LISP USERS' TEMPLATE

Updated by: Melissa Biggs (Biggs.PA @ Xerox.COM)

This document provides a template and instructions for formatting the Lisp Users' module documentation. This template applies primarily to standalone workstation users. Using the Lisp Library module TEdit, and this document, you should be able to create a standard Lisp Users' module for the Lisp Users' manual. This document gives you the written specifications for formatting your document. The specifications are given in the order in which you would most likely use them to format a document, with the basic text and margins described first, then the various levels of headings, then special elements such as page numbers.

RULES FOR CONTENT

Documentation should always include the name of the module, the name of the author (and Xerox, Arpanet, CSNET or other electronic mailing address, when available—otherwise US mail address), the names of all other Lisp Users' modules required, the names of all files which are part of the module (data files, other Lisp files, etc.), and enough detail to allow someone to effectively use it**.** A sample Lisp Users' template appears at the end of this document.

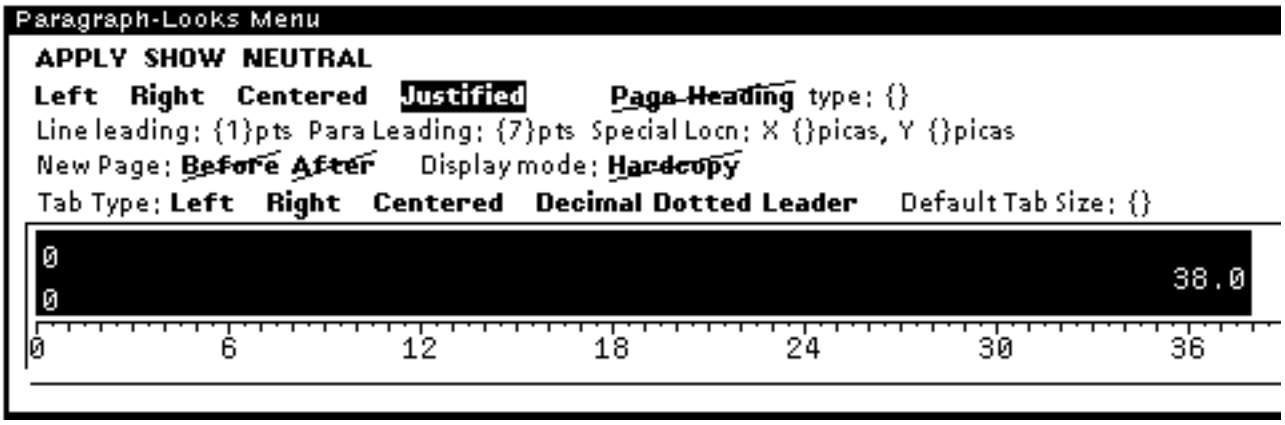
BASIC SPECIFICATIONS

It is wise to apply the specifications for the body text, headings, functions, variables, and page numbers as you write the document.

Font, Type Size, Leading, and Margins

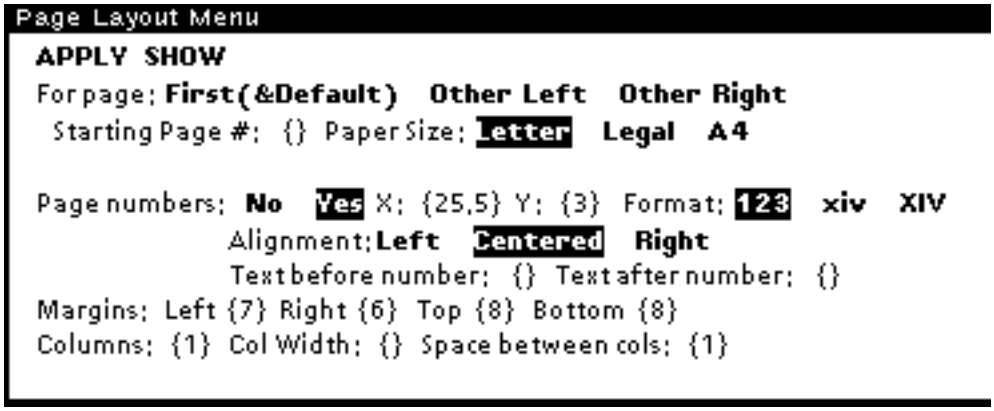
For the text, choose a 10-point Modern font and Apply it to the appropriate text using the Character Looks menu.

