/Ruler/Justification.

Once you select the Justification command you want, all text that you type (until you insert a new ruler line with a different justification setting) will be aligned as you specified.

Tips

You can also include a *Justify* parameter as part of a Style command at the top of your file or add a *Justify* parameter to any Format command. For example,

```
STYLE Justify no
```

tells the formatter you want your entire document left-justified.

If you modify a format and add *Justify yes*, the formatter will left- and right-justify the text in this format and then return to the justification set by the document's Style command or the justification setting on the current ruler line. If you set justification to Off with a parameter and later insert a ruler with *J* on the ruler line, the ruler setting takes precedence.

When you're editing a ruler onscreen, you can achieve two special text alignments by typing C and J or by typing R and J. The *CJ* combination fully justifies all text governed by that ruler except for the last lines of paragraphs, which are centered. Similarly, the *RJ* combination sets all text governed by the ruler to full-justification except for the last lines of paragraphs, which are right-justified.

See Also Modifying Formats, Ruler, and Chapter 2 (Style)

# Keep with Following Text

Keystrokes *Alt-L* (or *F10*, Layout), Page Breaks, Keep with Following Text

Function Prevents a page break where one might normally occur.

How To Choose Keep with Following Text wherever you want to prevent a page break that could ordinarily happen (for example, after a subhead). Sprint inserts the command name KEEPFOLLOWING at the end of the current paragraph. For example,

```
What's the current value of our stock?KEEPFOLLOWING

Our stock is moderately priced for the serious
investor.
```

The question and its answer will appear on the same page, no matter what.

You can keep a subhead with the text that follows by entering the command on the line following the subhead. For example,

```
Page BreaksKEEPFOLLOWING
Gives you control over how Sprint will break your
pages when you print.
```

Here, the subhead and following text will always appear on the same page.

See Also Page Breaks

# KeyCaps

Refer to the Graphics entry for details.

# Keyboard Record

Refer to the Glossary entry for details.

# Large

Refer to the Typestyle Menu entry for details.

# Last Bad Word

Refer to the Spelling Menu entry for details.

# Layout Menu

Keystrokes *Alt-L* (or *F10,* Layout)

Function Displays the **Layout** menu.

The **Layout** menu lets you set various global properties of your document (like margins and page size), as well as use several commands that affect text flow (like multiple columns and page breaks).

**Ruler**
Displays a list of commands to affect the ruler in your file: **Insert**, **Edit on Screen**, **Precise Settings**, **Justification**, and **Line Spacing**. These settings affect your current ruler, as well as all text following the ruler. You will not be able to see how unless you toggle **Codes** to **On**.

**Page Breaks**
Choosing **Page Breaks** displays a list of commands to affect page breaks: **Insert (Unconditional)**, **Conditional Page Break**, **Reserve Space**, **Blank Page(s)**, **Group**

Together on Page, Keep with Following Text, and Orphan-Widow control settings for your document.

**Columns**
Choosing Columns displays this list of commands: Snaking Columns, Column Break, and Gutter Between. These choices allow you to set the format for columns in your file.

**Document-Wide**
Displays a list of commands for you to choose: Right Margin, Left Margin, Top Margin, Bottom Margin, Offset, Paper Size, Word Spacing, Inter-Paragraph Spread, and Style Sheet. These choices make global changes to a document.

**Header**
Inserts a fill-in-the-blanks page header command that displays the following choices: All Pages, Even Pages, Odd Pages, Title Page, and Position.

**Footer**
Inserts a fill-in-the-blanks page footer command that displays the following choices: All Pages, Even Pages, Odd Pages, Title Page, and Position.

**Title Page**
Centers a new first page that has all the text vertically centered.

How To — To access the Layout menu, press *F10* and then choose Layout. You can then select one of its commands or submenus.

See Also — See the individual entries on these commands for further information.

# Left Indent

Refer to the Precise Settings Menu entry for details.

# Left Margin

Refer to the Margins entry for details.

# Length

Refer to the Tone entry for information.

# Letter File

Refer to the Merge entry for details.

# Line

Refer to the Block Select Menu entry for details.

# Line Drawing

| | |
|---|---|
| Keystrokes | *Alt-U* (or *F10,* **Utilities**), **Line Drawing** |
| Function | Lets you draw simple lines on the screen. |
| How To | When you choose this command, Sprint displays a menu of the types of characters you can use to draw lines. After you choose a character, your cursor becomes a drawing tool to create simple line drawing using the character you chose.<br><br>While you are in drawing mode, the status line reminds you of the following commands: |

| | |
|---|---|
| 0 | Pressing *0* lifts the "pen." That is, you can use the arrow keys to move the cursor without drawing. Pressing 0 again puts the pen back down to resume drawing. |
| 1-7 | Pressing a number from *1* to *7* changes the drawing character (see the next section). |
| 8 | Pressing *8* changes your cursor into an "eraser," so that any character it passes over turns into a space character. |
| arrows | Pressing any of the four arrow keys moves the cursor for drawing or erasing. |
| ESC | Pressing *Esc* takes you out of drawing mode. |

You can choose from these seven characters: **1.** Single (a single line), **2.** Double (a double line), **3.** Asterisk, **4.** Block, **5.** Degree, **6.** Period, or **7.** Shade. After that, you can use the cursor keys or a mouse to have the cursor draw these characters in its wake. Press *Esc* to end the drawing session.

Following is a table illustrating the various characters you can use to draw lines in your Sprint documents:

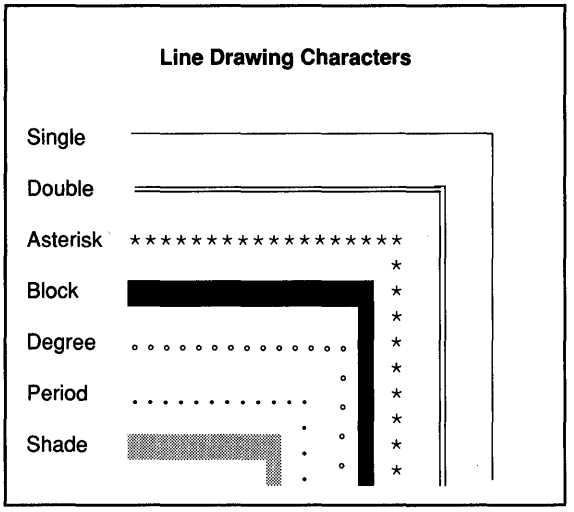



Figure 1.1: Sprint's Line-Drawing Characters

Tips

The results of this kind of graphics greatly depend on the printer you're using. Sprint uses the standard DOS graphic character set (high-ASCII characters) for most of the line characters, so you need a printer that supports these characters to get predictable results. (A PostScript printer, for example, will usually have different characters mapped to these spots.) Moreover, for best results you should use the Line Drawing command only if you're printing in a fixed-width font.

If you're using a two-button mouse with Sprint, you can press the left button to draw freeform (cursor can pass anywhere at any angle), or press the right button to draw straight lines only. With no button pressed, you can move the cursor without drawing.

If you have a printer that supports PostScript, you'll have better results if you use the commands in the **Style/Graphics** menu, not the **Line Drawing** command.

See Also — Graphics, Mouse Commands

# Line Spacing

Keystrokes — *Alt-L* (or *F10*, **Layout**), **Ruler**, **Line Spacing**

*Ctrl-OS* (sets ruler line spacing)

*Alt-L* (or *F10*, **Layout**), **Document-Wide**, **Inter-Paragraph Spread**

Function — Sets the spacing between lines.

The **Line Spacing** command lets you specify the space between lines in a paragraph. The **Inter-Paragraph Spread** command lets you specify the spacing between paragraphs.

How To — Choosing **Line Spacing** allows you to set how close together the lines of your printed document will be. You can select single (the default distance between lines, as determined by your printer), 1.5, double, or other. These dimensions are in terms of *lines*.

When you select **Other** from the **Line Spacing** menu, Sprint displays the following prompt:

```
Line spacing:
```

A shortcut for **Other** is *Ctrl-OS*.

When you choose a line spacing, Sprint inserts a special command (like *"spacing 12 picas"*) into the topmost ruler, which you can only see when you choose to show the hidden control codes in your document by pressing *Alt-Z*.

To set the spacing between paragraphs (if you want it to be different from the setting between lines), choose **Document-Wide/Inter-Paragraph Spread**. Sprint displays the following prompt:

```
Spacing to use between paragraphs:
```

When you enter an inter-paragraph spread, Sprint inserts a special command (like *"spread 14 picas"*) at the top of your file.

**Note:** You can use any valid vertical dimension for the commands, but if you omit a dimension, Sprint uses *lines*. Use *points* to select more precise spacing values. There are 72 points per vertical inch; so if you enter 72, your lines will appear 1 inch apart. Likewise, entering 9 yields 8 lines per inch (72/9 equals 8).

Tips

Note that the editor won't display the effects of your spacing commands, but when you print your file, Sprint will format the text properly. For example, choose the **R**uler/**L**ine Spacing/**D**ouble command and then type

```
This command lets you specify line spacing--how close
together the lines of your document will appear. From
the menu, you can select Single, 1.5, Double, or
Other.
```

Your printed text looks like this:

This command lets you specify line spacing—how close together the lines of your document will appear. From the menu, you can select Single, 1.5, Double, or Other.

When you want to end the effect of your **L**ine Spacing command, insert another ruler line in your file. This ruler will contain the same settings as those on the ruler line at the top of your file. If your want the spacing value to be different from that set on your original ruler, select the desired value on the **L**ine Spacing menu.

The formatter command Style lets you specify a spacing and inter-paragraph value for your entire document, regardless of ruler line settings. If you insert a Style command at the top of your file and include the parameter *Spacing 9 points*, the formatter will insert 9 points between each line of text, unless it finds a format that specifies a different *Spacing* parameter. The formatter will space lines within the format as specified in the format and then return to the spacing value set by Style.

For example,

```
STYLE LeftIndent 2 picas, Spread 0, Spacing 9 points
Pages of text...

More text...

BEGIN TEXT, SPACING 12 POINTS
This line of this example will be spaced further
apart than surrounding lines of text.
END TEXT

Here line spacing returns to 9 points between each
printed line.
```

This example sets overall line spacing at 9 points (approximately 8 lines per inch). When we modified the format, though, we changed the line spacing to 12 points (approximately 6 lines per inch). Once we end the modified format, line spacing returns to the value set with the Style command (9 points).

The Style command used to format the text of this manual includes the parameter *Spread .6*, which inserts 6/10 of a line between each paragraph.

See Also — Document-Wide Menu, Modifying Formats, Ruler, and Chapter 2 (Style)

# List (Glossary)

Refer to the Glossary entry for details.

# List Directory

Refer to the File Manager Menu entry for details.

# Lists

Keystrokes — *Alt-S,* (or *F10,* Style), Lists

Function — Displays a menu of commands for creating different types of lists.

**Outline**
Creates a outline-style list with successive levels of nested lists (uses I., A., 1., a.,...).

**Numbered**
Numbers the paragraphs of a list (uses 1, 2, 3,... for the first level; a, b, c for the second; i, ii, iii for the third).

**Multilevel**
Numbers the paragraphs of a list (uses 1, 2, 3 for the first level; 1.1, 1.2, 1.3 for the second; 1.1.1, 1.1.2, 1.1.3 for the third).

**Description**
Creates a two-column formatted list. A "subject" prints in the left column, and descriptive text appears next to the subject in the right column.

**Asterisks**
Prints an asterisk before each paragraph in an unnumbered list. A PostScript printer prints a diamond (♦) instead of an asterisk.

**Bullets**
Prints a bullet (either o or •) character before each paragraph in an unnumbered list.

**Hyphens**
Prints a hyphen before each paragraph in an unnumbered list.

Text can be affected by more than one Lists command at a time. Let's say you want paragraphs numbered, but within the numbered text, you want a bulleted list. For example,

```
BEGIN NUMBERED
The new car loan application must include the
following items:

BEGIN HYPHENS
Name

Rank

Serial number
END HYPHENS

Complete the forms in triplicate.

Sign all forms in magenta ink (or blood, if
preferred).
END NUMBERED
```

The printed text looks like this:

1. The new car loan application must include the following items:
   - Name
   - Rank
   - Serial number
2. Complete the forms in triplicate.
3. Sign all forms in magenta ink (or blood, if preferred).

As shown in the previous example, *you must end nested formats in reverse order of entry*!

How To — For examples of how to create these lists, refer to the Description, Bulleted Lists, Multilevel, and Numbered entries in this menu encyclopedia.

See Also — Bulleted Lists, Description, Multilevel, Numbered

# Load (Macro File)

Refer to the Macros entry for details.

# Load (User Interface)

Refer to the User Interface Menu entry for details.

# Log Errors to File

Refer to the Advanced Options entry for details.

# Macros Menu

Keystrokes *Alt-U* (or *F10,* Utilities), Macros

Function Displays the Macros menu.

When you choose the Macros menu, you have a choice of three commands:

**Load**
This command loads and compiles a macro file, adding it to the current overlay (.OVL) file that Sprint is using for its editor instructions.

**Enter**
This command allows you to enter the name of a macro, which is then executed immediately or assigned to a key. If you choose to assign the macro to a key, you can press the specified key at any time and have the macro run immediately.

**Run**
This command automatically saves the currrent file, loads it, compiles it, and executes it. If the current file is not a definition of a macro, has a ruler in it, or contains errors, Sprint gives you an error message and informs you of the offending line number.

Every action that takes place in the editor, every menu you see, and every command you issue is really a Sprint editor macro in disguise. This underlying power is what makes possible the alternative user interfaces, which can totally transform the keystrokes you use to run Sprint.

If you are experienced in programming or are curious about how you can program Sprint, you should read Part 2 in the *Advanced User's Guide*, called "Programming Editor Macros." That part has a tutorial and a complete reference section to all the built-in macro language components.

Tips

Don't confuse these macro commands with the formatter command called Macro. That command is for creating macro commands that the *formatter* uses, not the editor. The Macro command is discussed in the *Advanced User's Guide.*

# Main Dictionary

Refer to the Spelling Menu entry for details.

# Manuscript (Variable)

Refer to the Variables entry for details.

# Margins (General Information)

Keystrokes

*Alt-L* (or *F10,* Layout), **Ruler,** Precise Settings, **Left Indent** (or **Right Indent**)

*Alt-L* (or *F10,* Layout), **Ruler,** Edit on Screen, or **Insert** and type [ or ] anywhere on the ruler line.

*Alt-L* (or *F10,* Layout), Document-Wide

*Alt-S* (or *F10,* Style), **Modify,** and type `LeftIndent` *dimension,* `RightIndent` *dimension,* `LineLength` *dimension,* `LeftMargin` *dimension,* or `RightMargin` *dimension*

*Alt-A* (edits onscreen ruler)

*Alt-R* (inserts ruler)

*Ctrl-OL* (sets left margin)

*Ctrl-OR* (sets right margin)

*Ctrl-OF* (sets margin and tabs from text line)

Function

Sets up margins for your Sprint document.

Choosing the **D**ocument-Wide menu lets you choose from its five margin commands to change the margins for your entire document. The new margins will not be obvious until you print. The choices are

| | |
|---|---|
| **L**eft Margin | Sets a new left margin. |
| **R**ight Margin | Sets a new right margin. |
| **T**op Margin | Sets a new top of page margin. |
| **B**ottom Margin | Sets a new bottom of page margin. |
| **O**ffset | Sets a binding margin (an extra indent that alternates from right margin to left margin). |

If you don't make any settings in the **D**ocument-Wide menu, Sprint uses its default settings of 1 inch for left and right and 1 inch for top and bottom. When you make settings from this menu, Sprint inserts a Style command at the top of your document (appropriately modified).

If such a Style command exists at the top of your file, the first ruler in your document (which is below the Style command) automatically takes on the settings of the Style command. For example, if you set your left margin to 3 inches, the first ruler's setting of, say, column 10 *automatically stands for that setting* (3 inches), even though nothing has visibly changed on the screen. If there is no Style command above the first ruler, the ruler's settings are just like all your other rulers—have an offset effect against the default page margins.

Note that, strictly speaking, the only way to change *margins* in Sprint is through the Document-Wide menu; all other changes are relative to these margin settings and are properly referred to as *indents*, not margins. Nonetheless, common parlance prefers to call even temporary indentations margins. For this reason, we include some indenting commands here as well. Be sure to also refer to the Indenting entry in this chapter for more information.

The **Layout/Ruler/Edit on Screen** command lets you change a ruler in your file. You can set the left (type [ on ruler line) or right (type ] on ruler line) margins. You can also set **Left Indent** and **Right Indent** from the **Precise Settings** menu. Choosing **Left Indent** or **Right Indent** from the **Ruler/Precise Settings** menu lets you specify a margin for the text governed by the current ruler. You can use any horizontal dimension (inches, points, characters, and the like) when specifying margins in this way.

You can also set a temporary margin (that is, an indent) for a block of text governed by a particular format command. You enter such margins by adding a parameter to a format after choosing **Style/Modify** or to the **Style** command after choosing **Style/Other Format**. These parameters follow:

- ***LeftMargin*** *dimension* sets the left margin to the specified dimension for the rest of the format.

  If the dimension includes a + or – symbol (for example *LeftMargin+5 picas*), the left margin will be moved inward (toward the right margin) or outward from the left margin set outside the environment. For example, if you have a left margin of 6 picas and want the format to print at 9 picas, select **Style/Modify** and then modify the format to include the parameter *LeftMargin+3 picas*. Text following the command resumes printing 6 picas from the left edge.

- ***RightMargin*** *dimension* sets the line length to the specified dimension for the rest of the current format. For example, *RightMargin 32 picas* sets the right margin 32 picas to the right of the current left margin.

  The dimension can also include a + or – symbol. If a + dimension is entered, it moves the right margin in (leftward, from the current right margin setting) by the specified dimension, and if the dimension is negative, it moves the right margin to the right of the current right margin setting by the specified dimension. For example, *RightMargin –6 picas* extends the line length by 6 picas for the rest of the current format. Once you end the command, the right margin returns to the setting that existed outside the format.

- ***LineLength** dimension* sets the line length to the specified dimension, calculated from the *LeftIndent* parameter specified in the current command. For example,

  ```
  LeftIndent 2 picas, LineLength 39 picas
  ```

  sets a left margin 2 picas from the printer's default left margin setting, and the printed line will extend 39 picas to the right of this margin.

  *LineLength 0* eliminates the right margin; the text will go off the right edge of the paper.
- ***LeftIndent** dimension* sets the indentation to the right of the left margin by the specified positive *dimension.*
- ***RightIndent** dimension* sets an indentation to the right or left of the right margin by the specified positive or negative *dimension.*

The *Ctrl-O* commands also affect only the onscreen ruler settings. When you choose these commands, Sprint prompts for the column number for left (*Ctrl-OL*) or right (*Ctrl-OR*) margin, and then inserts a ruler into your file with the dimension you specified at the prompt.

Sprint provides a variety of commands and parameters to vary the left and right margin settings in your document. Use the following guidelines to determine which of these commands to use for a particular formatting situation:

- The first ruler's margin settings—regardless of what they are—always match the **Document-Wide/Margins** settings (which insert Style commands above the first ruler).
- Subsequent rulers and margin modifications to formats are relative to the surrounding format (the format set *outside* the format you're modifying).

Also, be sure to refer to the Indenting entry for related information.

If you want to leave some extra space on the *inside* margin to make binding easier, use the **Layout/Document-Wide/Offset** command and specify the desired offset. The specified amount will be added to the

left margin on odd pages and subtracted from the left margin on even ones.

See Also — Document-Wide Menu, Indenting, Justification, and Chapter 2 (Style)

## Master Keyword

Refer to the Index entry for details.

## Match Words Only

Refer to the Searching entry for details.

## Menu Display Delay

Refer to the Options Menu entry for details.

## Menu Shortcuts

Keystrokes — *Alt-C* (or *F10,* Customize), Menu Shortcuts

Function — Displays onscreen abbreviations for Sprint's shortcuts.

If Yes, Sprint displays in abbreviated form the shortcut equivalents for the menu commands.

How To — Choosing **O**ptions/**M**enu Shortcuts toggles from Yes to No as you press *Enter.* Press *Esc* to return to the **O**ptions menu or *Shift-Esc* to continue editing your file.

**Note to two-floppy system users:** Sprint may have to prompt you to insert the correct Program Disk into Drive A in order to use this command.

Tips — When a menu is displayed, you can toggle the menu shortcuts for that menu only by pressing the plus key (+) on the numeric keypad (to show the shortcuts) and the minus key (–) on the keypad (to hide them).

**Note:** Don't confuse this command with **Reset Shortcuts**, which returns your shortcut assignments to their "factory settings."

For a complete overview of the default menu setup in Sprint, refer to the fold-out menu-tree chart at the front of this manual.

See Also — QuickCard, Function Keys

# Merge

Keystrokes — *Alt-P* (or *F10*, **Print**), **Merge**

Function — Displays the SprintMerge menu, whose commands let you create form letters by merging a letter file with a record file.

This command instructs the formatter to merge a letter file with a record (or template) file. It's typically used to merge a form letter with a list of names and addresses. The record file contains a template that defines how your records should be formatted, as well as the records themselves. You can append additional records with the @Include command.

**Note to two-floppy system users:** Sprint may have to prompt you to insert the correct Program Disk into Drive A in order to use SprintMerge.

How To — To run SprintMerge, you must have a *record file* with a template and records, and a *letter file* with variables to be filled in.

First, create a form letter using the Sprint editor. Press *F10* and choose **Insert/Merge Field** when you come to a variable that the SprintMerge program will fill in with the actual value. Save your letter file; if you don't specify a name, SprintMerge will look for the default file DEFAULT.SPR.

Next, open a record file: either open a new, empty file or open a file that contains records. At the beginning of the file, set up a template by pressing *F10* and choosing **Insert/Template for Data**. The commands

```
@Template{ }
@Data
```

show up on the screen. Begin typing your template within the braces.

The template should contain all the fields you will need for your letter files. Here's a typical one:

```
@Template{<honorific>/<firstname>/<lastname>/
<address -multiline>/
<city>/<state>/<zip -num>}
```

Each item within angled brackets is a *field*. A complete set of fields makes up one *record*. The angle brackets are *field delimiters*. The slashes are *field separators*. Visible field separators are handy when you have empty fields: You can see at a glance what's missing.

Now you can enter your actual records, set up exactly like the template. Enter your records after the @Data command that Template for Data inserted. Here's a typical record:

```
Ms./Carlita/Montilla/
123 Main Street,
Apartment F/
Los Gatos/CA/95030
```

You type in the field separators, but not the delimiters.

You're almost ready to merge. Press *Alt-P* and choose Merge to view the SprintMerge menu, which has these options:

| | |
|---|---|
| Go | Starts SprintMerge and prints. |
| Screen Preview | Sends the merged letters to the screen. |
| Letter File | Selects the letter file you want merged. |
| Record File | Selects the record file you want merged. |
| Data Format | Lets you choose the format for setting up your files: Standard or Basic (uses quotes as delimiters). |

| | |
|---|---|
| Initial Record | Prompts for the the first record you want merged. |
| Ending Record | Prompts for the last record you want merged. |
| Order | Lets you specify sort order (ascending or descending in alphabetical, numerical, ASCII, or default format). |
| Criteria | Selection specifications, using logical or relational operators. |

To run SprintMerge, type in the appropriate letter file and record file entries, then choose **Go**.

You can use the following options with the SprintMerge program:

Table 1.4: The SprintMerge Program Commands

| @ Command | Menu Equivalent | Command Line Equivalent |
|---|---|---|
| @Template(*spec*) | *F10*/Insert/Template for Data | None |
| @Value(*spec*) | *F10*/Insert/Merge Field | None |
| @Include(*spec*) | None | None |
| @Sort(*spec*) | *F10*/Print/Merge/Order* | –Sort(*specifications*) |
| @Select(*spec*) | *F10*/Print/Merge/Criteria | –Select(*specifications*) |
| @Startrec(*number*) | *F10*/Print/Merge/Initial Record | –Startrec *n* |
| @Endrec(*number*) | *F10*/Print/Merge/Ending Record | –Endrec *n* |
| @Sortfile(*filename*) | None | –Sortfile(*filename*) |
| @Outfile(*filename*) | None | –Outfile(*filename*) |
| @Print(*options*) | *F10*/Print/Options | –Print or –p |
| @Comment(*notes*) | None | None |
| @Remditto() | *F10*/Print/Merge/Data Format | –Remditto |

*If the command string is too long to enter into the status line, use the @-form of the command in the file.

See Also Insert Menu, and the *User's Guide* (Part 3, "SprintMerge")

# Merge Field

Refer to the Insert Menu and Merge entries for details.

# Merge (Glossary)

Refer to the Glossary entry for details.

# Microsoft Word (Translate)

Refer to the Translate entry for details.

# Minimum Word Length

Refer to the Hyphenation Menu entry for details.

# Minute (Variable)

Refer to the Variables entry for details.

# Modifying Formats

Keystrokes — *Alt-S* (or *F10*, Style), Modify, and type *new parameters*

Function — Changes the effect of a predefined format command.

There are a variety of ways to modify a format. If you want to change a format's effect in a given instance, you should use the Style/Modify command.

If you want to modify the format's effect throughout a given file, the formatter command Modify, typed at the top of the file, saves you some time. You only need to type the modifications once, rather than for each occurrence of the command.

You must insert a Modify command before the first time you use the altered format. Also, you can't use the Description format, for example, a couple of times and then modify it. If you want to affect only certain occurrences of your Description format, use the **S**tyle/**M**odify command, or define a new format that creates a modified Description format.

When you want to change a format's effect on *all of your files*, you have several additional options:

1. You can add a Modify command to the default STANDARD.FMT file and place this command on a line *after* the format definition. This Modify command tells the formatter how you want the format's effect to differ from the format's original definition.
2. You can edit the @Define command that defines the format in STANDARD.FMT.
3. You can define a new command based on a current one using @Define in STANDARD.FMT.
4. You can define a new format in STANDARD.FMT. Instead of modifying an existing definition, you can add a format definition that is similar but not identical to the format you want to modify. That way, you can take advantage of both formats.

Refer to the Modify entry in Chapter 2 for more information on using Modify. Also see the *Advanced User's Guide* for detailed information on modifying formats in this manner.

There are actually several ways to modify menu formats. For example, if you want to print text in the Hyphens format as double-spaced, you could use three different ways to accomplish this:

1. Enter Hyphens from the **O**ther Format menu. Type `R` for Region, then `B`. Backspace into the highlighted text and add the parameter *spacing* 2 after typing a comma (the comma is optional if you're adding only one parameter). Type the text as usual, then select **O**ther Format to end the format (type `E`).
2. Choose **O**ther Format, type `Hyphens, spacing 2`, type `R` for Region, then `B` for Begin Command. Sprint enters

and highlights your command exactly as entered. Type the text as usual, then select Other Format to end the format (type E).

3. Choose Modify from the Style menu and choose This Format or Previous Format.

How To

*One-Time Change*

You can change the effect of a particular format by choosing Style/Modify. When you choose this command, Sprint prompts with a new menu:

| | |
|---|---|
| This Format | The format command at which the cursor is presently located. |
| Previous Format | The format command that appears before the current format command. |

Once you select a format to modify, Sprint displays the following prompt:

```
Modify by adding:
```

If you enter more than one parameter, type a comma to separate the parameters. (The comma is optional when you enter only one parameter.) Once you've entered all parameters, press *Enter* to end the command. Sprint automatically inserts the parameter(s) at the end of the Begin format command and returns your cursor to the original location.

**The List of Parameters**

Sprint has over 60 different parameters available, all of which are listed in Table 1.8.

Note that the Style command is unique in that it sets global settings in a document. Because of this, there are certain parameters that can *only* be used with the Style command. These parameters are listed in Table 1.6.

There are also some parameters that can *only* be used with formats that affect regions of text. These commands include any format that starts with the keyword Begin, any that is editable with the Style/Modify command, or any command you create yourself using the Define or Modify commands. (You enter the Define and Modify commands using the Style/Other Format menu

command.) Parametes valid only for these formats are in Table 1.5.

**Note:** The Style/Modify command recognizes only commands entered through the menus, not commands you entered using the @-sign method.

A third group of parameters are those that are valid anywhere: both in Style commands and in all other commands. This group of parameters appears in Table 1.7.

After the three short tables, comes a complete alphabetical list (Table 1.8) of all the parameters with short descriptions about each.

Table 1.5: Parameters Used Only with Formats Affecting Regions

| | | |
|---|---|---|
| Above | Centered | Inline |
| AbovePage | Column | Invisible |
| After | Columns | LeadingSpaces |
| AfterEntry | Divider | Margins |
| AfterExit | FlushLeft | Numbered |
| Before | FlushRight | Overstruck |
| BeforeEach | Free | Script |
| BeforeExit | Group | Strikeout |
| Below | IfNotFound | Underline |
| BelowPage | Index | WithEach |
| BlankLines | Initialize | Within |

Table 1.6: Parameters Used Only with Style Commands

| | | |
|---|---|---|
| BindingOffset | Increment | RightMargin |
| BottomMargin | LeftMargin | TabSize |
| Comments | Offset | TopMargin |
| Counter | Paper | WidowPrevent |
| FormFeed | PaperWidth | WordSpacing |

Table 1.7: Parameters Used Anywhere

| | | |
|---|---|---|
| Fill | Justify | Size |
| Font | LeftIndent | Spacing |
| Gutter | LineLength | Spread |
| Indent | NoTCT | TCT |
| Justification | RightIndent | |

Table 1.8: Format Parameters (Complete List)

| Field | Description |
|---|---|
| ***Typeface Parameters*** | |
| Font <*names*> | Uses the <*name*> font. The *Font* parameter can be one font name or a list of names. For example, *Font courier pica elite* allows any of these fonts to be used. The first match is the one used. |
| IfNotFound | Ignores any script, size, overstruck, underline, strikeout, or invisible fields if the most recent font was matched by something from the printer. For example, *Font bold, IfNotFound, Overstruck* will overstrike only if the printer does not have a *bold* font. This command, if at the end of a definition, also prevents the error message that is normally printed if a specified font is not supported by the printer; for example, *@Define(Typewriter, Font courier, IfNotFound).* |
| Invisible | Does not print the specified text. However, the text still takes up space and gets underscored and struck out if appropriate. |
| Overstruck | Prints the text once, offsets slightly, and prints again. It is similar to the bold format. |
| Script +/– <*dimension*> | Moves up or down by the given dimension. (The dimension must be in lines.) |
| Size <*dimension*> | Specifies the point size. Size can be given in absolute units such as inches or points, or it can be given in lines (the width of the line depends on the current font's point size). If the dimension is in lines, nesting such formats will cause the point size to grow or shrink geometrically. |
| Strikeout <*type*> | The specified text will be struck out. For example, ~~STRIKE SOMETHING OUT...~~ For a list of acceptable types, refer to *Underline.* |
| Underline <*type*> | Underlines the specified text. There are four different types of underline formats:<br>*all* — Everything will be underlined.<br>*alphanumeric* — All letters and numbers will be underlined.<br>*nonblank* — Everything except blanks will be underlined.<br>*off* — No underlining will occur.<br>If no *type* is specified, then *all* will be used. |
| ***Formatting Parameters*** | |
| Above <*dimension*> | At least this much blank space will be put above the format. |
| AbovePage | "Floats" this format to the top of the page. |

Table 1.8: Format Parameters (Complete List), continued

| | |
|---|---|
| Below <*dimension*> | At least this much blank space will be put below the format. |
| BelowPage | "Sinks" this format to the bottom of the page. |
| BlankLines <*type*> | Defines how the formatter will view blank lines entered in the text. Valid types are<br>*break* Multiple blank lines are ignored. Together they will result in a single *spread* line.<br>*kept* Formats each blank line (opposite of *break*). *Kept* is the default.<br>*hinge* Similar to *break*, but an automatic Hinge command is inserted on each blank line. |
| BottomMargin <*dimension*> | This is the space between the end of the text area (which includes the footer if there is one) and the bottom edge of the paper. Default value is 1 inch. Set this value only once, at the beginning of the document, and only as a modifier to the Style command. |
| Centered | Centers the text within the defined margins (same as *Justify center*. |
| Column | Sets up a format to print parallel (not snaking) columns. The lines that come after this format start printing at exactly the same spot as the lines in it. (For this reason, you need to set a new left indent after the command that has the *Column* parameter.) |
| Columns *n* | Divides the page up into this many columns (maximum is 6). If you set *n* equal to 0 (the default value), the text will be formatted into one column. |
| Comments {*yes/no*} | Tells the formatter whether to hide comments (that is, any line starting with a semicolon) entered in your file. Yes means to omit the comments from the printed version of the file; no means to print the comments. See the Comments entry for instructions on entering comments. The default value of this parameter is *Comments no* and is set on the last line of STANDARD.FMT. Used only as a modifier to the Style command. |
| Fill {*on/off*} {*yes/no*} | Turns *Fill* mode on or off for this format. Turning *Fill* on causes the formatter to wordwrap a paragraph, ignoring single hard returns if necessary. Setting *Fill* to off means Sprint always starts a new line if it encounters a single hard return character. (If you do not wish to left justify, you can override this by inserting another justify command after the fill command.) |
| FlushLeft | Forces all lines to begin at the left margin (same as *Justify left*). |
| FlushRight | Forces the end of all lines to the right margin (same as *Justify right*). |
| FooterSpacing <*dimension*> | Defines the distance between the bottom of the page and the place where the footer begins printing. |

| | |
|---|---|
| FormFeed {*on/off*} | Defines whether the formatter will send form feed characters to the printer to advance the printer to the top of the next page. If you specify *off*, the formatter sends line feed characters to the printer instead of form feeds. You may need to used this parameter if your paper length is not 11 inches (the usual default paper length). Default is *FormFeed on*. Used only as a modifier to the Style command. |
| Group {*yes/no*} | Groups this format. You can also disable grouping with *Group no*. |
| Gutter <*dimension*> | Defines the distance between columns in a multi-column format. Default value is .5 inch. |
| HeaderSpacing <*dimension*> | Defines the distance between the top of the page and the place where the header begins printing. |
| Indent {+/–}<*dimension*> | Defines the amount of space the first line of a paragraph will be indented (or outdented) relative to the left margin. Default value is 0 (no indent). If this parameter is a positive number (for example, 3 picas), the formatter indents the first line of *every* paragraph by this amount. If you specify a negative number, the first line of every paragraph will be outdented by the specified amount (printed to the left of the remaining text in the paragraph). The *Indent* parameter has the same effect as the Layout/Ruler/Precise Settings/Initial (First Line) Indent command.<br><br>If an area of text is affected by a command that indents by default, or if you specify an indent value in a Style command and don't want an area of your file indented, modify the command affecting the text that shouldn't be indented. |
| Justification <*type*> | Same as *Justify*. |
| Justify <*type*> | Defines the type of justification. Valid types are *left, right, no, yes, off, on, both,* and *center*. *Right* and *center* types also set the fill command to off. The default value is *Yes*; all paragraphs justified to the left and right margins. If you set this parameter to *No*, Sprint prints with ragged-right margins. |
| LeadingSpaces <*type*> | Defines how the formatter will treat indentation by tabs or spaces at the start of a paragraph. Valid types are<br><br>*kept* — Formats each blank space.<br><br>*ignored* — Formats two or more blank spaces as a single blank space. Manual indentation by tabs or spaces at the start of a paragraph is ignored. |
| LeftIndent {+} <*dimension*> | Defines the new left margin relative to the current format's left margin. For example, *LeftIndent 1 inch* starts the left margin 1 inch from the previous format's left margin. You cannot use negative numbers with *LeftIndent*. |
| LeftMargin | Defines the left margin. You can use this parameter only once in a document and only at the top of the file modifying a Style command. The default value is 1 inch. |

Table 1.8: Format Parameters (Complete List), continued

| | |
|---|---|
| LineLength {+/–} <*dimension*> | Defines the length of a line of text (that is, the placement of the right margin relative to the left margin). When the dimension is set to 0, paragraph wordwrap ("fill") does not occur (so long lines can go off the right side of the page). |
| Margins {+/–} <*dimension*> | Simultaneously defines both the left and the right margin. For example, *margin .5 inch* creates a left and right margin of 1/2 inch. |
| NoTCT | Same as *TCT no.* |
| Offset <*dimension*> | Adds this much space to the *inner* margin (that is, alternating between the left and right margins) to facilitate binding. Use this parameter only once in a document and only as the first parameter modifying a Style command. |
| Paper <*dimension*> | Changes the paper length as defined in the selected printer'sprinter definition (typically 11 inches). Be sure this parameter matches the form length of the paper in the printer, or the text will "drift" over the pages.<br><br>**Warning:** Many printers take a *form feed command* and will not adjust to the longer or shorter paper unless the form length switch on the printer is changed to the correct position. If this is necessary with your printer, you can tell Sprint not to use the form feed command by running SP-SETUP and answering *N* to the question *Use Form Feed (Control-L)?* You can also insert a Style command with the *Formfeed No* parameter. Default value is read from the printer definition, usually set at 11 inches.<br><br>**Note:** Set this value only once, at the beginning of your document, as a modifier to the Style command. |
| PaperWidth <*dimension*> | Changes the paper width, as specified in the selected printer's printer definition (default is typically 8-1/2 inches), thus moving the right edge of the text further right or left. The length of the lines increases or decreases without changing the margins. Default value is read from the printer definition, usually set at 8-1/2 inches. Used only as a modifier to the Style command. |
| RightIndent {+/–} <*dimension*> | Defines the new right margin relative to the current format's right margin. A positive number moves the indent to the left of the right margin; a negative margin moves it to the right. |
| RightMargin | Defines the right margin. You can use this parameter only once in a document and only at the top of a file modifying a Style command. |
| Spacing <*dimension*> | Changes the distance between each line of the text. Spacing can be given in absolute units such as inches or points, or it can be specified in lines (the depth of the line depends on the current font's point size). Default is 1 line. |
| Spread <*dimension*> | Defines the depth of a single blank line. If this value is equal to the *spacing* value, then blank lines in the input look just like blank lines in the output. The default is |

| | |
|---|---|
| | 1 line. Often the printout looks better if this value is set to less than one line. |
| TabSize <*dimension*> | Determines the distance between ASCII tabs. You should only use this value if you're creating an ASCII file and don't have any ruler lines in your file. Default value is 8 characters. Used only as a modifier to the Style command. |
| TCT {*yes/no*} | Disables/enables TCT translations inside this format. |
| TopMargin | Defines the top margin. You can use this parameter only once in a document and only at the top of a file modifying a Style command. The default is 1 inch. |
| WordSpacing <*dimension*> | Determines the maximum extra space the formatter can insert between words during justification. When justifying text, Sprint stretches the spaces between words first; if this stretching has reached a maximum, and the line is still not justified, Sprint spreads out the letters of individual words. If your printer can handle microspacing between each letter without slowing down considerably, you may want to set this to 2 or 3, so that words will also be stretched. Some people even like to set *WordSpacing* to 1, which inserts space evenly across all the letters and spaces in a line with no special consideration for stretching word spacing first. Default value is *WordSpacing 10,000*, which in effect disables this feature. Used exclusively as a modifier to the Style command. |
| WidowPrevent {*on/off/N*} | Prevents widows (a partial paragraph at the bottom of a page) and orphans (a partial paragraph at the top of a page). *N* is the minimum number of lines permissible at the bottom or top of a page. *On* is the same as entering 1; *off* is the same as entering 0. Setting this parameter to a large number (like 100) is a good way to prevent paragraphs from ever being split across pages. Used exclusively as a modifier to the Style command. |

***Enumeration Parameters***

| | |
|---|---|
| Counter <*variable*> | Uses the *variable* as the counter (if *variable* is not specified, a "local" counter will be used, which can be referenced by the name *counter*). Used exclusively as a modifier to the Style command.<br><br>Sets the built-in variable *Counter* to the specified variable for this format only. This command is used mainly to affect the number the formatter assigns to text affected by the **D**efine a Tag and **R**eference a Tag commands. The STANDARD.FMT file, by default, sets this counter to *SectionNumber*, but you can use this parameter to override this *Counter* setting. Note, however, that the formatter doesn't increment each paragraph if you set the counter with a Style command.<br><br>Basically, *Counter* tells tags what variable to save for later reference. Therefore, *Style counter SectionNumber* causes tags to save the current section number for later reference. Similarly, *Style counter Figure* causes tag to reference the *Figure* variable. |

Table 1.8: Format Parameters (Complete List), continued

| | |
|---|---|
| Increment *<variable>* | Same as *Counter*. |
| Numbered {*"string"*} | Increments the counter for each paragraph. *"String"* is an optional template for the counter. |
| Within *<variable>* | Sets a global variable to be the "parent" of the counter. (If *variable* is not specified, the enclosing format's counter is used.) |
| ***Macro Parameters*** | |
| AfterEntry *"text"* | Defines a macro to be executed immediately after the format is started. |
| AfterExit *"text"* | Defines the macro to be executed in the format that called this one. This command is usually used to print reference numbers to floating environments. |
| BeforeEach *"text"* | Defines the macro to be executed before each paragraph (as long as the paragraph is separated from the previous paragraph by a blank line and is not indented). |
| BeforeExit *"text"* | Defines a macro to be executed at the end of the format. |
| Divider *"text"* | Executes a macro to produce the "divider line" between a footnote or figure and the regular text. *Divider* is used only by float and sink formats. |
| Initialize *"text"* | Same as *AfterEntry*. |
| WithEach *"text"* | Similar to *BeforeEach*, except all leading text up to the first tab is read in and placed in the variable *"text"*. The macro is then executed. |
| ***Floating Parameters*** | |
| Above | "Floats" this format to the top of the column. |
| After/Before | Saves text to be printed either at the end of the document or at the very start. The text saved is exactly what was placed in the format call. You can imagine this as a file into which all text is written, then that file is reread by the formatter and formatted. New lines are not appended after each item. *Initialize* and *BeforeExit* are done when the command is formatted. |
| Below | "Sinks" this format to the bottom of the column. |
| Free | Formats a "free" format in the column after the current line is finished. |
| Inline | Turns off any floating switches (*Free, Above, Below,* etc.). |
| Index | Used in creating indexes. *Initialize* and *BeforeExit* are executed when the format is executed. Refer to the explanation in the *Advanced User's Guide* for details. |

See Also Chapter 2 (Style), and the *Advanced User's Guide*

# Monochrome Set

Refer to the Colors entry for details.

# Month (Variable)

Refer to the Variables entry for details.

# MonthName (Variable)

Refer to the Variables entry for details.

# Mouse Commands

Function

If you are using a standard two- or three-button mouse, you can conveniently make text selections, choose menu commands, and change windows with it.

The following list explains how you can use a mouse with Sprint:

***When a Menu Is Visible***

| | |
|---|---|
| Left Button | Makes a menu choice. |
| Right Button | Cancels the menu (same as *Esc*). |

***When a Menu Is Not Visible***

| | |
|---|---|
| Left Button | Toggles text selecting (same as *F3*). You can press the button and drag across the text to select it (the Mouse menu appears when you quit dragging), or you can click once and then call up the Edit menu (click the right button three times) to extend the block.<br><br>Mouse text selection also works in column mode. |

Clicking the left button in an area to the right of the onscreen margin (that is, where there is no text) affects the current window. (If you have the right margin set to column 65, this non-text zone is 15 characters wide. If you have a file with no ruler, the non-text zone is any position that's to the right of a hard return paragraph mark.) Clicking in the upper quarter of the non-text zone switches to the previous window. Clicking in the lower quarter of the non-text zone switches to the next window. Clicking in the middle part of the non-text zone zooms or unzooms the current window.

Right Button

Displays the **Mouse** menu if you have moved the mouse without selecting text; displays the **Edit** menu after three clicks if there is text selected; displays the main Sprint menu after three clicks if you have not moved the mouse and have not selected text.

Middle Button

If your mouse has a middle button, you can drag to select text by *lines* instead of by characters. Double-clicking the middle button allows you select text by *paragraphs*.

