

CONTENTS

NAME.....	1
SYNOPSIS	1
DESCRIPTION	1
OPTIONS	1
USAGE	2
MACROS TO SET FONTS	3
MACROS TO DESCRIBE HYPERLINKS AND EMAIL ADDRESSES.....	4
MACROS TO DESCRIBE COMMAND SYNOSES.....	5
MISCELLANEOUS	5
EXAMPLES	6
PORTABILITY AND TROFF REQUESTS	7
SEE ALSO.....	7
COPYING.....	7
AUTHORS.....	7

NAME

man – man macros to support generation of manual pages

SYNOPSIS

```
nroff -man [options ...] [files ...]
troff -man [options ...] [files ...]
nroff -mandoc [options ...] [files ...]
troff -mandoc [options ...] [files ...]
```

DESCRIPTION

The **man** macros are used to generate *man pages* with *nroff* or *troff*. This document provides a brief summary of the use of each macro in that package.

OPTIONS

The **man** macros understand the following command line options (which define various registers).

-dToc=filename

Generate table of contents, hyperlinks and PDF bookmarks. Table of contents entries generated from **.SH** and **.SS** macros are output on **stderr** with “**Toc:**” prepended. See **EXAMPLES** section for example commands using this option.

-rcR=1 This option (the default if in *nroff* mode) creates a single, very long page instead of multiple pages. Say **-rcR=0** to disable it.

-rIN=width

Set body text indentation to *width*. The default is 7n for *nroff*, 7.2n for *troff*. For *nroff*, this value should always be an integer multiple of unit ‘n’ to get consistent indentation.

-rLL=line-length

Set line length. If this option is not given, the line length is set to respect any value set by a prior ‘.ll’ request (which *must* be in effect when the ‘.TH’ macro is invoked), if this differs from the built-in default for the formatter; otherwise it defaults to 78n in *nroff* mode and 6.5i in *troff* mode.

Note that the use of a ‘.ll’ request to initialize the line length is supported for backward compatibility with some versions of the **man** program; direct initialization of the ‘LL’ register should *always* be preferred to the use of such a request. In particular, note that a ‘.ll 65n’ request does *not* preserve the normal *nroff* default line length, (the **man** default initialization to 78n prevails), whereas, the ‘-rLL=65n’ option, or an equivalent ‘.nr LL 65n’ request preceding the use of the ‘.TH’ macro, *does* set a line length of 65n.

-rSxx Base document font size is *xx* points (*xx* can be 10, 11, or 12) rather than 10 points.

USAGE

This section describes the available macros for manual pages.

.EX

.EE Example/End Example. After **.EX**, filling is disabled and the font is set to constant-width. This is useful for formatting code, command, and configuration-file examples. The **EE** macro restores filling and restores the previous font.

.HP [nnn]

Set up a paragraph with hanging left indentation. The indentation is set to *nnn* if that argument is supplied (the default unit is ‘n’ if omitted), otherwise it is set to the previous indentation value specified with **.TP**, **.IP**, or **.HP** (or to the default value if none of them have been used yet). Font size and face are reset to its default values. The following paragraph illustrates the effect of this macro with hanging indentation set to 4 (enclosed by **.RS** and **.RE** to set the left margin temporarily to the current indentation):

This is a paragraph following an invocation of the **HP** macro. As you can see, it produces a paragraph where all lines but the first are indented.

Use of this presentation-level macro is deprecated. While it is universally portable to legacy Unix systems, a hanging indentation cannot be expressed naturally under HTML, and many HTML-based manual viewers simply interpret it as a starter for a normal paragraph. Thus, any information or distinction you tried to express with the indentation may be lost.

.IP [designator] [nnn]

Set up an indented paragraph, using *designator* as a tag to mark its beginning. The indentation is set to *nnn* if that argument is supplied (the default unit is ‘n’ if omitted), otherwise it is set to the previous indentation value specified with **.TP**, **.IP**, or **.HP** (or to the default value if none of them have been used yet). Font size and face of the paragraph (but not the designator) are reset to its default values.

To start an indented paragraph with a particular indentation but without a designator, use “” (two doublequotes) as the second argument.

For example, the following paragraphs were all set up with bullets as the designator, using ‘.IP \(bu 4’. The whole block has been enclosed with **.RS** and **.RE** to set the left margin temporarily to the current indentation value.