Then, in the Paragraph Looks menu, set the leading, the spacing between lines of type; the justification; and the left and right margin settings. Set line leading to 1 point and paragraph leading to 7 points. Apply all paragraph looks to the appropriate text.



(Because 7 points paragraph leading is all you need, you should use only one carriage return between paragraphs when typing in text.)

Finally, in the Page Layout menu, set the left margin to 7 picas, the right margin to 6, and the top and bottom margins to 8. Apply these to all types of pages (first, other left , and other right).

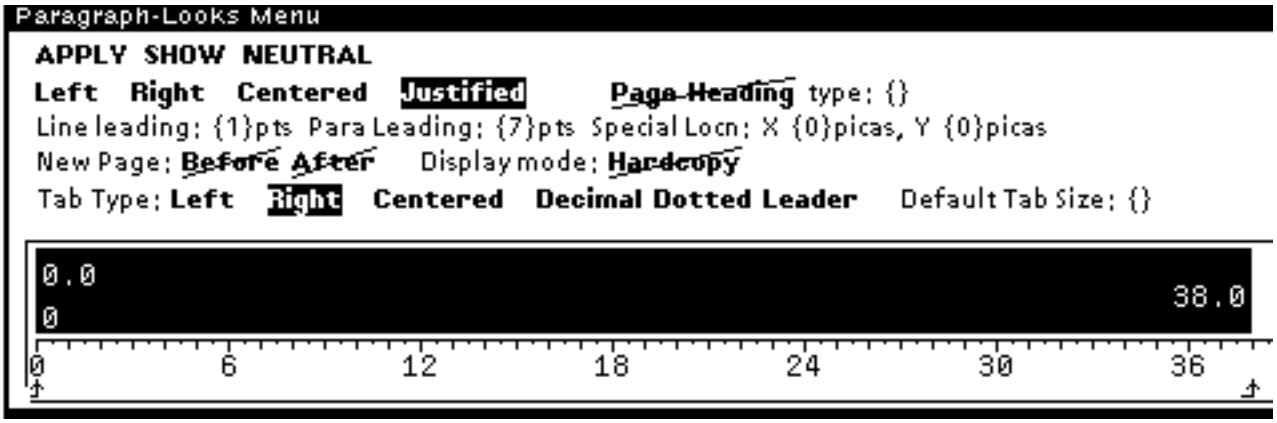


Functions, Variables, and Lisp Code Examples

It is usual to first give the name of a function, then describe its purpose and each of its arguments. When the name of a function is first given, it is set off like this:

(IMAGEFNSCREATE *DISPLAYFN IMAGEBOXFN PUTFN GETFN COPYFN*) [Function]

The Paragraph Looks menu for a function is set up like this:

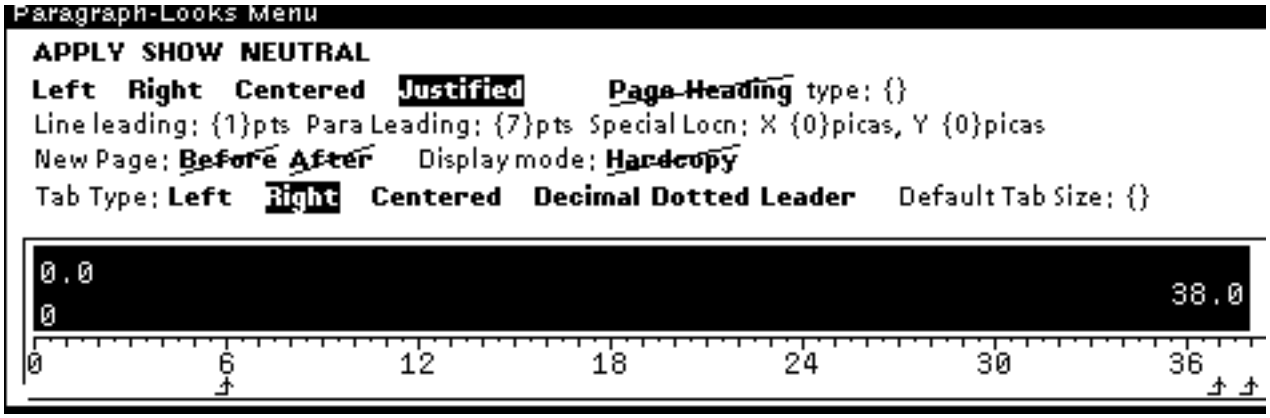


The function name is in 10-point regular Modern, all caps. Arguments are in 10-point italic Modern, in all caps, mixed case, or lowercase, as they appear in the system.

If the function description is more than one line long, the runover arguments should be indented under the function name and the word [Function] placed on the last line of the argument list, like this:

(MAKE-ARRAY *INDICESLIST &KEY :ELEMENT-TYPE: ˆ*
INITIAL-ELEMENT :INITIAL-CONTENTS :ADJUSTABLE: ˆ
FILL-POINTER :DISPLACED-TO :DISPLACED-INDEX-OFFSET) [Function]

The Paragraph Looks menu should be set as follows to produce this type of argument format:



Long function descriptions should have two points leading between lines. To space them correctly, hold down the meta key when typing the carriage returns so that TEdit breaks the lines without creating separate paragraphs.

Parentheses are in 10-point regular Modern type. The type of definition is in 10-point Modern, caps and lowercase, enclosed in square brackets, and flushed right with a right tab set to 38.0. Note that tabs are used to indent the second and third lines of arguments.

Variables look like functions, except that the word "Variable," enclosed in square brackets, follows the variable name.

Examples of code should be in 10-point Terminal font (but not function names, commands, file names, and the like). If 10-point Terminal is not available on your printer, use 8-point Terminal. Code examples should have two points line leading and no paragraph leading.

Quotation Marks, Bullets, and Dashes

We recommend using TEdit's "expanded abbreviations" to produce professional-looking quotation marks, bullets, em-dashes—used to separate text phrases—and en-dashes (used to indicate inclusive numbers, as in "pages 3–6").

· To produce a bullet, type a lowercase b, select it, and type Control-X.

· To produce an em-dash, type a lowercase m, select it, and type Control-X.

· To produce an en-dash, type a lowercase n, select it, and type Control-X.

HEADS

There are four levels of heads in the Lisp Users' documentation: chapter (level 1) heads, level 2, level 3, and level 4 heads.

Note: A head that falls at the bottom of a page (a "widow") is undesirable. You eliminate a widow by selecting it, then applying the Before option of the New Page command in the Paragraph Looks menu.

The Chapter Head

The chapter head appears at the beginning of the document and identifies it. The heading "Lisp Users' Template," above, is a correctly formatted Lisp Users' chapter head.

When submitting a Lisp Users' module on floppy disk format the chapter head as follows:

Module Name: your Lisp Users' Module (all caps, 12-point bold Modern)

Your name and ARPANET address (if you have one) in 10-point Modern

The Level 2 Head

Level 2 heads identify major sections of a document. The level 2 heads for the Lisp Users' documentation are in 10-point bold Modern, all caps.