Clicking the middle button in an area to the right of the right onscreen margin (that is, where there is no text) affects the current file. Clicking in the upper quarter of the non-text zone switches to the previous file. Clicking in the lower quarter of the non-text zone switches to the next file. Clicking in the middle part of the non-text zone calls the Pick from List file menu.

| | |
|---|---|
| Double-clicking | Double-clicking the left button select text by *words*. Double-clicking the middle button allows you to select text by *paragraphs*. Double-clicking the right button has no special effect. Subsequent double-clicks extend the selection. |

The Mouse menu consists of the following commands:

| | |
|---|---|
| Copy | Same as *F10*/Edit/Copy |
| Move-Cut | Same as *F10*/Edit/Move-Cut |
| Paste | Same as *F10*/Edit/Insert-Paste |
| Erase | Same as *F10*/Edit/Erase |
| Write Block | Same as *F10*/Edit/Write Block |
| Select BLOCK/ COLUMN | Same as *F10*/Edit/Block Select/ Column Mode |
| Typestyle | Same as *F10*/Typestyle |
| Index Word | Same as *F10*/Style/Index/Word |
| Cancel | Same as *Esc* |

Tips

If you move the mouse pointer to the top two lines of the screen, Sprint will scroll the document up; if you move the mouse pointer to the bottom two lines, Sprint will scroll down. Moving the mouse out of the two-line region stops the scrolling.

# Move-Cut

Keystrokes

*Alt-E* (or *F10*, Edit), Move-Cut

*F5*

*Del*

*Ctrl-KK*, and type D

*Ctrl-KY*

Function

Moves a marked block to the Clipboard.

This command works on selected text. When you move a block, you are instructing Sprint to remove the block from its present location and place it on Sprint's Clipboard (a memory buffer). You can then move the cursor to another area in your file, or to another file, and *paste* this block into a new location.

How To

To move a block to Sprint's Clipboard, you must first select the block. After you've done this, press any of the shortcuts listed in "Keystrokes" or choose **Edit/Move-Cut**. Sprint removes the block from its current location and places it on the Clipboard.

When you've moved the cursor to where you want to paste this block, press *F6* or choose the **Edit/Insert-Paste** command.

Tips

Sprint's Clipboard ordinarily can hold only one block at a time. This means that when you move a block of text to the Clipboard, you should *quickly* paste the block into the desired location. If you move, copy, or erase another block before you paste the first block you marked, you'll *lose* the first block you intended to move.

See Also

Block Select Menu, Clipboard, Deleting, Edit Menu

# MS Word (User Interface)

Refer to the User Interface Menu entry for details.

# Multilevel

Keystrokes

*Alt-S* (or *F10*, Style), **Lists**, **Multilevel**

Function

Numbers nested paragraphs.

The **Multilevel** command acts like **Numbered** in that it creates numbered paragraphs, but **Multilevel** uses a different type of numbering scheme that's sometimes called "military spec" numbering. Instead of **Numbered**'s ordinal, alphabetic, Roman numeral sequence, **Multilevel** numbers nested paragraphs 1., 1.1., 1.1.1., and so on.

This type of numbering works well for any highly organized document with many levels of paragraphs that must be referred to (such as contracts, bids, reference manuals, specifications, and military documentation).

You will find this command defined in the Sprint file STANDARD.FMT.

How To

Choose Multilevel from the Style/Lists menu. Type B for Begin Command when Sprint prompts and then type your text. Choose Multilevel again and type E for End command when you have finisthed typing your text.

You can also enter the text first, mark it as a block, and then select the Multilevel command. For example,

```
BEGIN MULTILEVEL
Paragraphs are numbered. So far, this looks just like
the Enumerate format.

TabAs with other list formats, if a second paragraph
continues the same subject as the preceding one, and
shouldn't be numbered, indent the first line of that
paragraph with a Tab.

BEGIN MULTILEVEL
Level(s) can be placed one inside the other.

Multi-part paragraph numbers are generated
automatically, one for each Level nested.
END MULTILEVEL

When you end the enclosing format, you return to the
previous one.

If you don't want spaces between the paragraphs in a
Multilevel format, modify it by adding the parameter
Spread 0.
END MULTILEVEL
```

The printed result looks like this:

1. Paragraphs are numbered. So far, this looks just like the Enumerate format.

   As with other list formats, if a second paragraph continues the same subject as the preceding one,

and shouldn't be numbered, indent the first line of that paragraph with a *Tab*.

1.1. Level(s) can be placed one inside the other.

1.2. Multi-part paragraph numbers are generated automatically, one for each *Level* nested.

2. When you end the enclosing format, you return to the previous one.

3. If you don't want spaces between the paragraphs in a MultiLevel format, modify it by adding the parameter *Spread 0*.

See Also Lists, Modifying Formats, Numbered

# MultiMate/MultiMate Advantage

Refer to the Translate entry for details.

# New (File)

Refer to the File Menu entry for details.

# Next Occurrence

Refer to the Searching entry for details.

# Next (Window)

Refer to the Window entry for details.

# Non-Breaking Space

Keystrokes *Alt-I*, (or *F10*, Insert), Non-Breaking Space

*Ctrl-Spacebar*

Function

Inserts a non-breaking, fixed-width space character that will not be altered or adjusted during formatting.

The Non-Breaking Space command tells Sprint to insert a special space character at the current cursor position. Sprint treats this special space character just like any other text character, and ignores it both when it looks for places to add spaces to justify lines and when it looks for legitimate spots to break a line. If necessary, the formatter will insert extra space between *characters* rather than between the words affected by Non-Breaking Space when it needs extra space to justify the line.

How To

Choose Insert/Non-Breaking Space when you want to keep words together on a line.

Tips

You often need a non-breaking space when typing company names, personal names, and abbreviations to avoid "bad breaks." For example, all of the following should have non-breaking spaces where the bullets are:

```
J.•D. Salinger
Texas A•&•M
World War•II
2•1/2 inches
```

You might want to assign company or propers names, complete with Non-Breaking Space commands, to a key on your keyboard. See the Glossary entry in this menu encyclopedia.

You can also use this command to leave space to paste in a character that your printer can't print. Since you don't want the formatter to use this space to break the line, select Non-Breaking Space from the Insert menu.

See Also

Blank Space (Horizontal), Glossary

# Non-Breaking Spaces

Refer to the Screen entry for details.

# Normal (Typestyle)

Refer to the Typestyle Menu entry for details.

# Notes

Refer to the Footnote entry for details.

# Numbered

Keystrokes — *Alt-S* (or *F10*, Style), **Lists**, **Numbered**

Function — Creates a numbered list.

This command tells Sprint to number the paragraphs comprising a list. If you add or delete items from the list, Sprint automatically renumbers the items when printed.

You can also create *sublists* within a list (as you do when creating an outline) by nesting Numbered formats. Sprint numbers each paragraph of the main list with ordinal numbers (1, 2, 3,...); assigns lowercase letters (*a-z*) to items in the first sublist; and numbers items of a second sublist with Roman numerals (i, ii, iii,...). If you create a third sublist (three Numbered commands within an Numbered format), Sprint starts repeating the numbering sequence (ordinal numbers, lowercase alphabetic characters, Roman numerals).

You will find this command defined in the Sprint file STANDARD.FMT.

How To — If you've already typed the text of your list, select the list and then choose **Numbered** from the **Style/Lists** menu. Sprint automatically inserts the BEGIN NUMBERED command on the line above your list, and the END NUMBERED command on the line following your marked text.

If you haven't yet entered the text to be numbered, choose **Numbered** from the **Style/Lists** menu. Sprint prompts:

```
Press (B) for Begin command, (E) for End command,
or ESC for cancel:
```

Type B, and Sprint automatically inserts the BEGIN NUMBERED command. Enter the text of your list. Type E for End command after you've typed the list.

Make sure there's a blank line between each paragraph. Sprint uses these blank lines to determine where one paragraph ends and the other begins. Press *Tab* as the first character of any paragraph that you *don't* want numbered within the list.

```
BEGIN NUMBERED
Remove the front cover.

Tab(Use a flat-head screwdriver for this.)

Connect the widget cable to the widget connector, and
lock the latch.

Replace the front cover.
END NUMBERED
```

Your printed list looks like this:

1. Remove the front cover.
   (Use a flat-head screwdriver for this.)
2. Connect the widget cable to the widget connector, and lock the latch.
3. Replace the front cover.

Note that the paragraph under step 1 is not numbered because it starts with a *Tab* character.

If you don't want spaces between the paragraphs, *don't* delete the blank lines. Instead, modify the Numbered format by adding *Spread 0*. Once you enter the *Spread* parameter, your file prints like this:

1. Remove the front cover.
   (Use a flat-head screwdriver for this.)
2. Connect the widget cable to the widget connector, and lock the latch.
3. Replace the front cover.

If you want to cross-reference text in a Numbered format, use the commands listed on the X-Reference menu. For example, let's say in the last step of an example, you want to refer the reader to the first step. Use the Define a Tag command to create a new tag named "remove" and make it equal to the variable named *Counter*. Then use the Reference a Tag command to insert a reference by section number (Sprint lets you choose a reference either by page number or section number.) Here's how to do this:

```
BEGIN NUMBERED
Remove the four screws securing the front cover,
slide the back panel off the unit, and then
disconnect the ribbon cable.TAG remove=counter

Insert the widget in slot 2.

Connect the widget cable to the widget connector, and
lock the latch.

Replace the front cover by reversing the steps listed
in Step remove.
END NUMBERED
```

Your printed list looks like this:

1. Remove the four screws securing the front cover, slide the back panel off the unit, and then disconnect the ribbon cable.
2. Insert the widget in slot 2.
3. Connect the widget cable to the widget connector, and lock the latch.
4. Replace the front cover by reversing the steps listed in Step 1.

Tips

You can nest Numbered commands, which is useful in creating detailed procedures or outlining text. Sprint uses letters to enumerate nested Numbered commands. This is especially useful for contracts, bids, and specifications. (See Multilevel in this menu encyclopedia for details.)

See Also

Modifying Formats, Multilevel, Tags

# Number of Copies

Refer to the Print Menu entry for details.

# Number of Passes

Refer to the Advanced Options entry for details.

# Odd Pages

Refer to the Footer Menu or Header Menu entry for details.

# Offset

Refer to the Document-Wide Menu entry for details.

# Open (File)

Keystrokes *Alt-F* (or *F10*, File), Open

*Ctrl-F3*

Function Opens a file.

Before you can edit an existing file, you need to *open* it. The Open command tells Sprint to look for the file on your disk, read it into the swap file, and display the file in the current window. Once you open a file, you can view, edit, and print the contents.

When you ask Sprint to open a file without specifying a file extension, Sprint first looks for a file with no extension. If it can't find a file name like that, it looks for a file with the Sprint default extension .SPR. Note that if you have two files, one named TEST.SPR and one named TEST (with no extension), you have to include the extension when opening it. Otherwise, Sprint will find and open TEST but not TEST.SPR.

How To

There are two ways to open a Sprint file:

1. From the DOS command line.
2. By choosing **O**pen from the **F**ile menu (or using the *Ctrl-F3* shortcut).

For details on opening Sprint files from DOS, see Chapter 4.

To open a file from Sprint's menus, choose **F**ile/**O**pen (or press *Ctrl-F3*). Sprint displays the `File to open:` prompt. Enter the name of the file you want to open and press *Enter*. Sprint then displays the file on your screen.

If you can't remember the exact spelling of the file you want to open, you can use wildcards in your file name at Sprint's prompt. Sprint displays a list of all file names that match your criteria. Once you select a file from the list, Sprint opens and displays the selected file.

You can also just press *Enter* at the `File to open:` prompt to choose a file from a list of files whose extension is .SPR.

If you enter a file name that does not currently exist, Sprint displays the following prompt:

```
Create new file FILENAME?
```

If you want to create a file by that name, type `Y`. Sprint creates the file for you and inserts a ruler line at the top of the file. If you got this prompt because you made a mistake in typing the file name, type `N`.

Note that if you want to create a new file that does *not* have the .SPR extension, you must enter a period (.) after the file name.

If the specified file is already open when you select the **O**pen command from within Sprint, Sprint will simply display the open file.

Tips

You don't have to close a file before opening another; Sprint lets you open and edit up to 24 files at the same time. If you are using multiple windows, you can display a different file in each window (up to six) by

moving to the window and then choosing the Open (file) command.

See Also File Manager Menu, File Menu, Pick from List

# Open (Window)

Refer to the Window entry for details.

# Options Menu (Customize)

Keystrokes *Alt-C* (or *F10*, Customize), **O**ptions

Function Displays the **O**ptions menu.

This menu lets you make several customization settings:

**Preserve Editing Session**
If set to Yes, Sprint retains the swap (backup) file every time you quit and automatically reopens it when you start Sprint again. If set to No, Sprint still creates a swap file while you are working (so you'll still have a backup if the power goes out or your system crashes) but does not preserve this file from one Sprint session to another.

**Background Save Period**
Lets you set the number of seconds of inactivity that Sprint must wait before it saves the current state of all your open files to the swap file. If you set the number to 0, Sprint does not save your work to the swap file at all, except to occasionally free up some memory for editing tasks. Default setting is 3 seconds; the maximum setting is 60 seconds.

**Menu Display Delay**
Lets you set the amount of delay time (in tenths of seconds) before Sprint displays a menu. Entering 0 means there is no delay (the default setting). Making the number larger means you will be able to type ahead of the menu displays; that is, you can choose menu commands without having to show them!

**Insert Mode**
If Insert, typing pushes existing text to the right (status line shows *Ins*); if Overwrite, typing overwrites existing text (status line shows *Ovr*).

**Tone**
Displays a menu that lets you change the sound (pitch and length) of Sprint's beep. (Use the *Right* and *Left arrow* keys to change the length; use the *Up* and *Down arrow* keys to change the pitch.)

How To

Press *Alt-C* to view the **C**ustomize menu and then choose **O**ptions.

**Note to two-floppy system users:** Sprint may have to prompt you to insert the correct Program Disk into Drive A in order to use the **C**ustomize/**O**ptions menu.

See Also

Preserve Editing Session, Background Save Period, Insert Mode, Tone

# Options Menu (Print)

Refer to the Print Menu entry for details.

# Order

Refer to the Merge entry for details.

# Other Format

Keystrokes

*Alt-S* (or *F10*, Style), **O**ther Format

Function

Lets you enter your own formats.

This command instructs Sprint to invoke a particular format. The format can be one of lesser-used, built-in commands, or it can be one you may have defined with Sprint's Define or Macro commands.

When you type in the name and press *Enter*, Sprint prompts you to choose whether the format is applicable

to a Region of text or not (then it's a Command). If you choose Region (press *R*) and don't have text selected, Sprint prompts you to decide whether this is the beginning or end of the format block. If you select a text block first, however, Sprint inserts the correct BEGIN and END commands around the block. If you choose Command (press *C*), Sprint enters and highlights your format onscreen with the hidden control codes ^O and ^N beginning and ending your format command.

For information about how to modify the formats defined in the STANDARD.FMT file, refer to Part 1 of the *Advanced User's Guide*, called "Advanced Formatting."

If you enter a Begin format and forget to enter the corresponding End format, Sprint displays an error message during formatting and will not print your file until you correct the error.

See Also — For a detailed list of the commands you can enter using **O**ther Format, see Chapter 2 of this guide.

# Outline

Keystrokes — *Alt-S* (or *F10*, Style), Lists, Outline

Function — Creates an outline-style list.

This command tells Sprint to number the paragraphs comprising a list as if it were an outline.

That is, the first-level items are numbered with Roman numerals (I, II, III); the second-level items (which comprise all the items in the first nested Outline command) are numbered in capital letters (A, B, C); the third-level items (in the second-nested group) are numbered with Arabic numbers (1, 2, 3); the fourth-level are in lowercase letters (a, b, c); the fifth are in lowercase Roman numerals (i, ii, iii).

You'll find this command defined in the Sprint file STANDARD.FMT.

How To — If you've already typed the text of your list, select the list and then choose **O**utline from the Style/Lists menu.

Sprint automatically inserts the BEGIN OUTLINE command on the line above your list, and the END OUTLINE command on the line following your marked text.

If you haven't yet entered the text to be numbered in outline style, choose **Outline** from the **Style/Lists** menu. Sprint prompts:

```
Press (B) for Begin command, (E) for End command,
or ESC for cancel:
```

Type `B`, and Sprint automatically inserts the BEGIN OUTLINE command. Enter the text of your list. Type `E` for End command after you've typed the list. Sprint inserts the END OUTLINE command in your text.

If you already have typed the text of your list, you can select the text first and then choose **Outline**. Sprint then automatically inserts the BEGIN and END OUTLINE commands around your text.

Repeat the procedure to create as many sublists as needed in your outline.

Make sure there's a blank line between each paragraph. Sprint uses these blank lines to determine where one paragraph ends and the other begins. Press *Tab* as the first character of any paragraph that you *don't* want numbered within the list.

Tips

If you prefer to work in a full-featured outline program, you can use the Outlook outliner in SideKick Plus. When you have finished your outline there, you can import it into Sprint using the **File/Translate/Import/SideKick Plus** command. When you choose this command, Sprint prompts you whether you want to have the Outlook outline converted using the **Outline** command or using tabs.

See Also

Lists, Translate

# Page (Variable)

Refer to the Variables entry for details.

# Page Breaks

Keystrokes

*Alt-L* (or *F10*, Layout), Page Breaks

*Alt-N* (insert unconditional page break)

Function

Displays the Page Breaks menu.

The Page Breaks menu gives you control over how Sprint will break you text into pages when printing.

When you choose the Page Breaks commands, Sprint displays a menu with these commands:

**Insert (Unconditional)** *Alt-N*
Inserts a mandatory page break (which shows onscreen as a bold horizontal line). Sprint will always start a new page when it encounters this command.

**Conditional Page Break**
Inserts a page break that tells Sprint it *can* (but doesn't have to) break to a new page at that point. This command appears onscreen as PGBREAK.

**Reserve Space**
Inserts a command telling Sprint to create a specific block of vertical blank space. Sprint will try to put the entire space on a single page. If it doesn't fit, Sprint puts the whole amount of blank space on the next page. This command appears onscreen as RESERVE.

**Blank Page(s)**
Inserts a command telling Sprint to create one or more blank pages (but headers and footers still print on them). This command appears onscreen as PGBLANK.

**Group Together on Page**
Lets you select text that should not be broken across separate pages. This command shows up onscreen as BEGIN GROUP and END GROUP.

**Keep with Following Text**
Inserts a command that tells Sprint it cannot break to a new page at that spot no matter what. This command appears onscreen as KEEPFOLLOWING.

**Widow-Orphan Control**
Lets you determine what the minimum number of lines of a paragraph can appear at the top (= "orphan") or bottom (= "widow") of a page. Sprint will only fix one-line orphans (by moving an extra line from the previous paragraph). But you can prevent widows of any size. Setting Widow-Orphan Control to a large number (like 100) is a good way to prevent paragraphs from *ever* being split across pages.

See Also

In addition to the commands in the Page Breaks menu, you might refer to the entries on pagination in this chapter (Paginate and Remove Formatter Page Breaks), and the commands HaveSpace and NeedSpace in Chapter 2.

# Page Numbers (General Information)

Keystrokes

*Alt-E* (or *F10*, Edit), Go to Page

*Alt-I* (or *F10*, Insert), Variable, *Page*

Function

Go to Page moves the cursor to the specified page number.

The Go to Page command lets you move the cursor to a specific page number in the current file. This is especially helpful when you are working with large files. You have to use Sprint's Paginate command to generate onscreen page breaks before you can choose Go to Page.

The Variable command can be used to reference the variable *Page*.

The variable *Page*, referenced by choosing Insert/Variable, prints the current page number. When the formatter encounters this command, it determines the current page number and replaces the command with the actual page number. This is useful in page headers and footers.

How To

When you choose the Go to Page command, Sprint displays the following prompt in the status line:

```
Page number: 1
```

Enter the number of the page (1 is default). Sprint immediately moves the cursor to the first character on the page you specified.

The current position of the cursor doesn't matter. The cursor can be positioned either before or after the specified page number when you select the Go to Page command.

Sprint automatically numbers each page of your document. By default, the formatter prints page numbers as Arabic numbers (numerals like 1, 2, and 3), but you have the option to print Roman numerals, cardinal, or ordinal numbers, or words. See the Variables entry for details.

*Page Numbers in Headers and Footers*
By default, the first page of your document is not numbered, the second page is numbered 2, the third is 3, and so on. The page number appears in the center of the page footing (at the bottom of the page). If you select the **F**ooter command on the **L**ayout menu, you can override this automatic page numbering. If you don't enter any text in your footer command, the formatter leaves the footer blank. If you enter text, the formatter prints the text instead of the page number. If you want to print text *and* a page number in either your header or footer, you need to use the **I**nsert/**V**ariable command and reference the variable *Page*. See the Footer, Header, and Variable entries for details.

*Chapter/Section Numbers in the Page Number*
If you want your page numbers to include the current chapter number (like 1-1, 1-2, 2-1, and so on), you need to tell the formatter to include the *"parent"* in the page number. Near the top of the file, choose **S**tyle/**O**ther Format and type `Parent page = chapter` (press *C* for a Command).

Likewise, if you want page numbers preceded by a section number, type the Parent command and make *Page* equal to *Section*. See the Parent entry in Appendix C of the *Advanced User's Guide* for details on this command.

*Page Number Reference*

When you want to reference text that appears on a particular page of your document, use the **Define a Tag** and **Reference a Tag** commands. For example, if you want to reference the page on which a table of phone numbers appears, set a tag for the table and then reference the tag when you want to reference the page on which the table appears. For example:

- Select the Table format (type *Alt-S*, then choose **Table**). Type the caption in response to Sprint's prompt or leave it blank if there is no caption.
- Choose **Define a Tag** from the **Style/X-Reference** menu.
- Type the following in response to Sprint's prompt:

```
Name for new tag: phone=table
```

- Type the text of your table (for example, a list of phone numbers).
- Continue typing the text of your document. When you want to reference the page on which the table appears, select the **Reference a Tag** command from the **Style/X-Reference** menu. For example,

```
Contact Pat Cole to get a copy of this document.
Her phone number is listed on page
```

  At this point, choose the **Reference a Tag** command and type the following response to Sprint's prompt:

```
Tag to reference: phone
```

- When Sprint displays the **Reference By** menu, choose **Page number**. Sprint inserts the command `PAGEREF phone` after the text "...listed on page." The text now looks like this:

```
Her phone number is listed on page PAGEREF phone.
```

- When your document prints, the formatter replaces the PageRef command and the tag name *phone* with the page number on which the table appears.

*Renumbering Pages*

Sprint automatically begins numbering with page number 2 and continues numbering each page sequentially. If you have a document that's made up of several

files, and you want the files to have sequential page numbers, you can

- Create one large document that contains the text of all your files.
- Create a *master* file that uses the formatter command Include to merge your separate files.

  For example, let's say your document is made up of three files: INCOME.SPR, INVEST.SPR, and LOSSES.SPR. Your master file could be named MYMONEY.SPR, and could include the text of your headers and footers and the following Include commands:

  ```
  INCLUDE INCOME.SPR
  INCLUDE INVEST.SPR
  INCLUDE LOSSES.SPR
  ```

  To insert an Include command, move the cursor to a blank line and choose **S**tyle/**O**ther Format. When Sprint prompts for a formatter command, type `Include` plus the file name. For example,

  ```
  Include INCOME.SPR
  ```

  Press *Enter* to end the formatter command and then press *Enter* again to create a blank line. Choose **S**tyle/**O**ther Format again and type the text of your next Include command.
- You could also maintain your separate files without creating a master file.

  At the top of each file, type the formatter command Set and specify a beginning page number. Then choose **L**ayout/**P**age Breaks/**I**nsert (Unconditional) (or type *Alt-N*). You must set the page number before the next page starts.

  For example, let's say that your document is made up of three separate files, and the first file ends on page 12. You don't need to do anything with the first file if you want it to begin printing with page 1.

  At the top of the second file, however, choose **S**tyle/**O**ther Format and type

  ```
  SET page=12
  ```

  Now choose **L**ayout/**P**age Breaks/**I**nsert (Unconditional) (or type *Alt-N*). The first page of your second

file will be numbered 13. If this file ends on page 28, edit the third file to include the following formatter command:

```
SET page=28
```

Be sure to choose the **Insert** (Unconditional) Page Break command after typing the Set command. The third file will begin on page 29.

See Also — Page Breaks, Footer Menu, Header Menu, Tags, Variables, and Chapter 2 (Include, Set)

# Page Range

Refer to the Index entry for details.

# Paginate

Keystrokes — *Alt-P* (or *F10,* **Print**), **Paginate**

*Ctrl-F7*

Function — Shows onscreen where pages will begin and end when the file prints.

This command tells Sprint to format your file and determine where it will break the pages (that is, end one page and begin the next one). Sprint inserts a bold line to indicate each new page. If you want to remove these page indicators, you can select the **Remove Formatter Page Breaks** command. If you don't like where Sprint breaks a particular page, you can use the **Page Breaks** commands to override the automatic page break.

How To — When you choose this command, Sprint begins formatting the file as if it were going to output the file to the currently selected printer; instead of printing, however, Sprint displays a bold line wherever it will start a new page.

Tips — If you want to display the page breaks as they would appear on an alternate printer, use the **Current Printer** command in the **Print** menu to select the desired printer. Now when you select the **Paginate** command, Sprint for-

mats the file and displays the page breaks for the printer you selected.

You can use the **Paginate** command for an easy way to spot formatting errors in your file. If Sprint finds any formatting errors when paginating, it inserts a bold line and an error message at the location of each error. This makes it easier to spot and fix errors in your document before printing.

**Note:** Sprint automatically saves your file to disk as part of the **Paginate** command.

See Also — Page Breaks, Remove Formatter Page Breaks, Current Printer

# Paper Size

Refer to the Document-Wide entry for details.

# Paragraph

Keystrokes — *Alt-S* (or *F10*, **Style**), **Headings**, **Paragraph**

Function — Prints a numbered paragraph heading.

The **Paragraph** command prints a numbered title in boldface type, flush with the left margin, and makes an indented entry in the table of contents. **Paragraph** sectioning is the lowest level of sectioning in the STANDARD.FMT file.

The number Sprint assigns to the paragraph title marked with the **Paragraph** command depends on the number and type of sectioning commands entered before this command. For example, if you don't enter any other sectioning commands and enter a **Paragraph** command, Sprint numbers the paragraph 1. If you enter one or more numbered headings commands (**Chapter**, **Section**, or **Subsection**) before the **Paragraph** command, Sprint includes the number of each sectioning command in the paragraph number. See "How To" for an example that illustrates this.

You will find this command defined in the Sprint file STANDARD.FMT.

How To

To create a new, numbered paragraph, type the title of your paragraph on a separate line, press *Alt-S*, and then choose **Headings/Paragraph**. A paragraph heading command might look like this:

```
PARAGRAPH Artificial Intelligence
Artificial Intelligence is, loosely speaking, the
effort to make computers more like people.
```

This results in:

#### 3.2.1.7 Artificial Intelligence

Artificial Intelligence is, loosely speaking, the effort to make computers more like people.

This example is based on a file that has three **Chapter** commands; after the third **Chapter** command, there were two **Section** commands and then one **Subsection** command. This example is the seventh **Paragraph** command following the **Subsection** command. If the file didn't have any of these table of contents commands in it, and if this were the first **Paragraph** command in the file, the paragraph title would print like this:

#### 1 Artificial Intelligence

Tips

You can also create numbered paragraphs with the **Numbered** format. This format numbers each new paragraph with a single ordinal number, which is used when creating numbered lists.

See Also

Chapter, HeadingB, HeadingC, Paragraph, Section, Subsection

## Paragraph Marks

Refer to the Screen entry for details.

# Paragraphs (General Information)

Function

In Sprint terms, a *paragraph* is one or more lines of text preceded by a blank line (two hard return characters in a row). This concept is important because many of Sprint's formatting commands affect paragraphs, and we normally think of a paragraph as being several sentences in length. With Sprint, that's not necessarily the case.

*Formatting Paragraphs*

The **S**tyle menu lists several ways to format paragraphs in your file. For example, the **H**yphens format places a hyphen before the text of each paragraph. Therefore, if you want to start a line with a hyphen, the line must be preceded by at least two hard return characters. Likewise, the **N**umbered format numbers each paragraph within the format. If you want text to be numbered, it must be preceded by two hard return characters.

*Spacing between Paragraphs*

The *Spread* parameter lets you specify the distance between paragraphs. You can modify formats to include this parameter or add this parameter to the Style command at the top of your file. For example, if you want your paragraphs to be separated by two blank lines, include the parameter *Spread 2* in a Style command at the top of your file. If you're creating a list with the Numbered format and don't want blank lines between the lines of your list, you can modify the Numbered format command so it looks like this:

```
BEGIN NUMBERED Spread 0
```

*Spread 0* removes the blank lines between paragraphs, *Spread .5* replaces a blank line with one-half of one line, and so on. The *Spread* parameter accepts a variety of other dimensions (inches, points, centimeters, millimeters, portions of a page, and so on). For a complete list of valid dimensions, see page 58.

To adjust the spacing between paragraphs for your *entire* file, choose **L**ayout/**D**ocument-Wide/**I**nter-Paragraph Spread.

*Keeping One or More Paragraphs Together*
The **G**roup Together on Page command keeps a section of text from being split over two pages. You can enclose a single paragraph within this format to make sure that the entire paragraph appears together on a page, or you can group multiple paragraphs. If the current page does not have sufficient space to print all paragraphs within a grouped format, the formatter will automatically begin a new page.

There's one other command that can force the formatter to keep paragraphs together: **L**ayout/**P**age Breaks/**K**eep with Following Text. Normally, the formatter will split two paragraphs at the blank line between them. Most times, this is an acceptable break, but when it's not you can enter this command.

See Also — Keep with Following Text, Group Together on Page, Page Breaks, Style Menu, Modifying Formats

# Paragraph Selecting

Refer to the Block Select Menu entry for details.

# Paragraph (Variable)

Refer to the Variables entry for details.

# ParagraphTitle (Variable)

Refer to the Variables entry for details.

# Pause Between Pages

Refer to the Print Menu entry for details.

# Pick from List

Keystrokes — *Alt-F* (or *F10*, File), Pick from List

*Ctrl-F9*

Function — Lists all open files.

This command lists all files that are currently open, and lets you pick the one you want to display. The name of the current file appears at the top of the list, followed by the "next" file, all the way to the bottom of the list, which displays the "previous" file. By default, the previous file is selected. If an asterisk (*) appears to the left of a file name, it means that the file has been modified, but not saved.

How To — Once you enter the Pick from List command, Sprint displays a menu of open files. This menu lists all open files in the swap file; file names include the drive name and directory. Move the cursor to the file you want to display and press *Enter*. Sprint then displays the selected file.

See Also — Open (File)

# Pitch

Refer to the Tone entry for details.

# Place Mark

Keystrokes — *Alt-E* (or *F10*, Edit), Place Mark

*Alt-G* (goes to a mark)

*Alt-M* (sets a mark)

Function — Allows you to set and jump to specific spots in your documents.

When dealing with long documents, it is often desirable to be able to define a location, so you can later go back to

that spot quickly and easily. Sprint's place markers let you do that. You can have up to 10 such marks.

How To

Choose **P**lace Mark from the **E**dit menu. Choosing **S**et lets you insert an invisible mark at your cursor's present location (at the exact spot the cursor is flashing). Adding or removing text in front of the place mark moves the mark along with the character it was set on. Sprint prompts you for a marker number (from 0 to 9). Type a number.

To quickly go back to that spot, choose **P**lace Mark again, and choose **G**o To. When prompted, enter the same number. Immediately Sprint places the cursor at the invisible marker location.

Tips

You'll find the shortcuts *Alt-G* and *Alt-M* especially handy for quickly setting and jumping to numbers.

You don't have to delete a mark to set a new one. If you've already used 1 as a mark, you can use it again by resetting it at a new spot.

If you choose to go to a place mark that's been set in a different file, Sprint automatically makes the other file the current one.

You'll lose your place marks whenever Sprint exits from the editor. This means, for example, you'll lose all place marks you've set whenever you print, paginate, or quit.

See Also

Edit Menu

# Plain (Variable)

Refer to the Variables entry for details.

# Position

Refer to the Header and Footer entries for details.

# Potpourri

Keystrokes *Alt-U* (or *F10,* **Utilities**), **Potpourri**

Function Displays the **Potpourri** menu.

These handy, miscellaneous commands (which are really Sprint editor macros in disguise) have names descriptive of their functions.

Table 1.9 lists all the commands available on the **Potpourri** menu and what they do.

Table 1.9: Potpourri Menu Commands

| | |
|---|---|
| Again | Repeats the last editing command you selected. |
| BottomOfFile | Moves the cursor to the bottom of the current file. |
| BottomOfScreen | Moves the cursor to the bottom of the current screen. |
| CaseRotate | Looks at the case of the current word and changes its letters first to all uppercase, next to all lowercase, and last to first-letter capitalized only. That is, it changes the word "red" to *Red,* then *RED,* then *red.* But it changes the word "IBM" first to *ibm,* then *Ibm,* then back to *IBM.* This commands works on single words only, not blocks of text. |
| CaseSwitch | Changes the selected text to all uppercase or all lowercase. If the first letter of the selected block is uppercase, Sprint changes the entire block to lowercase; if the first letter is lowercase, Sprint switches the block to all uppercase. For example, "My IBM" switches to "my ibm," and "your Apple" switches to "YOUR APPLE." |
| CenterTab | Centers text between two tab stops on a ruler by inserting wide spaces (springs). |
| DecimalTab | Moves your cursor to the next tab stop and allows to use it as a *decimal* tab. When your cursor is at the decimal tab stop, Sprint treats new numbers or characters you type as belonging to the left of the decimal place. In other words, the characters are inserted to the *left* of the cursor position. When you type a period (a decimal place), Sprint prints it at the tab stop. Any characters typed after you entered the period are inserted to the *right,* as usual. Note that you must type a period (or press *Esc*) to end the DecimalTab command. |
| DeleteLine | Deletes the current line. |
| DeleteLineBeg | Deletes the current line from the current cursor position to the beginning of the line. |
| DeleteLineEnd | Deletes the current line from the current cursor position to the end of the line. |
| DeleteParagraph | Deletes the current paragraph. |

| | |
|---|---|
| DeleteSentence | Deletes the current sentence. |
| DeleteToChar | Allows you to specify to which character Sprint should delete text to the right. Upper and lowercase letters are treated as equivalent. |
| DeleteWord | Deletes the current word. If the cursor is in the middle of a word, deletes the cursor to the end of the word. |
| DeleteWordLeft | Deletes the current word from the current cursor position left. |
| FileCloseAll | Closes all opened files. Sprint prompts whether to save the file(s) before closing the files and leaves you with an unnamed blank file. |
| FileNext | If you have more than one file open, this command displays the next open file. |
| FilePrevious | If you have more than one file open, this command displays the previous opened file. |
| FindCharBack | Moves the cursor back (to the left) to the specified character. Upper and lowercase letters are treated as equivalent. |
| FindCharFwd | Moves the cursor forward (to the right) to the specified character. |
| FileSaveAll | Automatically saves all currently open files. |
| GotoNextPage | Advances the screen one page forward. |
| GotoPrevPage | Displays the previous screen page. |
| MoveParagraphBack | Places your cursor at the beginning of the current paragraph. If you are not within a paragraph, this command moves the cursor to the beginning of the previous paragraph. |
| MoveParagraphFwd | Moves the cursor to the beginning of the next paragraph. |
| MoveSentenceBack | Moves the cursor to the beginning of the sentence. |
| MoveSentenceFwd | Moves the cursor to the beginning of the next sentence. |
| MoveWordBack | Moves the cursor to the previous word. |
| MoveWordFwd | Moves the cursor to the next word. |
| RegionIndent | Indents the selected text to the nearest tab stop by adding a tab to the beginning of every line. |
| RegionOutdent | "Outdents" the selected text (that is, creates a hanging indent). |
| RepeatCount | Allows you to set the number of times to repeat a keystroke. |
| ReplaceAll | Allows you to search for all occurrences of a string and then specify the desired replacement string. |
| RightTab | Inserts a right tab (a wide space) at the current cursor position. |
| ScrollDown | Scrolls the screen down. |
| ScrollLeft | Scrolls the screen to the left. |

| | |
|---|---|
| ScrollRight | Scrolls the screen to the right. |
| ScrollUp | Scrolls the screen up. |
| TemporaryIndent | Indents the current paragraph by moving the left indent to the next tab stop (same as pressing *Shift-Tab*). The paragraph following the current one returns to the original indentation. |
| TimeDate | Inserts today's date at the current cursor position. |
| TimeTime | Inserts the current time at the current cursor position. |
| TimeWeekDay | Inserts the day of the week at the current cursor position. |
| TopOfFile | Moves the cursor to the top of the file. |
| TopOfScreen | Moves the cursor to the top of the current screen. |
| TransposeChars | Switches the position of the current letter and the previous letter. |
| TransposeLines | Switches the current line and the line following it. |
| TransposeWords | Switches the position of the current word and the previous word. |
| WindowCloseAll | Closes all currently open windows. |

How To

Choose **Potpourri** from the **Utilities** menu. You can use the arrow keys and *PgUp* and *PgDn* to see more of the list. Choose the command you want and press *Enter* to execute the command.

Tips

If you find one or several Potpourri commands particularly useful, you should assign them to a shortcut keystroke. Select the command name and then press *Ctrl-Enter*. You can then press the key you want to assign to the command. Be sure you don't assign a command to an existing shortcut key that you may want to use later (*Shift-Alt* combinations are usually "safe").

See Also

Deleting, Window, Scrolling, Function Keys

# Precise Settings Menu

Keystrokes

*Alt-L* (or *F10*, **L**ayout), **R**uler, **P**recise Settings

Function

Displays a menu of commands that affect your current ruler.

**Font**
Lets you specify a particular font for printing your file.

**Size**
Lets you specify type size (for PostScript printers only).

**Initial (First Line) Indent**
Inserts a hidden ruler command for indenting only the first line of a paragraph.

**Left Indent**
Inserts a hidden ruler command for setting a new left margin relative to the previous one.

**Right Indent**
Inserts a hidden ruler command for setting a new right margin relative to the previous one.

**Tab Stops**
Inserts a command (showing up onscreen as TABSET) that sets tabs at precisely the spots you indicate. Sprint prompts you: `Set tabs at:`, and you respond with exact tab stops like

```
2 picas, 1 inch
```

This sets the first tab at 2 picas and the second one at 1 inch. The results of these tab stops is only evident at print time.

These commands allow you to make "precise" settings for your file in terms of any horizontal measurement (see Table 1.2 on page 58 for Sprint's formatting dimensions), font, or type size.

See Also — Font, Font Size, Indenting, Tab Stops

# Preserve Editing Session

Keystrokes — *Alt-C* (or *F10*, Customize), **Options**, **Preserve Editing Session**

Function — Determines whether Sprint erases the swap file from disk when you quit Sprint.

This command tells Sprint whether it should erase the contents of the swap file (backup file) from disk when you leave Sprint. If **Preserve Editing Session** is Yes, and you do not close a file before leaving Sprint, the file will still be open when you access Sprint the next time. If

Preserve Editing Session is No, Sprint automatically closes all saved, open files when you select the **Quit** command. In either case, if an open file has not been saved, Sprint asks if you want to save the file before you exit.

Even if you have **Preserve Editing Session** set to No, Sprint continues to generate a backup file *while* you work in Sprint. This means that if you have a power failure or system crash, you'll still be able to recover your work when you start Sprint again.

How To

To display or change the **Preserve Editing Session** setting, choose **Customize/Options/Preserve Editing Session**.

This command is a toggle, so selecting the command automatically changes its value. Once the command displays the desired setting, you can select another command on the **Options** menu, press *Esc* to return to the **Customize** menu, or press *Shift-Esc* to resume editing.

**Note to two-floppy system users:** Sprint may have to prompt you to insert the correct Program Disk into Drive A in order to use this command.

Tips

Consider setting this command to No if you use Sprint on more than one computer. Set this command to Yes if you work on the same files often and use only one computer to edit these files.

See Also

Background Save Period

# Previous

Refer to the Window entry for details.

# Previous Format

Refer to the Modifying Formats entry for details.

# Print Menu

Keystrokes

*Alt-P* (or *F10,* Print)

*Ctrl-F7* (paginates)

*Ctrl-F8* (prints to screen)

*Ctrl-KP* (prints active file)

Function

Displays the **Print** menu.

The **Print** menu contains commands and submenus that let you format and print your files in a variety of ways, such as telling Sprint to print five copies or to start printing at a particular page number. You can also display Sprint's automatic page breaks on the screen, so you can make any necessary changes before printing. The **Print** menu also provides access to the SprintMerge utility.

**Go!** *Ctrl-KP*
Saves the file to disk, formats the file, and then outputs the formatted text to the device specified by the **Current Printer** command.

**Note to two-floppy system users:** Sprint may have to prompt you to insert the correct Program Disk into Drive A in order to print a file. Moreover, if your computer has two 360K floppy disks and no hard disk, and if you are printing to a PostScript printer, you must insert Program Disk B into Drive A before choosing **Go** or **Screen Preview**.

**Destination**
A toggle to format your file to one of two places: Printer or File. Choosing Printer sends your output to the device specified by the **Current Printer** command. Choosing File sends your output to a disk file instead of the printer. The file name defaults to the name of your file, but with the extension .PRN. When you choose **Go**, you can override this default file name if you want.

**Screen Preview** *Ctrl-F8*
Formats your document but displays it to the screen instead of sending it to the printer. Page breaks are shown by a bold horizontal line across the screen.

**Note:** Because it is impossible to accurately display proportionally spaced fonts on a fixed-width screen, Sprint will have to "squish" words and sentences together when you choose Screen Preview. This allows you to preview line breaks and spacing with better accuracy. This word compression happens only if you have installed Sprint to use a proportionally spaced font (like most laser fonts) as the default.

**Merge**
Accesses the Merge menu to print SprintMerge files.

**Paginate** *Ctrl-F7*
Causes Sprint to format the file, write the page breaks to a log file, and then display with a solid bold (or colored) line and number the automatic page breaks in your file as found in the log file. Lines with errors will also be inserted with bold lines so you can readily locate them.

**Remove Formatter Page Breaks**
Removes the page break markers (the solid bold line(s) in your file) inserted with the Paginate command. This does *not* remove page breaks inserted with the Insert/Page Breaks commands.

**Options**
Displays the Options menu of commands, which include:

| | |
|---|---|
| Starting Page | Lets you tell Sprint the number of the first page you want to print. Sprint will format the entire file but will begin printing with the page you specified. It will continue to print the rest of the file, unless you specify an Ending page. If you enter 0 for the starting page, Sprint prints the table of contents only. |
| Ending Page | Tells Sprint when to stop printing. Sprint will print all pages up to and including the page specified. The default setting is Last. |
| Number of Copies | Lets you specify the number of copies you want to print. Unless you specify otherwise, Sprint will print one copy of your file. |

| | |
|---|---|
| Pause Between Pages | Lets you pause output after each page is printed so that you can manually insert single sheets into your printer. |

**Advanced Options**
Lets you select from a list of print options, which include Number of Passes, Formatted Print, Wordwrap ASCII Files, and Log Errors to File.

**Current Printer**
Lets you specify output to an alternate (but already configured) printer.

How To — To reach the **Print** menu, press *F10* and then choose **Print**. You can use Sprint's shortcut *Ctrl-KP* to format and print the current file.

Tips — If your printer supports PostScript as a page description language, you can use the **Print/Destination** command and toggle it to File to generate the PostScript code on a disk file. After that, you can actually change the PostScript to achieve special effects (like printing in gray instead of black)—assuming you know what you're doing and know how to download the PostScript file to your printer. (You can also achieve PostScript special effects by editing the file called POSTSCR.HDR or by using the Escape command.)

See Also — Advanced Options, Current Printer, Paginate, Remove Formatter Page Breaks

# Printer (Variable)

Refer to the Variables entry for details.

# Print Options

Refer to the Print Menu for details.

# QuickCard

Keystrokes — *Alt-U* (or *F10,* Utilities), QuickCard

Function — Creates and then displays a file summarizing all shortcuts for the function keys, numeric keypad keys, and all *Ctrl, Alt,* and *Shift* versions of these keys.

Note that the QuickCard file reflects any key you have reassigned—not just the "factory settings."

How To — Choose QuickCard from the Utilities menu. After a few moments, Sprint displays a file called QCARD.TXT, consisting of a summary of all the Sprint shortcuts currently available.

Tips — You can print out the QuickCard file that Sprint generates for a handy reference. If your printer can handle it, choose a monospaced font in a small point size (7 or 8 points) and format it in two columns (choose Layout/Columns/Snaking Columns, and then type 2).

See Also — Function Keys, Help, Macros Menu, Potpourri

# Quit

Keystrokes — *Alt-Q* (or *F10,* Quit)

*Alt-X*

*Ctrl-KX* (saves file and then quits)

Function — Exits Sprint and returns you to DOS.