- **IP** is one of the three macros used in the **man** package to format lists.
- **HP** is another. This macro produces a paragraph with a left hanging indentation.
- **TP** is another. This macro produces an unindented label followed by an indented paragraph.

.LP**.PP**

.P These macros are mutual aliases. Any of them causes a line break at the current position, followed by a vertical space downwards by the amount specified by the **PD** macro. The font size and shape are reset to the default value (normally 10pt Roman). Finally, the current left margin and the indentation is reset to the default values.

.RE [nnn]

This macro moves the left margin back to level *nnn*, restoring the previous left margin. If no argument is given, it moves one level back. The first level (i.e., no call to **.RS** yet) has number 1, and each call to **.RS** increases the level by 1.

.RS [nnn]

This macro moves the left margin to the right by the value *nnn* if specified (default unit is ‘n’); otherwise it is set to the previous indentation value specified with **.TP**, **.IP**, or **.HP** (or to the default value if none of them have been used yet). The indentation value is then set to the de-

fault.

Calls to the **RS** macro can be nested.

.SH [*text for a heading*]

Set up an unnumbered section heading sticking out to the left. Prints out all the text following **.SH** up to the end of the line (or the text in the next input line if there is no argument to **.SH**) in bold face (or the font specified by the string **HF**), one size larger than the base document size. Additionally, the left margin and the indentation for the following text is reset to the default values.

.SS [*text for a heading*]

Set up a secondary, unnumbered section heading. Prints out all the text following **.SS** up to the end of the line (or the text in the next input line if there is no argument to **.SS**) in bold face (or the font specified by the string **HF**), at the same size as the base document size. Additionally, the left margin and the indentation for the following text is reset to the default values.

.TH *title section [extra1] [extra2] [extra3]*

Set the title of the *man page* to *title* and the section to *section*, which must take on a value between 1 and 8. The value *section* may also have a string appended, e.g. ‘.pm’, to indicate a specific subsection of the *man pages*. Both *title* and *section* are positioned at the left and right in the header line (with *section* in parentheses immediately appended to *title*). *extra1* is positioned in the middle of the footer line. *extra2* is positioned at the left in the footer line (or at the left on even pages and at the right on odd pages if double-sided printing is active). *extra3* is centered in the header line.

For HTML output, headers and footers are completely suppressed.

Additionally, this macro starts a new page; the new line number is 1 again (except if the ‘-rC1’ option is given on the command line) -- this feature is intended only for formatting multiple *man pages*; a single *man page* should contain exactly one **TH** macro at the beginning of the file.

.TP [*nnn*]

Set up an indented paragraph with label. The indentation is set to *nnn* if that argument is supplied (the default unit is ‘n’ if omitted), otherwise it is set to the previous indentation value specified with **.TP**, **.IP**, or **.HP** (or to the default value if none of them have been used yet).

The first input line of text following this macro is interpreted as a string to be printed flush-left, as it is appropriate for a label. It is not interpreted as part of a paragraph, so there is no attempt to fill the first line with text from the following input lines. Nevertheless, if the label is not as wide as the indentation the paragraph starts at the same line (but indented), continuing on the following lines. If the label is wider than the indentation the descriptive part of the paragraph begins on the line following the label, entirely indented. Note that neither font shape nor font size of the label is set to a default value; on the other hand, the rest of the text has default font settings.

The **TP** macro is the macro used for the explanations you are just reading.

.TQ The **TQ** macro sets up header continuation for a **TP** macro. With it, you can stack up any number of labels (such as in a glossary, or list of commands) before beginning the indented paragraph. For an example, look up the documentation of the **LP**, **PP**, and **P** macros.

To summarize, the following macros cause a line break with the insertion of vertical space (which amount can be changed with the **PD** macro): **SH**, **SS**, **TP**, **TQ**, **LP** (**PP**, **P**), **IP**, and **HP**. The macros **RS**, **RE**, **EX**, and **EE** also cause a break but no insertion of vertical space.

MACROS TO SET FONTS

The standard font is Roman; the default text size is 10 point.

.B [*text*]

Causes *text* to appear in bold face. If no text is present on the line where the macro is called the text of the next input line appears in bold face.

.BI *text* Causes text on the same line to appear alternately in bold face and italic. The text must be on the same line as the macro call. Thus

.BI this "word and" that

would cause ‘this’ and ‘that’ to appear in bold face, while ‘word and’ appears in italics.