The Level 3 Head

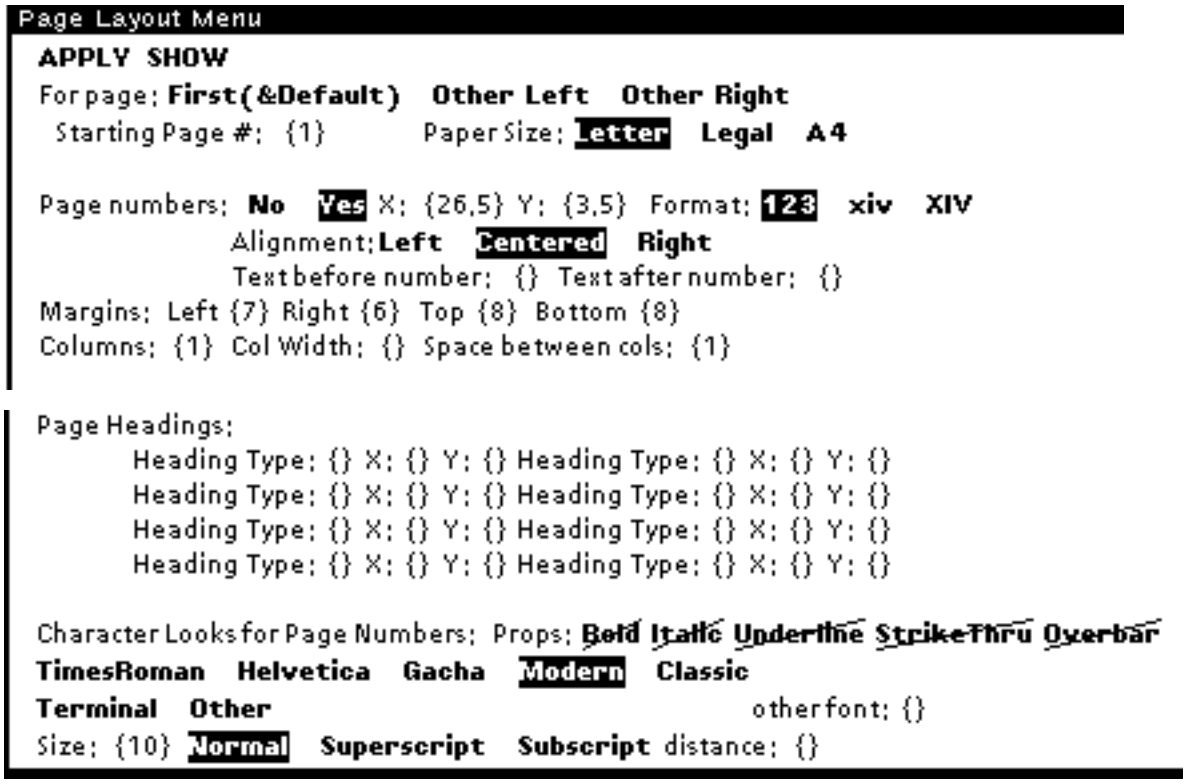
Level 3 heads identify subsections of a document. For the Lisp Users' manual, they are in 10-point bold Modern, caps and lowercase.

The Level 4 Head

Level 4 heads identify the lowest level of subsection in the Lisp Users' documentation. They are in 10-point regular Modern, caps and lowercase, underlined.

PAGE NUMBERS

Page numbers are specified and applied in the Page Layout menu. First, specify the alignment of the page numbers to be centered, with the *X* position being 26.5 picas and the *Y* position 3.5. Then specify the character looks to be 10-point regular Modern. Finally, apply the page numbers to the First(&Default) pages.



After you submit your document, XEROX AIS will add running heads, put in additional formatting, and provide final page numbering to assemble it into the Lisp Users' manual.

MANUAL TEMPLATE

The following page has a sample template for the information required in a Lisp Users' module. Use this template to produce your module documentation.

Module Name:

>>Module Name<<

By: >>Your Name<< (>>Your net address<<)

Uses: >>Other modules necessary to run this one<<

>>Type INTERNAL here if the file is for Internal Use Only<<

This document last edited on >>DATE<<.

INTRODUCTION

>>This paragraph should be replaced by an overview of your module. The information on the previous pages explains the documentation conventions to be used for each Lisp Users' module.<<

MODULE EXPLANATIONS

>>Functions, Variables, and Lisp Code Examples<<

It is usual to first give the name of a function, then describe its purpose and each of its arguments.

Module explanations may have several level headings.

Be sure to include the following information in any module explanations:

• any file dependencies

• definitions of all arguments

• module, function variable, etc. limitations

• a liberal number of examples for all functions, variables, etc.