These commands allow you to leave Sprint and return to the operating system. *Be sure to choose the Quit command before you shut off your computer or remove your Sprint work disk.* Sprint checks to see if you have any open files that haven't been saved to disk before it "quits."

How To — To exit (quit) the Sprint program, press *F10* and select Quit (or type *Alt-Q* or *Alt-X*). Sprint checks the swap file for any open, unsaved files. If it finds such a file, Sprint displays it on the screen and prints the following message in the status line:

```
The file C:\SPRINT\FILENAME.EXT has not been saved;
save it (Y,N,ESC)?
```

If you type `Y`, Sprint saves the file and then exits. If you type `N`, Sprint won't save the file before it returns you to DOS. If you change your mind about quitting, press *Esc*.

If Sprint does not find any open, unsaved files when you select **Quit**, it immediately returns you to the DOS prompt.

**Note to two-floppy system users:** Sprint may have to prompt you to insert the correct Program Disk into Drive A in order to exit Sprint.

Tips

If you want to temporarily leave Sprint to perform a DOS command, you don't have to select **Quit**. Instead, choose **Utilities/DOS Command** and type the text of the command you want to perform. When DOS completes the command, you'll see the `Press any key to continue` message. When you press a key on the keyboard, you'll return to Sprint, and the cursor will be positioned in the file you were editing when you chose the **DOS Command**.

See Also

Close (File), DOS Command, File Menu, Save

# Recall

Refer to the Glossary entry for details.

# Record File

Refer to the Merge entry for details.

# Reference a Tag

Refer to the Tags entry for details.

# Reference By

Refer to the Page Number or Tags entry for details.

# Reference Word

Refer to the Index entry for details.

# References Menu

Keystrokes — *Alt-S* (or *F10,* Style), **R**eferences

Function — Displays the **R**eferences menu.

The **R**eferences menu contains commands for entering footnotes and endnotes into your documents.

**Footnote**
Prints a numbered footnote at the *bottom* of the current page.

**Endnote**
Creates a numbered note at the *end* of the document.

**Notes**
Prints a note at the bottom of the page with an asterisk (*) reference marker.

See Also — Endnote, Footnote

# Remove Formatter Page Breaks

Keystrokes — *Alt-P* (or *F10,* **P**rint), **R**emove Formatter Page Breaks

Function — Removes the onscreen page breaks inserted with the **P**aginate command.

If you select the **P**aginate command, Sprint shows you (by inserting a bold line) where the formatter will create page breaks when you print your file. The **R**emove Page Breaks command removes all of these page-break lines

in your file. It *does not* remove any page breaks that you manually inserted with the **Insert/Page Breaks** commands.

See Also — Page Breaks, Paginate, Print Menu

# Rename-Move

Refer to the File Manager Menu entry for details.

# Repeating Character

Keystrokes — *Alt-I* (or *F10,* **Insert**), **Repeating Character**

*Alt-U* (or *F10,* **Utilities**), **Potpourri**, RepeatCount (allows you to set the number of times to repeat a keystroke)

*Ctrl-QQ*

Function — Repeats the specified character(s) from the current cursor position to the next tab or to the right margin.

This command lets you repeat text in a line so that it fills up the space between the margins. This is useful when you want to repeat leader dots (...) in a table, or draw a line across the page.

If the text entered in the **Repeating Character** command is longer than one character and does not fit evenly in the space created, the formatter cuts the last repetition short.

Sprint has two different, but related, commands for repeating characters: *Ctrl-QQ* and **Potpourri/RepeatCount**.

The *Ctrl-QQ* shortcut prompts for the character to repeat on the screen. (Press *Esc* to interrupt.) Note that this command inserts characters on the screen and *not* at print time, which means it is less exact than the **Repeating Character** command.

The RepeatCount command from the **Potpourri** menu lets you set the number of times to repeat a certain keystroke. You can also assign this command to a keystroke of your choice.

How To

When you want to repeat one or more characters from the current cursor position to the end of the line, choose the **R**epeating Character command. Sprint prompts you

```
Character to repeat:
```

Enter the character you want to repeat. Sprint inserts the text you typed in your text, displaying it after the special greater-than symbol (>). For example,

```
Name >_
```

prints like this:

Name _____

Let's say you want to create a list of names and addresses. Between the name and address text, you want to print dot (.) characters. Choose **R**epeating Character from the **I**nsert menu and enter the following:

```
Jessie Marigold > . 219-111-1111

Lawrence Daisy > . 714-222-2222
```

Your printed text will look like this:

Jessie Marigold ..... 219-111-1111

Lawrence Daisy ..... 714-222-2222

See Also

Potpourri

# Reselect Block

Refer to the Block Select Menu entry for details.

# Reserve Space

Keystrokes

*Alt-L* (or *F10*, **L**ayout), **P**age Breaks, **R**eserve Space

Function

Inserts a specific amount of vertical blank space.

This command reserves a blank area on the page, which is useful when you want to add a figure after printing your document. You can use any vertical dimension (for example, lines, inches, points, picas, portions of a page,

and so on) to specify how much blank space to leave between the line preceding the command and the line of text following the command.

If the desired amount of blank space doesn't fit on the current page, Sprint breaks the page, starts a new page, and places the entire amount of blank space at the top of the new page.

How To

Enter the **R**eserve Space command where you want the blank space to begin. After entering the command from the **P**age Breaks menu, the word appearing onscreen is RESERVE. For example,

```
This line precedes a block of reserved space.
RESERVE 75 points
The text to follow the reserved space appears here.
```

This prints as:

This line precedes a block of blank space.

The text to follow the blank space appears here.

Tips

Don't specify more blank space than will fit on a single page. If you need more than one page of blank space, use the **I**nsert (Unconditional) command from the **P**age Breaks menu or the **B**lank Page(s) command to create a blank page, and then use the **R**eserve Space command to reserve the remaining blank space.

Refer to the Dimensions entry for a list of the other dimensions and abbreviations you can use.

See Also

Blank Page(s), Figure, Page Breaks

# Reset Shortcuts

Refer to the User Interface Menu entry for details.

# Resize

Refer to the Window entry for details.

# Rest of File

Refer to the Spelling Menu entry for details.

# Revert to Saved

Keystrokes — *Alt-F* (or *F10,* File), **R**evert to Saved

Function — Throws away all changes made to a file since you last saved the file to disk.

This command *cancels* every action you've performed on a file since you last saved it. It's useful when you've made changes to a file and then decide you really don't want to save these changes.

How To — To discard the changes made to a file since you last saved, press *Alt-F* and then choose **R**evert to Saved. When you do this, Sprint displays the following prompt in the status line:

```
Discard changes (Y,N,ESC)?
```

Sprint is telling you that you have made changes since you last saved the file to disk. If you enter `N`, Sprint ignores the command and doesn't change anything. If you enter `Y`, Sprint deletes the current version of the file and redisplays the original version of the file (the file as it appeared when you last entered a **S**ave or **P**rint command).

Tips — Before choosing this command, remember that all changes you may have made to the file since you last saved will also go away.

Also remember that the commands to print, preview, or merge your file also *save* the file. You can't print your file

and then decide you want to abandon the changes you've made.

See Also — Block Select, Deleting, File, Save

# Right Indent

Refer to the Precise Settings Menu entry for details.

# Right Margin

Refer to the Margins entry for details.

# Ruler

Keystrokes — *Alt-L* (or *F10,* **Layout**), **Ruler**

*Ctrl-OT* (toggles edit ruler)

Function — Displays a menu for commands to insert and edit a ruler line.

Sprint uses *rulers* to set many onscreen formatting characteristics of your text, like tabs, justification, and margins. In addition, other formatter settings that occur when printing (but aren't noticeable onscreen) also get invisibly placed in the onscreen ruler.

When you choose **Layout/Ruler**, these menu commands are available:

**Insert** *Alt-R*
Inserts a ruler into your file at the current cursor position.

**Edit on Screen** *Alt-A*
Moves the cursor to the most current ruler so you can edit it onscreen. Type *Esc* to abort the command and return to the place in the file you were before choosing this command.

**Precise Settings**
Displays a menu of commands that affect your current ruler. These commands allow you to make "precise" settings for your file in terms of any horizontal measurement.

**Justification**
Displays a menu that lists the various ways you can justify text affected by the current ruler: **L**eft, **R**ight, **B**oth, and **C**enter.

**Line Spacing**
Displays a menu that lists the various ways you can space your text: **S**ingle, 1.5, **D**ouble, or **O**ther. Your choice is automatically entered in the current ruler.

Sprint automatically inserts a ruler line at the top of each new file. This ruler line sets up the following format for text in your file:

| | |
|---|---|
| Left Margin | Column 0 (text prints with a 1-inch left margin). |
| Right Margin | Column 65 (sets a 6.5-inch line length). |
| Paragraph Indent | Column 0 (paragraphs are not indented). |
| Justification | Left only (right margin is "ragged"). |
| Tabs | A tab set at column 5. |

All text entered below this ruler will be formatted according to the commands you place on this line, until you insert another ruler.

If the default ruler works well for your document, you won't need to change any of its settings. If your text requires a different format, you'll need to change the ruler line settings accordingly (see the "How To" section following). If Sprint's default ruler works well for one part of your file but not for another, you'll need to insert a second ruler and modify its settings.

You can also have Sprint insert a customized ruler every time you open a new file. Simply change a ruler to your liking, then select it as a marked block. Choose **D**efine from the **G**lossary menu and give the glossary entry the name AUTOEXEC. (Sprint saves the file as

AUTOEXEC.SPG.) Now whenever Sprint opens a new file, it prompts you if you want to use your own settings, that is, your customized ruler.

There is no limit to the number of rulers you can insert in your files (as long as there is some text between separate rulers). Whenever you choose **R**uler/**I**nsert (or press *Alt-R*), Sprint inserts a ruler line with the same settings as the ruler at the top of your file. This ruler can be modified as desired. Each ruler line sets up the basic format for text following the ruler, until Sprint sees another ruler with different settings.

How To

When you choose the **L**ayout/**R**uler/**I**nsert, Sprint will automatically place a ruler line at the current cursor position. This ruler is always a copy of the first ruler in your file. (You can change the ruler to a copy of the previous ruler by pressing *F4* immediately after inserting a new ruler.) If the margin, tab, paragraph indent, justification, font, type size, and spacing settings are acceptable, you won't have to change anything.

Chances are, though, you'll be inserting another ruler line because you want to enter text that should be formatted differently from the previous text. For example, maybe you want the text at the top of your file to be formatted according to Sprint's default ruler settings, but then want part of your file to have a left margin of 10 and a right margin of 55. If you change the ruler line at the top of your file, all your text will be affected. Therefore, you'll want to insert a new ruler line immediately before the text that should have the different margin settings.

*Editing a Ruler*

Editing the default ruler settings entails either choosing **R**uler/**E**dit on Screen (or the shortcut for this command, *Alt-A*) or choosing **L**ayout/**R**uler/**P**recise Settings. The difference between the two sets of editing commands is this: The **E**dit on Screen command lets you make changes to the onscreen ruler, which is *measured in screen columns*. The **P**recise Settings command, however, lets you make changes to your ruler settings, which you do not see onscreen but are only realized when you print your document. (These settings are measured not in

screen columns, but in the more precise terms of points, picas, inches, and so on).

If your printer supports proportionally spaced fonts or can measure distances in precise amounts, you should edit your ruler with the Layout/Ruler/Precise Settings command. The changes you make using this ruler will not change the look of the document *onscreen* but will change it when it's printed.

If your printer uses only monospaced fonts, however, you should edit the ruler using the Ruler/Edit command. When you choose this command, the first ruler above your cursor is highlighted, and your cursor jumps to the ruler line. Once the ruler is in this "editable state," you can press *F4* to copy all the settings from the previous ruler. You can also type any of the following symbols right on the line to make changes to the text following the ruler line:

[ Sets the left margin.

Using the spacebar or arrow keys, move the cursor to where you want your left margin, and then type [ on the ruler line. If you do not type a left bracket ([), the left margin will be the same as the left margin set on the ruler line at the top of your file.

] Sets the right margin.

Using the spacebar or arrow keys, move the cursor to the column you want for your right margin, and then type ] on the ruler line. If you do not type a right bracket (]), the right margin will be the same as the right margin set on the ruler line at the top of your file.

I Indents (or *outdents*) the first word of each paragraph.

Some document formats, such as those for modified-block letters or memos, require indenting the first word of each paragraph a certain number of characters from the left margin. You can always space over or press the *Tab* key, but if you want Sprint to do this for you automatically, you can set a paragraph indent on your ruler line. Move to the

column where you want to begin the first word of each paragraph (for example, column 5), and enter I.

You can also set the paragraph indent to be less than the left margin. For example,

The paragraph indent in this sample paragraph is set to 0 (we typed an *I* at column 0 on the ruler line). The left margin is set at column 10.

T Sets a tab stop.

Editor tabs are tabs set at specific column numbers on your ruler line. When you set an editor tab and then press the *Tab* key, the cursor moves to the nearest editor tab stop set on the ruler.

The new tab stops will not change anything you have already typed; they only determine where Sprint will place text when you press the *Tab* key. If you enter two or more tab stops on the ruler line, Sprint will automatically repeat the tab interval between the last two tab stops. For example, if you type T at columns 5, 15, and 30, Sprint looks at the last two tab stops you set (15 and 30), determines how many spaces there are between the two (15), and then repeats the tab stops every 15 columns thereafter. This means that you would actually have tabs set at columns 5, 15, 30, 45, 60, 75, 90, and so on.

J Specifies left and right (both) justification.

Type J anywhere on the ruler line to get left- and right-justified text. Sprint automatically adjusts the words on a line so that they line up neatly at both the left and right margins. When you enter the *J*, Sprint places it immediately after the right margin setting. This paragraph shows you how left- and right-justified text looks on your screen.

L Specifies left-justified text.

Type L anywhere on the ruler line if you want text aligned at the left margin and "ragged" (not justified) at the right margin. (Sprint automatically places the letter immediately after the right margin setting.) When you enter text following this ruler line, Sprint creates lines

that are *about* the same length; when a word is going to go past the right margin setting, Sprint automatically begins a new line for you. This paragraph shows you how left-justified text looks on your screen.

R Specifies right-justified text.

Type R anywhere on the ruler line if you want text aligned at the right margin and "ragged" (not justified) at the left margin. Sprint automatically places the letter immediately after the right margin setting. This paragraph shows text that is right-justified.

C Specifies centered text.

Type this character anywhere on the ruler line if you want your text centered between the right and left margins. Sprint will automatically place the letter immediately after the right margin setting. This paragraph shows text that is centered.

RJ Specifies a fully justified paragraph whose *last* line is flushright.

CJ Specifies a fully justified paragraph whose *last* line is centered.

***Making a Precise Ruler Entry***

If your printer is capable of formatting in precise measurements (not just in terms of characters or column spaces) or if it supports many fonts, you'll want to edit your ruler with the **Layout/Ruler/Precise Settings** command. When you choose this, you see a menu with the following commands that affect the text following the ruler line:

Font — Allows you to select a font by name. Choose the font name and press *Enter*. All text following this ruler will be printed in the font you choose.

Size — Allows you to select a font size. When you select this, Sprint prompts you for the new size. Be sure to type in the dimension you want (for example, 10 points). Only printers that can scale fonts (like PostScript printers) can use this command.

| | |
|---|---|
| Initial (First Line) Indent | Allows you to set the paragraph indent in precise amounts. When Sprint prompts you for the indent, type, for example, 1.5 picas or 2 ems. A negative number will *outdent* the text from the left margin. |
| Left Indent<br>Right Indent | Lets you set the ruler indent (temporary margins). When prompted, type the margin setting, for example, 0 picas or 1.5 inches. |
| Tab Stops | Lets you set tab stops at precise locations. Again, be sure to type in the unit of measurement (for example, 5 picas, 2.5 inches, or 40 points). Sprint inserts a TABSET command in your document. |

To see the precise settings placed on the ruler line, you have to press *Alt-Z* to show the hidden Sprint control codes.

The commands you type on your inserted ruler line will remain in effect until you insert another ruler line. For example, let's say you insert a ruler line and set the left margin at column 10. You enter a few paragraphs of text, and then want to go back to a left margin of column 0 for the next few paragraphs of your file. When you insert this new ruler line, Sprint displays a left margin equal to the left margin set on the ruler at the *top* of your file. If that left margin isn't at column 0, you'll need to edit the ruler so that it is.

**Note:** Remember that the onscreen margin settings made from the rulers are *relative* settings, not absolute. The margin settings made from the Document-Wide menu determines the true margin from the paper's edge to where text starts. Ruler margins are *relative* to the Document-Wide settings. If you have not made any changes to the Document-Wide menu, Sprint uses its *default* margin settings of 1 inch.

But if you have made Document-Wide settings, the *first* ruler automatically conforms to them, regardless of how you set the ruler's margins. In other words, changing the

right or left margins of the first ruler is—in this case—for onscreen convenience only; your printout looks the same regardless. Subsequent rulers, however, are relative to the first (that is, they add to or subtract from the first ruler's margins) and therefore do affect the printed margins.

See Also — Font, Glossary, Indenting, Justification, Margins, Precise Settings Menu, Line Spacing, Tabs

# Run (Macro)

Refer to the Macros Menu entry for details.

# Save

Keystrokes — *Alt-F* (or *F10*, **F**ile), **S**ave

*Ctrl-F2*

*Ctrl-KS* (saves and resumes editing)

*Ctrl-KD* (saves and closes file)

*Ctrl-KX* (saves and quits Sprint)

*Alt-U* (or *F10*, **U**tilities), **P**otpourri, FileSaveAll (automatically saves all currently open files)

Function — Writes the current file or files to disk.

Although your file changes are automatically stored and saved in the swap (backup) file, they are *not* ordinarily saved to your named file on disk until you explicitly choose one of the **S**ave commands.

How To — To save a file to disk, press *Alt-F* and then choose **S**ave. Sprint writes your file to disk. The text you are saving overwrites the text in your file on disk.

If you don't want to overwrite your text on disk but would rather save your changes in another file, you can choose the **F**ile/**W**rite As command. Once you choose this command, Sprint displays the following prompt in the status line:

```
Write file as:
```

Enter the name of the file in which you want to store your text (for example `DRAFT.2`). **Write As** saves the current file to disk under the new name and leaves the original file (with the original name) unchanged.

If Sprint finds a file with the same name as the one you entered in the `Write file:` prompt, the status line displays the following prompt:

```
Overwrite existing file?
```

If you weren't aware that there was already a file by that name on disk, you can type `N` to cancel the command. You can then enter the **Write As** command again and enter a different name for your file.

If you want to overwrite the file (delete the existing text and replace it with the text of your current file), type `Y`. Sprint copies the current file text over the existing text.

**Note to two-floppy system users:** If you save your work by pressing *Ctrl-KX*, Sprint may have to prompt you to insert the correct Program Disk into Drive A.

Tips

When you choose **Print/Go**, **Merge/Go**, or **Print/Screen Preview**, the editor also saves your file to disk.

If you want to save a Sprint file in another format, choose **File/Translate**.

See Also

Close (File), File Menu, Write As, Revert to Saved, Translate

# Save (User Interface)

Refer to the User Interface Menu entry for details.

# Screen Menu

Keystrokes

*Alt-C* (or *F10*, **Customize**), **Screen**

*Alt-Z* (toggles screen codes)

*Ctrl-OD* (toggles screen codes)

Function

Displays the Screen menu.

**Note to two-floppy system users:** Sprint may have to prompt you to insert the correct Program Disk into Drive A in order to use commands from the Customize/ Screen menu.

The Screen menu displays commands to tell the editor to either display or hide a variety of *screen elements*:

**Paragraph Marks**
If On, Sprint displays a hard return character (which represents a new paragraph) as a left-pointing triangle.

**Tabs**
If On, Sprint displays a small, right-pointing triangle at the location of each editor or ASCII tab character inserted in your file. If Off (the default), you will only see what appears to be space characters between tabbed text.

**Indents**
If On, Sprint displays a small, right-pointing arrow wherever you chose to indent a region of text by pressing *Shift-Tab*. The default is Off.

**Spaces**
On means Sprint will display a small, raised dot wherever a space exists in your text (between words, for example). The default setting is Off, which means you'll see a blank space, not the small dot.

**Non-Breaking Spaces**
If On, Sprint displays an upward-pointing triangle for every non-breaking ("hard") space you insert in your document. Default is Off.

**Wide Spaces (Springs)**
On means Sprint displays a small double-headed arrow to indicate that text will be moved to the right margin (or between tab stops). If Wide Spaces is Off (the default), you will see the amount of blank space required to perform the Insert/Wide Space (Spring) command.

**Codes** *Alt-Z*
On means Sprint displays the control codes that make up Sprint commands. Normally, these codes are hidden.

Note that if Codes is On, you won't see your ruler lines at all; you will instead see the *codes* comprising the ruler. The shortcut for this command is *Alt-Z*.

**Bottom Status Line**
If On (the default), the editor displays a status (information) line at the bottom of the screen. This line displays the drive, directory, and name of the file you're editing, the mode(s) you're in (Column, Insert, or Overwrite), current time of day, the current line number, and the current column number. When you are in the process of selecting text, the letters "Sel" appear in the status line. When you're in column mode, the letters "Col" appear in the status line. The status line displays an asterisk (*) if you've modified but not saved your file. If you have paginated your file, the status line also shows the number of pages and the number of lines on the current page.

How To

To toggle the value of any Screen command, press *Enter* or type the highlighted letter. Once the command is set the way you want it, press *Shift-Esc* to end the command or press *Esc* to return to the Customize menu.

The following figure shows you what the screen elements look like for these commands both with screen codes on and off.

**Sprint's Special Screen Characters**

| | With Screen Codes Off | With Screen Codes On |
|---|---|---|
| Paragraph mark (hard return): | ◀ | ◀ |
| Space character: | · | · |
| Non-breaking space: | ▲ | ^\ |
| Wide space (spring): | ◆▶ | ^F |
| Tab: | ▶ | ^I |
| Indent mark: | → | ^G |

Figure 1.2: Sprint's Special Screen Characters

See Also Codes, Hard and Soft Returns, Options Menu, Ruler, Status Line, Tabs, Wide Spaces

# Screen Preview (SprintMerge)

Refer to the Merge entry for details.

# Screen Preview (Printing)

Refer to the Print Menu entry for details.

# Scrolling

Keystrokes

*Ctrl-W* (one line up)

*Ctrl-Z* (one line down)

*Shift-F7* (all windows one line up)

*Shift-F8* (all windows one line down)

*Ctrl-QW* (continuous up)

*Ctrl-QZ* (continuous down)

*PgUp* (or *Ctrl-R*) (one screen up)

*PgDn* (or *Ctrl-C*) (one screen down)

*Alt-U* (or *F10*, Utilities), Potpourri, ScrollDown

*Alt-U* (or *F10*, Utilities), Potpourri, ScrollLeft

*Alt-U* (or *F10*, Utilities), Potpourri, ScrollRight

*Alt-U* (or *F10*, Utilities), Potpourri, ScrollUp

Function

Scrolls the screen either up, down, right, or left.

Press *Ctrl-W* or *Ctrl-Z* to scroll your screen one line up or down. Pressing the *Up arrow* or *Down arrow* key also scrolls the screen one line at a time, but *only* once the cursor has reached the top or bottom of the screen.

To have the screen scroll continuously, press *Ctrl-QW* or *Ctrl-QZ*. You can change the rate of the scrolling by then pressing a number from 0 (fastest scroll) to 9 (slowest scroll). Pressing any other key stops the scrolling.

If you're using a mouse, you can scroll up by moving the pointer to the top two lines of the screen. If you move the mouse to the bottom two lines, Sprint will scroll down.

# Searching

Keystrokes

*Alt-E* (or *F10,* Edit), Find

*Alt-E* (or *F10,* Edit), Search-Replace

*F7* (search)

*F8* (search and replace)

*Ctrl-L* (finds next occurrence)

*Ctrl-QF* (forward search)

*Ctrl-QA* (forward replace)

Function

Allows you to search (and optionally replace) the text of your Sprint document.

The Edit menu has an important command used for searching: Search-Replace. The Search-Replace command displays another menu with several options.

From the Search-Replace menu, you can look forward or backward in your file to locate a particular string of text (called a *search string*) and also replace that text with something else (the *replacement string*). Search and replacement strings can contain from 1 to 70 characters and can include plain text, control characters (entered with the *Ctrl-Q* prefix), @-sign commands, and wildcard characters.

The editor preserves the case of the replaced text. For example, if you're searching for the word *further* and want to replace it with the word *additional,* and Sprint finds the word *Further,* it will replace it with *Additional.* The editor looks at the first two characters of the word it's replacing: If both characters are lowercase, the replacement text is lowercase; if the the first character is capitalized and the second character is lowercase, the replacement text begins with a capital letter, followed by lowercase letters; if the first two characters of the text you're replacing are uppercase, Sprint replaces the text with all capital letters.

The Search-Replace menu also provides options that let you determine whether the editor searches the entire file, searches from the top or from the bottom of the file,

searches forward or backward, pays attention to case (capitalization), searches for characters or words only, or accepts wildcard characters in your search string.

How To

When you want to quickly locate a string in your file, press *F7*. Sprint displays the prompt

```
Forward search:
```

This command uses the current settings in the Search-Replace menu, so if you've toggled Direction to Backward, the prompt is

```
Backward search:
```

The Search-Replace menu displays the following commands:

**Find** *F7*

Causes Sprint to prompt for the text you want to find. Once you enter the text, Sprint looks for the first string of text that matches the text you specified. When Sprint finds the text, it highlights the text and ends the command. If Sprint doesn't find the specified text, it displays a `Not found.` message and returns the cursor to where it was when you selected the command. You can abort the search by pressing *Ctrl-U* anytime after Sprint has started its search. Your cursor is returned to where you began the search.

**Next Occurrence** *Ctrl-L*

Tells Sprint to search for the next occurrence of the text you entered in your last Find command.

**Search & Replace** *F8*

Causes Sprint to search for and then replace text. When you enter this command, Sprint prompts you for the text you want to find and then asks you for the replacement text. When you have entered this information, Sprint starts searching forward (or backward depending on the direction you have set) from the current cursor position. You can abort to search by pressing *Ctrl-U*. When Sprint finds the text, it highlights it and displays the Replace this? menu of options:

| | |
|---|---|
| Yes | Tells Sprint to replace the text. |
| No | Tells Sprint to skip this one. |
| And the rest | Tells Sprint to search and replace all subsequent occurrences, without further prompting. You can also press *Esc* to cancel any further replacing. |

**Direction**
You can tell Sprint in which direction you wish your search to be performed with this selection. You can toggle between Forward and Backward by pressing *Enter* or by pressing *D* (the first letter of the selection). The default is Forward.

**Case Sensitive**
When set to No (the default), the editor ignores the case (capitalization) of text typed in response to the `Search for:` prompt. For example, if you search for *Text* the editor will find *text*, *Text*, *TEXT*, and *tExt*.

When set to Yes, the editor will only find text that is capitalized as specified in the search string.

**Match Words Only**
When set to No (the default), the editor searches for the text regardless of its location in the line. For example, if you search for *at*, the editor will find the *at* in loc*at*e and *at*tention, as well as the word *at*.

When set to Yes, the editor finds the text only if it is followed by a space character, a dash, period, or other non-alphanumeric character. It looks for the text as a *word*, not text included as *part* of a word. For example, if Match Words Only is Yes, and you search for *the*, the editor will find the word *the*, but it won't find *these* or *them*.

**Use Wildcards**
When set to Yes (the default), you can include *wildcard* characters in your search and replacement strings.

You can use the following wildcards when searching or replacing text in Sprint:

* Matches any set of characters (0 or more). This includes all letters, foreign characters, the

numbers 0 to 9, underscore, dollar sign, and the percent sign. For example, *let** will find words that start with the letters *let*, such as letter, lettuce, and lethal. Searching for *a*t* finds words that contain the letter *a* and the letter *t*, such as locate, character, and capitalized.

If you include an asterisk in both the search and replacement strings, the editor will place the text matching the search string's asterisk at the same location in the replacement string. For example, if you're editing a price list and select the command to search for **.49* and replace it with **.50*, the editor will search for any number ending with .49 and replace these three characters with .50.

? Matches any *single printing* character; the editor matches all characters except space, tab, and soft and hard return characters. For example, *d?g* finds dig and dog, but won't find ding.

You can use a question mark to find unknown letters. For instance, if you search for *a?*, the editor will search for all two-letter (minimum) combinations that start with an *a*. If you want to search for the first character of a paragraph, but you don't know what the character is, you can search for *^J?* (entered as *Ctrl-Q Ctrl-J?*).

If you type a question mark in your replacement string, and your search string also contains a question mark, the editor inserts the letter matched by the question mark at that location in the replacement. For example, replacing *1?0* with *2?0* will change 100, 110, 120, 130, and so on to 200, 210, 220, and so on.

*<space character>*

Except when typed at the beginning of a word or phrase, this wildcard matches any *nonprinting* character (space, tab, carriage returns, and the end of a file). This is the opposite of using the *?* wildcard. For example, if you search for a space, the editor will find the space character, a tab, a carriage return, or the end of the file (whichever comes first). To search for a space character only,

use the wildcard \ (backslash) followed by a space.

The *space* wildcard can be used to locate sequential words, even if they're on separate lines. For example, the search string *alpha beta* will search for the words *alpha* and *beta*, regardless of whether they're separated by a space, a hard return, or a soft return character.

If you type a space in a replacement string, and you've typed one in the search string, the editor will insert the matched space character.

[*set of characters*]

Lets you specify a set of characters that can be matched. The editor searches for any characters between the brackets. You can specify a range of characters in the set by separating two characters with a dash. For example, *[0-9]* searches for any number 0 through 9; *[a-d]2* searches for any *a*, *b*, *c*, or *d* followed by the number 2. To identify specific characters in the set, separate each character with a comma. For example *d[o,u]g* finds *dog* and *dug*, but won't find *dig*; *415-[9,5,2]* finds all words that begin with the text 415- and are followed by either the number 9, 5, or 2 (for example, 415-978-7158, 415-559-8700, and so on).

If you add a plus symbol (+) to the set wildcard, the editor will match any sequence (1 or more) of characters in the set. It functions as a "repeating" set. For example, *A[0-9]+B* will search for *A* followed by any number, but ending in a *B* (like A123B, A9B, A4B). You can also use [*set*] in replacement text.

If you place a caret (^) as the first character of the set, Sprint will match all characters *except* those in the set. For example, *[^a-Z]* finds all characters that *aren't* letters (like whitespace and punctuation). To search for a caret itself, don't use it as the first character in the brackets.

Matches the character exactly, no matter what it is. You can use the backslash wildcard to specify

characters that would normally be used as a wildcard. For example, * searches for an asterisk.

You can use this wildcard to search for a space character. Type the backslash and press the spacebar; the editor will find a space character between words, but unlike the <space> wildcard, it will not find a tab, newline, or end of file.

**Entire File**
When set to No (the default), the Search (Forward), Search (Backward), and Replace commands function as described in this section.

When set to Yes, the editor searches the entire file for the text specified in your search string. The search begins at the current cursor position and continues in the direction specified by the Direction command. Instead of ending at either the top or bottom of the file, however, the editor continues searching until it reaches the point from which you entered the Search command. For example, if the cursor is in the middle of your file, and you enter a Forward Search command, the editor will search forward for the specified text. If it doesn't find the text by the time the cursor reaches the end of the file, the editor continues from the top of the file toward the middle of the file.

Tips

Whenever possible, avoid starting a search string with a wildcard character. The search will be much slower than if the search string starts with a normal character.

If Use Wildcards is set to No, the editor will not recognize the asterisk, question mark, left bracket, or backslash as wildcard characters. Instead, it will treat these characters as normal text to be located or replaced.

With Case sensitive set to No, as long as you don't include an asterisk (*), question mark (?), left square bracket ([), or backslash (\) in your search string, Sprint will search for (and replace, if you select this command) the characters exactly as you typed them in the status line. Uppercase letters, punctuation marks, and foreign characters are matched exactly. Lowercase letters match both the lowercase and uppercase versions of the string.

To enter control characters into your search or replacement text, you should press *Ctrl-Q* and then the desired control character. The character will appear in the text with a caret, like ^B or ^N.

The following table lists the control characters you need to search for to find certain special characters.

| To find: | Search for: |
|---|---|
| Hard return (paragraph mark) | ^J |
| Soft return | ^_ |
| Tab | ^I |
| Hard (non-breaking) space | ^\ |
| Soft space | \<space> |
| Discretionary hyphen | ^^ |

See also Table 1.1 on page 46 for a full list of Sprint control codes.

See Also Control Characters

# Search & Replace

Refer to the Searching entry for details.

# Search-Replace Menu

Refer to the Searching entry for details.

# Section

Keystrokes *Alt-S* (or *F10*, Style), Headings, Section

Function Prints a numbered section heading.

This command prints a large, bold, numbered heading, flush with the left margin, to start a new section of a document and makes an entry in the table of contents. Sprint leaves two blank lines above and below the section heading.

Section is a second-level command (after Chapter, and before Subsection and Paragraph.) If you use Section

after you have used the Chapter command, you get a two-level numbering scheme such as 3.3. If you don't use Chapter, you get a single-level number like 3, and Section becomes the top level of sectioning in the document. You may wish to do this with shorter documents where you do not want each division to start a new page the way Chapter does.

You will find this command defined in the Sprint file STANDARD.FMT. Refer to the Modifying Formats entry in this menu encyclopedia for details.

How To

There are two ways to create a new, numbered section from the **H**eadings menu. You can type the title of the section, mark it, and then choose **S**ection, or you can choose **S**ection and then type the section title. For example,

```
SECTION Word Processing Word processing is a tool for
the rapid production of high-quality documents. It
offers several major advantages over a typewriter.
```

results in:

## 3.1 Word Processing

Word processing is a tool for the rapid production of high-quality documents. It offers several major advantages over a typewriter.

This example is based on a file that has three **C**hapter commands preceding this **S**ection command. If the file didn't have any **C**hapter commands in it, and this was the first **S**ection command in the file, the section title would print like this:

## 1 Word Processing

Tips

If you don't want a section numbered, use the **S**ection command. This formats the title just as the **S**ection command does, but eliminates the number and does not make an entry in the table of contents. If you do want the section title to appear in the table of contents, but don't want the section title numbered, use the **H**eadingB

command and make sure your file contains the MakeTOC command.

See Also Chapter, Headings Menu, HeadingA, HeadingB, and Chapter 2 (MakeTOC)

## Section (Variable)

Refer to the Variables entry for details.

## SectionNumber (Variable)

Refer to the Variables entry for details.

## SectionTitle (Variable)

Refer to the Variables entry for details.

## See

Refer to the Index entry for details.

## Sentence (Selecting)

Refer to the Block Select Menu entry for details.

## Set (Mark)

Refer to the Place Mark entry for details.

## Shut All

Refer to the Window entry for details.

## Simple Line

Refer to the Line Drawing entry for details.

## Size

Refer to the Variables entry for details.

## SK Plus Outlook

Refer to the Translate entry for details.

## Snaking Columns

Refer to the Columns entry for details.

## SourceFile (Variable)

Refer to the Variables entry for details.

## Space Allowable

Refer to the Hyphenation Menu entry for details.

## Spaces (Horizontal)

Refer to the Non-Breaking Space entry for details.

## Spaces (Screen)

Refer to the Screen entry for details.

# Special Hyphen

Keystrokes

*Alt-I* (or *F10,* Insert), Special Hyphen

*Alt-U* (or *F10,* Utilities), Hyphenation, Word

*Ctrl-O E*

*Ctrl-*

Function

Specifies a discretionary (conditional) hyphen.

**Note to two-floppy system users:** Sprint may have to prompt you to insert the correct disk into Drive A in order to use the Utilities/Hyphenation commands.

A discretionary hyphen prints as a hyphen *only* when the word falls at the end of a line and when Sprint needs to break the word for good spacing.

Use the Special Hyphen command to specify where the formatter can split a polysyllabic word over two lines during formatting. The formatter will ignore any hyphenation point(s) it doesn't need to justify a line.

You won't see the hyphen onscreen, but if you press *Alt-Z* or select the Customize/Screen/Codes command, you'll see two carets (^^) where you inserted the command. (Two-floppy system users: Sprint may have to prompt you to insert the correct Program Disk into Drive A in order to show codes.) This is the discretionary (conditional) or "soft" hyphen. If the formatter needs to break the word, it will break it at the point you specified.

A soft hyphen eliminates the large amounts of whitespace between words by letting the formatter put as much of the lengthy word on one line as possible.

How To

Choose Special Hyphen from the Insert menu and hyphenate your word accordingly.

Here's an example where we've placed a discretionary hyphen in the word "antidisestablishmentarianism":

> This is what happens when the super-long word antidises-tablishmentarianism falls at the end of a line, and has been conditionally hyphenated.

You can also let Sprint decide where the hyphenation points should be by choosing the **Hyphenation/Word** command. When you choose this command, Sprint displays some suggested hyphenation points. Select the one that looks best, and Sprint inserts the discretionary hyphen for you.

Tips

If your document has a longish word that's used a lot, you might want to globally search for the word and replace it with the same word with discretionary hyphens inserted in it. This can improve Sprint's ability to make good-looking line breaks when it formats your text.

Sprint also treats a hyphen within a word as if it contains a soft hyphen; if necessary, Sprint will split the word over two lines. If you do not want this to happen (as in the word *co-op*), select the word, choose **Style/Other Format**, and type the formatter command Word. For example,

```
WORD co-op
```

See Also

Hyphenation Menu, Justification, and Chapter 2 (Word)

# Spelling Menu

Keystrokes

*Alt-U* (or *F10*, **Utilities**), **Spelling**

*Shift-F1*

Function

Displays the **Spelling** menu.

This command displays the **Spelling** menu, which lists the following commands:

**Word**
Checks spelling of the current word.

**Block**
Checks spelling of all words contained in a selected block of text.

**File**
Checks the spelling of every word in your file, starting at the top and continuing to the bottom.

**Rest of File**
Checks the spelling of every word in your file. When your cursor reaches the bottom of your file, Sprint asks if you want to check spelling from the top. The check stops when the cursor returns to where it started.

**Last Bad Word**
If AutoSpell is On, moves the cursor to the last word recorded as misspelled and displays a list of spelling options.

**Every Bad Word** *Ctrl-F1*
If AutoSpell is On, moves the cursor backward to the last word recorded as misspelled and gives you a choice to correct it or leave it as is. Then the cursor moves backward to the next suspect word.

**Autospell**
Turns the automatic spelling checker on and off. Default is Off.

**Main Dictionary**
Displays a list of files with the .LEX extension. (LEX files are specially encoded dictionary files.) You can select a different dictionary from this list (to switch from an American word list to a British one).

**User Dictionary**
Displays a list of files with the .DIC extension. You can select from this list which supplementary ("user") word list you want loaded.

You can create a supplementary word list using Sprint. The words must be in strict ASCII order (which is *not* the same as alphabetical order!) and capitalization is taken literally. Refer to Appendix B for a list of ASCII characters. Then save the file with the .DIC extension.

How To

Choose Spelling from the Utilities menu.

**Note to two-floppy system users:** Sprint may have to prompt you to insert the correct Program Disk into Drive A in order to use Spelling commands.

If you turn Autospell On, Sprint loads its dictionary and begins checking the accuracy of each word you type *as* you type. It also gives you access to the Last Bad Word and Every Bad Word commands. As long as AutoSpell is

On, Sprint records each word that it considers misspelled (a misspelled word in Sprint is a word that is not in the Sprint dictionary or in the user dictionary).

Regardless of whether **A**utoSpell is on or off, you can always choose the **W**ord, **B**lock, and **F**ile commands on the **S**pelling menu. When Sprint comes across a word that's not in its dictionary, it displays the following list of spelling options:

| | |
|---|---|
| **S**kip Once | Lets you skip the word this time, but Sprint continues to check for it in the rest of the session. |
| **I**gnore | Ignores the word for the entire correcting session and goes on checking for spelling errors. (To be precise, words stay ignored until quit, paginate, print, preview, hyphenate, or load the thesaurus.) |
| **A**dd to Dictionary | Inserts the current word in the user dictionary so that it won't be considered a misspelled word. |
| **L**ookup | Displays a list of similarly spelled words. If the word you need is listed, select it and Sprint will automatically replace the misspelled word with the selected word. If the correct spelling of the word is not listed, select **R**eplace with and enter the correct spelling. |
| **R**eplace With | Prompts you for the correct spelling of the word. Enter your response in the status line. Sprint also checks the spelling of the replacement words and offers replacements if it deems the replacement is misspelled. |

You can press *Esc* at any time to stop the spelling check. The cursor remains at the spot you left off, not where you started the check.

Tips

Don't worry about the unusual words that Sprint itself puts in your documents. The spell checker will not flag

these words (such as ENDF, FOOTERO, and TCAPTION) as misspelled.

## Starting Page

Refer to the Print Menu entry for details.

## Status Line

Keystrokes — *Alt-C* (or *F10*, Customize), Screen, Bottom Status Line

Function — The status line informs the user of the current file name, line number, time, and mode. It also doubles as the "prompter," where Sprint gives messages and asks for input. You can choose to hide the status line (thereby letting you see an extra line of your document text) by setting Screen/Bottom Status Line to Off.

**Note to two-floppy system users:** Sprint may have to prompt you to insert the correct Program Disk into Drive A in order to use this command.

The status line changes somewhat, depending on whether you've paginated your file. Here are the possible elements of a status line, going from left to right:

| | |
|---|---|
| `FILENAME` | The name of your document, complete with path name. |
| * | An asterisk indicating that the file contains unsaved changes. |
| `Ins`<br>`Ovr` | Indicates whether you're currently in Insert or Overwrite mode. |
| `Sel` | Indicates whether you're in Select mode (like when you've pressed *F3*). |
| `Col` | Indicates whether you're in Column mode. |
| `time am/pm` | The current time (according to your computer's clock). |
| `Ln.x of y` | Displays the current line and total number of lines in the document (this is present *only* if you haven't paginated). |
| `Pg.x Ln.y` | Displays current page and line (shows *only* if you've previously paginated). |
| `Col.x` | The current column. |

See Also Screen, Insert Mode, Block Select Menu, Paginate

# Strikethrough

Refer to the Typestyle Menu entry for details.

# Style Menu

Keystrokes *Alt-S* (or *F10,* Style)

Function Displays the Style menu.

This menu displays a list of commands that help organize a document and enter formats.

**Center**
If no text is selected, centers the line the cursor is on between the left and right margins. If a block of text is

selected, inserts commands before and after the block to center each line of the selected text.

**Modify**
Moves the cursor to the last format command and displays a menu that lets you choose whether to modify This format or Previous format.

**Headings**
Displays a menu of formats to create different types of numbered and unnumbered heads in your documents.

**Lists**
Displays a menu of commands for creating numbered, unnumbered, and two-column descriptive lists in your documents.

**Table**
Tells Sprint to insert a table at the current cursor position. Sprint inserts the BEGIN and END TABLE commands and prompts for a caption to print below the table.

**Figure**
Tells Sprint to insert a figure at the current cursor position. Sprint inserts the BEGIN and END FIGURE commands and prompts for a caption that Sprint ordinarily prints below the figure.

**Graphics**
Displays a menu of drawing commands: EPS Picture, KeyCaps, Bar, and Draw Box. These commands are recommended only if you've a PostScript printer.

**Index**
Displays a menu of commands that let you create an index of your Sprint document at print-time.

**References**
Displays a menu of commands for creating types of notes in your documents.

**X-Reference**
Displays a menu of commands used to define and tag cross-references in your Sprint documents.

**Other Format**
Lets you type in the format commands not included by Sprint's menus or those you create. These are primarily lesser-used, advanced commands. (See Chapter 2 on Other Format in this manual for a detailed list of these commands.)

See Also — Refer to the individual entries on these commands for details.

# Style Sheet

Refer to the Document-Wide Menu entry for details.

# Subscript

Refer to the Typestyle Menu entry for details.

# Subsection

Keystrokes — *Alt-S* (or *F10*, Style), **H**eadings, **S**ubsection

Function — Creates and prints a numbered subsection heading.

This command prints a bold, numbered heading, flush with the left margin, and makes an entry in the table of contents. The **S**ubsection command is the next level of section heading after **S**ection. If you've selected a **S**ection command before the **S**ubsection command, Sprint assigns a two-level number (for example, 2.1) to the subsection. If you also enter a **C**hapter command before it, Sprint assigns a three-level number (for example, 2.1.1) to the subsection. If you don't enter any of these higher-level sectioning commands before your **S**ubsection command, Sprint assigns a single-level number (for example, 1) to the subsection.