.BR *text*

Causes text on the same line to appear alternately in bold face and roman. The text must be on the same line as the macro call.

.I [*text*] Causes *text* to appear in italic. If no text is present on the line where the macro is called the text of the next input line appears in italic.

.IB *text* Causes text to appear alternately in italic and bold face. The text must be on the same line as the macro call.

.IR *text* Causes text on the same line to appear alternately in italic and roman. The text must be on the same line as the macro call.

.RB *text*

Causes text on the same line to appear alternately in roman and bold face. The text must be on the same line as the macro call.

.RI *text* Causes text on the same line to appear alternately in roman and italic. The text must be on the same line as the macro call.

.SB [*text*]

Causes the text on the same line or the text on the next input line to appear in boldface font, one point size smaller than the default font.

.SM [*text*]

Causes the text on the same line or the text on the next input line to appear in a font that is one point size smaller than the default font.

MACROS TO DESCRIBE HYPERLINKS AND EMAIL ADDRESSES

Using these macros helps ensure that you get hyperlinks when your manual page is rendered in a browser or other program that is Web-enabled.

.MT *address*

.ME [*punctuation*]

Wrap an email address. The argument of **.MT** is the address; text following, until **.ME**, is a name to be associated with the address. Any argument to the **ME** macro is pasted to the end of the link text. On a device that is not a browser,

```
contact
.MT fred.foonly@\:fubar.net
Fred Foonly
.ME
for more information
```

usually displays like this: “contact Fred Foonly <fred.foonly@fubar.net> for more information”.

.UR *URL*

.UE [*punctuation*]

Wrap a World Wide Web hyperlink. The argument to **.UR** is the URL; thereafter, lines until **.UE** are collected and used as the link text. Any argument to the **UE** macro is pasted to the end of the text. On a device that is not a browser,

```
this is a link to
.UR http://\:randomsite.org/\:fubar
some random site
.UE ,
given as an example
```

usually displays like this: “this is a link to some random site <http://randomsite.org/fubar>, given as an example”.

MACROS TO DESCRIBE COMMAND SYNOSES

These macros are a convenience for authors. They also assist automated translation tools and help browsers in recognizing command synopses and treating them differently from running text.

.OP *key value*

Describe an optional command argument. The arguments of this macro are set surrounded by option braces in the default Roman font; the first argument is printed with a bold face, while the second argument is typeset as italic.

.SY *command*

Begin synopsis. Takes a single argument, the name of a command. Text following, until closed by .YS, is set with a hanging indentation with the width of *command* plus a space. This produces the traditional look of a Unix command synopsis.

.YS

This macro restores normal indentation at the end of a command synopsis.

Here is a real example:

```
.SY troff
.OP \-afizV
.OP \-d as
.OP \-d a\fB=\fPs
.OP \-F fontdir
.OP \-m name
.OP \-o list
.OP \-r aN
.OP \-r a\fB=\fPN
.OP \-s N
.OP \-T device
.OP \-u N
.OP \-x N
.RI [ file
.IR .\|.\|.
.YS
```

produces the following output:

```
troff [-afizV] [-d as] [-d a=s] [-F fontdir] [-m name] [-o list] [-r aN] [-r a=N] [-s N]
[-T device] [-u N] [-x N] [file ...]
```

If necessary, you might use **br** requests to control line breaking. You can insert plain text as well; this looks like the traditional (unornamented) syntax for a required command argument or filename.

MISCELLANEOUS

The default indentation is 7.2n in troff mode and 7n in nroff mode.

.AT [*system [release]*]

Alter the footer for use with AT&T *man pages*. This command exists only for compatibility; don't use it.

.BT

Print the footer string. Redefine this macro to get control of the footer.

.DT Set tabs every 0.5 inches. Since this macro is always called during a **TH** macro, it makes sense to call it only if the tab positions have been changed.

Use of this presentation-level macro is deprecated. It translates poorly to HTML, under which exact whitespace control and tabbing are not readily available. Thus, information or distinctions that you use **.DT** to express are likely to be lost. If you feel tempted to use it, you should probably be composing a table using **tbl(1)** markup instead.

.PD [*nnn*]

Adjust the empty space before a new paragraph or section. The optional argument gives the amount of space (default unit is ‘v’); without parameter, the value is reset to its default value (1 line in nroff mode, 0.4v otherwise). This affects the macros **SH**, **SS**, **TP**, **LP** (resp. **PP** and **P**), **IP**, and **HP**.