How To — There are two ways to create a numbered section from the **H**eadings menu. You can type the title of the section, mark it, and then choose **S**ubsection, or you can choose **S**ubsection and then type the section title. For example,

```
SUBSECTION Parasites
```

prints as:

### 1.1 Parasites

See Also HeadingC

## SubSection (Variable)

Refer to the Variables entry for details.

## SubSectionTitle (Variable)

Refer to the Variables entry for details.

## Superscript

Refer to the Typestyle Menu entry for details.

## Tab Expansion

Keystrokes *Alt-C* (or *F10*, Customize), ASCII File Handling, Tab Expansion

Function Specifies the distance between ASCII tab stops.

This command specifies how far apart the ASCII tab stops should be. The default is 5 (every five columns).

**Note to two-floppy system users:** Sprint may have to prompt you to insert the correct Program Disk into Drive A in order to use this command.

After choosing this command, type the desired tab size (tabs set every *n* columns). If you're editing an ASCII file with no ruler line, and you press *Tab*, the cursor moves to the nearest ASCII tab (default is column 5, 10, 20, 25, ...). If the file contains a ruler line, and you press *Tab*, the cursor moves to the nearest ruler line tab (not the

column specified with Tab Expansion), so the screen display will not reflect the actual printed output.

Tips

This command is typically used in conjunction with program files or pure ASCII files, not files with rulers. If you want to set specific tab stops in a file that has a ruler, at columns 20 and 45 for example, move the cursor to the appropriate column on the ruler line and type `T`.

See Also

Tabs

# Table

Keystrokes

*Alt-S* (or *F10*, Style), Table

Function

Creates a Table format for text.

Use this command when you want to format text as a table. It also automatically "groups" the text, which means that Sprint will keep the text of the table together on a single page. If the text doesn't fit on the current page, Sprint starts the table on the following page. If the table contains more than one page of text, Sprint displays a `Warning: Group too big for page` message when it formats your file, but will let you print the file anyway.

Sprint retains any leading spaces that you enter in a table. This means that it will not add to or delete any space characters appearing at the beginning of a line in your table. Leading spaces in a table are equivalent to entering Non-Breaking Space commands.

You will find this command defined in the Sprint file STANDARD.FMT.

How To

If you've already typed the text of your table, select the text and then choose Style/Table from the main menu. When Sprint prompts for the caption title, enter the caption and press *Enter*. If you haven't yet entered the text of your table, Sprint places the BEGIN TABLE command with your caption immediately preceding the END TABLE command. Your cursor remains on the blank line between the BEGIN TABLE and TCAPTION commands, ready to input your table text.

If you want to add a caption at the *top* of your table, you can type in your table *after* the TCAPTION command that Sprint inserts, instead of before it, as is usual. Delete the blank line Between the BEGIN TABLE and TCAPTION commands. Move your cursor to the end of the TCAPTION line, press *Enter*, and type your table text.

If you need a special set of tabs for your table, insert a new ruler after the BEGIN TABLE command and set the tabs by either editing the ruler and typing `T` whenever you want a tab set or by choosing **Layout/Ruler/Precise Settings/Tab Stops**.

The following example shows you how to create a table. Every time *Tab* appears in the table, you should press the *Tab* key.

```
BEGIN TABLE
@TabSet(7.5 picas)

Command Name TabFunction

Table TabCreates a table format.

TCaption TabCreates a table caption and enters the
caption in the table of contents.

TCapt TabCreates a table caption but does not enter
the caption in the table of contents.
TCAPTION Sprint Table Commands
END TABLE
```

Here's the result:

**Table:** Sprint Table Commands

| Command Name | Function |
|---|---|
| Table | Creates a table format. |
| TCaption | Creates a table caption and enters the caption in the table of contents. |
| TCapt | Creates a table caption but does not enter the caption in the table of contents. |

There are several ways to create blank space between columns:

1. Use a ruler with a "hanging indent."

   If you set the indent marker (the *I* on the ruler) to the *left* of the left margin marker (the [ on the ruler), the first line of each paragraph will "hang" off to the left of the subsequent lines. Be sure to insert a "normal" ruler after the table.

2. Use a Description format command.

   If you type your table within a Description format, the formatter will print the text in the left column in bold type, move over one-fourth of your line length, and begin the second column of text in plain type. If you want to change the gap between the two columns, you can modify the default *indent* parameter defined by the Description format. As in the preceding example, use the a *Tab* between the text of the two columns.

3. Use the **Wide Space** command.

   If you want to align the second column at the right margin, you could type the text of the first column, then choose **Wide Space (Spring)** from the **Insert** menu, and then type the text of the second column.

Tips

If you want to cross-reference a table, you can use the **Define a Tag** command, type a tag name for your table, and choose **Reference a Tag** when you want to reference this table.

If your table prints on more than one page, you can use the formatter commands PageFoot or HaveSpace to print a message at the bottom of the first page of your table, such as "Continued" or "Continued on the following page."

See Also

Caption, Figure, Tags, TCapt and TCaption, Wide Space (Spring)

# Table (Variable)

Refer to the Variables entry for details.

# Tabs (General Information)

Keystrokes

*Alt-L* (or *F10,* Layout), **Ruler,** Precise Settings, Tab Stops

*Alt-U* (or *F10,* Utilities), Potpourri, CenterTab (Inserts a tab, centered between the left and right margins, and moves text at the current cursor position to this tab stop)

*Alt-U* (or *F10,* Utilities), Potpourri, DecimalTab (moves to the next tab stop and treats it as a decimal tab)

*Alt-L* (or *F10,* Layout), **Ruler,** Edit on Screen, and type `T` on ruler line

*Alt-L* (or *F10,* Layout), **Ruler,** Insert, and type `T` on ruler line

*Alt-C* (or *F10,* Customize), ASCII File Handling, Tab Expansion

*Alt-S* (or *F10,* Style), Other Format, and type `TabDivide` *n*

*Alt-A* (edits onscreen ruler)

*Alt-R* (inserts ruler)

Function

Lists tab-related commands.

There are a variety of tab-related commands that you can use with Sprint:

| | |
|---|---|
| Tab Stops | Allows you to set tabs on the current ruler line, using any horizontal unit of measurement. |
| Tab Expansion | Specifies the distance between ASCII tabs. |
| TabDivide *n* | Creates *n* evenly divided columns. |

Precise Settings/Tab Stops affect *ruler* tabs—the tabs set on the current ruler line. The only difference between this kind of tab and the kind you set by choosing **Ruler/** Edit on Screen or by inserting a ruler and typing a `T` as needed is that Precise Tabs can be defined in any horizontal dimension (points, ems, and so on), while the tabs you set by typing `T` are measured in characters (onscreen columns) only. As a result, you will not see your "precise tabs" at work until you print your job, but the onscreen tabs affect the document immediately.

When you want to print text at a ruler tab stop, press *Tab* and then type the text.

The Tab Expansion command on the ASCII File Handling menu affects ASCII tabs only. These tabs are generally used in ASCII files (typically computer programs or files you want to transfer with electronic mail packages). The **Tab Expansion** command specifies the number of characters between ASCII tab stops. For example, if you type `10`, the editor sets ASCII tab stops every 10 columns. These tabs are not set on a ruler line—they're used in files that don't contain rulers.

**Note to two-floppy system users:** Sprint may have to prompt you to insert the correct Program Disk into Drive A in order to use the **Tab Expansion** command.

The DecimalTab command moves your cursor to the next tab stop and allows to use it as a *decimal* tab. When your cursor is at the decimal tab stop, Sprint treats new numbers or characters you type as belonging to the left of the decimal place. In other words, the characters are inserted to the *left* of the cursor position. When you type a period (a decimal place), Sprint prints it at the tab stop. Any characters typed after you entered the period are inserted to the *right*, as usual. Note that you must type a period (or press *Esc*) to end the DecimalTab command.

The advantage of using decimal tabs is that you can easily line up columns of numbers according to their decimal points. This is the preferred format for any list of numbers that are being added or compared.

How To — For examples of how to use most of the tab commands explained here, refer to its entry in this menu encyclopedia.

See Also — ASCII Files, Tab Expansion, Tab Stops, Ruler, and Chapter 2 (TabDivide)

# Tabs (Screen)

Refer to the Screen entry for details.

# Tab Stops

Keystrokes — *Alt-L* (or *F10*, Layout), **Ruler**, **Precise Settings**, **Tab Stops**

Function — Allows you to set tabs on the current ruler line in terms of any horizontal measurement.

This command lets you set and clear tabs on the current ruler line. It is the equivalent of choosing **Layout/Ruler/Edit** on Screen and typing `T` to set a tab stop or clear an existing tab stop. The only difference is the **Tab Stops** can use any measurement (points, ems, and so on), while the other kind are in terms of characters (columns) only. As a result, **Tab Stops** do not show up on your screen; you see their effect only when you print.

When the editor displays the prompt for the character position at which you want to set a tab stop, it assumes the current cursor position as the default column.

How To — To set or clear a tab stop, press *Alt-L* and then choose **Ruler/Precise Settings/Tab Stops**. Sprint prompts you:

```
Place tab at:
```

Type in your desired tab stops and press *Enter*. For example,

```
Place tabs at: 10 points, 20 points, 35 points
```

Sprint inserts your precise tabs in the form of the TabSet command placed just below the ruler line. You can change this command by cursoring onto it and editing it.

The tabs will be active until you insert a new ruler with different settings or until you enter a command like TabDivide.

See Also — Tabs and Chapter 2 (TabDivide)

# Tags

Keystrokes — *Alt-S* (or *F10*, **Style**), **X-Reference**, **Define a Tag**

*Alt-S* (or *F10*, **Style**), **X-Reference**, **Reference a Tag**

Function — Lets you make cross-references in your text.

Sprint's cross-reference commands let you avoid *hard references* in your documents. By "hard" we mean entering an actual chapter, figure, or section number. For example, if you enter "For more information, see Chapter 2", that's a hard reference. If you use hard references throughout your document and then rearrange your document so that Chapter 2 becomes Chapter 5, and Chapter 5 becomes Chapter 3, and Chapter 1 becomes Chapter 2, and so on, you have to go back through your document, find all the erroneous hard references, and change them.

Sprint's cross-reference commands let you create *soft* references. These references let you tag (mark with a **D**efine a Tag command) text that you want to refer to, without having to know the number Sprint will assign to the text when it formats and prints your file. Using tags and then referring to these tags by name (with the **R**eference a Tag command) means that no matter how often you change a document's organization, your cross-references will always be accurate.

How To

To display the Cross-Reference menu, choose **S**tyle/**X**-Reference and then choose one of the following commands:

**D**efine a Tag prompts you for a name to give the new tag. **R**eference a Tag lets you cross-reference a tag that you've set in your file. For example, at the `Name for new tag:` prompt, you might type

```
MarkTwain
```

to mark the spot where Mark Twain is quoted. When you choose the **R**eference a Tag command in your paragraph and type `MarkTwain` and then choose **P**age number to reference it, the formatter will automatically replace this reference with the actual page number.

You can also reference the running section numbers of tags you've set. In those cases, you should explicitly tell Sprint what kind of number you'll be needing. You do this by equating the tag name to one of Sprint's variables, like *Figure*, *Table*, or *Chapter*. So if "Mark Twain" were a tag to a chapter number, you would have typed

```
MarkTwain = Chapter
```

Now when you reference the tag called *MarkTwain*, Sprint substitutes the correct chapter number instead of the page number.

You can use any of the variables listed in the Variables entry of this menu encyclopedia (for example, *Chapter, Section, Figure, Table, Month, Day, Year,* and so on), as long as the variable stands for a *number*. You can also use a variable you defined yourself using the **I**nsert/**V**ariable command—as long as it stands for a number.

Once you choose the **R**eference a Tag command and enter the name of the tag you want to reference, Sprint displays the **R**eference By menu. The **A**ssigned Number command tells the formatter to print the *number* of the tagged item (like the figure number, chapter number, section number, and so on). If you had tagged the item as equalling one of the Sprint variables (*tagname=chapter*, for example), Sprint will substitute the correct number (the chapter number, for example). If you did not explicitly equate the tag name to a variable, Sprint prints the value of the variable called *Counter* when you choose **R**eference By/**A**ssigned Number.

*Counter* keeps track of a variety of levels in a particular format and might not be the number you wanted. When in doubt, always explicitly equate the tag name to exactly the type of number you want printed. To print page numbers, the tag name needn't be equaled to anything.

The **P**age Number command tells the formatter to replace the command with the page number on which the tagged text appears. For example, let's say your file contains a figure that you want to reference in a paragraph following it:

```
BEGIN FIGURE
RESERVE 3 INCHES
CAPTION Radio Frequency SignalsTAG radio=figure
TAG radion
END FIGURE

As shown in Figure radio on page PAGEREF radion, the
system is adversely affected by radio frequency
signals. When you shield the system from these signals,
performance increases dramatically.
```

The command lines `TAG radio=figure` and `TAG radion` (for the page reference) are the result of the **Define a Tag** command. The text beneath the figure entry refers to these tags using the **Reference a Tag** command. In this example, we choose both the **Reference By/Section Number** and **Reference By/Page Number**, respectively.

The example prints like this:

**Figure 1.1: Radio Frequency Signals**

As shown in Figure 1.1 on page 238, the system is adversely affected by radio frequency signals. When you shield the system from these signals, performance increases dramatically.

See Also — Page Numbers, Variables

# TCapt and TCaption

Keystrokes — *Alt-S* (or *F10*, Style), **Table**

*Alt-S* (or *F10*, Style), **Other Format**, and type `TCapt Text`

Function — Numbers and creates a title for the current table.

These commands work in conjunction with the Table format. When you enter a Table command in your file, Sprint prompts you for a caption, which it inserts as a TCaption command. When printing your document, the formatter determines the current value of Table and then automatically prints the word *Table*, followed by the correct table number and caption. Moreover, Sprint generates a special "table of tables" entry (comparable to a table of contents page) when it prints the file. The TCapt command, however, does *not* create the table of

tables entry. In all other respects, though, it is the same as TCaption.

These commands are defined in the Sprint file STANDARD.FMT.

How To

Choose Style/Table and, when prompted, enter the text of your caption. Sprint inserts the TCAPTION command automatically. Move the command to wherever you'd like the table number and title (caption) to print. If you want the table number and caption to print at the top of the table, move the TCAPTION command above the table text. If you want the table number and caption to print at the end of the table, move the TCAPTION command *below* the table text. (This is actually the default setup.)

Tips

If you want to change the TCaption command to a TCapt command, cursor to the command name and delete the last three letters (the *ion*).

To make sure the formatter keeps the table text and the caption together on the same page, type the TCapt or TCaption command inside the Table format.

If your file is set up such that your page numbers include a chapter or section number (for example, page 3-3, 4-1, and so on), you can specify that your table numbers also follow this format (for example, Table 1-1, Table 1-2, Table 2-1, and so on). Add the following definition anywhere in your .FMT file:

```
@Parent[table=chapter]
```

(STANDARD.FMT already is set up with this command in it.)

**Remember:** Use TCapt if you want to avoid printing a Table of Tables entry; otherwise, use TCaption.

See Also

Tags

# Template for Data

Refer to the Insert Menu and Merge entries for details.

# Thesaurus

Keystrokes

*Alt-U* (or *F10*, Utilities), Thesaurus

*Alt-F1*

Function

Loads the Sprint thesaurus word list.

This command loads the thesaurus word list and either displays a list of synonyms for you to choose from or, if your word is not found, gives you the message `Not in thesaurus`.

**Note to two-floppy system users:** Sprint may have to prompt you to insert the correct disk into Drive A in order to use this command.

How To

Place the cursor next to or within the word you would like a synonym for. Press *Alt-F1*, and Sprint displays an extensive list of words with similar meanings. If the word can be several parts of speech (noun, adjective, and so on), the list is divided in this way. The list is also usually divided by primary synonyms and then broken down further. Use *PgUp* and *PgDn* and the arrow keys to see more of the list.

When you find a good word, select it and press *Enter*. Sprint automatically substitutes the chosen word for the word originally in your text. Sprint usually then displays another list of words—this time of synonyms for the first-chosen word. You can continue substituting words in this way indefinitely. You can also choose either **Previous Word** to go back a level or **Original Word** to go back to the word you started from. Press *Esc* when you're happy with your selection.

# This Format

Refer to the Modifying Formats entry for details.

# Title Page (Footer/Header)

Refer to the Footer Menu or Header Menu entry for details.

# Title Page

Keystrokes — *Alt-L* (or *F10*, Layout), Title Page

Function — Centers all text vertically on the first page.

How To — After choosing the command, Sprint inserts the CENTERPAGE command at the top of your document and puts a page break right after it. Your cursor remains between the command and the page break for you to type in some text. For example,

```
CENTERPAGE .5 page
```

The center point is one-half (.5) the page length, measured from the top margin. The text will print in the middle of the page. (Page length is defined in the STANDARD.FMT file; it is usually 11 inches.)

Tips — The **Title Page** command does *not* center text horizontally. So if you want your text centered between the left and right margins *and* centered vertically on the page, put your cursor on the line of text and choose **Style/Center**. For example,

```
CENTERPAGE .5 PAGE
^FSprint: The Professional Word Processor^F
```

**Title Page** is also convenient for short, one-page business letters. All the text will be centered with equal blank space above and below it.

**Note:** If you have a header or footer on the page, it prints normally, not centered.

See Also — Centering Text, Justification

# Tone

Keystrokes — *Alt-C* (or *F10*, Customize), **Options, Tone**

Function — Lets you change the sound of Sprint's beep.

This command displays a list of options that let you adjust Sprint's warning tone. Sprint generates this tome if it can't find the string you specified as part of a Search-Replace command, if you type a word that's not

in Sprint's dictionary and Autospell is On, and so forth. The commands include Pitch and Length. This menu also shows which arrow key to use to affect the change; that is, *Down arrow* lowers the pitch.

How To

To change the tone, press *Alt-C* and choose Options/ Tone.

You can then change the tone by using the arrow keys to alter the pitch and length. Press *Esc* when you are satisfied with the sound.

You can change the pitch from 20 to 2000Hz; the length from 10 to 400 milliseconds.

**Note to two-floppy system users:** Sprint may have to prompt you to insert the correct Program Disk into Drive A in order to use this command.

# Top Margin

Refer to the Document-Wide Menu entry for details.

# Translate

Keystrokes

*Alt-F* (or *F10,* File), Translate

Function

Displays commands to import and export files between word processing program formats.

The Translate menu displays a menu of commands to import (read in) a file that is currently not a Sprint file and export (write to) a Sprint file to a foreign format.

**Note to two-floppy system users:** Sprint may have to prompt you to insert the correct Program Disk into Drive A before you can translate files.

How To

Choosing the Import command lets you convert files in the following formats into Sprint:

ASCII file
DisplayWrite 4 (DCA RFT)
MS Word
MultiMate
MultiMate Advantage
Wang (IWP)
WordPerfect 4.2
WordStar
WordStar 2000
SideKick Plus (Outlook)

Choosing the **Export** command from the **Translate** menu lets you save your Sprint file in any of the same formats, except SideKick Plus. Sprint prompts you for the original file name (the file to be imported or exported) and then the new file name (the file after it is imported or exported).

Choosing SideKick Plus from the **Import** menu displays this prompt on the status line:

```
Press (T) for tab indent, (O) for outline, or ESC to
 cancel:
```

If you press *O*, your Outlook outline will converted into a Sprint document formatted as an **Outline** list (that is, the **Style/Lists/Outline** command). If you press *T*, your Outlook outline will be converted into a Sprint documented that uses tabs to indent your topics and subtopics.

**Note:** Even though Sprint uses control codes for specialized formatting, the files it creates are still ASCII (that is, any other word processor or editor can open them up directly and read them). Similarly, Sprint can open any ASCII file *directly* (without using the **Import** command).

You only need the **Import** and **Export** commands when dealing with ASCII files, and you need to strip out or insert single hard return characters at the end of every line.

Tips — Sprint *does not* retain a copy of the file in ASCII format. If you want to keep a copy of the ASCII file, copy the file before you import it.

See Also — ASCII Files, Outline, Write As, Save

# Turn Select Mode

Refer to the Block Select Menu entry for details.

# Typestyle Menu

Keystrokes — *Alt-T* (or *F10*, Typestyle)

*Ctrl-P*

Function — Displays the Typestyle menu.

This command displays the Typestyle menu, which lists many of the Sprint-supported typestyles. You can select Normal, Bold, Underline, Word Underline, Italic, and Strikethrough. This menu also provides commands to subscript, superscript, as well as large text. Choosing Font lists available typestyles for you to choose from.

**Normal**
Turns off the selected typestyle and can be used with any of the typestyles listed in this table. Pressing the *Right arrow* key when you're at the last character of the typestyle performs the same function as this command.

**Note:** The font defined as "normal" is the default font you selected when you created a printer driver using the Sprint install program.

**Bold**
Prints text in boldface type. If your printer can't print bold characters, it will double-strike the text.

**Italic**
Prints text in italics. If your printer can't print italic characters, it will underline the text.

**Underline**
Underlines all text (words and the spaces between them).

**Word Underline**
Underlines words only, not the spaces between them.

**Large**
Prints selected text in a big, bold font, if your printer has this capability.

**+ Superscript and – Subscript**
Raises and lowers text (respectively) by one-half of one line. Sprint will also use a smaller font to print the superscripted character (if your printer has that capability).

If your printer does not support vertical microspacing, Sprint will create a blank line above the text for a superscript or blank line below for a subscript.

**Strikethrough**
Prints text with dashes (--) through it. This is useful for showing deleted text, such as in legal documents.

**Font**
Displays a list of available fonts for selected text.

**Character Size**
Lets you specify a font size for the selected text. You can use any legal dimension, but *points* are the most common.

You can also make the new size relative to the current size by omitting a dimension. For example, entering just 2 makes the selected text *twice* the current size.

This command works only if you have a printer that can scale characters (as PostScript printers can).

**Hidden**
Displays text on your screen that will *not* appear when printed.

Table 1.10 lists the commands that affect your text onscreen and includes the corresponding @-sign commands and control code characters to begin a typestyle command. Note that the ^N character *always* ends a typestyle format.

Table 1.10: Typestyle Commands

| Menu Selection | Control Character |
|---|---|
| **Bold** | ^B |
| *Italic* | ^E |
| Underline | ^U |
| Word Underline | ^W |
| Large | ^A |
| + superscript | ^S |
| – subscript | ^Q |
| ~~Strikethrough~~ | ^X |

How To

There are two different ways to change the typestyle of your text. It depends on whether you are inserting new text that you want to be affected, or whether you want to change an existing typestyle.

1. For new text, choose the desired typestyle from the Typestyle menu, type the text to be affected, and then choose the Normal command from the Typestyle menu to end the format. You can also use *Right arrow* to end the typeface.

   For example, choose Bold from the Typestyle menu, type the text you want in boldface type (look at the column number on the status line to see how Sprint has affected the number), and then press *Right arrow*. The cursor won't move, but look at the column number on the status line. Now when you start typing, the new text will be in regular type.

2. For existing text, select the block of text you want to change, then choose the desired typestyle from the Typestyle menu. Since you've marked a block, Sprint knows that the typestyle applies to that block and changes the entire block to the typestyle you selected.

   For example, if you want your company name to appear in bold type, select the company name (using the Block-Select commands or *F3*) and then choose Bold from the Typestyle menu. Since you've marked the beginning and end of the block (the company name), you don't have to select the Normal com-

mand. Sprint automatically ends the bold typestyle at the end of the marked block.

You can also use *F3* to mark text to be affected.

Tips

If you already selected a typestyle for a particular block of text but want to change that typestyle to something else, you need to return the text to Normal, then reselect it, and apply the new typestyle to it. For example, to change your company name from bold to italics, you would select the type, choose **Typestyle/Normal**, select the text again, and choose **Typestyle/Italic**. If you simply chose **Italic** without first choosing **Normal**, your company name would be in bold-italic typeface.

If you're ever unsure about the typestyle affecting your text, move the cursor to the text in question and select **Typestyle**. The status line will display the name of the **Typestyle** affecting the current word.

If your computer uses the Hercules InColor card or the Hercules Graphics Card Plus, your typestyles can be made to be quite varied. Choose **Customize/Colors/Modify/Typestyle** to choose from a palette of fonts and font styles. Also, italic fonts will show in a more accurate display of italics (they'll be slanted onscreen instead of underlined) if you are using these cards.

See Also

Codes, Colors, Control Characters, Font, Font Size

# Undelete

Keystrokes

*Alt-E* (or *F10*, Edit), **Undelete**

Function

Returns the last-cut text to the exact spot it was cut from.

Note that the difference between **Undelete** and **Paste** is that **Paste** inserts the last-cut or last-copied text *at the cursor location*, but **Undelete** inserts the text *back where it started*—even going to a different file, if necessary.

See Also

Canceling, Deleting

# Underline

Refer to the Typestyle Menu entry for details.

# Undo

Refer to the Colors entry for details.

# Unzoom

Refer to the Window entry for details.

# Use Wildcards

Refer to the Searching entry for details.

# User Dictionary

Refer to the Spelling Menu entry for details.

# User Interface Menu

Keystrokes — *Alt-C* (or *F10,* Customize), User Interface

*Shift-Alt-U*

Function — Displays the User Interface menu.

Choosing this commands displays a menu with the following commands:

**Function Keys**
Allows you to redefine the meaning of your function keys and any *Ctrl-Alt* or *Shift* version of them. You can also change any Ctrl-letter or Alt-letter shortcut using this command.

The Function Keys entry in this chapter has a complete list of the standard function key assignments.

**Reset Shortcuts**
Choosing this command immediately clears all loaded macros and shortcuts that you've loaded "on top of" the user interface. This command returns you to the "factory settings" of the user interface you're using.

**Save**
Choosing the Save command lets you save all your shortcuts into a "user interface" file (with the extension .UI), so you can reload it whenever you want.

The Save command is especially handy if several different people use the same computer. Using this command, each person could have a separately saved set of shortcuts that could easily be reloaded whenever needed.

Note that if you have already saved a setup, you can reload it to discard any new shortcuts you created in the current session. This technique, in effect, can be used as a "revert to saved" command for user interfaces.

**Load**
Choosing the Load command displays a list of user interface files (which always have the .UI extension). Selecting one of these UI files loads a user interface.

For example, if you are familiar with WordStar, you can tell Sprint to respond to commands that you're accustomed to using in WordStar. Once you do, you'll find a help menu by pressing *Ctrl-J*, a block menu by pressing *Ctrl-K*, a quick menu by pressing *Ctrl-Q*, and so on. If you're not familiar with WordStar, however, we strongly suggest using one of the Sprint user interfaces, which are much more powerful than the alternative.

The alternative UIs available are MSWORD (MS Word-compatible), WORDPERF (WordPerfect-compatible), SIDEKICK (SideKick-compatible), WORDSTAR (WordStar-compatible), and EMACS (EMACS-compatible).

Note: The Reset Shortcuts, Save, and Load commands are not available if your computer has two 360K floppy

disks and no hard disk—unless you insert Program Disk B into Drive A before choosing the command.

There are three Sprint UIs: SPTUTOR (used only with the Sprint tutorial in the *User's Guide*), SPBASIC (for anyone who's just starting out using a word processor), and SPADV (for advanced users).

When you choose SPADV, you have access to the complete Sprint menu system. When you choose SPBASIC, however, some of the more esoteric commands are unavailable. Following is a list of commands that are *not* found in the menus when you choose the basic (SPBASIC) user interface instead of the advanced (SPADV) interface.

*Edit Menu*

Edit/Erase

*Insert Menu*

Insert/Variable
Insert/Define Text Variable

*Style Menu*

Style/Modify

Style/Graphics
Style/Graphics/EPS Picture
Style/Graphics/KeyCaps
Style/Graphics/Bar
Style/Graphics/Draw Box

Style/Index
Style/Index/Word
Style/Index/Reference Word
Style/Index/Master Keyword
Style/Index/See
Style/Index/Also See
Style/Index/Index Under
Style/Index/Page Range

Style/References
Style/References/Footnote
Style/References/Endnote
Style/References/Notes

Style/X-Reference
Style/X-Reference/Define a Tag

Style/X-Reference/Reference a Tag

Style/Other Format

*Layout Menu*

Layout/Page Breaks/Conditional Page Break
Layout/Page Breaks/Group Together on Page
Layout/Page Breaks/Keep with Following Text
Layout/Page Breaks/Widow-Orphan Control

Layout/Document-Wide/Offset
Layout/Document-Wide/Word Spacing
Layout/Document-Wide/Inter-Paragraph Spread
Layout/Document-Wide/Style Sheet

*Print Menu*

Print/Advanced Options
Print/Advanced Options/Number of Passes
Print/Advanced Options/Formatted Print
Print/Advanced Options/Wordwrap ASCII Files
Print/Advanced Options/Log Error to File

*Utilities Menu*

Utilities/Spelling/Main Dictionary
Utilities/Macros
Utilities/Macros/Load
Utilities/Macros/Enter
Utilities/Macros/Run

*Customize Menu*

Customize/Screen/Non-Breaking Spaces
Customize/Screen/Wide Spaces (Springs)

Customize/Options/Menu Display Delay
Customize/Options/Tone

Tips

The shortcut *Shift-Alt-U* is available in *all* user interfaces. So if you load, say, Sprint's WordPerfect user interface and then want to bail out, press *Shift-Alt-U* and choose a regular Sprint interface (like the basic or advanced).

You should use the tutorial interface (SPTUTOR) only when you're working with the tutorial lessons in the *User's Guide* manual.

See Also

Potpourri, Function Keys, Glossary

# Utilities Menu

Keystrokes — *Alt-U* (or *F10*, **Utilities**)

Function — Displays the **Utilities** menu.

The **Utilities** menu gives you access to several handy Sprint features, which have their own entries in this chapter:

**Spelling**
**Hyphenation**
**Thesaurus**
**Glossary**
**Arrange-Sort**
**Line Drawing**
**Potpourri**
**QuickCard**
**Macros**
**DOS Command**

See Also — Refer to the individual entries for more information.

# Variables (General Information)

Keystrokes — *Alt-I* (or *F10*, **Insert**), **Variable**

*Alt-I* (or *F10*, **Insert**), **Define Text Variable**

Function — Lists commands for inserting and defining variables.

A *variable* is just what its name implies: Something whose *value* (meaning) varies. The Sprint program supports a variety of variables, including the ones for day, month, year, chapter number and title, section number and title, figure number, table number, and page number. These variables each contain a *value*—the *current* meaning of the variable. To print the value of a variable, you use the **Variable** command from the **Insert** menu.

Sprint also allows you to define, name, and reference your *own* variables.

Why use variables? Variables allow you to reference elements in a document (such as the current chapter number, page number, figure number, and so on) without knowing the actual number or text assigned to the element.

For example, if you want the formatter to print the current chapter number and title in the header of each page, you can reference the variables *Chapter* and *ChapterTitle* in your **H**eader command. That way, you don't have to type a new header at the beginning of each chapter or change your headers if you reorganize the document. Likewise, if you want the footer in a document to contain the current page number and, date, and file, you can reference the variables *Page*, *MonthName*, *Day*, *Year*, and *Manuscript* in a single **F**ooter command.

***Built-in Variables***

Variables for the date, time, and the current page number are *built-in* variables. They're a part of the Sprint program and cannot be changed. Table 1.11 lists each of Sprint's built-in variables.

Table 1.11: Built-in Formatting Variables

| | |
|---|---|
| Day | The day of the month (1-31). The formatter gets this information from DOS as soon as it begins formatting the file.<br><br>If you want to print the name of the day (for example, Sunday, Monday, Tuesday, etc.), reference the variable *WeekDay* (a variable defined in STANDARD.FMT) and choose None for the template. |
| FirstPage | The last page number of the introductory matter created by the formatter (such as the table of contents, etc.). For example, if the table of contents is three pages long, the value of the *FirstPage* variable is 4.<br><br>Normally, Sprint prints page numbers on the introductory matter and then resets the page counter when it begins the body of the document. This means that the first page of your document begins on page 1. If you don't want Sprint to reset the counter and would like your pages numbered continuously, choose **O**ther Format from the **S**tyle menu and type *Set page=firstpage* at the beginning of your document. |

Table 1.11: Built-in Formatting Variables, continued

| | |
|---|---|
| Font | The full name of the current font, including dots for attributes (e.g., Times.bold.italic). |
| Hour | The hour of the day (0-23). The formatter gets this information from DOS as soon as it begins formatting the file.<br><br>If you want to specify pm, set up a formula containing `Hour >= 12`. |
| Manuscript | The name of the *main* file being printed. The main file is the file that contains all of the Include commands that tell the formatter to merge other files. |
| Minute | The minute of the hour (0-59). The formatter gets this information from DOS as soon as it begins formatting the file. |
| Month | The number of the current month (1-12). The formatter gets this information from DOS as soon as it begins formatting the file.<br><br>If you want to print the *name* of the current month instead of the number, reference the variable *MonthName* (defined in the STANDARD.FMT file) and choose None as the template. |
| Page | The current page number. The formatter automatically increments this value every time it begins a new page. You can set the page number to any desired value, however, using the formatting command Set *page=value*. For example, type `Set page=101` if you want to force the formatter to number the current page 101. Subsequent pages will be incremented from the value 101 (e.g., the next page will be numbered 102, unless you type another, different Set page command). |
| Plain | Normally 0, if the –P print option was selected. You can use the If command in a definition in STANDARD.FMT to test for whether plain printing is being done: `If(plain=0, Y...)`. |
| Printer | The name of the printer being used. This is not the name of the .SPP file, but the name of the device SP-SETUP put in the file (e.g., the name you assigned to the printer when you installed the printer). |
| Size | Current point size in vertical printer units. |
| SourceFile | The name of the current file being printed. |
| SourceLine | The current line number in the file being printed. |
| WeekDay | The day of the week (Sunday = 0). The definition of this variable in STANDARD.FMT |

Table 1.11: Built-in Formatting Variables, continued

| | |
|---|---|
| | creates a template which prints the names of the days instead of the number. |
| Words | The number of words printed so far. The formatter increments this value for every word formatted in the main text (not including format commands, page headers, page footers, and so on). If you want to determine the number of words in a document, use the formatter command Message and reference the variable *Words*. The formatter will display the number of words formatted thus far. For example, inserting this line at the end of your document:<br><br>`BEGIN MESSAGE`<br>`This file has WORDS words in it.`<br>`END MESSAGE`<br><br>results in a tally of the total words formatted being displayed on the screen after the file is sent to a printer. The message does not *print*, however. |
| Year | The last two digits of the current year (the year minus 1900). The formatter gets this information from DOS as soon as it begins formatting the file. |

### *Variables Defined in STANDARD.FMT*

Sprint has other variables that are defined in the STANDARD.FMT file, which means you can change the way the formatter prints the value of these variables. For example, *Chapter* is a variable. The number Sprint assigns to a particular chapter depends on how many chapter commands you entered before it. If you wanted your chapter numbers to print in Roman numerals or print with letters rather than numbers, for example, you could modify the variable's definition in the STANDARD.FMT file. Table 1.12 lists all variables defined in the STANDARD.FMT file.

Table 1.12: Variables Defined in STANDARD.FMT

| | |
|---|---|
| Chapter | The current chapter number. This variable contains the chapter number only if you've selected the Chapter command prior to referencing this variable; if you haven't created any chapters, the formatter prints a 0 in place of the reference to the variable. |
| Section | The current section number. |
| Subsection | The current subsection number. |
| Paragraph | The current paragraph number. |
| Appendix | The current appendix letter. |
| SectionNumber | The number of the last chapter, section, subsection, or paragraph started. |
| SectionTitle | The name of the last chapter or level started. |
| ChapterTitle | The name of the last chapter or appendix started. |
| SubsectionTitle | The name of the last subsection started. |
| ParagraphTitle | The name of the last paragraph started. |
| AppendixTitle | The name of the last appendix started. |
| Figure | The number of the last figure, which included a Caption command. |
| Table | The number of the last table, which included a TCapt command. |
| Footnote | The number of the last footnote or endnote. |
| MonthName | The name of the current month. |
| WeekDay | Modified to print the name of the day. |

How To

When you want to print the value of a variable listed in Table 1.11 or Table 1.12, choose the **Insert/Variable** command. Sprint lists available variables, as well as **Other**. If you select **Other**, Sprint prompts for the name of the variable you want to reference. Type the name of the desired variable (for example, *Paragraph, Chapter, Figure*). Sprint highlights the name of the variable; when the formatter prints your file, it will replace the command with the value of the variable you referenced. For example, type

```
This is page
```

then choose *page* from the Insert/Variable menu. The Pick Template for Page menu appears on your screen. Choose the specifications desired. You're telling the formatter to print the current page number when it sees this *Page* variable. For example,

```
This is page PAGE, t="%d"
```

For a detailed discussion of variables (how to define your own text variables, how to insert variables in headers and footers, how to reference date and time, and much more), refer to the *Advanced User's Guide.*

See Also — Tags, Header Menu, Footer Menu, Page Numbers, and Chapter 2 (Case, TCT, Template)

# Wang (IWP)

Refer to the Translate entry for details.

# WeekDay (Variable)

Refer to the Variables entry for details.

# Wide Space (Spring)

Keystrokes — *Alt-I* (or *F10,* Insert), Wide Space (Spring)

Function — Pushes text to the right margin or next tab.

This command tells the formatter to insert as much blank space as needed to align text at the right margin or a predefined tab stop. This is quite useful when creating telephone directories, tables of information, play bills, and price lists.

How To — When you want to create two or more columns of text and don't want to figure out where to set tabs or how many spaces you should insert, you can use the Wide Space (Spring) command.

Let's say you want to create a two-column telephone directory. The column on the left should contain personnel names, and the column on the right should list phone numbers. You can enter the first name, select the **Wide Space (Spring)** command, and then enter the phone number. After you press *Enter* to start a new line, you can repeat these steps for each person you want listed in the directory. When you print this directory, the names will be aligned at the left margin, and the phone numbers will be aligned at the right margin. For example (note the ^F represents a wide space),

```
Jessie Marigold      ^F219-111-1111
Lawrence Daisy       ^F714-222-2222
Maren Carnation      ^F603-333-3333
Arlin Rose           ^F408-444-4444
```

produces the following:

Jessie Marigold 219-111-1111
Lawrence Daisy 714-222-2222
Maren Carnation 603-333-3333
Arlin Rose 408-444-4444

You can also use this command to create more than two columns of text, but the printed result might not line up correctly. Using the example above, you could add a column for Department number. The middle column is centered on the whitespace and so does not necessarily line up perfectly. For example,

| Name | Department | Phone Number |
|---|---|---|
| Jessie Marigold | 10 | 219-111-1111 |
| Lawrence Daisy | 5 | 714-222-2222 |
| Maren Louise Carnation | 5 | 603-333-3333 |
| A. Rose | 7 | 408-444-4444 |

To create an *aligned* three-column table, you should use the Table command and set up tab stops where you want them.

Tips

Wide spaces act like springs pushing against each other. If you want text precisely placed two-thirds across the page, you can insert *two* wide spaces before the text and

only *one* after it. (Two wide spaces exert twice the "push" than one.)

See Also Justification, Table, Tabs

# Wide Spaces (Spring)

Refer to the Screen entry for details.

# Widow-Orphan Control

Refer to the Page Breaks entry for details.

# Window

Keystrokes *Alt-W* (or *F10,* **W**indow)

*Shift-F2* (resizes a window)

*Shift-F3* (opens a window)

*Shift-F4* (closes a window)

*Shift-F5* (zooms or unzooms a window)

*Shift-F6* (goes to next window)

*Shift-F7* (scrolls all windows up one line)

*Shift-F8* (scrolls all windows down one line)

*Shift-F9* (closes all windows)

*Alt-U* (or *F10,* **U**tilities), **P**otpourri, **W**indowCloseAll

Function Displays the **W**indow menu.

Windows are an easy way to display either different parts of the same file or more than one file at a time. When you open a window, Sprint splits the screen and separates the windows by a status line. The window in which the cursor is positioned is called the *current* window.

**Open** *Shift-F3*
Tells Sprint to split the screen and display part of the current file in one window and part of the same file in another window. If you want a window to contain a different file, simply move to the window and open the file (*Alt-F*, **O**pen).

**Close** *Shift-F4*
Closes the active window.

**Shut All** *Shift-F9*
Closes all open windows except for the active one (the one that your cursor is in).

**Zoom** *Shift-F5*
Expands the current window so that it takes up the entire screen. Once you have zoomed a window, the command name changes to **U**nzoom.

**Resize** *Shift-F2*
Lets you expand or reduce the size of the current window. Press the + or – key to expand or shrink the window by one line for each press. Press *Esc* or *Enter* to exit.

**Next** *Shift-F5*
Moves the cursor from the current window to the next window.

**Previous**
Moves the cursor from the current window to the previous window.

How To

You can have up to six windows open on your screen at one time. Using the commands listed on the **W**indow menu, you can create, open, close, scroll, enlarge, reduce, and move between windows. Using commands from the **B**lock menu, you can also copy and move text between windows.

When you want to use windows to view more than one open file at a time, you'll first need to open the window(s) and then open the desired file(s) to appear in those windows. Let's say you're working on a file called LETTER1.SPR and want to view LETTER2.SPR at the same time.

First create a window and then open LETTER2.SPR (LETTER1.SPR is in the top window; the first opened file). Sprint displays LETTER2.SPR in the current window (the window containing the cursor, or the one you just opened). The following steps explain how to do this.

1. From the currently open file (LETTER1.SPR), press *Alt-W* and choose **O**pen. Sprint splits the screen and displays part of the file in each window. The cursor will be positioned in the bottom window.
2. To open a different file (LETTER2.SPR) in the bottom window, press *Alt-F* and choose **O**pen.
3. At the prompt, type the name of the file you want to open (LETTER2.SPR). Sprint opens and displays the file in the bottom window. The file you were working on before entering these commands (LETTER1.SPR) is displayed in the top window.

Tips

Windows are useful when you need to move information within a large document. You can display part of the document in one window and other parts in up to five other windows. If a document is comprised of multiple files, you can view and edit several of these files at once. You use Sprint's editing commands within windows just as you do when there is only a single file displayed on the screen. You use the shortcuts listed above (or the commands on the **W**indow menu) to move between windows.

If you are using a mouse, you can use it as a shortcut for several window commands. Clicking the left button in an area to the right of the right onscreen margin (that is, where there is no text) affects the current window. (If you have the right margin set to column 65, this non-text zone is 15 characters wide. If you have a file with no ruler, the non-text zone is any position that's to the right of a hard return paragraph mark.) Clicking in the upper quarter of the non-text zone switches to the previous window. Clicking in the lower quarter of the non-text zone switches to the next window. Clicking in the

middle part of the non-text zone zooms or unzooms the current window.

See Also Insert (File), Open (File)

# Word (Hyphenation)

Refer to the Hyphenation Menu entry for details.

# Word (Index)

Refer to the Index entry for details.

# WordPerfect 4.2

Refer to the Translate entry for details.

# WordPerfect (User Interface)

Refer to the User Interface Menu entry for details.

# Word Selecting

Refer to the Block Select Menu entry for details.

# Word Spacing

Refer to the Document-Wide Menu entry for details.

# Word (Spelling)

Refer to the Spelling Menu entry for details.

# Word Underline

Refer to the Typestyle Menu entry for details.

# Words (Variable)

Refer to the Variables entry for details.

# WordStar/WordStar 2000

Refer to the Translate entry for details.

# WordStar (User Interface)

| | |
|---|---|
| Keystrokes | *Alt-C* (or *F10*, Customize), User Interface, WordStar<br>*Shift-Alt U*, WordStar |
| Function | Loads Sprint's WordStar-compatible menus and shortcuts. |
| Tips | There are many WordStar-compatible commands supported in the regular Sprint user interfaces, too. These shortcuts all use the *Ctrl* key and start with either *Ctrl-K, Ctrl-O, Ctrl-P, Ctrl-Q,* or *Ctrl-J.*<br>**Note to 360K two-floppy system users:** To load the WordStar-compatible user interface, you must rerun SP-SETUP. |
| See Also | User Interface Menu |

# Wordwrap ASCII Files

Refer to the Advanced Options entry for details.

# Wordwrapping

Keystrokes — *Ctrl-OW* (toggles wordwrapping)

Function — Inserts a new ruler above the current line and temporarily removes the right margin setting to show your document without wordwrapping.

Pressing *Ctrl-OW* goes back and forth between your current ruler and a version of that ruler with no right margin set. This is a handy way of spotting hard returns in your file, since lines with hard returns will not change when wordwrap is turned off. Removing all rulers from your file has the same effect.

See Also — Advanced Options, Hard and Soft Returns

# Wrap Long Lines

Refer to the ASCII Files entry for details.

# Write As

Keystrokes — *Alt-F* (or *F10*, **F**ile), **W**rite As

Function — Saves the current file under a new file name.

This command writes the contents of the current file under a new name and then uses that name as the current file. For example, if you're editing the file ARTICLE.SPR and select the **W**rite As command, you can write the contents of the file to a file named ARTICLE.NEW. You can continue editing the ARTICLE.NEW file and have a "clean" version of the older version of the file safely stored on disk.

How To — To write a file to a different file name, press *Alt-F* and choose **W**rite As. Sprint displays the following prompt in the status line:

```
Write file as: FILENAME.EXT
```

Type the name of the file to which you want to copy your text. If the file name already exists, Sprint prompts

```
Overwrite existing file?
```

If you type Y, Sprint replaces the text with the text of your current file. If you type N, Sprint cancels the command.

You can also use this command to write an unnamed file to disk. If you typed SP at the DOS command line, didn't enter a file name as part of the command, and there were no open files in your swap file, Sprint would display a blank, unnamed file. If you entered text in this file, you could write it to disk by selecting the **Write As** command. Sprint displays the following prompt:

```
Write file as:
```

If you don't choose this command and try to save your unnamed file, Sprint will prompt you for the name you want to give this file.

Tips

If you want to write a portion of your file (not the entire file), select the text you want to write and choose **Edit/ Write Block**.

When Sprint prompts for a file name, enter a name that does not yet exist in your current directory. Sprint leaves the block in its present location, writes the block to the new file, and then returns you to the file from which you selected the **Write Block** command. If you enter an existing file name, Sprint asks if you want to overwrite the existing file. If you reply Y, you'll replace the existing text with your marked block. If you answer N, Sprint cancels the command.

See Also

Save, Write Block

# Write Block

Keystrokes

*Alt-E* (or *F10*, Edit), **Write Block**

*Ctrl-KW*

Function

Writes a marked block of text to a specified file name.

This command works on a marked block of text. Once you select the block and enter the **Write Block** command, Sprint prompts you for the name of the file to

contain the block. After you enter a file name, Sprint writes the block to the specified file, closes the file, and then returns you to the file from which you selected the Write Block command. The block you marked remains in your current file, and it remains selected so you can delete it or move it if necessary.

How To

To write a block to a file, you must first select the block you want to write to a file. After you have done this, press *Alt-E* and choose **W**rite Block. Sprint displays the following prompt:

```
Name of file to write block to:
```

If you enter the name of an existing file, Sprint prompts

```
Overwrite existing file?
```

If you enter `Y`, Sprint replaces the text in the specified file with the block you've marked. If you don't want to overwrite the file, enter `N` to cancel the command.

See Also

Block Select Menu, Save, Write As

# X-Reference Menu

Refer to the Tags entry for details.

# Year (Variable)

Refer to the Variables entry for details.

# Zoom

Refer to the Window entry for details.

C H A P T E R

# 2

# Miscellaneous Formats

This chapter describes about 30 commands that are not directly available using the Sprint menus.

You'll need to use these commands only rarely (if ever), and you may find the explanations of them a bit rough going. Nonetheless, you may also find the perfect solution to a problem you're having, or you may be one of those people who just loves to squeeze the last drop of power out of a product. If so, read on.

Remember, the primary reference to Sprint formats and concepts is not this chapter but Chapter 1, "The Sprint Menu Encyclopedia." Refer to that chapter for a list of acceptable dimensions (Table 1.2 on page 58), parameters (Table 1.8 on page 149), and variables (Table 1.11 on page 253).

## How to Enter Non-Menu Formats

You enter the commands in this chapter by choosing **Style/Other Format** and then following the prompts in the status line.

When you choose the **Other Format** command, you're first greeted with a simple prompt:

```
Format:
```

You type the name of the format (as listed in this chapter) and press *Enter*.

Sprint then asks if this format is a command or a format that affects a region.

```
Insert for Region (R) or Command (C):
```

The difference between these two choices is that some formats affect blocks of text (regions), but others are commands that Sprint carries out on the spot (without waiting to find the end of the region). You're already familiar with this distinction without really knowing it: The formats that Sprint inserts with a Begin and End command are "region" formats; the ones Sprint inserts without the Begin and End are not.

Common sense will usually correctly guide you to the right choice for the format you want to insert. But to take the guesswork out of it, the descriptions of the commands that follow all state whether you should press *R* or *C* in response to this prompt.

If you press *C*, Sprint inserts the command as you typed it at the current cursor position.

If you press *R*, however, Sprint displays another prompt:

```
Press (B) for Begin command, (E) for End command,
or ESC to cancel:
```

Sprint is asking you whether your cursor is at the start or the end of the block of text you want affected by the format. If it's at the start, press *B*. Let's say you're formatting some text in the Display format; Sprint would insert

```
BEGIN DISPLAY
```

If your cursor's at the end of the text, press *E*. In that case, Sprint would insert

```
END DISPLAY
```

Remember, every beginning needs an end. If you've inserted a Begin command in your text, you must also insert a corresponding End command. (You choose **O**ther Format again and go through all the same prompts, but this time press *E* for End command.)

If you're not particularly pleased by the prospect of repeating the series of prompts, here's a shortcut: Select the text you want affected by the format *before* you choose **O**ther Format. This way, Sprint automatically inserts both the Begin and End commands as soon as you enter the command name.

# Address

Keystrokes

Choose *Alt-S* (or *F10*, Style), **O**ther Format, type `Address`

Function

Creates a format for return addresses.

This command defines the format for addresses used in letters. It sets the left margin at the center of the page, inserts two blank lines above and below the address text, and left justifies it.

You will find this command defined in the Sprint file STANDARD.FMT.

How To

Type the text of the address (or any other text you want to begin printing in the center of the page). Select the text, choose **O**ther Format, and type `Address`. When you press *Enter*, Sprint inserts the BEGIN and END commands before and after your text. For example,

```
BEGIN ADDRESS
Frank Borland
Borland International, Inc.
P.O. Box 660001
4585 Scotts Valley Drive
Scotts Valley, CA   95066
END ADDRESS
```

produces

Frank Borland
Borland International, Inc.
P.O. Box 660001
4585 Scotts Valley Drive
Scotts Valley, CA 95066

(If you look carefully, you can confirm that this *is* centered on the page.)

Tips

If you want the return address to begin printing somewhere other than at the center of the page, you can either modify the definition of Address in a copy of STANDARD.FMT or modify the Address format itself

(choose Style/Modify) by adding the *LeftIndent* parameter. For example,

```
BEGIN ADDRESS, LeftIndent 0 Chars
Frank Borland
Borland International, Inc.
P.O. Box 660001
4585 Scotts Valley Drive
Scotts Valley, CA  95066
END ADDRESS
```

This address prints like this:

Frank Borland
Borland International, Inc.
P.O. Box 660001
4585 Scotts Valley Drive
Scotts Valley, CA 95066

# Begin and End (Formats)

Keystrokes — *Alt-S* (or *F10,* Style), Other Format, and type B or E at the prompt

Function — Begins and ends the current format.

These commands instruct Sprint to begin and end a format that you specifically selected with the Style/Other Format command. Usually these Other Format commands are those not available through the menus or are those that you have "defined" either in your file or in the appropriate .FMT file.

Commands on the Style/Lists menu also prompt for whether to begin or end the command, and if you choose Table or Figure from the Style menu, Sprint automatically inserts the BEGIN and END commands for you.

How To — Press *Alt-S* and choose Other Format to enter your formatter command. Sprint prompts you with:

```
Insert for Region (R) or Command (C)
```

If you press *R,* and you've already selected your text, Sprint inserts the BEGIN and END *format* commands for

you. You can also choose Region before typing in the format text. Sprint displays the following prompt instead:

```
Press (B) for Begin command, (E) for End command,
or ESC to cancel:
```

Type `B` to begin the format command, type your text, and then choose the **O**ther Format command and type `E` to end your format.

If you choose *C* (Command) from the Region/ Command prompt, you can type `Begin Format` (and any modifications to the format command), and Sprint inserts the BEGIN *format* command in your text at the current cursor position. Be sure to end your format as well (just type `End Format`).

If you modified a BEGIN *format* command to include formatting parameters, you *do not* have to include these parameters in the END *format* command. When the formatter sees the END *format* command, it automatically ends the format, parameters included.

**Note:** If you enter a BEGIN command and forget to enter the corresponding END command, Sprint displays an error message when you try to print your file and gives you the line number that the BEGIN command is on. Sprint will not print your file until you correct the error.

Tips

Usually it is easier to select as a block the text you want the format to affect *before* choosing **O**ther Format from the menus. (Sprint inserts the BEGIN and END commands for you, both at once.)

# Case

Keystrokes

Press *Alt-S* (or *F10,* **S**tyle), **O**ther Format, type `Case`, and press *C*

CASE *Selector, Name1 "version 1", Name2 "version 2", . . . Else "version if none above match"*

Function

Generates slightly different versions of the same document, based on your input. Also, when used in creating

or modifying .FMT files, this command can offer powerful branching capabilities.

The Case command lets you set up a document so you can produce several different versions of it. You create a file that contains text appropriate for several different versions. When the information to be printed depends on a certain case being true (for example, in that case, print this), you insert a Case command, which defines each possible case, and the text that should be printed in each case. The Case command looks like this:

```
Case Selector, Name1 "version 1",
               Name2 "version 2", ...
               Else  "version if none above match"
```

When you use the Case command from the Other Format command, the command must be able to fit in the status line (about 70 characters long). If you want to insert a Case command with many lines or paragraphs, you have to use the @-sign command @Case and enclose the lines in delimiters. When you're using the Case command in .FMT files, you always use the @-sign version of the command as well.

For example, let's say your company produces two types of blenders. The instruction manual that goes with each blender is identical except for the last page, which provides a list of replacement parts. You can create a single manual and, at the end of the manual, insert a Case command that tells the formatter "If Model A, print this list; if Model B, print this list." The Case command might look like this:

```
@CASE[Models, ModelA "Part 1A-000
                      Part 2A-000
                      Part 3A-909",
              ModelB "Part 1B-000
                      Part 2B-000
                      Part 3B-909"]
```

The Case command begins with the *Selector*, the "subject" on which the formatter will base its printing decision. In the previous example, the subject is *Models*—the parts list that should be printed depends on the model of the blender.

You tell the formatter which model applies with a **Define Text Variable** command (**Insert/Define Text Variable**), which *must* precede the Case command.

The next part of the Case command includes a *Name* for the first version (ModelA in the example), followed by the *version1* text (the quoted text that begins with "Part 1A-000"). The text must be enclosed in quotes. Following the *version1* text is a comma, and then the *Name* assigned to the second version (ModelB), followed by the text of that version (within quotes). You can repeat this *Name "version text"* syntax until you've covered each possible version.

When the formatter sees the Case command, it looks for a **Define Text Variable** command that contains the variable *Selector* (for example, *Models*) and a *Name* that matches one of the names in the Case command (for example, *Models=ModelA*). This tells the formatter which version you want to print.

You can also include the name *Else* and a different version of the text. The name *Else* means "none of the above": If the *Name* entered in the *Selector=Name* string doesn't match any of the names listed in the Case command, the formatter incorporates the *Else* version. You can omit the *Else* clause if you wish; if no *Name* matches, then no text at all will be produced by this Case command.

How To

Use the Case command when you want to produce slightly different versions of the same document.

To see how you would input the Case command, consider an example of a form letter, the closing paragraph of which you want to vary depending on the recipient's account status. Right before the last paragraph, type the @Case command, an opening delimeter, the text of the command, and a closing delimeter, like this:

```
@Case[AccountStatus,
Tardy "Perhaps this reminder and your payment have
crossed in the mail. If so, please disregard this
notice.",

Overdue "Please pay past due amount immediately.",

Delinquent "You have not responded to any of our
requests for payment. Please be advised that if we do
not receive the full amount within five days, we will
turn your account over to our attorneys for
collection.",

Else "If you have any questions regarding your bill,
please call 555-4444 between 8:30 a.m. and 4:30
p.m."]
```

You can then select the desired paragraph by assigning a string value *at the beginning of your document.* Choose Define Text Variable from the Insert menu and type *one* of the following string assignments:

```
AccountStatus = "Tardy"
AccountStatus = "Overdue"
AccountStatus = "Delinquent"
AccountStatus = "Okay"
```

**Note:** Although only *one* of the cases is selected for a given document, the commands in all of them must be legal.

If you typed `AccountStatus="Overdue"`, the last paragraph will contain the text, "Please pay past due amount immediately." If you typed `AccountStatus="Okay"`, the formatter would see that neither the *Tardy, Overdue,* nor *Delinquent* paragraphs apply, and would print the *Else* version instead.

You can use any delimiter pair around the version text, not just double quotes. (If a double quote is in one of the versions, of course, you *must* use some other delimiter.)

You can also use semicolons (;) in place of commas, and place equal signs (=) between the names and the version text. So another valid Case statement is

```
CASE Sex, F = <her>; M = 'his'
```

Note that because this Case command is only one line, you do not have to use the @-sign version of it; instead

you can type in the whole line after choosing Style/ Other Format.

If you want to include a comma, semicolon, slash, or equal sign as part of a version name, you must enclose the name within delimiters:

```
CASE Fraction, "1/2" "half", "2/3" "most"
```

This command states if the fraction is 1/2, print the word *half*; if the fraction is 2/3, print the word *most*. (The quotation marks are necessary around "1/2" because of the slash character, which Sprint will misunderstand unless you enclose the text that contains it in quotes.)

If you want two *Name* values to use the same version of the text, just put a comma after the first one:

```
CASE Time, 10am, 11am, 12 "hello", 1pm "good afternoon"
```

causes "hello" to print whenever the value of *Time* is 10am, 11am, or 12 pm, and "good afternoon" to appear when *Time* is 1pm.

***With .FMT Files***

You can also use the Case command when you're working in .FMT files. Basically, you use the command as in regular Sprint documents except that you ordinarily use @-sign commands when editing .FMT files. Here are some examples of the Case command at work in .FMT files:

```
@Case(printer,LaserWriter "@include(POSTSCR.TCT)")

@ifdef(paper, y "@Case{paper,
Letter  "@Style(paper 11 inches)",
Legal   "@Style(paper 14 inches)",
A4      "@Style(paper 297 mm,paperwidth 210 mm)",
A3      "@Style(paper 420 mm,paperwidth 297 mm)",
A5      "@Style(paper 210 mm,paperwidth 148 mm)",
B5      "@Style(paper 250 mm,paperwidth 176 mm)",
IntFan  "@Style(paper 305 mm,paperwidth 210 mm)",
Comput  "@Style(paper 11 inches,paperwidth 14.75 inches)",
else  "@Error(Paper type '@value(paper)' not known.)"}")
```

The first line simply says that if the printer chosen is called "LaserWriter," the file called POSTSCR.TCT should be included when formatting a document (remember, this command is in an .FMT file, which Sprint uses only at print-time).

The second example sets up a more elaborate condition. It says that if the variable called *paper* is defined and matches one of eight names, to use the corresponding @Style command to adjust for it. This example also includes an *else* clause that handles any other paper definition.

Tips

If you add a StringInput command to the example on page 273, you can eliminate the need to assign a string. At the top of the file, insert a StringInput command that asks the user for the customer's account status. For example,

```
StringInput "Please enter account status", AccountStatus
```

When you print your letter, the formatter automatically displays the StringInput message and waits for a response before formatting the file. Once you type one of the options (*Tardy, Overdue, Delinquent,* or *Okay*), the formatter decides which paragraph to print at the end of the form letter. This means you don't have to remember to change the string assignment each time a client's account status changes; the formatter prompts you for account status, and prints the appropriate closing paragraph each time you print.

# Char

Keystrokes

*Alt-S* (or *F10,* Style), **O**ther Format, and type `Char` *`number`*

Function

Prints the ASCII character designated by the number entered in the command text.

This command lets you print a character by specifying its ASCII-equivalent *number.* For example, the decimal number for the letter *A* is 65. If you type the command `Char 65`, the formatter prints the letter *A.*

This command is useful when your printer supports foreign character fonts, a mathematical symbol set, or special characters that don't show on your screen but are supported by your printer. (If your printer doesn't have this capability, you can create a facsimile of some types of special characters with the *O* (Overprint) command.)

How To

If you want to insert a special character that your printer can print but isn't on your keyboard, you need to first refer to your printer manual to find the ASCII character number that corresponds with the character you want to print. Generally, the numbers associated with special fonts are in the higher number range, 128 to 255.

Once you know the decimal value for the ASCII character, choose **O**ther Format from the **S**tyle menu and type `Char`, followed by that number.

```
The symbol CHAR 189 stands for per thousand.
```

If you print this example on an Apple LaserWriter, you get the following results:

```
The symbol ‰ stands for per thousand.
```

The decimal equivalent of the ASCII character ‰ is 189.

Tips

You can use this command in conjunction with the formatter command Set. This allows you to set a word equal to the decimal number of an ASCII character. This is useful because you often cannot get an accurate display onscreen of these special characters.

To easily tell when you're going to print this special character in your file, choose **O**ther Format at the top of your file and type the following command:

```
SET perthou = 189
```

Then, when you want to include the per thousand symbol as part of your text, type the word *perthou* in place of the variable *number*:

```
CHAR perthou
```

The printed result, if your printer has this capability, would be ‰.

**Note:** The Set command can appear anywhere in your document, so long as it comes before a reference to the text you're *setting*. We recommend, though, that you enter all Set commands near the top of your file.

You can also use the Char command in conjunction with special fonts that your printer might have. You can change fonts using the **T**ypestyle/**F**ont command. For example, the LaserWriter Plus supports a font called

"Dingbats." If you enter the command `CHAR 189` while you are in that font, you'll end up behind the 8-ball—🎱.

You can use decimal numbers, hexadecimal numbers, octal numbers, or binary numbers with the Char command. Sprint assumes the number is in decimal notation unless you end it with the letter *h* (for hex notation), *o* (for octal notation), or *b* (for binary notation). For example, the following three commands are equivalent:

```
CHAR 208
CHAR 0d0h
CHAR 320o
```

All of these commands access the em dash (—) in many PostScript fonts.

Another way of embedding specific character codes into your file is to press *Alt* while typing the decimal number on the numeric keypad (*NumLock* does not have to be on). When you release the keys, a character will appear there—but it might not look like the character that your *printer* has at that decimal location. (Note that if you have SuperKey loaded, you may have to press *Shift-Alt* in order to type ASCII decimal numbers on the keypad.)

# Closing

Keystrokes — Choose *Alt-S* (or *F10*, Style), **O**ther Format, and type `Closing`

Function — Formats text for creating complimentary closings in letters. The format is exactly the same as Address.

Closing sets the left margin at the center of the page, does not wordwrap lines, and inserts two blank lines above and below the closing.

You would usually use this format in conjunction with the Address format.

You will find this command defined in the Sprint file STANDARD.FMT.

How To

Type the text of the closing, select it, choose **Other Format** from the **Style** menu, type `Closing`, and press *Enter.*

# Column

Keystrokes

Choose *Alt-S* (or *F10,* **Style**), **Other Format**, and type `Column`

Function

Sets up a format to print parallel (not snaking) columns.

You can use this command to set up side-by-side columns of text. The text of one column does not automatically flow to the next, as it does when you use the **Snaking Columns** command from the **Column** menu.

The lines that come after this command start printing at the exact same spot as the lines in it. For this reason, you always need to set a new left indent after the Column command. The text that follows the Column command is the text of the parallel column.

The following example illustrates how you could use the Column command to create a "drop-cap" (often used as the first letter of book chapters) in your Sprint document.

```
                    Frank Borland: A Biography

BEGIN COLUMN, linelength 2.1 picas, font helvetica,
size 48 points
F
END COLUMN
STYLE leftindent 2.5 picas, linelength 25.5 picas
rank Borland is more mystique than mystic. Even at
Borland Intl., his namesake, few people have ever
seen him. The old-timers recognize him for his
remarkable algorithms--still the fastest in the West.

STYLE leftindent 0 picas, linelength 28 picas
Borland lives deep in the Santa Cruz mountains with
his transportable computer, his burro, and his dogs.
Although he doesn't have a permanent homestead, he
keeps a couple of semi-permanent camps deep in the
redwood groves, where the sun's glare rarely reflects
on his monitor. He'll occasionally drop into Scotts
Valley for provisions, but he communicates with the
rest of the world only rarely, and mainly by modem.
```

Here is how this would print out:

Frank Borland: A Biography

Frank Borland is more mystique than mystic. Even at Borland Intl., his namesake, few people have ever seen him. The old-timers recognize him for his remarkable algorithms—still the fastest in the West.

Borland lives deep in the Santa Cruz mountains with his transportable computer, his burro, and his dogs. Although he doesn't have a permanent homestead, he keeps a couple of semi-permanent camps deep in the redwood groves, where the sun's glare rarely reflects on his monitor. He'll occasionally drop into Scotts Valley for provisions, but he communicates with the rest of the world only rarely, and mainly by modem.

Note that it is up to you to explicitly tell Sprint where the columns start and stop. As shown in this example, typically you have to use the Style command three times to set and reset the *LeftIndent* and *LineLength* parameters: first for the text of the first parallel column (which in this case is only a single letter), next for the text of the second parallel column, and finally for the text that returns to normal single-column format.

# Display

Keystrokes

Choose *Alt-S* (or *F10*, Style), **O**ther Format, type `Display`, and press *R*

Function

Sets marked text apart from the body of the text.

This command adds a half inch to the left margin and inserts one blank line above and below the text. You have to press *Enter* whenever you want a line to break when within this format.

This command is useful when you have a line or block of text that you wish to set apart from the main text, and that does not naturally fall into wrapped paragraphs (compare the Quotation command entry).

You will find this command defined in the Sprint file STANDARD.FMT.

How To

Select the text you want to offset, choose **S**tyle/**O**ther Format, type `Display`, and press *Enter*.

```
BEGIN DISPLAY
As always, if you have any trouble,Enter
feel free to write or call.Enter
END DISPLAY
```

prints as:

> As always, if you have any trouble,
> feel free to write or call.

Tips

If you want the same, offset effect, but want the text to print in a fixed-width font, use the Example format. If you want the formatter to indent both the left *and* right margins and wordwrap your lines, use the Quotation format.

# Escape

Keystrokes

Choose *Alt-S* (or *F10*, Style), Other Format, type Escape, and press *C*

Function

Sends any sequence of characters to the printer without being interpreted.

This advanced command lets you send raw, unchanged data (any sequence of characters or an entire file) to the selected printer. This means you can use the formatter to print *anything* your printer can print.

**Caution:** This command is extremely printer-dependent and changes the control Sprint has over the printer. *We strongly discourage you from using this command;* look for other existing commands to perform the function you're attempting.

If you choose to use this command, be aware of the following problems associated with the Escape command:

- The formatter doesn't know *anything* about what the command string does; it just sends it to the printer. You can describe the string somewhat, telling the formatter how wide and tall the resulting graphic output is, and the movement of the print head, but the formatter "takes you at your word"—it can't confirm or deny this information.
- This command violates the control the formatter has over the printer. Formatting commands entered after the Escape command may not work properly. Once the string causes the formatter to do something (like change fonts, set an attribute, or move the paper backwards), the subsequent output may not be as expected. Likewise, any new printer definitions may not work properly. You can avoid this problem if you always restore the printer's original state once the Escape text has been sent.

How To

Because Escape does not affect blocks of your document's text, you need to press *C* (not *R* for a Region) after you enter the command. Escape does require text after it, however, as illustrated in the discussion that follows.

In order to enter this text, you need to type it after the name of the command on the status-line prompt.

If there is not enough room in the status line, enter just the name, press *C*, and then press *Alt-Z* to look at the usually hidden control codes. The Escape command Sprint inserts into your document looks like this:

```
^OESCAPE^N
```

Cursor to the left of the ^N code and type a space and then the full text of the Escape command. The text you type will be highlighted when you return to normal display by pressing *Alt-Z* again. Cursor to the right of the ^N to return to your normal text.

You can include the following fields in your Escape command. Note that the field names can be abbreviated to one character.

***Send (s)*** Followed by a quoted string. This string is sent to the printer. Ideally, the formatter will output the string exactly, but there are some limitations and guidelines:

- Variable reference, @Value, Char, and @Char commands will be performed.
- Any other formatter commands or @-sign commands included in the string will be sent as is, resulting most likely in garbage.
- Use the Insert/Non-Breaking Space or @_ command to insert multiple spaces (more than one in a row).
- Use CHAR 9 or @Char(9) to generate a tab character; don't use the *Tab* key or the @/ command to send tabs.
- Newlines are sent as CR (carriage return), LF (line feed).

For example, if your printer has a bell and you want to make the bell ring, you might try typing this:

```
ESCAPE s = "@char(7)"
```

***File (f)*** Followed by a quoted string that names a file. The contents of the named file will be sent to the printer *exactly*; however, a single

trailing ^Z, if it exists, will be removed. This field provides more exact control over the sent codes than does the *Send* field; you can also use *File* to dump graphic output from other programs. For example,

```
ESCAPE f="C:\QUATTRO\GRAPH.OUT", d=47 u
```

***Width (w)*** Followed by a dimension. This is a width for escape sequences that print some small graphic (such as a specially constructed character). The formatter assumes that the sent string moves the print head to the right by this amount. The Escape command will then act just like a character of the specified width. For example,

```
ESCAPE s="^[Y@char(4)@char(1)@char(2)
          @char(3)@char(4)", w=4 u}
```

The special value –1 indicates a zero-width escape sequence that moves the head to an unknown position horizontally; the printer driver repositions the print head after completing the command. The default width is 0.

**Note:** Sprint doesn't verify the existence or readability of the file until the printing pass, so if there are errors, you may waste a lot of time.

***Height (h)*** If your graphic is taller than the size of *normal* text, you can specify a dimension here for how high it is *above* the baseline. The formatter will move the current line the specified distance away from the previous line to leave room for the character.

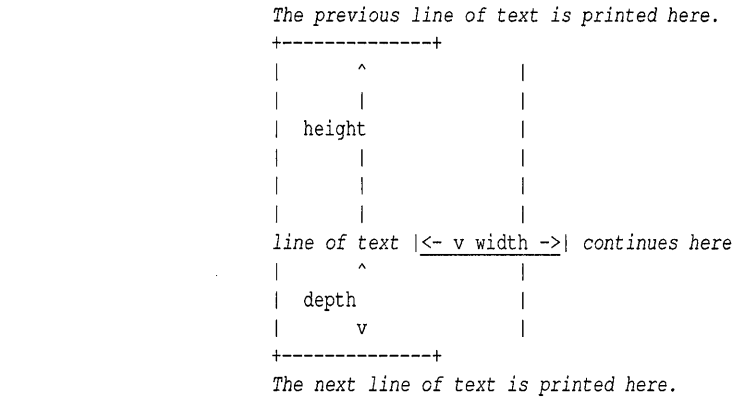
***Depth (d)*** Specifies how far the graphic character extends below the baseline. This is used for large, tall graphics, such as graphs output by other programs. The formatter assumes that when Escape sends the sequence, it also moves the print head *down* the page this amount and places the next line below it. Be sure this value is exact, or the printout may drift across page boundaries.

Refer to Table 1.2 on page 58 for a list of acceptable dimensions.

Tips

If your printer supports a page-description language (like PostScript), you can use Escape to embed commands for special effects (like printing in gray instead of black). In fact, the commands in the Style/PostScript menu all depend on extensive use of Escape in their execution. The following figure will help illustrate how the Escape dimensions affect your printout.

Result of the Escape between the words "text" and "continues":




# Example

Keystrokes

Choose *Alt-S* (or *F10*, Style), Other Format, and type `Example`

Function

Indents marked text and prints it in a typewriter (fixed-width) font.

The Example format adds a half inch to the left margin and inserts one blank line above and below the text. You have to press *Enter* wherever you want a line to break when you're working in this format. It is the same as the Display format, with one exception: Example prints the text of the example in a typewriter (fixed-width, like pica or elite) font, if your printer has this capability. The Example format is useful for approximating the look of a computer printout.

You will find this command defined in the Sprint file STANDARD.FMT.

How To

Select the text, choose **Style/Other Format**, type `Example`, and press *Enter*.

```
BEGIN EXAMPLE
As always, if you have any trouble, Enter
feel free to write or call. Enter
END EXAMPLE
```

prints as:

```
As always, if you have any trouble,
feel free to write or call.
```

Tips

If you want the same, offset effect, but want the text to print in your default type, use the Display format. If you want the formatter to indent both the left *and* right margins and wordwrap your lines, use the Quotation format.

# HaveSpace

Keystrokes

Choose *Alt-S* (or *F10*, Style), **Other Format**, type `HaveSpace` *`dimension, yes/no text`*, and press *C*

Function

Formats a page based on space remaining on the page.

The HaveSpace command causes Sprint to make a formatting decision based on the amount of space left on the page. Like the Case command, Sprint performs case *Y* if the amount of space you specify (or more) is available on the current page; otherwise, Sprint performs case *N*. You do not have to specify both cases; you can give an instruction for either *Y* or *N*, and the other case will simply leave the text unchanged.

This command appears in STANDARD.FMT as a component of the NeedSpace command.

How To

Type the HaveSpace command, enter the dimension and the yes/no text right after it on the status-line prompt, and type `C`. For instance,

```
HAVESPACE 3 inches, n "Continued on next page @NewPage"
```

This command says, "If there are 3 inches of space left on the page when this appears, do nothing; if there are fewer than 3 inches when this text appears, print the message:

```
Continued on next page
```

and continue printing the text on the next page."

If there is not enough room on the status line to print all your yes/no text, enter just the command name, press *C*, and then press *Alt-Z* to look at the usually hidden control codes. The HaveSpace command that Sprint inserts into your document will look like this:

```
^OHAVESPACE^N
```

Cursor to the left of the ^N code and type a space and then the full text of the HaveSpace command. The text you type will be highlighted when you return to normal display by pressing *Alt-Z* again. Cursor to the right of the ^N to return to your normal text.

Tips

When the formatter encounters this command, it counts the number of lines that will fit on the page (and divides this number by 2, if in a two-column format) to figure out the amount of space left on the formatted-and-to-be-printed page. If you place the HaveSpace command in the middle of a paragraph, Sprint does not count the current line, and hence Sprint's decision will be based on an estimate of the amount of space left. Therefore, we recommend that you place this command *between paragraphs only.*

# Hsp

Keystrokes

Choose *Alt-S* (or *F10*, **S**tyle), **O**ther Format, type `Hsp` *`dimension`*, and press *C*

Function

Moves the print head a specified horizontal distance.

The Hsp command moves the print head a specified distance horizonally. You can use this command to insert space between words on a line (to paste in special symbols, for example).

How To

Choose **O**ther Format from the **S**tyle menu and enter `Hsp` followed by a specified amount of space. Then press *C*. You can use any valid Sprint dimension (inches, points, characters, and so on). For example:

```
HAVESPACE 3 inches, n "Continued on next page @NewPage"
```

results in

Our new logo (  ) adds dignity to our image.

Tips

A *backward* distance (specified by a negative number) cannot be larger than the preceding word.

You can use any valid Sprint dimension (inches, points, characters, and so on).

See also the entry on the Tab command in this chapter.

# Include

Keystrokes

Choose *Alt-S* (or *F10,* **S**tyle), **O**ther Format, type `Include` *`filename`*, and press *C*

Function

Includes the specified file during formatting and printing.

This command allows you to combine multiple files during formatting. When the formatter sees an Include command, it looks for the specified file, reads it, and then formats and prints it. The formatter then returns to the file containing the Include command, and continues formatting and printing.

How To

Create a *master* file, which serves as an outline of sorts. Choose all **H**eadings commands within this file, and type the titles for your chapters, sections, subsections, and so on. If you'll be cross-referencing these table of contents entries, enter your **D**efine a Tag commands in this master file, too. Also enter your **H**eader and **F**ooter commands and, if you're going to use a **S**tyle command, enter it at the top of this master file.

Create a different Sprint file for each chapter or section. If your chapters are lengthy, you can create different files for each of your major sections or subsections. If you're not using **H**eadings commands, divide your

document into files that address a particular topic or area of discussion.

Within the master file, choose **O**ther Format on a blank line. When Sprint prompts for the Formatter command, type `Include` followed by the name of the first file you want to include (merge and print). Press *C* when Sprint prompts you. Enter the Include command wherever you want a file inserted. This command typically follows a **H**eadings command.

When you want to print your entire document, enter your **P**rint command from the master file. If you're printing from the DOS command line, enter the master file name as part of your SPFMT command. For example,

```
SPFMT MASTER.SPR
```

Tips

This command is extremely useful when creating and printing large documents. Let's say your document is 100 pages. If you type all the text in a single file, editing can be slowed down. It's better to split your document into separate files (say, by chapters), edit them separately, and merge them only at print time.

To do this, you might want to create a master file, which could contain an optional Style command at the top, your **H**eader and **F**ooter commands, and any **D**efine Text Variable strings you may have. Following this information, you could type a series of Include commands, which instruct the formatter to format and print your smaller files. Type the Include commands in the order in which you want your files to print.

If you want to include a file on a different disk or in a different directory, type the drive letter and/or directory name(s) before the file name. For example,

```
INCLUDE A:\FINANCE\LETTERS\PASTDUE.SPR
```

If the formatter can't find the specified file, it looks in the directory where your master file is located; if it doesn't find the file there, it searches the directories listed in your DOS PATH command.

If you begin a format command in a file, you must end it within the same file. For example, if you begin a **H**yphens format in one file, you must end it before

including another file in the format/print operation. Formatting commands cannot be stretched across files.

Although it's possible to nest up to six levels of Include commands (that is, include a file within a file that's included by another file, which is included by another file, which is included by another file, and so on), we strongly encourage you not to nest Include commands. In other words, *avoid entering Include commands within files that are included by other files.* The reason for this is basic housekeeping. If you can see all of the files you're including by viewing a single file, it's a lot easier to find information when you're editing from a printed copy. You can easily tell which file contains the information you're looking for. If you "bury" Include commands within "included" files, you can't tell where your information is without opening multiple files and searching through each.

# Incr

Keystrokes

Choose *Alt-S* (or *F10*, Style), **O**ther Format, type `Incr variable`, and press *C*

Function

Increments the specified variable and sets *SectionNumber* equal to the incremented value.

This command increments the value of a variable by one and makes the value of the variable *SectionNumber* equal to the value of the specified variable. Once you enter the Incr command, you can use the Label command to print the variable's incremented value.

The variable name *SectionNumber* can be somewhat misleading in that *SectionNumber* actually contains the value of the current *level* (either the current chapter, Section, Subsection, or Paragraph number). For example, if Sprint's formatting text within a Subsection, the value of the variable *SectionNumber* is actually the number of the Subsection head; likewise, if Sprint's formatting text and encounters a Chapter command, the value of *SectionNumber* is actually the current chapter number.

How To

Use the Incr command to force the formatter to add one to a specific variable. Choose **S**tyle/**O**ther Format, enter `Incr` followed by a variable, and press *C*. For example,

```
Incr chapter
```

tells the formatter to increment the current chapter number by 1. If the current chapter number is 3 and you enter this command, the formatter will number the chapter 4.

Tips

This command works similar to the Set command, in that it lets you change the value of a variable. With the Incr command, however, you don't specify the value. Instead, Incr adds 1 to the current value. If you want to assign a specific value to a variable, use the Set command.

# Justify

Keystrokes

Choose *Alt-S* (or *F10*, **S**tyle), **O**ther Format, type `Justify`, and press *C*

Function

Forces the text on the line preceding this command to extend from margin to margin, inserting as much space between characters as necessary.

This command spaces out the selected text so that is stretches from the current left margin to the current right margin.

How To

Justify must follow the text you want spread out. Typically, you would use this with a single line of text followed by Justify and then a hard return. You can, however, also use the command with the last line of a paragraph. Place the cursor after the last character of the line or paragraph, choose **O**ther Format, and enter `Justify`. Press *C* when prompted.

You can also use the Justify command with the words *left*, *right*, and *center* to justify a single line (or the last line of a paragraph) in a special way.

For an example of the effect, look at the chapter heads in this manual. The results are exactly the same as inserting wide spaces between each character of the line. For example,

F O R E W O R D

# Kern

Keystrokes

Choose *Alt-S* (or *F10*, Style), **O**ther Format, type `Kern` *dimension*, and press *C*

Function

Squeezes characters together to create special typographic effects.

This command specifies the amount of squeezing between two characters. You can use any of the dimensions listed in Table 1.2 on page 58, but the dimension cannot exceed the width of the character to the left of the command.

Some commonly needed kerning pairs are already defined for PostScript printers in the POSTSCR.TCT file. You can add your own kerning pairs to this file if you need to.

How To

From the **S**tyle menu, choose **O**ther Format and enter `Kern`, followed by the distance you want between the preceding character, and the character following the command. Press *C* when prompted. For example, certain letter combinations can leave too much space between them, especially in larger point sizes.

This sign:

```
TKERN.1 emoday Only: BLKERN.1 emTKERN.1 ems on
TKERN.1 emoast
```

prints like this:

Today Only: BLTs on Toast

Here's the difference between the kerned and the unkerned versions in a larger point size:

Today Only: BLTs on Toast

Tips

As with many of the advanced formatting commands, the results of this command will depend on the capabilities of your printer.

When defining kerning pairs, you should use the *em* dimension so that the kerning amount will be correct regardless of the point size used.

## Label

Keystrokes

Choose *Alt-S* (or *F10*, Style), Other Format, type `Label TagName`, and press *C*

Function

Sets tags to the value of the *SectionNumber* variable.

This command is used within a Numbered format to tag the current *section* number. It is functionally the same as choosing Define a Tag and then making the tag name equal to *SectionNumber*. To fully understand what Label does, you need to be familiar with a couple of concepts:

- The Define a Tag command automatically makes the specified tag equal to the variable *SectionNumber*. If you choose Define a Tag and type `install`, any reference to the *install* tag would cause the formatter to print the number of the sectioning command affecting the tagged text. For example, if the tagged text is in Chapter 3, Section 3.2, the formatter prints 3.2 wherever it sees a reference to *install*.
- Within a Numbered format, however, tags are *not* set to the section number; they are set equal to the current counter (typically the paragraph number). This allows you to tag items in a list and then reference these items by number.
- The Label command is used within a Numbered format when you want to refer to the current section number, rather than the current counter. For example, you might have a list of installation instructions and want to refer to the section containing this list later in

the document. You can't refer to any of the tags you set within the list because that causes the formatter to print the value of the counter (the number of the paragraph containing the tag name you set). You want to refer to the *section* containing the list, so you'd use Label to set the tag and then reference the tag name specified in the Label command.

- Of course, you could choose **D**efine a Tag directly after your Section command (not within the Numbered format), and the tag would be equal to the section number. The Label command, however, has an advantage over **D**efine a Tag: If you move the list to a different section, the *label* (since it's typed within the Numbered format) stays with the list text; when you reference the tag name specified in a Label command, the formatter prints the value of the section containing the list. If you use **D**efine a Tag to tag the section number and later move the list to another section, you can't reference the tag name you set, since the text is no longer in the tagged section.

How To

To tag a section number within an **N**umbered format, choose **O**ther Format from the **S**tyle menu, type `Label` *`tagname`*, and press *C*. Use the **R**eference a Tag command to reference the assigned number of the section that contains the formatted text.

Tips

The variable name *SectionNumber* is somewhat misleading in that *SectionNumber* actually contains the value of the current *division* (either the current chapter, Section, Subsection, or Paragraph number). For example, if you're formatting text within a Subsection, the value of the variable *SectionNumber* is actually the number of the Subsection head.

For example, let's say you're writing a report and create a list of recommendations in the *Executive Summary* section. The report might look like this:

```
SECTION Executive Summary
We propose the following recommendations:TAG recsection

BEGIN NUMBERED
Hire six to eight additional managers.LABEL hiringneeds

Offer mandatory management training classes (see Item
humanresdept).

Work with the Human Resources department to improve
staff training.TAG humanresdept

Increase the percentage of profit-sharing
contributions.TAG anitem

Eliminate all dead weight (no names please).
END NUMBERED

SECTION Problem Assessment
For a list of recommendations, see Section recsection.
For a list of hiring needs, see Section hiringneeds.
For referencing a particular item, see anitem
```

The example prints like this:

# 1 Executive Summary

We propose the following recommendations:

1. Hire six to eight additional managers.
2. Offer mandatory management training classes (see Item 3).
3. Work with the Human Resources department to improve staff training.
4. Increase the percentage of profit-sharing contributions.
5. Eliminate all dead weight (no names please).

# 2 Problem Assessment

For a list of recommendations, see Section 1.
For a list of hiring needs, see Section 1.
For referencing a particular item, see 4.

If you should move the list of recommendations to another section, the reference in Section 2 will still be accurate. This is because the Label *tagname* stays with the text you're referencing and prints the *current* section number.

# MakeTOC

Keystrokes — Choose *Alt-S* (or *F10*, Style), **Other Format**, type `MakeTOC`, and press *C*

Function — Tells the formatter to produce a table of contents.

This command is used when you want *unnumbered* headings in a document but also want Sprint to automatically produce a table of contents. *Numbered* headings (**Level1n**, **Level2n**, and so on) automatically generate a table of contents, so you don't need to insert this command in a file that has at least one numbered heading. *Unnumbered* headings do not generate a table of contents without the formatter command MakeTOC.

How To — To produce a table of contents that includes unnumbered headings, move the cursor to the top of the file, choose **Style/Other Format**, type `MakeTOC`, and press *C*.

If you use a numbered heading (like **Chapter**), Sprint automatically issues a MakeTOC command so that you can mix numbered and unnumbered heads and have them all appear in the table of contents. Therefore, you need to use the MakeTOC command if you have unnumbered heads *only*.

# Message

Keystrokes — Choose *Alt-S* (or *F10*, Style), **Other Format**, type `Message` *`text of message`*, and press *C*

Function — Displays onscreen messages while Sprint formats your document.

This command lets you type the text of a message you want displayed during formatting. You might use this command to monitor the progress of the formatting, or to remind yourself to paste-up a particular figure after the document prints. The message appears onscreen only; it does not print.

How To — To display a message during formatting, use the formatter command Message, type the text of your message, and press *C* at the prompt. For example, if you want to display a reminder that tells you to paste-up a

figure, move the cursor to a blank line within the figure format and enter the Message command. Like this:

```
BEGIN FIGURE
RESERVE .25 PAGE
CAPTION Sample Figure
MESSAGE Be sure to paste up original art on this page
END FIGURE
```

When Sprint formats the page containing these commands, it will display the text of your message on the screen. The message *will not* appear on the printed page.

You can use the **Variable** command to reference a variable in your message. For example, you might want to include a figure or page number in a reminder to paste up a particular figure. Enter the formatter command Message and, when you want to reference a variable, choose the **Variable** command from the **Insert** menu. For example,

```
MESSAGE Paste up original art on pg. PAGE, t="%d"
```

Note that the boldface text on this message line actually refers to the variable *Page*, which will print onscreen as a number.

This sample message reminds you to paste up original art on whatever page the figure will appear. The formatter determines the page number and adds it in your message. For example,

```
Paste up original art on pg. 3
```

Your message remains onscreen throughout formatting and printing.

Tips

If you want to save messages to a .LOG file, don't use the Message command. Instead use the Warn command, which is covered in the "Style Sheet Commands" appendix in the *Advanced User's Guide*.

# Modify

Keystrokes

Choose *Alt-S* (or *F10*, **Style**), **Other Format**, type `Modify formatname, new parameters`, and press *C*

Function

Changes the effect of a predefined format.

There are a number of ways to change the printed results of formats—the most convenient of which is to choose **Style/Modify**, as described in the entry called "Modifying Formats" in Chapter 1. But using the menu command modifies just particular occurrences of a format, not all occurrences of it. This is where the Modify command comes in.

How To

Because the Modify command makes a *global* change to a format, you have to enter the command before the first occurrence of the command you're changing. For this reason, you should enter all Modify commands near the top of your document. Enter the format name and the modifications you want to invoke. Parameters must be separated with commas.

Here's a typical example of what you would type in the status line prompt when you choose **Other Format**:

```
Modify Description, font Helvetica, indent -8 picas
```

When this line appears at the top of your document, every time you choose **Style/Lists/Description**, the text in the Description format prints in Helvetica (instead of your default font), and the first line of paragraphs begin printing 8 picas to the left of the current left margin.