Use of this presentation-level macro is deprecated. It translates poorly to HTML, under which exact control of inter-paragraph spacing is not readily available. Thus, information or distinctions that you use **.PD** to express are likely to be lost.

.PT Print the header string. Redefine this macro to get control of the header.

.UC [*version*]

Alter the footer for use with BSD *man pages*. This command exists only for compatibility; don’t use it.

The following strings are defined:

***R** The ‘registered’ sign (®).

***S** Switch back to the default font size.

***(lq**

***(rq** Left (“) and right (”) quote. This is equal to ‘\lq’ and ‘\rq’, respectively.

***(HF** The typeface used to print headings and subheadings. The default is ‘B’.

***(Tm** The ‘trademark’ sign (™).

If a preprocessor like **tbl** or **eqn** is needed, it has become common to make the first line of the *man page* look like this:

‘” *word*

Note the single space character after the double quote. *word* consists of letters for the needed preprocessors: ‘e’ for **eqn**, ‘r’ for **refer**, and ‘t’ for **tbl**. Modern implementations of the **man** program read this first line and automatically call the right preprocessor(s).

EXAMPLES

Formatting a manual page for text terminal output:

```
tbl -Tlocale filename | neqn | nroff -Tlocale -mandoc \
    | col -x | less
```

Formatting for PDF output with table of contents:

```
run_troff () {
    tbl -g $1 | eqn | troff -mandoc -dToc=/tmp/toc \
        2> /tmp/err | dpost | ps2pdf - $2
}
proc_toc () {
    sed -n 's/^Toc://p' /tmp/err > /tmp/toc
}
run_troff input.n output.pdf
proc_toc
run_troff input.n output.pdf
```

```
proc_toc
run_troff input.n output.pdf
```

Formatting for HTML output with table of contents:

```
run_troff () {
    tbl $1 | eqn | troff -Thtml -mandoc -dToc=/tmp/toc \
        2> /tmp/err | dhtml -t "$2" > $3
}
run_troff input.n 'input(n)' output.html
sed -n 's/^Toc://p' /tmp/err > /tmp/toc
run_troff input.n 'input(n)' output.html
```

PORATABILITY AND TROFF REQUESTS

Note that using raw troff requests is likely to make your page render poorly on the (increasingly common) class of viewers that render it to HTML. Troff requests make implicit assumptions about things like character and page sizes that may break in an HTML environment; also, many of these viewers don't interpret the full troff vocabulary, a problem that can lead to portions of your text being silently dropped.

For portability to modern viewers, it is best to write your page entirely in the requests described on this page. Further, it is best to completely avoid those we have described as ‘presentation-level’ (**.HP**, **.PD**, and **.DT**).

The macros we have described as extensions (**.EX/.EE**, **.SY/.OP/.YS**, **.UR/.UE**, and **.MT/.ME**) should be used with caution, as they may not yet be built in to some viewer that is important to your audience. If in doubt, copy the implementation onto your page.

SEE ALSO

tbl(1), **eqn(1)**, **refer(1)**, **man(1)**, **man(7)**, **mdoc(7)**

COPYING

Copyright © 1999-2014 Free Software Foundation, Inc.

Permission is granted to make and distribute verbatim copies of this manual provided the copyright notice and this permission notice are preserved on all copies.

Permission is granted to copy and distribute modified versions of this manual under the conditions for verbatim copying, provided that the entire resulting derived work is distributed under the terms of a permission notice identical to this one.

Permission is granted to copy and distribute translations of this manual into another language, under the above conditions for modified versions, except that this permission notice may be included in translations approved by the Free Software Foundation instead of in the original English.

AUTHORS

This manual page was originally written for the Debian GNU/Linux system by Susan G. Kleinmann <sgk@debian.org>.

It was corrected and updated by Werner Lemberg <wl@gnu.org>.

The extension macros were documented (and partly designed) by Eric S. Raymond <esr@thryrus.com>; he also wrote the portability advice.

The manual page had been changed to document the Heirloom doctools **-man** macros by others. The original file is found in path

tmac/groff_man.7.man

of the **groff** repository which can be downloaded with

```
git clone git://git.savannah.gnu.org/groff.git
```