For a full list of the parameters you can use with the Modify command, see Table 1.8 on page 149 in Chapter 1.

Tips

If you want to change all occurrences of a format in *all* your files (not just one or two), you should use the Modify command in the .FMT file you're using. (In this case, you would use the @-sign version of the command and enclose the format name and parameters in delimiters.)

If you want to add more parameters than will fit on the status line, type in what does fit, press *Enter*, and then press *C*. Now cursor within the command that Sprint inserts in your document and add the rest of the parameters (with commas in between each). It's OK to have Sprint wordwrap a long Modify command, but you should not press *Enter* until the command is over.

# NeedSpace

Keystrokes — *Alt-S* (or *F10*, Style), **Other Format**, type `NeedSpace` *dimension*, and press *C*

Function — Formats a page based on space remaining on the page.

The NeedSpace command specifies the amount of space that must be available before the formatter can print text following the command. For example, a NeedSpace command of 3 inches instructs the formatter to print text on the current page if there are at least 3 inches of blank space remaining on the page. If the formatter determines that there is less than the required amount of space, it breaks the current page and begins printing text on the following page.

The NeedSpace command is similar to the Group format in that it keeps text together on the page. The NeedSpace command specifies the amount of required blank space; the Group format sets off the text that should be kept together.

You will find this command defined in the Sprint file STANDARD.FMT.

How To — Enter this command along with a dimension (for example, 12 lines). Sprint then inserts in your document the command and dimension. For example,

```
NEEDSPACE 12 lines
```

This command states: If there are 12 lines left on the page when this command appears, print the text following the command; if there are fewer than 12 lines remaining when this command appears, end the current page and begin printing on the next page.

# NoTCT

Refer to the TCT entry in this chapter for details.

# NoteChapter

Keystrokes — Choose *Alt-S* (or *F10*, Style), **O**ther Format, type `NoteChapter`, and press *C*

Function — Prints the title of the current chapter in the endnotes.

This command tells the formatter to insert the number and title of the current chapter in the endnotes. If you type the NoteChapter command after each Chapter command in your file, the endnotes page will be organized by chapter; the formatter prints the number and title of the first chapter, lists the endnotes referenced in that chapter, prints the next chapter number and title, lists its endnotes, and so on.

This command also resets the *Footnote* variable to 0, so that footnotes and endnotes start numbering from 1 in each chapter.

How To — If you want to print the chapter number and title in your endnotes, insert the NoteChapter command after each Chapter command in your file. Choose **S**tyle/**O**ther Format, enter the NoteChapter command, and press *C* at the prompt. Here's what your screen might look like:

```
CHAPTER InstallationNoteChapter
```

When the formatter prints the endnotes page, it will include the chapter number and the title *Installation*, followed by the endnotes referenced in the *Installation* chapter.

Tips — If you also want to print section numbers and titles in your endnotes, use the NoteSection command. If you want the endnotes to appear somewhere other than at the end of your document, use the Place command.

# NoteSection

Keystrokes — Choose *Alt-S* (or *F10*, Style), **O**ther Format, type `NoteSection`, and press *C*

Function — Prints the title of the current section in the endnotes.

This command tells the formatter to insert the number and title of the current section in the endnotes. If you

type the NoteSection command after each Section command in your file, the endnotes page will be organized by section; the formatter prints the number and title of the first section, lists the endnotes referenced in that section, prints the next section number and title, lists its endnotes, and so on.

This command also resets the *Footnote* variable to 0.

How To

If you want to print the section number and title in your endnotes, enter the NoteSection command after each Section command in your file and press *C* when prompted. For example,

```
SECTION InstallationNoteSection
```

When the formatter prints the endnotes page, it will include the section number and the title *Installation*, followed by the endnotes referenced in the *Installation* section.

Tips

If you also want to print chapter numbers and titles in your endnotes, use the NoteChapter command. If you want the endnotes to appear somewhere other than at the end of your document, use the Place command.

# O and Ovp

Keystrokes

Choose *Alt-S* (or *F10*, Style), **O**ther Format, type `O` or `Ovp`, and press *C*

Function

Overprints one character on another to create special effects and accented letters.

The difference between the O and the Ovp command is that the former overprints only single letters and centers them, while the latter can overprint whole words starting at the left edge of the first letter.

Your printer must be able to backspace in order to use these commands.

Tips

Some trial and error is inevitably called for when overprinting characters. Here are a few examples of special characters you can create with most printers:

| To print this: | Enter this: |
|---|---|
| ÷ | O(:-) |
| ± | O(+_) |
| ñ | O(@+[~]n) |
| $^{2}/_{3}$ | Ovp(@+[2]) /@-(3) |

# Place

Keystrokes

Choose *Alt-S* (or *F10*, Style), **O**ther Format, type `Place`, and press *C*

Function

Prints text that would normally print at the end of the document.

This command is used to position notes that would normally print at the end of your document. For example, Sprint stores endnotes entries and inserts them at the end of your document. You can use the Place command to change that.

How To

To change where the formatter prints your endnotes use the Place command. For example, if you want the formatter to print your Endnotes at the end of each chapter instead of at the end of your document, insert the Place command before each new Chapter command. Choose Style/**O**ther Format and type

```
Place Notes
```

Respond to the next prompt by pressing *C*.

This command prints the endnotes accumulated so far (since the beginning of the document or since the last Place command). If you place the command at the end of each chapter (or add it to the @Chapter command in the STANDARD.FMT file), the collected notes for each chapter will print at the end of each chapter.

There must be some endnotes preceding the Place command for it to work.

**Note:** The Place command will affect any command that uses the *After* parameter in its definition. If you define a command that uses *After*, Place will affect text governed by your new command as well.

# Quotation

Keystrokes

Choose *Alt-S* (or *F10*, Style), **O**ther Format, and type `Quotation`

Function

Formats a block of text as a quotation set off from the surrounding text.

This command prints text in a justified, single-spaced block with wider margins, separated from the surrounding paragraphs by a blank line. As defined in STANDARD.FMT, this format automatically wordwraps paragraphs (ignoring single hard returns), and expands both the left and right margins by .5 inch each. This is the standard way of quoting a large piece of text from a book.

You will find this command defined in the Sprint file STANDARD.FMT.

How To

Mark the text of your quotation, choose **S**tyle/**O**ther Format and type `Quotation`. Press *R*, and Sprint inserts the Begin and End commands. If you haven't already typed the text to be affected, type `Quotation` and press *B* when prompted for Begin command. Then begin the text of the quotation. When you're done typing the text, type `Quotation` and press *E* for End command. For example:

```
He spent much of his time on the campaign trail with
the words that promised to please everyone:

BEGIN QUOTATION
...and, throughout my days in office, I have served
with the people, and that's people with a capital
'P,' and their values in mind. In this great country,
there is a place for everyone's values!
END QUOTATION

And the media and the public loved it.
```

The result:

He spent much of his time on the campaign trail with the words that promised to please everyone:

> ...and, throughout my days in office, I have served with the people, and that's people with a capital 'P,' and their values in mind. In this great country, there is a place for everyone's values!

And the media and the public loved it.

## Scale

Keystrokes — Choose *Alt-S* (or *F10*, Style), **O**ther Format, type `Scale n`, and press *C*

Function — Causes all dimensions of a document to be enlarged by the given factor of *n* (or reduced, if *n* is less than 1).

How To — You have to insert the Scale command before the first ruler line, any Style commands, or the text of your document. The scaling number must be a decimal number greater than 0.

Tips — This command is designed for documents that will be later photo-reduced to their final form. In other words, you could improve the resolution of your printer by producing your documents at an enlarged scale and then photo-reducing them back to normal, thereby giving you greater dots per inch than your printer is actually capable of.

You can also use this command to produce "thumbnails" of individual pages by scaling to .5 (half sized), .25 (quarter sized), or even smaller.

## Set

Keystrokes — Choose *Alt-S* (or *F10*, Style), **O**ther Format, type `Set variable=n`, and press *C*

Function — Assigns a number to a numeric variable.

This command lets you assign a number to either a Sprint-defined variable or a variable that you create. As

with all variables, you use the **V**ariable command to print the value of the variable.

You must use the Set command to assign a value to a variable before you can reference it with the Variable command. If you don't, Sprint will print the current value of the variable, instead of the number you want.

How To

There are several ways to use Set. In the examples that follow, a word set in angle brackets represents a variable name, for example, *<variable>* could be replaced by *page, month, chapter,* and so on. To enter a Set command, choose **S**tyle/**O**ther Format, type `Set` and the text of your Set command. Press *C* to implement the command.

To assign a specific value to a variable, set the variable equal to the number you want. For example, choose **O**ther Format and type

```
SET page = 123
```

This sets the current page number to page 123. You might want to use this form of the Set command if you're printing several files to create a single document, and you're not using a "master" file with Include commands. Place the *Set page=n* command at the top of the second and subsequent files and make *n* equal to the last page number printed in the previous file. For example, if FILE1.SPR completes printing after page 12, choose **O**ther Format and type the following command at the top of FILE2.SPR:

```
SET page = 12
```

If you want to increment the value of the variable by a positive or negative digit, type the + or - symbol before the number to be incremented or decremented, respectively. For example, choose **O**ther Format and type

```
SET chapter = +3
```

A good reason to increment a variable is to avoid having to know the value of a variable before you set it. For example, let's say that you're going to manually insert a page between two pages of your Sprint file (for example, between page 10 and page 11). If you want Sprint to account for the inserted page, you'd want the formatter

to increment the value of *page* on page 11. For example, after the text on page 10, enter the command `Set page = page+1`. The next page will be numbered one greater than it normally would be.

To make a defined variable equal to the value of another variable, or to create a new variable that's the same as another, use the Set command like this:

```
SET <variable1> = <variable2>
```

If you're creating *variable1*, *variable2* must already have a value. For example, choose Other Format and type

```
SET ThisPg = page
```

This example creates a new variable called *ThisPg* that is the same as the Sprint-defined variable *page*.

You can also add or subtract the value of one variable to the value of another, with the command form

```
SET <variable1> = + <variable2>
```

or

```
SET <variable1> = - <variable2>
```

The variables entered in your command must already be defined.

You can create a variable that's equal to another variable, plus or minus its current value, with the following form of the Set command:

```
SET <variable1> = <variable2> +/-n
```

If the variable whose value is assigned to another has a *parent* (see the Parent entry in this encyclopedia), then the new variable is also given a parent, which is *copied* from the original parent. There is no way to change the value of this *copied* parent.

Set can set any of the built-in counter variables (*Page*, *Footnote*, *Chapter*, *Section*, *Subsection*, *Paragraph*, *Appendix*, *AppendixSection*, and so on) to alter the normal numbering. The value of each of these counters is always the number of the *last* object it counts (0 if no object has been counted yet). For example, the value of *Footnote* is the number of the most recent footnote—the *next* footnote will be set to one number higher.

The following example shows three uses of Set: First we set the footnote number to 12, then add another variable to it, and then return it to whatever it orignally was. Note that whenever we change the value of the variable, the next footnote comes out *one higher.*

First, save the current footnote number in a new variable, *RealFoot*. Choose **O**ther Format and type

```
SET RealFoot = Footnote
```

Then set the *Footnote* variable to 11:

```
SET Footnote = 11

This is an example.FNOTE This will be footnote 12.
```

Now choose **O**ther Format and enter this command:

```
SET Footnote = Footnote+Page
```

The last footnote is 12, so the next one will be 12 plus the current page number.

```
This is an example. Don't panic, this is only an
example.FNOTE This footnote will have a rather high
number.
```

And now reset the variable to its original value (note that upper/lowercase doesn't matter):

```
SET Footnote = RealFoot
```

This results in:

This is an example.[12]

This is an example. Don't panic, this is only an example.[308]

Tips

You can create new variables by first naming them and then using the Set command. (**Note:** you must enter the variable correctly *before* entering Set. If an error is made, no error message will appear, you will just produce a new variable!)

12. This will be footnote 12.

308. This footnote will have a rather high number.

# StringInput

Keystrokes

Choose *Alt-S* (or *F10*, Style), Other Format, type `StringInput variable` or `StringInput "message", variable`, and press *C*

Function

Lets you enter a string during formatting.

This command tells Sprint that you want to enter text in a file *during formatting*. If you include a message as part of the StringInput command, the formatter will first prompt for the text you are to enter and then wait for you to enter it.

How To

There are two forms of the StringInput command:

- `StringInput variable`
- `StringInput "message", variable`

The first form collects input (whatever you type) from the terminal and assigns that text to the specified *variable*. The second form is similar, but prints the text of the quoted *message* before waiting for the input. For example:

```
STRINGINPUT "Please type your name", user
User was responsible for printing this document,
coordinating its production, and delivering the final
copy to the printer.
```

This example tells the formatter to prompt for a name during formatting. When the formatter reaches the line containing the StringInput command, it displays the message `Please type your name`. Once you enter your name, the formatter assigns your name to the variable *user*. When you reference the variable *user* (with the Variable command), the formatter replaces the variable reference with your name as typed from the keyboard during formatting. The formatter then continues to format the document as usual. If your name is Sue Jones, for example, the formatter prints:

> Sue Jones was responsible for printing this document, coordinating its production, and delivering the final copy to the printer.

Tips

The *message* referenced in this entry is *not* synonymous with the Message command. The Message command

prints a message during formatting but doesn't wait for or expect input. The message discussed as part of the StringInput command is a quoted string that asks the user to do something.

Sprint interprets a carriage return as the end of the *StringInput* text; therefore, input text *cannot* contain any hard returns (^M characters). Input text is limited to 80 characters maximum.

# Style

Keystrokes

Choose *Alt-S* (or *F10,* Style), **O**ther Format, type `Style` (followed by parameters listed in Table 1.8 on page 149), and press *C*

Function

Sets the parameters for a document's overall appearance.

This command specifies how your document, as a whole, should be formatted. It requires entry of at least one of the parameters listed in Table 1.8.

Since the Style command sets up the style and format of the rest of your document, it should be placed at the beginning of your document for global settings. To deviate from the format established with an initial Style command, you can mark a block of text and set a format command, modifying it as desired. You can also insert another Style command, remembering that the changes in it will affect the rest of the document. If an area doesn't lend itself to one of Sprint's commonly used formats (like Numbered, Hyphens, Display, and so on), select the "neutral" format Text, which is a "do-nothing" environment that accepts formatting parameters (see the Text entry for details).

Some parameters can *only* be changed with a Style command; these are parameters that apply throughout the document regardless of formats. For instance, the *Offset* parameter causes all pages to be offset from the left/right edge to allow for binding, and cannot be changed by a format. Table 1.8 on page 149 lists all possible parameters. A few of them are used only with Style (like *Comments, BottomMargin,* and *FormFeed*).

How To

The general form of the Style command is:

```
STYLE <parameter> <=> <value>,
      <parameter> <=> <value>, ...
```

where *parameter* is the built-in name of a formatting function (like *Offset*), and *value* is the value you assign to the parameter.

Note that Style is never used with Begin and End (that is, you never choose Region when prompted from the Other Format command).

There are two kinds of values: *dimensions* and *yes/no* values. A dimension is a distance or space to be allotted on a printed page, such as 2 lines or 1 character or 3 picas (see the Dimensions entry for a list of valid dimensions). A yes/no value is simply either *yes* or *no* (if neither is specified, the formatter assumes *yes*). You can also use 1 or 0 (where 1 equals yes, and 0 equals no.) Here are several examples of Style commands:

```
STYLE indent 5 chars

STYLE justify yes

STYLE fill

STYLE LeftMargin 2 picas, LineLength 35 picas,
      Indent 3 picas, TopMargin 9 picas,
      Spacing 1.2, Spread .6, Size 10 points,
      Justify, Fill, font Palatino
```

If you place a Style command in the STANDARD.FMT file, this command determines default settings—the settings that automatically take effect unless you specify otherwise—for *all* your documents. For instance, if you generally want your documents to be double-spaced, you can put the command

```
@Style(spacing = 2)
```

in STANDARD.FMT. Then every document prints out double-spaced, unless you add a command to a document that specifies single spacing. The same procedure will work for any of the Style parameters.

Note that some menu commands automatically insert a modified Style command into your document when you choose them. These commands are:

- Layout/Document-Wide/(all five margin commands)
- Layout/Document-Wide/Paper Size
- Layout/Document-Wide/Word Spacing
- Layout/Document-Wide/Inter-Paragraph Spread
- Layout/Page Breaks/Widow-Orphan Control

## Tab

Keystrokes — Choose *Alt-S* (or *F10*, Style), Other Format, type `Tab` *dimension*, and press *C*

Function — Moves the print head to the horizontal spot indicated by the dimension.

This command advances the print head to the spot indicated. Note that the movement is always to the right, which means that if the print head is already past the desired spot, Sprint creates a new line and starts from the left on the next line.

The difference between this command and Hsp is that Hsp always moves the print head relative to its current spot, while Tab moves it to an absolute position. See also the Hsp entry in this chapter.

## TabDivide

Keystrokes — Choose *Alt-S* (or *F10*, Style), Other Format, type `TabDivide` *n*, and press *C*

Function — Sets tab stops evenly across the line for a formatted table.

This command automatically creates *n* columns (maximum of 9) across the page, by setting *n*-1 tab stops. The first column begins at the current left margin. The formatter automatically places a gutter (an even amount of blank space) between each column.

How To — Choose this command when you want to set up a series of evenly divided columns defined with tab stops. For example, *TabDivide 3* creates three columns (2 tab stops); the first column begins at the left margin, followed by a gutter. The next column begins at a formatter-

determined tab stop, followed by a gutter, and the third column begins at the second formatter-determined tab stop. The text in the third column is roughly even with (but does not exceed) the right margin setting.

Tips

You should also set up tabs on your ruler line to approximate the look of the number of columns desired.

If you enter a TabDivide command within a format (like Table, Verbatim, and so on), the formatter will ignore any tabs set prior to the beginning of the format. Once you end the format, the formatter-set tabs will be cleared, and the previously set tabs will again be in effect.

Use TabDivide only if you absolutely need precisely divided columns. Use tabs on the ruler line or **Layout/Ruler/Precise Settings/Tab Stops** in all other cases.

# TagString

Keystrokes

Choose *Alt-S* (or *F10*, **Style**), **Other Format**, type `TagString tagname = "string"`, and press *C*

Function

Sets a tag to a string value.

Whenever you need to reference a bit of text associated with a variable, you can use the TagString command. Ordinarily, referencing a tag returns the number associated with the variable. (If a tag was not set equal to a variable, the default variable *SecionNumber* is used.) But you can also have the formatter substitute strings for variables that you define.

How To

When you want to set a variable equal to text, choose **Other Format** and enter `TagString` followed by a unique tag name and the string itslef (in quotes). For example, to access the title of a chapter (not the chapter *number*, which the *Chapter* variable does), you would enter

```
TagString FirstChapter="The Early Years"
```

Then when you want to print the name of the chapter, you could choose **Insert/Variable/Other** and then enter `FirstChapter` and choose **None** for the template.

Tips

The command

```
TagString name=ChapterTitle
```

always sets *name* to the current chapter title.

# TCT/NoTCT

Keystrokes

Choose *Alt-S* (or *F10*, Style), Other Format, type `TCT "string1" = "string2"`, and press *C*

Choose *Alt-S* (or *F10*, Style), Other Format, enter `NoTCT "string1" = "string2"`, and press *C*

Function

Automatically changes *"string1"* into *"string2"* if it is encountered while formatting.

This command translates one character sequence into another during formatting. (TCT stands for translation character table.) This is useful for special characters that don't display on the screen, but your printer can print.

*String1* is the *source* string, which can be any sequence of printing characters but can't contain commands. The source string can begin with a leading blank space, which means that the translation will only be done if the string immediately follows some whitespace (such as a blank space, the start of the line, or many formatting commands); no other whitespace characters are allowed, however. If the string contains any punctuation marks, the string must be quoted within delimiters.

*String2*, on the other hand, can be any string at all, including complex command sequences or whitespace characters. While *String1* doesn't have to be within quotes (unless it contains punctuation marks), *String2* must appear within quotes.

The character translation affects only *formatted text*; that is, the formatter will not change any matching string that's included as part of a command or command definition. For example,

```
TCT "verb" = "noun"
```

tells the formatter to change the letters *verb* (that appear as text to be formatted) to the letters *noun*, but the

translation won't affect the wording of a Begin *Verb*atim or End *Verb*atim command in the file.

How To

To translate one character sequence to another, choose Style/Other Format, type

```
TCT "String1" = "String2"
```

press *Enter*, then press *C* at the prompt.

*String1* is the input string (the text as it appears in your file), and *String2* is the desired output string (what you want the formatter to print when it sees *String1*). The equal sign is optional.

For example, let's say you've written a report about a product code-named *Magnum* and, before the report's published, the product name is changed to *Mega-Systems*. You could, of course, use the Search-Replace command to change each occurrence of *Magnum*, but it's easier to insert the following command near the top of your file:

```
TCT "Magnum" = "Mega-Systems"
```

If the name changes again, you have only to change the TCT command, not the entire file.

A more common use of TCT is to enable a printer to print characters that don't display on the screen. For example, the Apple LaserWriter can print a long dash (called an "em dash"), but your screen can't display this symbol. Traditionally, two hyphens in a row represent the em dash. So you could set up the TCT command so that it automatically translates "--" into "—":

```
TCT "--" = "@char[208]"
```

For example, if you type

```
That reminds me of my days as a rodeo clown--but I
digress.
```

it will print like this:

That reminds me of my days as a rodeo clown—but I digress.

A single TCT command can define any number of translations. For example,

```
TCT "e.g." = "for example", "i.e." "that is"
```

This example defines the character translation for the abbreviations *e.g.* and *i.e.*

If more than one source string matches at a certain point in the file, the formatter uses the longest one. For example, if you typed the command

```
TCT "a" = "Apple", "ab" = "Absolutely"
```

and the formatter saw the text *abcdac*, the following would print:

AbsolutelycdApplec

**Note:** The formatter performs translation straight across the boundaries between macros. For instance, if you've created the macro *M()* to be *abcda*, and you have a TCT command that specifies *ab* = *X*, when you format the text *M()b* the formatter will print *XcdX*.

Tips

For an extensive list of TCT command examples, look at the file POSTSCR.TCT on the Sprint distribution disks. This file lists numerous character translation strings for output on the Apple LaserWriter and other PostScript printers.

If you need to turn off the character translation table temporarily, you should mark the text as a block, then choose **S**tyle/**O**ther Format and enter `NoTCT`. Sprint will print that text just as it appears on the screen, with no character translation.

Note that some formats *automatically* turn character translation off, like the Example and Verbatim formats.

# Template

Keystrokes

Choose *Alt-S* (or *F10,* **S**tyle), **O**ther Format, type `Template` *`parameters`*, and press *C*

*Alt-I* (or *F10,* **I**nsert), **V**ariable, <*VariableName*>, **P**ick template

Function

Specifies how variables print.

This command lets you define how a text variable should print. For example, you can specify how the variable *Page* should print (for example, page numbers

can be Arabic, Roman, Ordinal, Arabic ordinal, or English ordinal numbers, or alphabetic characters or words). For a complete list of variables, see the Variables entry in the "Menu Encyclopedia" chapter of this manual. For a list of all Template options, refer to Table 2.1.

A *numbering template* can be used as a standalone command at the beginning of a file (for example, choose Style/Other Format, and then enter `Template` *parameters*), or it can temporarily affect a variable when typed as part of a Variable or Value command. If you enter the @Template command in the STANDARD.FMT file, the specified variable(s) will *always* print according your template.

Use the symbol % to indicate where you want the variable to print, followed by a letter that specifies *how* you want the variable to print as shown in the following table.

Table 2.1: Template Options

| Option | Effect |
|---|---|
| %d,%u | The variable prints as an Arabic number (e.g., 1, 2, 3,...), up to 65,535.<br><br>If you want to insert a uniform space in which the variable will print, place a number between the % and the *d* or *u*. For instance, %3u puts enough spaces before the number to make it three characters wide. %03u puts enough zeros before the number to make it three characters wide. This also works for %' (described next), and %A.<br><br>If you want the variable *Minute* to print as a two-digit number, you can type the following template in your STANDARD.FMT file:<br><br>`TEMPLATE minute = "%02d"`<br><br>This prints one minute after 11 as 11:01 instead of 11:1. |
| %' | The variable prints as an ordinal number (e.g., 1st, 2nd, 3rd, etc.). |
| %i,%I | The variable prints as a Roman numeral (e.g., I, II, II, IV,...), up to MMMCMXCIX. If you want lowercase Roman numerals (like vii), use %i; if you want uppercase Roman numerals, use %I. |
| %a,%A | The variable prints as an alphabetic character (e.g, 1 prints as A, 2 as B, etc.). |
| %o,%O | The variable prints as an English cardinal number (e.g., One, Two, Three...), up to Nine Hundred Ninety-Nine Thousand Nine Hundred Ninety-Nine. Use %o for all lowercase; %O if you want the first letter of the number capitalized. |
| %f,%F | The variable prints as an English ordinal number (e.g., First, Second, Third...), up to Nine Hundred Ninety-Nine Thousand Nine Hundred Ninety-Ninth. Use %o for all lowercase; %O if you want the first letter of the number capitalized. |
| %#*text*%] | This prints the *parent* of the variable being referenced, plus any text you want to print with the parent. The parent is printed according to its own template. If there is no parent, the formatter ignores the entire template command, from the %# to the %]. For example, Sprint precedes a subsection number with the chapter number, followed by a period, followed by the section number, followed by a period, like this:<br><br>2.4.5 Simulated Intelligence<br><br>The template for section and subsection number are built-in, but if printed would look like this:<br><br>`TEMPLATE Section="%#.%]%d"`<br>`TEMPLATE SubSection="%#.%]%d"`<br><br>These templates instruct the formatter to print the parent of the variable, plus a period, followed by an Arabic number. If you want the section numbers to print like this:<br><br>2-4-5 Simulated Intelligence |

Table 2.1: Template Options, continued

| Option | Effect |
|---|---|
| | You'd create a numbering template that looks like this:<br>`TEMPLATE Section="%#-%]%d"`<br>`TEMPLATE SubSection="%#-%]%d"` |
| *%&text%]* | This works like %#, but instead of affecting how the variable prints, it applies only to any *tag* that uses that variable. References to the original variable print only the variable itself. If you use the **Define a Tag** command and then reference the tag (using the **Reference a Tag** command), the references will print with copies of the parent variables as well. For example,<br>`TEMPLATE x="%&.%]%d"`<br>`PARENT x=page`<br>`SET x=1`<br>`SET y=x`<br><br>`x`<br>`y`<br>prints in a form like this:<br>1<br>318.1 |
| *%(text%)* | Sprint prints the text (variable) times, instead of the number of the variable. For instance, if you create a template of `%(*%)` for the variable *Chapter*, when you print Chapter 1, the formatter prints Chapter *; when you print Chapter 2, the formatter prints Chapter **; and so on.<br><br>You can use this template to create footnotes that are starred instead of numbered. |
| *%[text%; text%; text%:text%]* | This works like the formatter command Case, printing a different text for each different value of the variable. Zero goes to the first case (the first *text*), 1 goes to the second case (the second *text*), and so on. %: indicates a *default* case that should be used if no others are suitable. For example, if Sprint didn't have a variable called *MonthName*, you could create the following template to print the name of the month when using the variable *Month*:<br>`The number of the month is MONTH, t="%0",`<br>`but the name of the month is MONTH,`<br>`T="%[%;Jan%; Feb%; Mar%; Apr%; May%; June%;`<br>`July%; Aug%; Sept%; Oct%; Nov%; Dec%]".`<br>The printed example looks like this:<br>The number of the month is 4, but the name of the month is April. |

Table 2.1: Template Options, continued

| Option | Effect |
|---|---|
| *%<text%; text%; text%:text%]* | This prints different text for each number of *parents* for the variable. This is typically used for the counters in the Numbered and Hyphens formats. If there are no parents, the first case is printed. One parent causes the second case to be printed. If that parent has a parent, the third case is printed, and so on. The case %: indicates a default case that's used if no others are suitable. If there is no default %: case, it repeats with the first case as with %[ above.<br><br>For example, the Hyphens format marks different levels of paragraphs by "numbering" them first with dashes, then asterisks. The numbering template looks like this:<br><br>`t="%<-%;*%;o%]"`<br><br>To change the Hyphens format so that it marks levels with dashes, asterisks, equal signs, and plus signs, you can modify the format by using the Modify command. Choose Style/Other Format and enter<br><br>`Modify Hyphens, t="%<-%;*%;=%;+%]"` |
| *%%* | This is used to print a percent sign. |

How To

By far the easiest, safest, and quickest way to insert template commands is to use the menu commands off the Variable menu.

If you absolutely have to change the way a variable prints in the entire document or in all your documents, use the Template command in your file or the @Template command in the STANDARD.FMT file.

One of the best ways to become accustomed to the template codes is to choose variables from the menu and observe the commands that Sprint inserts in your file.

Tips

Included in STANDARD.FMT are a couple of templates. In this way, we've changed the way the following variables normally print:

**Day** Arabic ordinal numbers: 1st, 2nd, 3rd ...

**Minute** 2-digit decimal number: 01, 02, 03 ...

**Page number** Lowercase roman numerals in the table of contents pages, arabic numbers (1, 2, 3 ...) elsewhere.

You can change any of the templates in a backup copy of the STANDARD.FMT file or (if you're using a PostScript

writer) in the POSTSCR.TCT file by editing the file and modifying the template parameters. For a list of all valid *Template* entries, see the preceding table.

You can also include a numbering template when you modify a format. Use the parameter *numbered* and then type the template text. You don't need to enter the Template command in this instance. You can also change the template of any command "on the fly" by using the Modify command. Choose **S**tyle/**O**ther Format and type `Modify`, followed by the new template. For example, if you want to modify a particular **H**yphens format to print a circle (lowercase *o*) instead of a dash, you could type

```
Modify Hyphens, t="<o%;*%"
```

Only this particular occurrence of the Hyphens format will be affected.

# Text

Keystrokes

Choose *Alt-S* (or *F10,* **S**tyle), **O**ther Format, and type `Text`

Function

Creates a customized format for a block of text.

This command by itself has no affect on your text. In essence, it lets you set a "unique" format that will format your text according to the parameters you add to it. For example, if you want a paragraph formatted identical to surrounding text but also want it to print "grouped" (all lines on one page) with different margins, you'd use the Text format and modify it to include the *Group* and *Margins* parameters.

Most of the formatting commands available with the Text command are more easily gotten with menu commands, but if you want to really specialize the look of a block of text, the Text command is perfect because you can combine many changes into one command. For example, with one Text command, you could change the font, font size, margin, and spacing—rather than apply four different menu commands to do this.

Tips

For a list of parameters that you can include in the Text command, refer to Table 1.8 on page 149 in Chapter 1.

# Timestamp

Keystrokes

Choose *Alt-S* (or *F10*, Style), Other Format, type `Timestamp`, and press *C*

Function

Inserts the current date and time into your printed document.

When the formatter processes the Timestamp command, it reads the current date and time from your computer's DOS clock and inserts them into your document. The format is

```
MM/DD/YY hour:minute am/pm
```

How To

Here's how your document might look when you use the Timestamp command:

```
                    Memo to All Employees

TIMESTAMP

From:       Albert W.
Re:         Vacation Pay
```

This results in something like this:

Memo to All Employees

4/29/89 2:30pm

From: Albert W.
Re: Vacation Pay

Tips

Timestamp is defined in STANDARD.FMT, which means you could change the format as needed (for example to 24-hour time or to DD/MM/YY).

You might find this command useful in a header or footer to "stamp" when the printout was run. This command is also handy when creating memos and letters.

If your time or date is incorrect, your computer's clock must be off. Use the DOS commands TIME or DATE to correct it.

# Title

Keystrokes — Choose *Alt-S* (or *F10*, Style), **O**ther Format, type `Title VariableName`, and press *C*

Function — Inserts the title of a predefined variable like *Chapter*, *Section*, or *Subsection*.

Ordinarily, referencing a predefined variable like *Chapter* gives you the Sprint-assigned number of the variable. You can use the Title command, however, to reference the text associated with the title of the heading instead of the number.

How To — Whenever you want to insert the title of a numbered heading into your document, choose **O**ther Format, type `Title`, then enter the variable associated with the heading, and press *C*.

For example, to print the current appendix title, you could enter

```
The current appendix is called "TITLE APPENDIX."
```

This prints as:

The current appendix is called "Further Reading."

Tips — This command has the same effect as inserting the *SectionTitle*, *ChapterTitle*, *SubsectionTitle*, *ParagraphTitle*, or *AppendixTitle* variables into your document.

You can use this command only to refer to the title of the current heading. You should use tags to refer to titles and numbers of sections and headings that are not current (that is, they come before or after the section that the command appears in).

# Undent

Keystrokes — Choose *Alt-S* (or *F10*, Style), **O**ther Format, and type `Undent, indent = -dimension`

Function — Starts the first line of each paragraph at the left margin; the *rest* of the paragraph is indented (creates a *hanging indent*).

This command is the opposite of a normal paragraph indent, which indents the first line of each paragraph and prints the remaining lines at the left margin. By default, Undent (no dimension) prints the first line of the paragraph at the left margin and then indents the remaining lines one-half inch to the right of the left margin.

You will find this command defined in the Sprint file STANDARD.FMT.

How To

If you've already typed the text you want to affect, select the text and then choose **O**ther Format from the **S**tyle menu. Then type the following command: `Undent`. For example,

```
BEGIN UNDENT
Page 1 lists items you should have received in the
shipping container.

Page 2 explains how to get the unit out of the box.

Page 3 tells you everything you ever wanted to know
about the unit. It explains basic operation,
maintenance, and troubleshooting.
END UNDENT
```

The example prints like this:

Page 1 lists items you should have received in the shipping container.

Page 2 explains how to get the unit out of the box.

Page 3 tells you everything you ever wanted to know about the unit. It explains basic operation, maintenance, and troubleshooting.

Tips

If you want the second and subsequent lines of each paragraph to be indented more or less than the default one-half inch, modify the Undent format so that it *indents* the text by a negative dimension. You can use any of the dimensions listed in Table 1.2 on page 58. For example,

```
BEGIN UNDENT, indent = -.5 line
This example shows how to print the top line of a
paragraph at the left margin and print the
paragraph's remaining lines halfway across the page.

Tab This paragraph should also begin printing
halfway across the line, since it begins with a Tab.
END UNDENT
```

The printed result:

This example shows how to print the top line of a paragraph at the left margin and print the paragraph's remaining lines halfway across the page.

> This paragraph should also begin printing halfway across the line, since it begins with a *Tab*.

# Verbatim

Keystrokes — Choose *Alt-S* (or *F10*, Style), Other Format, and type `Verbatim`

Function — Prints text in a single-spaced, flush-left format.

This format sets line spacing to single and extends to right margin to the right edge of the paper. When you want the formatter to begin a new line, you have to press the *Enter* key.

You will find this command defined in the Sprint file STANDARD.FMT.

How To — Choose Style/Other Format, type `Verbatim`, and press *R* at the prompt. You can then type the text you want to appear in Verbatim format. When you've entered the text, choose Other Format, type `Verbatim`, press *R*, and then end the region by pressing *E*. If you mark the text before you enter the Verbatim command, all marked text will print in this format.

Tips — This command is rarely used, but if you want to create some unusually formatted text, Verbatim may lend itself to producing the desired output. For example,

```
BEGIN VERBATIM
     This textEnter
has an unusual format.Enter

You may never need this type ofEnter
                    format,Enter
                 butEnter
you can see howEnter
                                    it works.Enter
END VERBATIM
```

This example prints like this:

```
     This text
has an unusual format.

You may never need this type of
                    format,
                 but
you can see how
                                    it works.
```

Also, it's important to note that soft returns (the ^J return characters automatically inserted by Sprint) are *still ignored for wordwrapping*, even in Verbatim. If you want your output to look *just* like it does when you type it, put hard returns (press *Enter*) in the appropriate places in your text.

# Word

Keystrokes — Choose *Alt-S* (or *F10*, Style), Other Format, type `Word`, and press *C*

Function — Tells the Sprint formatter to treat text as a single "word."

The Word command lets you define a block of text as a single "word"—even if there are spaces, commas, or hyphens in it. This is convenient to do if you have a phrase that you never want broken at the end of a line. For example, if you are listing a series of punctuation marks, you probably do not want the list to be broken up.

How To

If you need to set aside a block of text to be treated as a word, select the text and choose **S**tyle/**O**ther Format. Type `Word` and press *C* at the prompt. For example:

```
The opening delimiters in Sprint, WORD [, {, <, ", `,
and (, must be matched by an appropriate closing
delimiter: WORD ], }, >, ", ', and )!
```

The result is

> The opening delimiters in Sprint, [, {, <, ", `, and (, must be matched by an appropriate closing delimiter: ], }, >, ", ', and )!

Tips

Use the Word command to group inverted names that you're indexing with the **R**eference Word command. For example,

```
WORD Einstein, Albert
```

is indexed without inserting an unwanted sublevel due to the presence of the comma.

## !

Keystrokes

*Alt-S* (or *F10*, Style), **O**ther Format, type !, and press *C*

Function

Creates a conditional line break.

This command acts like a discretionary hyphen, without ever printing a hyphen. It specifies where (besides at a space or hyphen character) Sprint can break the current line.

This is useful for phrases containing the slash (/) character (such as in "and/or"). Sprint normally sees such phrases as a single word; if the phrase cannot fit at the end of the current line, Sprint places the entire phrase on the following line. If you insert the ! command after the /, however, and there's enough room on the line, Sprint prints the first part of the phrase on the current line, breaks the line after the slash, and puts the remainder of the phrase on the next line. *Sprint does not, however, print a hyphen character when it breaks the line.*

How To

To allow Sprint to break a line at a particular spot, choose **S**tyle/**O**ther Format and type ! where a

conditional break is acceptable. Press *C* at the prompt. For example:

```
Here's the printed effect of the ! command: Look at
this/!that/!the other information in this section for
details. A user can always refer to this example as
needed and/!or try his/!her own examples.
```

The result looks like this:

> Here's the printed effect of the ! command: Look at this/that/the other information in this section for details. A user can always refer to this example as needed and/or try his/her own examples.

Tips

If you want to conditionally hyphenate *all* words that contain a slash, open a renamed backup copy of the STANDARD.FMT file and type the following formatter command:

```
@TCT["/"="/@!"]
```

## <

Keystrokes

*Alt-S* (or *F10*, Style), **O**ther Format, type <, and press *C*

Function

Acts like a hard return character and overprints text at the left margin with text following the command.

This command creates a hard return and places text following the command on top of text appearing at the left margin. In other words, it performs a carriage return without a linefeed. This is useful for creating special effects, such as formulas and scientific equations.

How To

To print the fraction

$$\frac{2}{3}$$

at Sprint's default left margin:

- Type 2, choose Style/**O**ther Format, and type <. Press *C* at the prompt.
- Type an underbar character (_) and press *Enter*.
- Type 3.

C H A P T E R

# 3

# @-Sign Commands

This chapter discusses Sprint's so-called @-sign commands. @-sign commands are an alternative to choosing formatting commands from the pop-up menus. For example, when you want text to print in italic, you can choose **Italic** from the **Typestyle** menu, or you can enter the @i command and enclose the text to be italicized in *delimiters* (which are special symbols used for enclosing text to be acted on by the command).

Unlike formatting commands chosen from Sprint's menus, @-sign commands do not display in reverse video nor does the text affected by them appear in a different color or attribute. Sprint's @-sign commands don't have any effect while you are typing and editing; instead, they are carried out only when you print your file.

## How to Enter @ Commands

There are two ways to enter @-sign commands. Let's look at a simple example:

```
@center[Bingo!]
```

This tells Sprint to center the word *Bingo!* between the left and right margins, like this:

Bingo!

This example demonstrates the method commonly used for a small amount of text, such as one or more words or a line of text. Type the @-sign, followed by the name of the command, and then type text within a

matched set of delimiters (that is, special symbols that Sprint recognizes as enclosing the text to be affected).

The second method looks like this:

```
@Begin[CommandName]
Any amount of
text to be affected
by the format command
@End[CommandName]
```

This second method is commonly used to format a large area of text. The `@Begin[Command Name]` serves as the beginning delimiter, and the `@End[Command Name]` is used as the closing delimiter. You type the text to be affected between the Begin and End commands. For example,

```
@Begin[Quotation]
"Lengthy quotation text"
@End[Quotation]
```

This command tells the formatter to print the *lengthy quotation text* in a Quotation format.

There are two major reasons for using the @Begin[*CommandName*] and @End[*CommandName*] delimiters:

- when you want a large area of text to be affected by a formatting command
- when you want to modify a format that you've chosen

If you're using the @-sign method of entering commands and want to modify a format, you *must* use the @Begin[]/@End[] delimiters. For example, to create two-column text, you would type:

```
@Begin[text, columns 2, gutter 1.5 picas]
Pages of text that
you want to appear
in two columns.
@End[text]
```

You *cannot* type `@text, columns 2[Paragraphs of text]` without getting an error message!

# Delimiters

@-sign commands always start with an @-sign and use *delimiters* to mark the text to be affected by the command. You can use the following sets of delimiters to mark text:

[ ] brackets

{ } braces

< > angle brackets

" " double quotes

‘ ’ single opening and closing quotes

’ ’ single closing quotes

( ) parentheses

Let's look at the *Bingo!* example again:

```
@center[Bingo!]
```

The brackets ([ and ]) are the delimiters: The left bracket ([) marks the beginning of the text you want centered, and the right bracket (]) marks the end of the text to be centered.

# Nesting @ Sign Commands

Text can be affected by more than one @-sign command. Let's say you want paragraphs numbered, but within the numbered text, you want an itemized list. For example:

```
@Begin[Numbered]
The new car loan application must include the following items:

@Begin[Bullets]
Name

Rank

Serial number
@End[Bullets]

Complete the forms in duplicate.

Sign all forms in magenta ink (or blood, if we prefer).
@End[Numbered]
```

The printed text looks like this:

1. The new car loan application must include the following items:
   - Name
   - Rank
   - Serial number
2. Complete the forms in duplicate.
3. Sign all forms in magenta ink (or blood, if we prefer).

When formatting a small area of text, you can include multiple @-sign commands on a single line. For example,

```
@Center[@B<bold, @I[bold italics]>, plain text]
```

The example prints like this:

**bold text**, ***bold italics***, plain text

# Nesting Delimiters

Sprint can correctly handle sets of delimiters that are nested inside one another. For example,

```
@Quotation(Outside parentheses are delimiters, and so are the brackets
surrounding the word @i[Sprint]; Sprint will not print these delimiter
characters. Parentheses around the word (inside) are not delimiters.
Since they're not part of an @-sign command, Sprint prints them as
text.)
```

Prints like this:

> Outside parentheses are delimiters, and so are the brackets surrounding the word *Sprint*; Sprint will not print these delimiter characters. Parentheses around the word (inside) are not delimiters. Since they're not part of an @-sign command, Sprint prints them as text.

Be careful, though, when typing a delimiter character as text. When doing so, make sure your begin and end delimiters are different from the delimiter character that you want to print as text or use the Begin/End pairs so Sprint doesn't accidentally end a format before you want it ended.

For example, if you wanted to print the text *3 > 2* in an Example format, you could type

```
@Example[3 > 2]
```

or

```
@Begin<Example>
3 > 2
@End<Example>
```

But, if you type this:

```
@Example<3 > 2>
```

the formatter will see the *greater than* symbol as the closing delimiter for the Example format. The angle bracket you intended to be the closing delimiter will actually print as text, like this:

3 2 >

Delimiters must always be a matched set. If you use a [ as the begin delimiter, for example, you must use a ] as the end delimiter. If the delimiters in your command don't match, Sprint displays an error message during formatting and will not let you print until you correct the error. One way to avoid this is to:

1. Load the macro file *Match* (*F10*, **U**tilities, **M**acros, **L**oad, and choose **M**atch).
2. Move the cursor to the first open delimiter (such as [, {, <, and so on).
3. Execute the macro called *Match* (*F10*, **U**tilites, **M**acros, **E**nter, type `Match`, then type `E`). (You can also just press *Alt-M* to enter this macro.)

Sprint then looks for a matching *end* delimiter. If it finds one, the cursor "bounces" between the begin and end delimiters. If it doesn't find the ending delimiter, Sprint displays the message `Mismatched or missing delimiter`. Press any key to return the cursor to normal editing mode.

You can enter an @-sign command in either uppercase or lowercase letters, or any combination of the two. Sprint is not case sensitive.

If you want to print an @-sign, be sure to follow the @-sign with a space, a number, a control character, or another @-sign. If you don't use one of these tricks, the formatter produces an error message indicating you've entered an unknown command. For example, to print the text, *This entry is about @-sign commands*, you could type

```
This entry is about @@sign commands, or
This entry is about @-sign commands, or
```

```
This entry is about @ sign commands
```

If you typed `This entry is about @sign commands`, the formatter would produce an error message, since "@sign" isn't a valid @-sign command.

**Remember:** Only use the @-sign commands if you have to. The menu commands are usually easier and clutter up your document far less because they stand out from the surrounding text. However, there are certain instances where you should use only @-sign commands; see the next section, "When to Use @-Sign Commands."

# When to Use @-Sign Commands

It's best to use @-sign commands when preparing text for electronic mail or in files that you want to edit with another word processor. Unlike formatting commands chosen from Sprint's menus, @-sign commands do not insert control codes in the file. They are pure ASCII text, so if you create a file, format it with @-sign commands, press *Enter* at the end of every line (or toggle **Wrap Long Lines** to **Yes** in the **Customize/ASCII File Handling** menu to automatically wordwrap long lines for you), and remove the ruler line at the top, you'll produce a pure ASCII file that will look the same in any text editor you open it up with.

Also, you ordinarily use @-sign commands if you are editing or creating a Sprint style sheet (like the supplied file STANDARD.FMT). Format files use only @ commands to improve readability.

# @-Signs vs. Menus

In a normal Sprint document, there is virtually never a time that you have to use an @-sign command over a menu command. There is always a way to achieve the same result through the Sprint menus. Most of the time, there is a unique menu command equivalent to the @-sign version. In rarer cases, you have to choose **Style/Other Format** and then enter the name (without the @ sign).

Refer to the "Sprint Menu Encyclopedia" chapter in this manual for details on the @-sign commands that have menu equivalents. Note that in some cases, the names are different. For example, the @-sign command @Format has a menu equivalent called **Style Sheet**. To find information on @Format, therefore, you should look up the Style Sheet entry in the "Menu Encyclopedia" (Chapter 1). The table that follows this section will tell you

the correct name corresponding to the @-sign command you're interested in.

Refer to Chapter 2 in this manual for explanations of the meaning of @-sign commands that have no unique menu equivalent. For example, look at the Address entry in that chapter for information on the @Address command. In the table that follows this section, such commands are marked with an asterisk (*).

In a few cases, the explanation of the @-sign commands are saved for the "Style Sheet Commands" section in the *User's Guide*. In the table, those rarities have nothing in the column marked "Menu Command."

A list of all @-sign commands and their pop-up menu equivalents follows.

# @-Command Tables

Table 3.1 is an alphabetical list of the @-sign commands. When there is a menu equivalent, that command follows the @-command. But the lack of an equivalent doesn't mean you can't enter the command from the menus. In most cases, you can choose **S**tyle/**O**ther Format, and then enter the command name (*without* the @-sign). Most of the @-commands with no menu equivalents are either esoteric rarities or for advanced Sprint users.

Table 3.1: @-Commands and Menu Equivalents

| @-Command | Menu Command | @-Command | Menu Command |
|---|---|---|---|
| @@ | | @Caption | (Caption prompt) |
| @_ | Non-Breaking Space | @Case | * |
| @\ | *Tab* | @Center | Center (or C on ruler) |
| @/ | | @CenterPage | Title Page |
| @< | | @Chapter | Chapter |
| @; | | @Char | * |
| @, | | @Closing | * |
| @~ | | @Column | * |
| @* | *Enter* | @ColumnBreak | Column Break |
| @! | * | @Columns1...6 | Snaking Columns |
| @\| | Special Hyphen | @Comment | Hidden |
| @/ | *Tab* | @D | (Index) Word |
| @[ | ([ on ruler) | @Define | |
| @] | (] on ruler) | @Description | Description |
| @^ | (T on ruler) | @Display | * |
| @$ | ([ on margin) | @E | Italic |
| @> | Wide Space | @End | (End prompt) |
| @>(text) | Repeating Character | @EndF | Font |
| | | @EndS | Character Size |
| @< | * | @ENote | Endnote |
| @= | Center | @EPS | EPS Picture |
| @+ | Superscript | @Error | |
| @- | Subscript | @Escape | * |
| @^*n* | Control Character | @Eval | Variable |
| @A | Large | @Example | * |
| @Address | * | @FCapt | * |
| @Appendix | Appendix | @Figure | Figure |
| @AppendixSection | AppendixSection | @FlushRight | (R on ruler) |
| @AtEnd | | @FlushLeft | (L on ruler) |
| @Bar | Bar | @FNote | Footnote |
| @B | Bold | @Font | Font |
| @Begin | (begin prompt) | @FontName | Font |
| @Box | Draw Box | @FooterE | Even Pages |

Table 3.1: @-Commands and Menu Equivalents, continued

| @-Command | Menu Command | @-Command | Menu Command |
|---|---|---|---|
| @FooterO | Odd Pages | | (Unconditional) |
| @FooterT | Title Page | @NoFloat | |
| @Format | Style Sheet | @NoTCT | * |
| @Group | Group Together on Page | @NoteChapter | * |
| | | @NoteSection | * |
| @HaveSpace | * | @Numbered | Numbered |
| @Header | All Pages | @O | * |
| @HeaderE | Even Pages | @Ovp | * |
| @HeaderO | Odd Pages | @PageFooting | Footer |
| @HeaderT | TitlePage | @PageFoot | Title Page (Footer) |
| @HeadingA | HeadingA | @PageHeading | Header |
| @HeadingB | HeadingB | @PageHead | Title Page (Header) |
| @HeadingC | HeadingC | | |
| @HeadingD | HeadingD | @PageInit | |
| @Hsp | * | @PageRef | Reference a Tag |
| @HUnits | | @Paragraph | Paragraph |
| @Hyphens | Hyphens | @Parent | |
| @I | Italic | @PassInit | |
| @If | | @PgBlank | Blank Page(s) |
| @IfDef | | @PgBreak | Conditional Page Break |
| @IfOdd | | | |
| @Include | * | @Place | * |
| @Incr | * | @Printer | Current Printer |
| @IxRef | Reference Word | @Quotation | * |
| @IxMaster | Master Keyword | @ReadEPS | |
| @IxSee | See | @Ref | Reference a Tag |
| @IxSeeAlso | Also See | @Reserve | Reserve Space |
| @IxRefUnder | Index Under | @Reset | |
| @IxRange | PageRange | @Scale | * |
| @Justify | * | @Section | Section |
| @KeepFollowing | Keep with Following Text | @Subsection | Subsection |
| | | @Set | * |
| @Kern | * | @Size | Character Size |
| @Key | KeyCaps | @SNote | Notes |
| @Label | * | @String | Define Text Variable |
| @Large | Large | | |
| @Level | | @StringInput | * |
| @Macro | | @Style | * |
| @MakeTOC | * | @Tab | * |
| @Merge_Init_ | | @TabDivide | * |
| @Message | * | @TabSet | Tab Stops |
| @Modify | | @Table | Table |
| @Multilevel | Multilevel | @Tag | Define a Tag |
| @NeedSpace | * | @TagString | * |
| @NewColumn | Column Break | @TCaption | (Caption prompt) |
| @NewPage | Insert | @TCapt | * |

Table 3.1: @-Commands and Menu Equivalents, continued

| @-Command | Menu Command | @-Command | Menu Command |
|---|---|---|---|
| @TCT | * | @V | Variable |
| @Template | Variable | @Value | Variable |
| @Text | * | @Verbatim | * |
| @Timestamp | * | @VUnits | |
| @Title | * | @W | Word Underline |
| @U | Underline | @Warn | |
| @UN | Word Underline | @Word | * |
| @Undent | * | @X | Strikethrough |
| @UX | Underline | | |

* These @-commands have menu equivalents through the Style/Other Format menu. See Chapter 2 for details.

The next table shows the menu commands matched up with their @-command equivalents—the previous table in reverse.

Table 3.2: Menu Commands and @-Command Equivalents

| Menu Command | @-Command | Menu Command | @-Command |
|---|---|---|---|
| Also See | @IxSeeAlso | KeyCaps | @Key |
| Appendix | @Appendix | Large | @A or @Large |
| AppendixSection | @AppendixSection | Master Keyword | @IxMaster |
| Bar | @Bar | Non-Breaking Space | @_ |
| (begin prompt) | @Begin | Notes | @SNote |
| Blank Page(s) | @PgBlank | Numbered | @Numbered |
| Bold | @B | Other Format | (see below) |
| (Caption prompt) | @Caption | Page Range | @IxRange |
| (Caption prompt) | @TCapt | Paragraph | @Paragraph |
| (Caption prompt) | @TCaption | Reference a Tag | @PageRef or Ref |
| Center | @= or @Center | Reference Word | @IxRef |
| Chapter | @Chapter | Repeating Character | @>(text) |
| Column Break | @NewColumn | Reserve Space | @Reserve |
| Conditional Page Break | @PgBreak | ([ on ruler) | @[ |
| Control Character | @^*n* | (] on ruler) | @] |
| Current Printer | @Printer | ([ on ruler) | @$ |
| Define a Tag | @Tag | (C on ruler) | @Center |
| Define Text Variable | @String | (L on ruler) | @FlushLeft |
| Description | @Description | (R on ruler) | @FlushRight |
| Discretionary Hyphen | @\| | (T on ruler) | @^ |
| Draw Box | @Box | See | @IxSee |
| Endnote | @ENote | Section | @Section |
| (End prompt) | @End | Snaking Columns | @Columns1...@Columns6 |
| *Enter* | @* | Strikethrough | @X |
| EPS Picture | @EPS | Style Sheet | @Format |
| Figure | @Figure | Subsection | @Subsection |
| Font | @FontName | Subscript | @- |
| Footer | @PageFooting | Superscript | @+ |
| Footnote | @FNote | *Tab* | @/ or @\ |
| Group Together on Page | @Group | Tab Stops | @TabSet |
| Header | @PageHeading | Table | @Table |
| HeadingA | @HeadingA | Title Page | @CenterPage |
| HeadingB | @HeadingB | Title Page (Footer) | @PageFoot |
| HeadingC | @HeadingC | Title Page (Header) | @PageHead |
| HeadingD | @HeadingD | Underline | @UX or @U |
| Hidden | @Comment | Variable | @Eval |
| Hyphens | @Hyphens | Variable | @Template |
| Index Under | @IxRefUnder | Variable | @Value |
| (Index) Word | @D | Wide Space | @> |
| Insert (Unconditional) | @NewPage | Word | @D |
| Italic | @I or @E | Word Underline | @UN or @W |
| Keep with Following Text | @KeepFollowing | | |

There are a few @-sign commands that have no menu equvalents:

| | | |
|---|---|---|
| @\ | @Error | @Parent |
| @/ | @HUnits | @PassInit |
| @< | @If | @Place |
| @; | @IfDef | @ReadEPS |
| @, | @IfOdd | @Reset |
| @~ | @Macro | @U |
| @AtEnd | @Merge_Init_ | @VUnits |
| @Case | @Modify | @Warn |
| @Define | @NoFloats | |

See the "Style Sheet Commands" appendix in the *Advanced User's Guide* for more information.

Table 3.3: Parameters and Menu Equivalents

| Parameter | Menu Equivalent |
|---|---|
| Spacing | Line Spacing |
| Spread | Inter-Paragraph Spread |

# @-Sign Syntax

The syntax for the @-sign commands in the previous lists usually follows the syntax for the menu equivalents. For example, because the EPS Picture command requires a file name to complete the command, you can correctly assume that the @EPS command also needs a file name, like this:

```
@EPS["C:\CHART.EPS]"
```

(@EPS also takes an optional height and width measurement, as does its menu version.)

The @ commands almost all follow one of the following patterns:

- @ commands followed by a dimension in delimiters. For example,

  ```
  @Bar[2 points]
  @Reserve[.5 page]
  @Kern[.2 cm]
  ```

- @ commands followed by a file name in delimiters. The file name should be in quotes if you include a path name with slashes. For example,

  ```
  @EPS[CHART.EPS]
  @Include["C:\MYBOOK\CHAPT1.SPR"]
  @Printer[EPSON.SPP]
  ```

- @ commands followed by a number in delimiters. For example,

```
@PgBlank[2]
@Scale[1.5]
@Char[138]
```

- @ commands followed by a region of text in delimiters. For example,

```
@I[Gone with the Wind]
@Comment[these figures might be wrong]
@Description[Apples @/A fruit
  ...
  Oranges @/Another fruit.]
```

This category of commands can also begin with an @Begin command and end with an @End command to improve readability and to allow you to modify the occurrence of the command. For example,

```
@Begin[Description, spread 0]
Apples @/ A fruit
...
Oranges @/ Another fruit.
@End[Description]
```

- @ commands followed by a line of text in delimiters. For example,

```
@Chapter[How to Make Money]
@Level1[How to Avoid Guilt]
@>[*]
```

- @ commands followed by nothing at all. For example,

```
@MakeTOC
@@
@/
@_
@Reset
```

- @ commands followed by a parameters in delimiters. For example,

```
@Style[topmargin 1.5 inches, bottommargin 2 inches]
@Text[group, leading 1.75 lines]
```

There are also a few commands that do not fall into these categories, especially @Set which accepts variables and numbers as its argument and @Case, which accepts multiple strings.

The syntax of four indexing commands calls for special mention when using @ signs:

@IxSee — Requires two strings in quote marks separated with a comma. Like this:

```
@IxSee["Mexican Americans","Hispanics"]
```

This results in a *See* reference that read "Mexican Americans *See* Hispanics" in the index under *M.*

@IxSeeAlso — Similarly requires two strings of text in quote marks separated with a comma:

```
@IxSeeAlso["minorities","ethnic groups; Hispanics"]
```

This results in a *See also* reference that read "Minorities *See also* ethnic groups; Hispanics" in the index under *M.*

@IxRefUnder — Requires that you embed this command in another indexing command. Like this:

```
@IxRef[@IxRefUnder[twentieth] 20th-Century Fox]
```

@IxRange — Requires a tag name, a comma, and then the index term in quote marks. Like this:

```
@IxRange[tagname,"philosophers, ancient"]
```

# A Note to FinalWord Users

Sprint is built on the foundation originally laid by FinalWord II. Even though Sprint's formatting capabilities and user interface are much more powerful than FinalWord's, the original @-sign command set from FinalWord II is still functional (but not documented here). For example, FinalWord users are accustomed to using the @NoHinge command to prevent page breaks, while Sprint uses the @KeepFollowing command. Similarly, FinalWord uses the @Big command where Sprint uses @Large. Both @NoHinge and @Big will work just fine in Sprint.

If you already know FinalWord II or if you have documents that already contain FinalWord @-sign commands, you should have virtually no conversion to do. We encourage FinalWord users, however, to learn the Sprint commands since only Sprint commands are documented and used in our examples—and there are many Sprint commands that are new (and better) than FinalWord II.

C H A P T E R

# 4

# Running Sprint from DOS

This chapter describes the several Sprint commands (and variations of those commands) you can issue from the DOS command line—that is, without having Sprint itself running.

The three main things you can do from DOS command line are

- start Sprint
- print Sprint documents
- run Sprint's SprintMerge utility

**Note to two-floppy system users:** Make sure your Program Disk is in Drive A before running Sprint from DOS.

## Running Sprint

There is one command issued from the DOS command line that every Sprint user must know:

SP *Enter*

This command starts Sprint. It's as simple as that.

There are two variations on this basic command—one is common, and the other is rarely used.

The common variation looks like this:

SP *FILENAME.EXT* *Enter*

This command tells Sprint you want to open a particular Sprint document (FILENAME.EXT) immediately after starting Sprint itself.

You can also open several files at once from DOS by typing

```
SP FILENAME1 FILENAME2
```

In this case, Sprint opens each of the files in turn and puts each in its own window. You can open as many files as fit on the DOS command line, but Sprint assigns only the last six files to their own windows.

Note that if document is already in the swap file, Sprint simply redisplays it from there and *does not read it in from disk.*

If you don't include an extension to the file name, Sprint first assumes the extension is .SPR and looks for a file with that name. If it can't find such a file, it looks for a file with no extension, and the name you typed.

The other variation of the SP command involves a far less common need. If you have installed Sprint to work with more than one monitor type (let's say you have an EGA monitor that works in both 25-line and 43-line modes), you can tell Sprint which monitor to expect. To do this you include the screen driver file name that the SP-SETUP program assigned it (these files always end with .SPS).

For example,

```
SP -S=<FILENAME.SPS> Enter
```

This command tells Sprint to use the information in the .SPS file to display correctly for a particular monitor type.

Remember, use this command only if you have already installed Sprint for that monitor using the SP-SETUP program.

# Printing from DOS

The most common way to print a Sprint document is to choose **Print/Go** from the Sprint menus. If, however, you are not running Sprint but want to quickly print a Sprint document (like when your boss suddenly needs another copy of a memo for a meeting that's starting in 10 seconds), you can set the wheels in motion right from the DOS command line.

All you do is type

```
SPFMT FILENAME.EXT Enter
```

FILENAME.EXT stands for, of course, the actual name of the file you want to print.

When you enter this command, the Sprint formatter takes over, looks through your file for errors and, if it encounters no difficulties, immediately starts to print your document to the default printer.

That's all there is to it—most of the time. Sometimes, you want to have more control over your printing options, just as you do when you're printing from within Sprint. In those cases, you can use the DOS command-line options.

## *Command-Line Options*

As you no doubt know by now, Sprint is no slouch when it comes to options. Sprint is one of the most flexible and feature-packed word processors available. So it should be no surprise that even when you are printing documents from DOS (something in itself that virtually no other word processor lets you do), there are a number of printing options you can make use of.

To see the list of options, type

SPFMT *Enter*

In other words, don't type a file name this time. When you enter the SPFMT command without a file name, the Sprint formatter displays a screen that looks like this.

```
Sprint Formatter   Copyright (c) 1988  Borland International, Inc.
Usage to format and print <file>.spr is:
spfmt {options} <file>

options:
  -#              number of copies to print (e.g. -2 for two)
  -dname{value}   define variable 'name'
  -f              fill paragraphs that have hard returns between lines
  -f{=name}       use alternate font
  -l              send error messages to <file>.LOG
  -o{=name}       send output to name.PRN
  -p              print unformatted
  -p#             print during pass # (1, 2, or 3)
  -p=name         use alternate printer (requires name.SPP)
  -page=#         print only page #
  -pause          pause to insert each page in printer
  -start=#        start printing with page #
  -stop=#         stop printing after page #
  -s{=name}       use alternative screen (requires name.SPS)
  -t=#            set default tab stop every # characters
  -v              preview formatted output
```

Figure 4.1: Sprint SPFMT Command-Line Options

This table summarizes the each of the command-line options available to you. Notice that each option starts with a hyphen, and that the options go before the file name. Let's look at each of them in a little more detail.

## –# (Number of Copies)

At the DOS prompt, you can enter the option –#, where # stands for the number of options you want printed.

For example,

```
SPFMT -2 MEMO.SPR Enter
```

prints two copies of the file called MEMO.SPR.

The menu equivalent for this command-line option is choosing **Print/Options/Number of Copies**.

## –dname{=value} (Defining Variables)

Using this option at the DOS command line lets you define a variable with a particular name and set it equal to a particular value.

For example,

```
SPFMT -2 MEMO.SPR -dproduct=SuperGloop Enter
```

prints two copies of the file called MEMO.SPR and defines the variable called *product* as *SuperGloop*.

This option lets you print slightly different versions of your documents without having to actually change anything in the file at all. **Note:** The variable has to already be defined in the document for this to work (in this case, the variable called *product* has to have already been defined).

The menu equivalent for this command-line options is **Insert/Define Text Variable**.

## –f (Fill Paragraphs)

Using this option at the DOS command line lets you choose to have Sprint ignore any single hard returns at the end of lines. Instead, Sprint will wordwrap the lines (using the setting of the **Print/Advanced Options/Wordwrap ASCII File** command) until it finds two hard returns in a row, which signals an end of paragraph.

For example,

```
SPFMT -f BBS.SPR Enter
```

prints the file called BBS.SPR so that single hard returns are ignored. Typically, you would use this command if you have received a file through telecommunications so that each line has a hard return but want to print it in continuous paragraphs.

The menu equivalent for this command-line option is choosing **Style/Modify** and then adding the *Fill yes* parameter.

## –f=name (Use Alternative Font)

This option lets you print a file in a font of your choosing.

For example,

```
SPFMT -f=Condensed REPORT.SPR Enter
```

prints the file called REPORT.SPR in a font called Condensed.

If your printer doesn't support the specified font, Sprint displays a warning message when it formats your file and prints your file in the default font.

Using the font option changes the entire document into the specified font, even if you had marked certain parts of the document to print in special fonts.

## –l (Create .LOG File)

Using this option at the DOS command line lets you write to file any error or warning messages that the formatter displays as it goes through your file.

For example,

```
SPFMT -l MYBOOK.SPR Enter
```

begins formatting the file called MYBOOK.SPR and—if there are any error messages generated—creates a file called MYBOOK.LOG and writes the messages to it.

The advantage of saving error messages to a .LOG file is that this facilitates fixing the errors, since you can then easily go back to the .LOG file to note the line numbers the errors were on.

The menu equivalent for this command-line option is choosing **Print/Advanced Options/Log Errors to File**.

## –o{=name} (Send Output to a File)

Normally, Sprint formats and sends the file to a printer. You can send your output to a file instead.

For example,

```
SPFMT -o MYBOOK.SPR Enter
```

prints the document called MYBOOK.SPR to a file instead of the default printer. Sprint automatically creates the file and names it (in this case) MYBOOK.PRN—in other words, the file name with the .PRN extension tacked on to it.

If you want a different name assigned to the print file, use this format:

```
SPFMT -o=TEST1.PRN MYBOOK.SPR Enter
```

This prints the document called MYBOOK.SPR to file called TEST1.PRN.

The menu equivalent for this command-line option is choosing **Print/Destination** and toggling to File.

## –p (Print Unformatted)

You can use this option to print your Sprint documents without formatting. In other words, instead of having Sprint interpret your formatting commands, it prints them as is. For example, if have marked a word to print in bold in your file, ordinarily it comes out **like this**. But if you print unformatted, the bold words print not in bold, but with the special control codes showing, ^Blike this^N.

For example,

```
SPFMT -p MYBOOK.SPR Enter
```

prints the document called MYBOOK.SPR without interpreting the formatting codes. Instead, the codes print out—even if they aren't visible when you view your file.

The menu equivalent for this command-line option is choosing **Print/Advanced Options/Formatted Print** and toggling to No.

## –p# (Number of Passes)

You can specify the number of passes that the Sprint formatter should perform prior to printing by using the –p# option.

For example,

```
SPFMT -p3 MYBOOK.SPR Enter
```

tells the formatter to go through the document called MYBOOK.SPR three times before it actually starts printing.

The formatter performs the following functions during each pass:

**1 Pass** The formatter prints while it formats, printing "???" for any forward reference it finds.

**2 Passes** The formatter goes through your document once and records any cross-references, then it goes through the file again replacing the **V**ariable and **R**eference a Tag commands with the appropriate variable or tag name. Then it prints.

**3 Passes** The formatter does the same two passes but then does a third "cleanup" pass to double-check all references.

You don't have to use three passes unless you have very long documents that have a lot of formatting commands (like the manual you're reading).

If you use one pass, and you print a file that has tags, the references print with three question marks instead of the right reference. For example, you might get "See page ???" instead of the correct page number. If you aren't concerned about incomplete references (perhaps you're printing a first draft), printing with one pass instead of two can save time.

For the quickest printing, you should use only the minimum number of passes necessary.

The menu equivalent for this command-line option is choosing **P**rint/ **A**dvanced Options/**N**umber of Passes.

## –p=name (Using Another Printer)

You can use this option to select an alternate printer to print to.

For example,

```
SPFMT -p=EPSON.SPP MYBOOK.SPR Enter
```

tells the formatter to print the document called MYBOOK.SPR using the printer driver file EPSON.SPP.

Before you can use this command, you must have already selected the printer with the SP-SETUP program. When you select the printer, Sprint creates a printer driver file with the extension .SPP, which it needs to properly format your file for output on the desired printer.

This print option overrides an @Printer command that might be in your file.

If you receive an error message when printing from the DOS command line, check the following things:

- Make sure you entered the printer name correctly. If you're not sure what you called the device when you installed it, choose **Current Printer** from the **Print** menu and check the list of printers displayed on the screen.
- Confirm that the printer driver file ( it ends with .SPP) is on your disk. If you're using a hard disk or multiple directories, make sure the printer definition file is either in the current directory, or in a directory on the system search path.
- Make sure you installed the printer with the SP-SETUP program. To check this, choose **Current Printer** from the **Print** menu and verify that the list of alternate printers includes the desired printer name. If it's not there, the printer was never installed for use with Sprint.

The menu equivalent for this command-line option is choosing **Print/ Current Printer** and then choosing an alternative printer.

## -page=# (Prints a Single Page)

This option lets you print a single page (and only that page) of a document.

For example,

```
SPFMT -page=50 MYBOOK.SPR Enter
```

prints just page 50 of the document called MYBOOK.SPR.

Note that the formatter still goes through the entire document at least once even when you are printing a single page.

The menu equivalent for this command-line option is choosing **Print/ Options/Starting Page** and then making the setting the same as **Print/ Options/Ending Page**.

## -pause (Pause between Pages)

This option makes the formatter stop printing after every page and wait for you to tell it to print the next page.

For example,

```
SPFMT -pause MYBOOK.SPR Enter
```

prints the entire document called MYBOOK.SPR but waits after every page for you to tell it to proceed.

This option is useful if you are hand-feeding the paper to your printer.

The menu equivalent for this command-line option is choosing **Print/Options/Pause Between Pages** and toggling to **Yes**.

## –start=# (Starting Page #)

This option lets you specify which page of your document to start printing from.

For example,

```
SPFMT -start=2 MYBOOK.SPR Enter
```

prints the document called MYBOOK.SPR starting on page 2.

The menu equivalent for this command-line option is choosing **Print/Options/Starting Page**.

## –stop=# (Ending Page #)

This option lets you specify at which page of your document to stop printing.

For example,

```
SPFMT -stop=3 MYBOOK.SPR Enter
```

stops printing the document called MYBOOK.SPR after page 3.

The menu equivalent for this command-line option is choosing **Print/Options/Ending Page**.

## –s{=name} (Use Alternative Screen)

This option lets you use an alternative screen for your print session.

For example,

```
SPFMT -s=EGA43.SPS -v GRAPH.SPR Enter
```

prints the file called GRAPH.SPR to the screen (not to a printer), but uses the monitor called EGA43.SPS.

Note that you have to already have created a screen driver file (it ends with .SPS) with the SP-SETUP program for this to work.

## –t=# (Set Tab Stops)

This option lets you set the number of spaces each tab stop should be expanded to when the document prints.

For example,

```
SPFMT -t=12 BBS.SPR Enter
```

prints the file called BBS.SPR and formats every tab stop as the same as 12 space characters.

You would normally use this option only when you're dealing with ASCII files, which have no rulers.

The menu equivalent for this command-line option is choosing **Print/Customize/ASCII File Handling/Tab Expansion**.

## –v (Preview File)

This option lets you preview your formatted document on the screen without sending it to a printer.

For example,

```
SPFMT -v MYBOOK.SPR Enter
```

formats the file called MYBOOK.SPR and displays it (as well as it can) on your screen.

The menu equivalent for this command-line option is choosing **Print/Screen Preview**.

# Merging Files from DOS

Sprint allows you to access the SprintMerge utility directly from the DOS command line, which merges a form letter with a data file. Here's a command line that shows all the options:

```
SPMERGE LETTER.SPR -record RECORD.REC options -out FILENAME -print
```

*Enter*

Here's what the entries in the command line mean:

- SPMERGE starts the Merge program.
- LETTER.SPR is your form letter file.
- LETTER.SPR specifies the letter file containing your form letter. If you don't specify a letter file name, the SprintMerge program looks for the file DEFAULT.SPR.
- -R tells SprintMerge to expect a record file next.
- RECORD.REC specifies your record file. If you don't specify a template file name, the Merge program looks for the file DEFAULT.REC.
- OPTIONS specify actions that should be performed in addition to filling in your form letters:
  - -SELECT filters out certain records based on your specifications.
  - -SORT arranges your records according to a specified field.
  - -OUT or O creates an intermediate output file with the file name that you specify. If you don't specify a file name with this command, the SprintMerge program uses the default output file SPM.O$$. If sorting occurs, SprintMerge copies this file into the default sort file SPM.S$$.
  - -PRINT indicates that you want to use Sprint's print options, which you should type in after this switch.
  - -STARTREC and -ENDREC specify the record to begin and end within a particular file.
  - -SORTFILE specifies a temporary sort file other than the default sort file SPM.S$$.
  - -REMDITTO removes the double quotation marks (aka, ditto marks) commonly found in BASIC-type database files.

These commands in the command line override any similar commands in your record.

To display the usage and options list on your screen, type SPMERGE and press *Enter*. Here's what you'll see:

```
Main Options
-select selspec       operate on only a subset of the records
-sort sortspec        process the records in a specified order

Output Options
-out filename         send a copy of output to a file
-o filename           same as -out
-print options        specify more print options at end of command line
-p options            same as -print

Other Options
-startrec n           ignore the first n-1 records
-endrec n             ignore all records after record n
-remditto             remove quotation marks from input fields
-record filename      specify name of record file
-r filename           same as -record
-sortfile filename    specify name of temporary sort file
```

Figure 4.2: SprintMerge Command-Line Options

**Note:** To stop the SprintMerge program at any time, press *Ctrl-C*. Command lines cannot have more than 128 characters.

For specific instructions and a tutorial on the SprintMerge utility, refer to the Sprint *User's Guide*.

A P P E N D I X

# A

# Formatter Error Messages

The formatter produces error messages that look like this:

```
document.spr 143 Error: Begin Hyphens on line 122 missing End.
```

where "document.spr" is the name of the file that the formatter found the error in and "143" is the line number that the error was found on in that file.

(In this instance, there is further information, namely that line 122 was the beginning of the unended command. Since the formatter processes its input files from beginning to end, this error message indicates that that the formatter got up to line 143 by the time it saw another contradictory command, which indicated there must have been a missing End command to the command that started on line 122. Thus, in this case, the error is probably closer to line 122 than line 143.)

By using the Jump to Line command in the editor (press *F9*), you can look at this line.

The word "Error" may instead say "Fatal" or "Warning," depending on the severity of the error. Warning errors are advisory messages only and can sometimes be ignored (printing will proceed if only Warnings are issued). Warnings indicate minor document problems, such as a missing reference tag or a font substitution.

"Fatal" errors, however, indicate that the formatter is unable to continue, such as when it runs out of memory.

Normal errors do not stop the formatter from completing the current pass. If the file already printing, the output might not be correct.

*For advanced users of MS-DOS: the formatter returns a value (the exit code) of 0 if it is successful, 1 if there are any errors other than warnings, and 100 if it is interrupted by a ^C or ^Z. A "shell" program can examine this return value to see what the formatter did.*

***command* can't have an argument.**
(error) The built-in command you just used is not allowed to have delimiters and an argument with it.

***command* may have missing close delimiter before it.**
(error) While processing the arguments to one command, the formatter encountered another command. This second command is not one of the ones (like @Value) that are evaluated automatically when supplying arguments to the first command. You probably forgot to put a close delimiter to the argument list of the first command.

***filename* is invalid.**
***filename* is version *n.n*, need version *n.n*.**
(fatal) More file opening and database errors about .SPS files.

***name* in *command* missing definition.**
***'name'* may have missing comma before it.**
(error) While reading fields for a command (as opposed to a format that affects regions), the formatter found missing fields, or, more likely, you left out delimiters on the previous field. For example, @Template(page=) will cause this.

***n* errors found.**
(warning) Generic warning message that tells you how many errors the formatter found.

***parameter* not allowed in Begin.**
(error) Some parameters allowed in @Style and @Define are not allowed in @Begin, namely, *Numbered, Counter Increment, Within, Divider, Initialize, AfterEntry, BeforeEach, WithEach, BeforeExit, AfterExit, Inline, Free, Before, After, Index,* and *Column.*

***'character'* may have missing comma before it.**
(error) The formatter encountered a character when it was expecting a comma. Since it did so while processing a command, the most likely cause is that you were missing a comma after the previous parameter.

**'@*command*' already defined.**
(error) You have defined a command more than once. Remove one of the Define commands, or use Modify for the second. This error can also be caused by creating a printer driver with a font or attribute with the same name as a command.

**'*name*' already has a parent.**
**'*name*' already has a template.**
(warning) Only one @Template command is allowed for any single variable in a document. The first template will be the only one stored. Use @Value(variable, template "...") for your less common templates and @Template for your most frequently used.

**'*name*' given more than once to *command*.**
(error) @Macro definitions can have named arguments, and you can call the macros by giving their names. You gave one of them more than once. For example,

```
@Macro(RIndex(tag,word) = "... do R-type Index...")

@RIndex(tag="5", word="foo bar", tag="6")
```

produces the error message `Tag given more than once to RIndex`.

**@*command* on line *nnn* missing '*delimiter*' in argument.**
(error) The close delimiter to a string argument to some command is missing. The argument was started on line *nnn*.

**@O too complex.**
(error) You can't put more than about 60 characters inside an @O or O command. (This is rarely a problem since you can't read that much overprinting anyway.) You must have made a delimiter matching mistake.

**@*command* on line *xxx* missing '*delimiter*'.**
(error) You just used an @-sign command with an open delimiter, and the matching close delimiter was not found. This error usually occurs when all other intervening commands have been processed, and the formatter discovered that it was still waiting for a close delimiter. Thus, the error message will come out on some later line, but notice that it refers you to the line the command started on.

***name*.spp is invalid.**
(fatal) The printer description appears to have garbage in it. Rerun SP-SETUP.

***name*.spp is version....**
(fatal) The printer driver was created with an old version of SP-SETUP. Only use the current version.

***name* not allowed in Begin.**
(error) The given format parameter cannot be changed in a Begin command. *BeforeEach* and other string parameters cannot be changed this way.

**^*x* on line *nnn* missing ^N.**
The format/command was opened with a control character (^O, ^A, etc.) but was never closed. Notice that formats must be opened and closed in the same source file.

**Bad .spp file.**
(error) The formatter could not get the information it needed from the printer definition file. Are you sure you ran the same version of Sprint as the version of the Sprint installation program (SP-SETUP)? The printer description (*name*.SPP) could also be bad. Usually this indicates that the file was truncated, perhaps by a file-transfer program.

**Balancing columns may have lost some text**
(error) There was an internal formatting error that should be reported to Borland's Technical Support department.

**Begin 'name' on line *nnn* missing End.**
(error) The format/command was opened with a Begin command, but was never closed. Notice that formats must be opened and closed in the same source file (they cannot extend past @Include boundaries, but can enclose them).

**Can't create *'filename'*.**
(warning) This happens when purging memory when attempting to create temporary files like TOC.$$$.

**Can't create *filename, reason.***
(error) For some DOS reason (listed as *reason*), the formatter couldn't create the log file named *filename*. Ensure legality of the name you gave it with the "-l=name" switch, make sure you have enough disk space, confirm that the disk name is valid and that the floppy door is closed, or fix whatever the *reason* of the error is.

**Can't find *'name*.spp'.**
(fatal) If this says 'default.spp', you forgot to run SP-SETUP, or you changed directories or disks after running it. If it says something else, you named a printer with the –p=switch that has not been selected yet by SP-SETUP.

**Can't modify *'name'* after use.**
(error) An attempt is being made to modify a command that has already been invoked at least once in this document. Format definitions cannot be changed between invocations (since they must be remembered for the next pass). Either move the Modify command to the top of the file or define a new format with the modifications (use the Define command).

**Can't open *filename, reason.***
(fatal) This error can appear whenever the formatter can't open a file. The *reason* is the reason DOS gave for being unable to open the file. For example, if it reads *filename*.SPP cannot be opened, you must have used the "-p=XXX" switch on the command line. There is a message that says the same thing for .EPS files, and so forth.

**Can't open output file *'name'*.**
(fatal) The file name supplied to a –o switch is illegal.

**Can't open STANDARD.FMT.**
(fatal) Similar to above, but has more dire consequences.

**Can't print a *'c'*.**
(warning) This character was encountered in the input, but the formatter does not know how to print it on the printer, and there is no TCT table entry for it. It is replaced by a space.

**Can't run *"command"* at exit – *reason.***
(error) The @AtExit argument is searched for as soon as the command is encountered. Either the argument isn't a program, or it could not be found. The message from DOS (the *reason*) will tell you why it couldn't be found.

**Command *'name'* not found.**
(error) @Macro(a = b) was attempted, and there is no command called b.

**Device cannot overprint.**
(warning) An attempt was made to overstrike, boldface, underscore, or strikeout on a printer type that cannot back up to do this.

**Download file *'filename'* not found.**
(warning) You did @Escape(f="...") and there was no file of that name in the current directory or anywhere on the path.

**End *command* has no Begin.**
(error) An End command was encountered unexpectedly. The formatter found @End(xxx) (or ^OEND xxx^N) without a matching @Begin (or ^OBEGIN xxx^N).

**File *'filename'* created.**
(warning) This happens when purging memory when attempting to create temporary files like TOC.$$$.

**Format ignored here.**
(warning) An @Format or Style Sheet command appears after the first printing text in the file. You must move it above the first printing text.

**Format *'name'* not found.**
(error) @Define(a = b), @Modify(b), or @Place(b) was attempted, and b is not a format name.

**Freeing memory.**
(warning) Occurs in conjunction with "Writing TOC.$$$" type of messages. When formatting large documents or documents with complicated Escape commands, the formatter often uses large amounts of memory and tries to free up extra memory whenever it can.

**Group too big for page.**
(warning) A Group Together on Page command has been done around too large of a block of text. Note that because many other commands (like Table) use the group attribute, this warning does not necessarily signal a problem with the actual use of Group Together on Page or the @Group command.

**HeaderSpacing>TopMargin; enlarging margin.**
**FooterSpacing>BottomMargin; enlarging margin.**
(warning) *TopMargin* is measured from top of page to top of text, and *HeaderSpacing* is the distance from the bottom of the header to the top of the text. Therefore, the top margin must be able to enclose the header distance. So too with footers. If the user makes the header or footer too big, the formatter tries to fix the problem by enlarging the top or bottom margin.

**Illegal character found following ^O.**
(error) Caused by some other control character following a Control-O. Usually this means that you (or your user interface) overlapped some control character commands.

**Internal command nesting error.**
**Internal command reading error.**
**Internal macro execution error.**
**Internal font switching error.**
(error) There was an internal error that should be reported to Borland's Technical Support department.

**Invalid argument name *'variable'*.**
(error) Just like variables in @Set, variables in a @Macro argument list must start with letters.

**Invalid expression.**
(error) When used in @Set(*var=<arithmetic expression>*), there was some syntax error in the expression.

**Invalid number *nnn*.**
(error) When used in @Set(var=*nnn*), the *nnn* wasn't a number.

**Invalid tct entry.**
(error) You can't use the TCT command with control characters or @-sign commands.

**Invalid variable name *'variable'*.**
(error) Variable names must begin with letters.

**Invalid *parameter*.**
(error) The value given for the command argument doesn't make any sense. For instance, "@Style(paper = "text")" will produce this error.

**Memory manager error.**
(error) There was an internal memory management error that should be reported to Borland's Technical Support department.

**Missing comma before *'word'*.**
(error) Caused by a missing comma in the argument list to a command, or by extraneous punctuation or spaces in the middle of a variable name when supplied to a command.

**Missing argument for *@command*.**
(error) The command requires an argument in delimiters. This error can be caused by a space between the command name and the open delimiter.

**Missing argument to macro?**
(error) A macro is attempting to do @Eval(text) (or just @Eval), and discovers that there's nothing there. It assumes that the user must have used a macro needing an argument, but didn't supply one.

**Missing variable name.**
(error) You forgot to use a variable name, as in @Value().

**Nested index commands.**
(warning) You cannot include an index command within another index command. This check is done while printing the index, not while processing the index commands, so the error line number is usually at the end of the document.

**No %%BoundingBox in *filename*.**
***filename* is not an EPS file.**
(error) These messages all come from the EPS reader. If it finds an error, it does its best to complain intelligently. Because there's a standard format for EPS files, if you don't follow it, you may get the "non-EPS-file" error. Minimally, an EPS file must have a %%BoundingBox comment in its header and must have %! as its first two characters.

**Numeric overflow.**
(error) Either a number you entered is larger than 65,535, or you tried to measure more than 65,535 printer units. Since most laser printers operate at 300 dots per inch (DPI), your calculations can go up to 210 inches. Even at 2,540 DPI, which is about what the fanciest typesetter can do, you're still free to go to about 25 inches.

**Out of Memory.**
(fatal) Ran out of memory space. Your document is too complicated. A possible cause of this is that you redefined figures as a "floating" format, and spaced them too close together. Try placing some text (a few lines is all that is necessary) between each figure.

A more common cause is a missing close delimiter on a command. All delimiters that have not yet been matched in the input are reported, so that such errors can be found.

Another possibility is that you're using a number of complicated Escape commands—which can greatly tax Sprint's memory management.

This error can also be caused by a very large index. Unfortunately, there is presently no way around this.

**Output file write error.**
(warning) Probably means the disk is full when using the –o switch to send output to a file. Delete something.

**Page headings too big for page.**
(fatal) The page header and footer together fill or overflow the page. Possibly a missing close delimiter to a page header command.

**Page layout may have lost a column of text.**
**Page layout may have lost a galley of text.**
**Page layout may have lost galleys of text.**
(error) There was an internal formatting error that should be reported to Borland's Technical Support department.

**Printer does not have *'name'* font.**
(warning) "@Style(font = *x*)" or "FONT *x*...ENDF" or the "-f=*x*" command or a precise ruler font was given, and *x* is not a font on this printer. The formatter uses the default installed font instead.

**Printer ignored here.**
(warning) A Printer command appears after the first printing text in the file. You must move it above the first printing text.

**Quote mark may have missing comma before it.**
(error) The formatter encountered a quoted string when it wasn't expecting it. Since it did so while processing a command, the most likely cause is that you were missing a comma after the previous parameter.

**Running *'command'* at exit returned code *nnn*.**
(warning) The AtEnd command, which allows you to specify a command to run just before the formatter exits, ran the command successfully, but the command itself returned an error code *nnn*.

**Script must be expressed in lines.**
(error) The *Script* format parameter cannot be given in any units except lines or fractions of lines.

**Tag *'name'* already defined.**
(warning) You aren't allowed to do more than one @Tag(x) with the same name; in other words, tags must reference a single specific counter or location. The second and subsequent tags will be ignored.

**Template expansion too long.**
(error) An @Value command overflowed the buffer used to expand the numerical value into text. The template for that variable was too long and complex. The maximum length of the @Value of a variable is 60 characters.

**Template too long.**
(warning) In an @Value command only, a template cannot be longer than 60 characters. Permanent templates (via @Template) can be much longer. The end of the template will not be executed in this @Value.

**Text in .FMT file.**
(fatal) A printing character or a format invocation was found in the format file. These are not allowed, because the format file is read only during the first pass. Only format definitions, macros, and Style commands are allowed in STANDARD.FMT or any other .FMT file. Anything that actually places a character on the page must be in your manuscripts, not in your .FMT files.

**Too many @/ commands.**
(warning) You set some number of tabs (with @^ or @Tabset) on a line, and gave more than that number of @/ commands on a line.

**Too many formats open at once.**
(fatal) Usually, no more than 16 formats can be open at once. Exceeding this number normally generates the error message `Too many nested commands`. The formatter, however, is smart enough to give you some leeway. If it runs out of room anyway, you might get this message. Reduce the nesting of your commands.

**Too many nested commands.**
(fatal) Too many commands are nested in the input. About 16 levels are allowed.

**Too many nested Include files.**
(fatal) No more than 20 nested Include commands can be handled. Create a "master file" that lists all the Include commands instead of nesting them.

**Too many nested macros.**
(error) There were too many macros nested inside each other. No more than 30 arguments to macros can be present, including all the nested macros that were executed to get to this one.

**Too many tabs.**
(error) Only 30 tabs total are allowed at any one time, in *all* nested formats. (In other words, using the TabDivide 4 command and then doing a TabDivide 6 command within a Numbered format is a total of 10 active tabs while in the Numbered format.) You may want to do a TabSet command to clear all tabs.

**Too many templates.**
(warning) You can't have more than 255 templates for variables.

**Unable to open *'name'*.**
(fatal) The main input file was not found.

**Undefined tag *'name'* used in *filename* line *nnn*.**
(warning) @Ref(x) was done somewhere, but there was no matching @Tag(x). The reference will print as "???".

**Unexpected code *nnn* found while assembling lines of text.**
(error) There was an internal formatting error that should be reported to Borland's Technical Support department.

**Unknown argument *'name'* to *@command*.**
(error) An unknown field name was supplied to a command. For instance, "@style(boing)" will produce this error.

**Unknown command: *@command*.**
(error) The given command is undefined.

**Unknown option *xxx*.**
(error) You typed (probably, you mistyped) *xxx* at the DOS command line, but the formatter did not recognize that as a legal option here. The formatter displays a list of all the valid options when entering commands from the DOS command line.

**Unknown unit of measure *'name'*.**
(error) Not a legal dimension. The legal units of measure are in, inch, inches, cm, mm, point, points, p, pt, pts, pica, picas, em, ems, en, ens, char, chars, character, characters, line, lines, page, u, unit, and units.

**Use eval for macro argument *'argumentname'*.**
(error) The formatter is warning you that you are doing a math expression in a macro, and you cannot delay the binding of a named macro argument. You must do @Eval(*argumentname*) instead.

**Variable *'name'* cannot be changed.**
(error) The variable is *locked* (either it's a tag, a built-in variable such as *hour*, or it was set in the .FMT file), and an attempt is being made to change it, such as with the Set command.

**Variable *'name'* not found.**
(error) The variable, referenced in a command such as **@Value**, is currently undefined.

**Word too long to wrap.**
(warning) A word was too big for the formatter's internal storage, and was treated as two words. Generally this is harmless.

**Write error on *filename*.**
(warning) There was some problem continuing to write data to one of the .$$$ files that the formatter creates to hold TOC, NOTES, or the like. The usual cause is running out of disk space, or, on a network, someone else is in the same directory trashing the TOC.$$$ file.

**Writing *name* to *name*.$$$.**
(warning) The memory is filling up, and the formatter is attempting to free some of it by writing the endnotes, the table of contents, or another end matter to the disk.

A P P E N D I X

# B

# ASCII Character Set

The American Standard Code for Information Interchange (ASCII) is a code that translates alphabetic and numeric characters and symbols and control instructions into 7-bit binary code. Table B.1 shows both printable characters and standard control characters.

You need these decimal numbers in Sprint when you're using the Char command. You also need to note the order of the characters, since the Utilities/Arrange-Sort command arranges according to ASCII order.

Table B.1: ASCII Table

| DEC | HEX | CHAR | | DEC | HEX | CHAR | DEC | HEX | CHAR | DEC | HEX | CHAR |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | ^@ | NUL | 32 | 20 | | 64 | 40 | @ | 96 | 60 | ` |
| 1 | 1 | ^A | SOH | 33 | 21 | ! | 65 | 41 | A | 97 | 61 | a |
| 2 | 2 | ^B | STX | 34 | 22 | " | 66 | 42 | B | 98 | 62 | b |
| 3 | 3 | ^C | ETX | 35 | 23 | # | 67 | 43 | C | 99 | 63 | c |
| 4 | 4 | ^D | EOT | 36 | 24 | $ | 68 | 44 | D | 100 | 64 | d |
| 5 | 5 | ^E | ENQ | 37 | 25 | % | 69 | 45 | E | 101 | 65 | e |
| 6 | 6 | ^F | ACK | 38 | 26 | & | 70 | 46 | F | 102 | 66 | f |
| 7 | 7 | ^G | BEL | 39 | 27 | ' | 71 | 47 | G | 103 | 67 | g |
| 8 | 8 | ^H | BS | 40 | 28 | ( | 72 | 48 | H | 104 | 68 | h |
| 9 | 9 | ^I | TAB | 41 | 29 | ) | 73 | 49 | I | 105 | 69 | i |
| 10 | A | ^J | LF | 42 | 2A | * | 74 | 4A | J | 106 | 6A | j |
| 11 | B | ^K | VT | 43 | 2B | + | 75 | 4B | K | 107 | 6B | k |
| 12 | C | ^L | FF | 44 | 2C | , | 76 | 4C | L | 108 | 6C | l |
| 13 | D | ^M | CR | 45 | 2D | - | 77 | 4D | M | 109 | 6D | m |
| 14 | E | ^N | SO | 46 | 2E | . | 78 | 4E | N | 110 | 6E | n |
| 15 | F | ^O | SI | 47 | 2F | / | 79 | 4F | O | 111 | 6F | o |
| 16 | 10 | ^P | DLE | 48 | 30 | 0 | 80 | 50 | P | 112 | 70 | p |
| 17 | 11 | ^Q | DC1 | 49 | 31 | 1 | 81 | 51 | Q | 113 | 71 | q |
| 18 | 12 | ^R | DC2 | 50 | 32 | 2 | 82 | 52 | R | 114 | 72 | r |
| 19 | 13 | ^S | DC3 | 51 | 33 | 3 | 83 | 53 | S | 115 | 73 | s |
| 20 | 14 | ^T | DC4 | 52 | 34 | 4 | 84 | 54 | T | 116 | 74 | t |
| 21 | 15 | ^U | NAK | 53 | 35 | 5 | 85 | 55 | U | 117 | 75 | u |
| 22 | 16 | ^V | SYN | 54 | 36 | 6 | 86 | 56 | V | 118 | 76 | v |
| 23 | 17 | ^W | ETB | 55 | 37 | 7 | 87 | 57 | W | 119 | 77 | w |
| 24 | 18 | ^X | CAN | 56 | 38 | 8 | 88 | 58 | X | 120 | 78 | x |
| 25 | 19 | ^Y | EM | 57 | 39 | 9 | 89 | 59 | Y | 121 | 79 | y |
| 26 | 1A | ^Z | SUB | 58 | 3A | : | 90 | 5A | Z | 122 | 7A | z |
| 27 | 1B | ^[ | ESC | 59 | 3B | ; | 91 | 5B | [ | 123 | 7B | { |
| 28 | 1C | ^\ | FS | 60 | 3C | < | 92 | 5C | \ | 124 | 7C | \| |
| 29 | 1D | ^] | GS | 61 | 3D | = | 93 | 5D | ] | 125 | 7D | } |
| 30 | 1E | ^^ | RS | 62 | 3E | > | 94 | 5E | ^ | 126 | 7E | ~ |
| 31 | 1F | ^_ | US | 63 | 3F | ? | 95 | 5F | _ | 127 | 7F | |

Table B.1: ASCII Table, continued

| DEC | HEX | CHAR | DEC | HEX | CHAR | DEC | HEX | CHAR | DEC | HEX | CHAR |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 128 | 80 | Ç | 160 | A0 | á | 192 | C0 | └ | 224 | E0 | α |
| 129 | 81 | ü | 161 | A1 | í | 193 | C1 | ┴ | 225 | E1 | β |
| 130 | 82 | é | 162 | A2 | ó | 194 | C2 | ┬ | 226 | E2 | Γ |
| 131 | 83 | â | 163 | A3 | ú | 195 | C3 | ├ | 227 | E3 | π |
| 132 | 84 | ä | 164 | A4 | ñ | 196 | C4 | ─ | 228 | E4 | Σ |
| 133 | 85 | à | 165 | A5 | Ñ | 197 | C5 | ┼ | 229 | E5 | σ |
| 134 | 86 | å | 166 | A6 | ª | 198 | C6 | ╞ | 230 | E6 | µ |
| 135 | 87 | ç | 167 | A7 | º | 199 | C7 | ╟ | 231 | E7 | τ |
| 136 | 88 | ê | 168 | A8 | ¿ | 200 | C8 | ╚ | 232 | E8 | Φ |
| 137 | 89 | ë | 169 | A9 | ⌐ | 201 | C9 | ╔ | 233 | E9 | θ |
| 138 | 8A | è | 170 | AA | ¬ | 202 | CA | ╩ | 234 | EA | Ω |
| 139 | 8B | ï | 171 | AB | ½ | 203 | CB | ╦ | 235 | EB | δ |
| 140 | 8C | î | 172 | AC | ¼ | 204 | CC | ╠ | 236 | EC | ∞ |
| 141 | 8D | ì | 173 | AD | ¡ | 205 | CD | ═ | 237 | ED | ∅ |
| 142 | 8E | Ä | 174 | AE | « | 206 | CE | ╬ | 238 | EE | ∈ |
| 143 | 8F | Å | 175 | AF | » | 207 | CF | ╧ | 239 | EF | ∩ |
| 144 | 90 | É | 176 | B0 | ░ | 208 | D0 | ╨ | 240 | F0 | ≡ |
| 145 | 91 | æ | 177 | B1 | ▒ | 209 | D1 | ╤ | 241 | F1 | ± |
| 146 | 92 | Æ | 178 | B2 | ▓ | 210 | D2 | ╥ | 242 | F2 | ≥ |
| 147 | 93 | ô | 179 | B3 | │ | 211 | D3 | ╙ | 243 | F3 | ≤ |
| 148 | 94 | ö | 180 | B4 | ┤ | 212 | D4 | ╘ | 244 | F4 | ⌠ |
| 149 | 95 | ò | 181 | B5 | ╡ | 213 | D5 | ╒ | 245 | F5 | ⌡ |
| 150 | 96 | û | 182 | B6 | ╢ | 214 | D6 | ╓ | 246 | F6 | ÷ |
| 151 | 97 | ù | 183 | B7 | ╖ | 215 | D7 | ╫ | 247 | F7 | ≈ |
| 152 | 98 | ÿ | 184 | B8 | ╕ | 216 | D8 | ╪ | 248 | F8 | ° |
| 153 | 99 | Ö | 185 | B9 | ╣ | 217 | D9 | ┘ | 249 | F9 | ∙ |
| 154 | 9A | Ü | 186 | BA | ║ | 218 | DA | ┌ | 250 | FA | · |
| 155 | 9B | ¢ | 187 | BB | ╗ | 219 | DB | █ | 251 | FB | √ |
| 156 | 9C | £ | 188 | BC | ╝ | 220 | DC | ▄ | 252 | FC | ⁿ |
| 157 | 9D | ¥ | 189 | BD | ╜ | 221 | DD | ▌ | 253 | FD | ² |
| 158 | 9E | ₧ | 190 | BE | ╛ | 222 | DE | ▐ | 254 | FE | ■ |
| 159 | 9F | ƒ | 191 | BF | ┐ | 223 | DF | ▀ | 255 | FF | |

# Index

% (glossary symbol) 93
-p (DOS print option) 359
headings
  unnumbered 107
templates 257
! command 326
"" (quotes) delimiters 331
$$$ files 358
% (template symbol) 316
' ' (single quote) delimiters 331
‘ ’ (single quote) delimiters 331
( ) (parentheses) delimiters 331
{} (braces) delimiters 331
* (asterisk)
  in Pick from List 181
  in status line 208, 226
  search wildcard 213
-# (DOS print option) 346
-d (DOS print option) 346
-f (DOS print option) 346
-f=name (DOS print option) 347
-l (DOS print option) 347, 359
-o (DOS print option) 348, 363
-p (DOS print option) 348
-p# (DOS print option) 348
-p=name (DOS print option) 349
-page=# (DOS print option) 350
-pause (DOS print option) 350
-s (DOS print option) 351
-start=# (DOS print option) 351
-stop=# (DOS print option) 351
-t=# (DOS print option) 352
-v (DOS print option) 352
/ (slash) character 326
; (glossary symbol) 93
; (one-line comment command) 42
< (less than) command 327
<> (angle brackets) delimiters 331
> (greater than) symbol 333
? (question mark)
  search wildcard 214
??? (undefined tags) 70
@ signs, printing 333
@' command 61
@* command 61
@, command 61
@-sign commands **329**
  ASCII files 18
  entering 329
  errors 70
  FinalWord 342
  nesting 331
  syntax 340
  vs. menus 334
  when to use 334
  with no menu equivalents 61, 340
@; command 61
@@ command 333
@` command 61
@~ command 61
[ (ruler mark) 201
[] (brackets)
  delimiters 331
  search wildcard 215
\ (backslash), search wildcard 215
] (ruler mark) 201
^ (caret), search negator 215
^] character 100
^^ (double caret) 221
^_ character 99
^J character 99
^L character 121
^M character 99

## A

Above parameter 149, 154
AbovePage parameter 149
accented letters 301
Add to Dictionary command 224
Address command **269**
Advanced Options menu **10**, 190
After parameter 154, 302
AfterEntry parameter 154
AfterExit parameter 154
alignment *See* justification
All Pages command 81, 100
Allen, Woody 117
Also See command 114
Alt-Z 208
alternative user interfaces 49, *See also*
  User Interface menu
  reverting 249
  shortcuts 8
apostrophes *See* ' ' (close quote)
  delimters
Appendix command **12**

Appendix variable 256
AppendixSection command **13**
AppendixTitle variable 256
Arrange-Sort command **14**, 367
ASC extension 15
ASC files 15
Ascending Order command 14
ASCII
  characters
    embedding 278
    printing 276
    sort order 14, 223
    table 367
  codes 367
  files **15**
    @-sign commands in 334
    creating 17
    hard returns 100
    indenting 18
    pure 16
    rulers 17
    tabs 153, 229, 234
    vs. Sprint files 243
    wordwrapping 11
ASCII File Handling menu 15, 49
Assign to Key command 92
Assigned Number command 237
Asterisks command 26, 134
at-sign commands *See* @-sign commands
AtEnd command 61, 364
AtExit command 359
Auto-Indent command 15, **18**
AUTOEXEC.SPG 94, 199
AutoSpell command **19**, 223

## B

Background Save Period command **20**, 167
backup file 3, 20, 167
  saving 167, 186
Bar command 95
BASIC 353
Before parameter 154
BeforeEach parameter 154
BeforeExit parameter 154
Begin and End commands 268, **270**, 330, 341
  error messages 358
  errors messages 69
Below parameter 149, 154
BelowPage parameter 150
Big command 342
binary numbers 278
binding margins 60, 138, 141, 152
Blank Page(s) command **21**, 171, 196
blank space (horizontal) **22**, 287, 311
blank space (vertical) 195, 299
BlankLines parameter 150
Block Select menu **23**, 65
blocks *See also* Block Select menu
  cutting 157
  deleting 51
  formats 87
  moving 157
  quotations 303
  saving 265
  selecting **23**
    by lines or paragraphs 156
    column mode 24
    extending by typing 25
    mouse commands 155
  typestyles 246
Bold command 244
Borland, contacting 4
Borland, Frank, biography 279
Bottom Margin command 59, 138
Bottom Status Line command 208, 225
BottomMargin parameter 150
BottomOfFile command 183
BottomOfScreen command 183
BoundingBox comment 362
BOX command 95, 96
boxes 95
bulleted lists **26**
  nesting 28
Bullets command 26, 134

## C

C (ruler mark) 203
canceling **29**, *See also* deleting
capitalization, searching 211, 213
Caption command 71
captions
  figures **30**

tables 231, **238**
Case command **271**
with FMT files 275
Case Sensitive command 213
CaseRotate command 183
CaseSwitch command 183
Center command 32, 226
Centered parameter 150
centering text **31**, 123, 226
tabs 183
vertically 241
CenterPage command 241
CenterTab command 32, 183
centimeter dimension 58
Change Directory command 74
Chapter command **33**, 300
chapter titles in endnotes 300
Chapter variable 83, 256
ChapterTitle variable 256
Char command **276**, 367
Character Size command 80, 245
character translation table 313
characters (dimension) 58
CHDIR (DOS command) 74
CJ (ruler mark) 203
clearing menus 29
Clipboard **34**, 47, 54, 121, 158
appending to 35
defined 25
Close command **35**, 76
Closing command **278**
closing files *See* files, closing
closing windows *See* Window menu
Codes command **37**, 44, 207
Col in status line 226
Colors command **38**, 49
Column Break command 41
Column command 42, **279**
Column Mode command 24
column number in status line 226
Column parameter 150
columns **39**
creating with tabs 311
headings 104, 105
parallel 279
snaking 150
COLUMNS command 39
Columns menu 39, 127
Columns parameter 40, 150
commands *See also* formats
functional index of 5
comments *See* nonprinting text
Comments parameter 150
complimentary closings 278
CompuServe 4
conditional hyphenation *See* hyphens, soft
Conditional Page Break command **43**, 171
context-sensitive help *See* help
Control Character command 119
control characters 37, **44**, 119
ASCII files 16
deleting 53
error message 358
inserting 44
printing 10
searching 45, 217
showing 208
typestyles 246
control codes *See* control characters
COPY (DOS command) 74
Copy command 34, **47**, 64
copying a block *See* Block Select menu
copying between windows 260
Counter parameter 153
Counter variable 163, 237, 300
crash recovery *See* backup file
Criteria command 144
cross-referencing 236, *See also* tagging; X-Reference menu
captions 31
Ctrl-Q 217
Ctrl-U 212
Current Printer command **48**, 176, 190, 350
customization 49
beeps 241
colors 49
function keys 90
options 167
screen elements 206
Customize menu **49**

# D

Data command 143
Data Format command 143
date-stamping 321
Day variable 253, 319
decimal numbers 278
decimal tabs 183, 234
DecimalTab command 183, 234
DEFAULT.REC 353
DEFAULT.SPR 142, 353
Define a Tag command 236, 293
Define command 61, 92, 146
  error message 357
  parameters 147
Define Text Variable command **50**, 119
DEL (DOS command) 74
DeleteLine command 183
DeleteLineBeg command 183
DeleteLineEnd command 183
DeleteParagraph command 183
DeleteSentence command 183
DeleteToChar command 184
DeleteWord command 184
DeleteWordLeft command 184
deleting **51**, **52**, 183, *See also*
  canceling
  blocks 157
  changes 197
  headers 102
  page breaks 122, 193
  shortcuts 249
  Undelete command 247
delimiters 331
  error message 357
  field 143
  nesting 332
Descending Order command 14
Description command **54**, 112, 134
  in tables 232
Destination command 188, 348
DIC files 223
dictionaries 223
dimensions **56**
  default 132
  error messages 366
  with Style command 310
DIR (DOS command) 74
Direction command 213
directories 74
discretionary hyphens *See* hyphens, soft
disks
  distribution 2
  floppy 20, 63
Display command **281**
DisplayWrite 243
Divider parameter 154
Document-Wide menu **59**, 127
DOS
  commands **62**, 74
  exiting to 192
  printing 344
    alternate printer 349
    defining variables 346
    ending page 351
    error messages 366
    filling paragraphs 346
    font changing 347
    number of copies 346
    number of passes 348
    pausing 350
    previewing 352
    set tabs 352
    single pages 350
    starting page 351
    to a file 348
    to alternative screen 351
    unformatted 348
    with log file 347
  running Sprint **343**
  SprintMerge **352**
  wildcards *See* wildcards, DOS
Draw Box command 73, 95
drawing 95, 128
drop-caps, creating 279
Duplicate-Copy command 74

# E

Edit menu **64**
Edit on Screen command 111, 138, 200
editing 64
  rulers 111, 138, 200

electronic mail 334
else clauses 273, 276
em dashes 278
em space dimension 58, 293
EMACS 249
en space dimension 58
encapsulated PostScript *See* EPS Picture command; PostScript
End command *See* Begin and End commands
ENDF command 78
Ending Page command 189, 350, 351
Ending Record command 144
Endnote command **66**, 193, 300, 301, *See also* footnotes
  placing endnotes 302
ENDS command 80
enlarging documents 304
ENOTE command 67
Enter command 136
Enter key 98
Entire File command 216
EPS Picture command 72, 95
  error messages 362
  inserting Quattro graphics 96
ERASE (DOS command) 74
Erase (file manager command) 74
Erase (glossary entry) command 92
Erase command 35, 54, 65, **68**
Error command 61
error messages **69**, **355**
  logging 11, 69
  missing End command 271
  preventing with IfNotFound 149
Esc key 29
Escape command **282**
  error messages 360
Eval command 61
Even Pages command 81, 101
Every Bad Word command 19, 223
Example command **285**
exiting Sprint *See* Quit command
Export command 243

## F

fatal error messages 355
FCapt command 31, 72
field separators 143
Figure command **71**, 227
  with captions 30
  with graphics 95
Figure variable 256
File Manager menu **73**, 76
File menu **75**
FileCharBack command 184
FileCharFwd command 184
FileCloseAll command 184
FileNext command 184
FilePrevious command 184
files
  ASCII *See* ASCII, files
  closing 35
  closing all 184
  creating 166
  default extension 165, 344
  default settings 93
  EPS 95
  error messages 359
  exporting 242
  extensions
    $$$ 358
    ASC 15
    DIC 223
    LEX 223
    LOG 11, 69, 347
      error message 359
    OVL 136
    PRN 188, 348
    SPG 93
    SPP 48, 349
      error message 359
      error messages 358
    SPR 76, 165, 344
    SPS 344, 352
    TXT 92
    UI 249
  importing 242
  in @ commands 340
  inserting 118
  letter 142
    default 353
  listing 75, 76, 156, 181
  marking a place 181
  master 175, 288
  merging 118, 288

mouse commands 75, 156
naming 76
opening 165
from DOS 343
overwriting 206
picking 75, 156, 181
record 142, 353
renaming 74
reverting to saved 197
saving 20, 177, 205
source 254
template 142
temporary 360
translating 242
working with 73
FileSaveAll command 184, 205
Fill parameter 150, 347
FinalWord II 342
Find command 212
FirstPage variable 253
FlushLeft parameter 150
FlushRight parameter 150
FMT files 60
and Case command 275
errors 364
modifying 298
with Style command 310
FNOTE command 85
Font command **77**, 185, 245
Font parameter 149
Font variable 253
fonts *See also* typestyles
blocks 78
changing 77
changing from DOS 347
default 78
Dingbats 277
document-wide 78
error messages 363
fixed-width 109, 111, 130, 201, 285
formats 79
kerning 292
large 80
parameters 149
PostScript 278
printing special characters 277
proportionally spaced 111, 189
rulers 185, 203
sizes **79**, 186, 203
Small 80
FOOTER command 81
Footer menu **81**, 127
FOOTERE command 81
FOOTERO command 81
footers 81, *See also* headers
all pages 84
alternating 82
first page 84
page numbers 83, 173
positioning 81, 361
FooterSpacing parameter 150
FOOTERT command 81
Footnote command 84, 193
Footnote variable 256, 301
footnotes **84**, *See also* Endnote command
resetting 85
unnumbered 85, 86
form feeds 152
form letters 142, *See also* SprintMerge
with Case command 273
Format command 334
error message 360
formats **87**, *See also* commands
"other" 168, 267
defining 146
document-wide 88
modifying 88, **145**, 320
error messages 359
global 146, 297
unique 147
not in menus 267
regions vs. commands 268
Formatted Print command 10, 348
formatting *See also* formats; printing
errors messages 355
global 310
input during 308
messages during 296
paragraphs 179
rulers 198
FormFeed parameter 150
Free parameter 154
Function Keys command **88**, 248
function keys, redefining 88

# G

G 44
glossaries 91
  AUTOEXEC.SPG 93, 199
  macros in 93
Glossary menu **91**
Go command (Print) 188
Go command (SprintMerge) 143
Go to Page command 65, 172
GotoNextPage command 184
GotoPrevPage command 184
graphics 72, 95, 128
  dimensions 284
Graphics Card Plus 39, 79, 247
Graphics menu **95**, 227
graphs, inserting into Sprint 96
Group parameter 151
Group Together on Page command 97, 171, 180
  error messages 360
grouping 97
Gutter Between command 41
gutter margins *See* binding margins
Gutter parameter 151

# H

hanging indents *See* indenting, hanging indents
hard returns 11, 16, **98**, 325, *See also* soft returns
  showing 207
hard space *See* Non-Breaking Space command; spaces, non-breaking
hardware requirements 3
HaveSpace command 232, **286**
HEADER command 100
Header menu 127
HEADERE command 101
HEADERO command 101
headers **100**, *See also* footers
  all pages 102
  alternating 102
  even pages 101
  odd pages 100
  page numbers 173
  positioning 101, 102, 361
HeaderSpacing parameter 151, 361
HEADERT command 100
HeadingA command **103**, 105
HeadingB command **104**
HeadingC command **105**
HeadingD command **106**
headings
  columns 105
  HeadingA 103
  HeadingB 104
  HeadingC 105
  HeadingD 106
  modifying 105
  multilevel 218, 228
  numbered 107
  Paragraph 177
  Section 217
  Subsection 228
  unnumbered 103
    in table of contents 296
Headings menu **107**, 227
help **108**
  function key template 91
Hercules 39, 79, 247
hexadecimal numbers 278
Hidden command 42, 245, *See also* nonprinting text
highlighting text *See* blocks, selecting
hot keys *See* shortcuts
Hour variable 254
Hsp command **287**
  vs. Tab command 311
HUnits command 61
hyphenation 108
  preventing 325
Hyphenation menu **108**
HyphenNL 100
Hyphens command 26, 134, 293
hyphens, soft 109, 119, 221
  vs. conditional line break 326

# I

I (ruler mark) 201
If command 61
IfDef command 61
IfNotFound parameter 149
IfOdd command 61
Ignore command 224
Import command 242, 243
inches dimension 58

Include command 142, 175, **288**
nesting 290, 365
InColor card 39, 79, 247
Incr command **290**
Increment parameter 153
Indent parameter 56, 111, 151
indenting **110**, *See also* margins
first lines 18, 111, 186, 204
hanging indents 232, 322
parameters 139
regions 111
showing 207
Undent command 322
vs. margins 138
Index menu **112**, 227
Index parameter 154
Index Under command 114, 115
indexing
multilevel 116
preventing 326
nesting 362
page ranges 115
sort order 115
with @ commands 341
Initial (First Line) Indent command 111, 186, 204
Initial Record command 144
Initialize parameter 154
Inline parameter 154
Ins in status line 226
Ins key 120
Insert (file) command 76, 118
Insert (Unconditional) Page Break command **121**, 171
Insert menu **118**
Insert Mode command **120**, 168
Insert-Paste command 34, 64, **120**
inserting 121
files 76, 118
variables 119
installation 48, 49, 78, 152
Inter-Paragraph Spread command 60, 131
Invisible parameter 149
Italic command 244
IXMASTER command 113
IXRANGE command 114
IXREF command 113
IXREF UNDER command 114
IXSEE command 113
IXSEEALSO command 114

## J

J (ruler mark) 202
Jump to Line command 65
justification **123**, 151
forced 291
rulers 202
Justification menu 123, 199
Justify command **291**
Justify parameter 124, 151

## K

Keep with Following Text command **125**, 171, 180
KEEPFOLLOWING command 171
Kern command **292**
KEY command 95
Keyboard Record command 92
keyboard shortcuts *See* shortcuts
KeyCaps command 95

## L

L (ruler mark) 202
Label command **293**
Large command 80, 245
LaserWriter printer 29, *See also* PostScript
Last Bad Word command 19, 223
Layout menu **126**
leader dots 194
leading
between lines 131
between paragraphs 60, 131
LeadingSpaces parameter 151
Left Indent command 111, 186, 204
Left Margin command 59, 138
LeftIndent parameter 151
LeftMargin parameter 139, 151
Length command 242
Letter File command 143
letters 269, 278
LEX files 223
License Statement 3
line breaks, conditional 326
Line Drawing command **128**
line numbers in status lines 226

Line Spacing command **131**, 202
linefeed characters 99
LineLength parameter 152
lines (dimension) 58
List command 92
List Directory command 74
lists 133
  bulleted 26
  Description 54
  nesting 28, 134, 165, 167
  numbered 161, 165
  outline-style 172
Lists menu **133**, 229
Load command 136
loading Sprint *See* starting Sprint
Log Errors to File command 11, 69
LOG extension 11, 69
LOG files 11, 69, 349
  error messages 361
Lookup command 226

## M

Macro command 61, 137
  error message 359
macros
  deleting 251
  Match 335
  parameters 156
Macros menu **136**
mail merge *See* SprintMerge
Main Dictionary command 225
MakeOdd command 61
MakeTOC command 103, 105, 106, 221, **298**
Manuscript variable 256
margins 59, 88, **137**, *See also* indenting
  bottom 150
  default settings 138
  vs. indents 138
  vs. rulers 207
Margins parameter 152
marking a block *See* blocks, selecting
Master Keyword command 113, 116
Match macro 335
Match Words Only command 215
memory
  error messages 360, 362, 365, 368
Menu Display Delay command 170
Menu Shortcuts command **141**
Merge (Glossary) command 92
Merge (Print) menu 191
Merge Field command 119, 142
Merge menu **142**
Merge_Init_ command 61
merging files 118, 290
  from DOS 354
Message command 257, **298**, 311
messages during formatting 310
millimeters dimension 58
Minimum Word Length command 109
Minute variable 256, 321
Modify (colors) command 38
Modify (formatter command) 145
Modify (menu command) **145**
Modify command 229, **299**, *See also* formats, modifying
monitors *See* screen
Month variable 256
MonthName variable 258
mouse commands 25, 158
  drawing lines 130
  scrolling 213
Mouse menu 159
Move-Cut command 34, 64, 160
MoveParagraphBack command 187
MoveParagraphFwd command 187
MoveSentenceBack command 187
MoveSentenceFwd command 187
MoveWordBack command 187
MoveWordFwd command 187
moving a block *See* blocks, moving
MS Word 245, 251
Multilevel command 134, **161**
MultiMate 245

## N

NeedSpace command **301**

nesting
@-sign commands 331
lists 26, 28, 134, 162, 164
New command 75
Next Occurrence command 212
NoHinge command 342
Non-Breaking Space command 22, 119, **160**
non-breaking spaces screen element 207
nonprinting text **42**, 150, *See also* Hidden command
Normal command 244
NoTCT command 313
NoTCT parameter 152
NoteChapter command 66, **300**
Notes command 85, 193, *See also* Endnote command; footnotes
NoteSection command 66, **300**
Number of Copies command 189, 346
Number of Passes command 10, 349
Numbered command 134, **162**, 293
Numbered parameter 154, 320

# O

O command **301**
error message 357
octal numbers 278
Odd Pages command 81, 100
Offset margin command 60, 138, 141
Offset parameter 152
Open command 76, **165**
opening files *See* files, opening
opening windows *See* Window menu
Options Menu **167**, 189
Order command 144
Original Word command 240
orphans *See* Widow-Orphan Control command; WidowPrevent parameter
Other Format command **168**, 228
using **267**
outdenting 111, 322
Outline command 134, **169**
Outlook outliner 170, 243
output file (SprintMerge) 353
overprinting 301, 327
error message 360
Overstruck parameter 149
overwrite mode *See* Insert Mode command
OVL files 136
Ovp command **301**
Ovr in status line 226

# P

page breaks 97, 121, 171
conditional 43, 286, 299
deleting 193
preventing 125, 153, 180
previewing 188
unconditional 21
unwanted 122
Page Breaks menu 126, **171**
page footers *See* footers
page headers *See* headers
Page Number command 237
page numbers 84, **172**, 237, 254
first page 253
in footers 83
in status lines 226
incrementing 305
renumbering 174
section numbers with 173
Page Range command 114, 115
Page variable 83, 254, 319
PageFoot command 232
PageInit command 61
pages (dimension) 58
Paginate command 122, **176**, 189
Paper parameter 152
Paper Size command 60
PaperWidth parameter 152
Paragraph command **177**
paragraph marks *See* hard returns
Paragraph Marks command 207
paragraph spacing *See* Inter-Paragraph Spread command; Spread parameter
Paragraph variable 256
paragraphs **179**
ending 11
ParagraphTitle variable 256
parameters
Above 149, 154

AbovePage 149
After 154
AfterEntry 154
AfterExit 154
Before 154
BeforeEach 154
BeforeExit 154
Below 149, 154
BelowPage 150
BlankLines 150
BottomMargin 150
Centered 150
Column 150
Columns 40, 150
Comments 150
Counter 153
dimensions 56
Divider 154
enumeration 153
error messages 356, 358
Fill 150, 347
floating 154
FlushLeft 150
FlushRight 150
Font 149
FooterSpacing 150
formatting 149
FormFeed 150
Free 154
Group 151
Gutter 151
HeaderSpacing 151, 361
IfNotFound 149
Increment 153
Indent 56, 111, 139, 151
Index 154
Initialize 154
Inline 154
Invisible 149
Justify 124, 151
LeadingSpaces 151
LeftIndent 151
LeftMargin 151
LineLength 151
macro 154
Margins 152
NoTCT 152
Numbered 154, 320
Offset 152
Overstruck 149
Paper 152
PaperWidth 152
RightIndent 152
RightMargin 152
Script 149, 364
Size 149
Spacing 132, 146, 152
Spread 28, 56, 152, 179
Strikeout 149
Style command 147
table 149
TabSize 153
TCT 153
template 316
TopMargin 153, 361
typeface 149
Underline 149
WidowPrevent 153
with regions of text 147
with Style command 309
WithEach 154
Within 154
WordSpacing 153
Parent command 61, 173, 239, 317
pasting 121, *See* Insert-Paste command
vs. undeleting 247
PATH (DOS command) 289
Pause Between Pages command 190, 351
PGBLANK command 171
PGBREAK command 171
picas dimension 58
Pick from List command 76, **181**
Pick Template menu 257, 315
Pitch command 242
Place command 67, **302**
Place Mark command 65, **181**
Plain variable 254
pm variable 254
points dimension 58
Position command 81, 101
POSTSCR.HDR 190
POSTSCR.TCT 275, 292, 315
PostScript 285
capturing to a file 190

graphics 72
Potpourri menu **183**
delete commands 53
Precise Settings menu 139, **185**, 199, 201
Preserve Editing Session command 36, 167, **186**
previewing *See* Screen Preview command
Previous Format command 147
Previous Word command 240
Print menu **188**
Printer command 350
error message 363
Printer variable 254
printers
alternate 48
changing 48, 176
drivers 357
escape codes 282
graphics 130
names 254
paper size 152
PostScript 72, 315
units 58
printing
accented letters 301
fonts 77
from DOS 344
alternate printer 349
defining variables 346
ending page 351
filling paragraphs 346
font changing 347
number of copies 346
number of passes 348
options 345
pausing 350
previewing 352
set tabs 352
single pages 350
starting page 351
to a file 348
to alternative screen 351
unformatted 348
with log file 347
large documents 289
nondisplaying characters 314
options 10, 189
previewing 188, 352
special characters 276
SprintMerge 353
to a file 190
unformatted 254, 348
PRN files 188, 348

# Q

QCARD.TXT 191
Quattro 96
QuickCard command 91, **191**
Quit command **191**
quitting Sprint 191
Quotation command **303**
quotation marks in file names 340
quotation marks in text variables 51

# R

R (ruler mark) 203
ragged text *See* justification
RAM-resident programs 63
ReadEPS command 61
Recall command 92
Record File command 143
records 143
reducing documents 304
Reference a Tag command 236
Reference By menu 237
Reference Word command 113, 326
References menu **193**, 227
region *See* block
RegionIndent command 184
RegionOutdent command 184
Remove Formatter Page Breaks command 122, 176, 189, **193**
REN (DOS command) 74
Rename-Move command 74
RepeatCount command 184, 194
Repeating Character command 73, 119, **194**
repeating characters template 318
Replace With command 224
ReplaceAll command 184
replacing text *See* searching
Reselect Block command 24, 118
RESERVE command 171, 196

Reserve Space command 30, 72, 171, **195**
Reset command 61
Reset Shortcuts command 249
Resize command 260
Rest of File command 223
Revert to Saved command 29, 76, **197**
Right Indent command 111, 186
Right Margin command 59, 138
RightIndent parameter 152
RightMargin parameter 139, 152
RightTab command 184
RJ (ruler mark) 203
roup Together on Page command 44
ruler lines *See* rulers
Ruler menu 126, 198
rulers
  centering 32
  default 199
    setting 93, 199
  deleting 53
  editing 138, 198
  first 138, 140, 204
  fonts 203
  hanging indents 232
  indent 201
  inserting 198
  justification 202
  margins 201
  precise settings 185
  spacing 131
  symbols in 201
  tabs 202
  vs. margins 204
Run command 136
running Sprint from DOS 343

## S

Save (UI) command 249
Save command 76, **205**
saving
  blocks 265
  files *See* files, saving
  shortcuts 249
  under a different name 264
Scale command **304**
screen
  blinking 39
  colors *See* Colors command
  customization 38, 49
  drivers 344
  elements (table) 208
  refreshing 16
Screen menu **206**
Screen Preview command (Printing) 188, 352
Screen Preview command (SprintMerge) 143
Script parameter 149, 364
ScrollDown command 184
scrolling 184, **210**
  mouse commands 157
ScrollLeft command 184
ScrollRight command 184
ScrollUp command 185
Search & Replace command 212
Search-Replace menu 65, 211
searching **211**
  aborting 212
  direction 213
  ReplaceAll command 184
  search/replace 211
Section command **217**
section titles in endnotes 300
Section variable 13, 256
SectionNumber variable 256, 290, 293
SectionTitle variable 256
See command 113
Sel in status line 208, 226
select mode 23
selecting text *See* blocks, selecting
semicolon command 42
Set command **304**
  with Char command 277
  with footnotes 86
  with page numbers 175
setting margins *See* margins
setting tabs *See* tabs
setup *See* installation
shortcuts
  defined 8
  deleting 249
  function key 88
  menu 141
  Menu Shortcuts command 49

Potpourri 185
printing 191
WordStar 25, 263
Shut All command 260
SideKick 63, 249
SideKick Plus 39, 170, 243
SIZE command 80, 186
Size parameter 149
Size variable 254
Skip Once command 224
Small font command 80
Snaking Columns command 39
SNOTE command 86
soft hyphens *See* hyphens, soft
soft returns 17, 98, 325, *See also* hard returns
sorting *See also* Arrange-Sort command
in SprintMerge 353
SourceFile variable 254
SourceLine variable 254
SP-SETUP 3, *See also* installation
SP.EXE 2
SP.SPM 3
SP.SWP 3
Space Allowable command 109
SpaceNL 99
spaces
hard *See* spaces, non-breaking
in tables 230
non-breaking 22, 119, 160
showing 207
search wildcard 214
showing 207
spacing
between lines 131
between paragraphs 60, 131, 179
between words 60, 61, 109, 153
Spacing parameter 132, 146, 152
SPADV.UI 8, 250
SPBASIC.UI 8, 250
Special Hyphen command 119, **221**
spelling check 222
automatic 19
Spelling menu **222**
SPFMT command 344
SPG files 93
SPL extension 3
SPM extension 3
SPM.O$$ 353
SPM.S$$ 353
SPMERGE command 353
SPP extension 48
SPP files 48, 349
error message 358
error messages 358, 359
SPR files 76, 165, 344
spread between paragraphs *See* Inter-Paragraph Spread command; Spread parameter
Spread parameter 28, 56, 152, 179
SPRECOVE.COM 3, 63
springs *See* Wide Space (Spring) command
SprintMerge 119, **142**
canceling 29
from DOS **352**
options 144
print options 353
SPS files 344, 352
SPTUTOR.UI 8, 250
STANDARD.FMT 3, 60, 146
comments in 43
meaning of 9
STANDARD.SPG 93
Starting Page command 189, 350, 351
starting Sprint 343
status line **225**
components 225
showing 208
typestyles 247
Strikeout parameter 149
Strikethrough command 245
string variables *See* Define Text Variable command
StringInput command 276, **308**
Style command 61, 88, 132, **309**
and first ruler 138, 140
parameters 147
Style menu **226**
Style Sheet command 60, 334
error message 360
subheads *See* Headings menu
Subscript command 245
subscript parameter 149
Subsection command **228**

Subsection variable 256
SubsectionTitle variable 256
SuperKey 63, 278
Superscript command 245
superscript parameter 149
swap file *See* backup file
synonyms 240

# T

T (ruler mark) 202
Tab command **311**
Tab Expansion command 15, **229**, 233, 352
Tab key to prevent formatting 163
Tab Stops command 186, 204, 233, **235**
TabDivide command 233, **311**
Table command 227, **230**
table of contents 12, 13, 296
  figures 72
table of tables 239
Table variable 256
tables
  captions 238
  creating with wide spaces 257
tabs **233**
  ASCII files 17, 229
  decimal 183, 234
  dividing 311
  in tables 231
  showing 207
  too many 364, 365
TABSET command 186
TabSize parameter 153
tagging **235**, *See also* cross-referencing
  error messages 364, 365
  figures 72
  numbered lists 163
  numbers 153
  page numbers 174, 237
  page ranges 115
  printing tags 10
  section numbers 236
  SectionNumber variable 293
  tables 232
  variables 236, 318
  warning messages 70
TagString command **312**
TCapt command 238
TCaption command 238
TCT command **313**, 327
  error messages 361
TCT parameter 153
Template command **315**, 316, 319
  error message 357
Template for Data command 119, 142
templates
  conditional 318
  for tags 318
  numbering 316
  parameters 316
  parents 317, 319
  printing % 319
  SprintMerge 353
  too many 365
TemporaryIndent command 185
text
  alignment *See* justification
  centering 123
  nonprinting 42
  selecting *See* blocks, selecting
  substituting 314
Text command 41, 309, **320**
Thesaurus command 240
This Format command 147
time in status line 226
time variables 254
TimeDate command 185
Timestamp command **321**
TimeTime command 185
TimeWeekDay command 185
Title command **322**
Title Page command 32, 81, 100, 127, **241**
TOC.$$$ 358, 360
Tone command 168, 241
Top Margin command 59, 138
TopMargin parameter 153, 361
TopOfFile command 185
TopOfScreen command 185
Translate menu 76, **242**
translation character table 313
TransposeChars command 185
TransposeLines command 185

TransposeWords command 185
Turn Select Mode command 24
TXT extension 92
TXT files 92
typefaces *See* fonts
Typestyle menu **244**
typestyles **244**, *See also* fonts
  changing 246
  condensing 292
  control characters 246
  in this manual 2

## U

UI files 249
Undelete command 34, 54, 64, **247**
Undent command **322**
Underline command 245
Underline parameter 149
units (dimension) 58
unknown command (error message) 70
unnumbered headings 107
Unzoom command 260
Use Wildcards command 213
User Dictionary command 223
User Interface menu **248**
user interfaces *See* alternative user interfaces
Utilities menu **252**

## V

Value command 364
Variable menu 319
variables **252**
  Appendix 256
  AppendixTitle 256
  assigning numbers to 304
  built-in 253
  Chapter 83, 256
  ChapterTitle 256
  Counter 163, 237, 300
  creating 307
  Day 253, 319
  defined in STANDARD.FMT 255
  editing 255
  Figure 256
  FirstPage 253
  font 253
  Footnote 256, 301
  Hour 254
  in footers 83
  in onscreen messages 297
  incrementing 290, 305
  inserting 119
  Manuscript 254
  Minute 254, 319
  Month 254
  MonthName 256
  Page 83, 172, 254, 319
  Paragraph 256
  ParagraphTitle 256
  parents 317, 319
  Plain 254
  pm 254
  Printer 254
  printing 315
  quotation marks in 51
  Section 256
  SectionNumber 256, 290, 293
  SectionTitle 256
  Size 254
  SourceFile 254
  SourceLine 254
  Subsection 256
  SubsectionTitle 256
  Table 256
  tagging 236
  templates 257, 315
  text 50, 119, 312
  WeekDay 254, 256
  Words 255
Verbatim command **324**
VUnits command 61

## W

Wang (IWP) 243
Warn command 61
warning messages **69**, 355
weekday variable 185, 254, 256
Wide Space (Spring) command 22, 32, 119, **257**, 292
  in tables 232
wide spaces screen element 207
Widow-Orphan Control command 172
WidowPrevent parameter 153
wildcards

DOS 75, 77, 166
in searches 213
Window menu 259
WindowCloseAll command 185
windows
opening from DOS 344
switching with a mouse 155
WithEach parameter 154
Within parameter 154
Word command 117, 222, **325**
Word Spacing command 60, 61
Word Underline command 245
WordPerfect 249
import/export 243
Words variable 255
WordSpacing parameter 153
WordStar 243
user interface 249, **263**
Wordwrap ASCII Files command 11, 15, 100
wordwrapping 37, 98, 100, 150, **264**
while printing 346
Wrap Long Lines command 15, 17
Write As command 76, 205, **264**
Write Block command 65, **265**

## X

X-Reference menu 227, 236, *See also* cross-referencing

## Z

Zoom command 260

BORLAND INTERNATIONAL, INC., 4585 SCOTTS VALLEY DR., P.O. BOX 660001, SCOTTS VALLEY, CA 95066-0001 PART# 22 MN-SPT02 BOR 0822