

NAME

nanorc – GNU nano's rcfile

DESCRIPTION

This manual page briefly documents GNU **nano**'s rcfile.

nano is a small, free and friendly editor which aims to replace Pico, the default editor included in the non-free Pine package. Rather than just copying Pico's look and feel, **nano** also implements some missing (or disabled by default) features in Pico, such as "search and replace" and "go to line and column number".

The *nanorc* file contains the default settings for **nano**. It should not be in DOS or Mac format. During startup, **nano** will first read its system-wide settings from *SYSCONFDIR/nanorc*, and then user-specific settings from *~.nanorc*.

OPTIONS

The configuration file accepts a series of **set** and **unset** commands, which can be used to configure nano on startup without using the command line options. Additionally, the **syntax**, **color**, and **icolor** keywords are used to define syntax highlighting rules for different text patterns. **nano** will read one command per line.

Options in nanorc files take precedence over nano's defaults, and command line options override nanorc settings. Also, options are unset by default, except for those that take arguments.

Quotes inside string parameters don't have to be escaped with backslashes. The last double quote in the string will be treated as its end. For example, for the **brackets** option, *"")>J]"* will match *"*, *'*, *)*, *>*, *J*, and *J*.

The supported commands and arguments are:

set/unset autoindent

Use auto-indentation.

set/unset backup

Create backup files in *filename~*.

set backupdir *directory*

Set the directory where **nano** puts unique backup files if file backups are enabled.

set/unset backwards

Do backwards searches by default.

set/unset boldtext

Use bold text instead of reverse video text.

set brackets *string*

Set the characters treated as closing brackets when justifying paragraphs. They cannot contain blank characters. Only closing punctuation, optionally followed by closing brackets, can end sentences. The default value is *"")>J]"*.

set/unset casesensitive

Do case sensitive searches by default.

set/unset const

Constantly display the cursor position in the status bar.

set/unset cut

Use cut to end of line by default.

set fill *n*

Wrap lines at column number *n*. If *n* is 0 or less, the maximum line length will be the screen width less *n* columns. The default value is -8.

set/unset historylog

Enable `~.nano_history` for saving and reading search/replace strings.

set matchbrackets *string*

Set the opening and closing brackets that can be found by bracket searches. They cannot contain blank characters. The former set must come before the latter set, and both must be in the same order. The default value is "`(<[{}>])`".

set/unset morespace

Use the blank line below the titlebar as extra editing space.

set/unset mouse

Enable mouse support, if available for your system. When enabled, mouse clicks can be used to place the cursor, set the mark (with a double click), and execute shortcuts. The mouse will work in the X Window System, and on the console when gpm is running.

set/unset multibuffer

Allow inserting files into their own buffers.

set/unset noconvert

Don't convert files from DOS/Mac format.

set/unset nofollow

Don't follow symlinks when writing files.

set/unset nohelp

Don't display the help lists at the bottom of the screen.

set/unset nonewlines

Don't add newlines to the ends of files.

set/unset nowrap

Don't wrap text at all.

set operatingdir *directory*

nano will only read and write files inside *directory* and its subdirectories. Also, the current directory is changed to here, so files are inserted from this directory. By default, the operating directory feature is turned off.

set/unset preserve

Preserve the XON and XOFF keys (`^Q` and `^S`).

set punct *string*

Set the characters treated as closing punctuation when justifying paragraphs. They cannot contain blank characters. Only closing punctuation, optionally followed by closing brackets, can end sentences. The default value is "`!.?'`".

set/unset quickblank

Do quick statusbar blanking. Statusbar messages will disappear after 1 keystroke instead of 25.

set quotestr *string*

The email-quote string, used to justify email-quoted paragraphs. This is an extended regular expression if your system supports them, otherwise a literal string. The default value is "`^[\t]*[#:>\\]]+`" if you have extended regular expression support, or "`>`" otherwise. Note that `'\t'` stands for a literal Tab character.

set/unset rebinddelete

Interpret the Delete key differently so that both Backspace and Delete work properly. You should only need to use this option if Backspace acts like Delete on your system.

set/unset rebindkeypad

Interpret the numeric keypad keys so that they all work properly. You should only need to use this option if they don't, as mouse support won't work properly with this option enabled.

set/unset regexp

Do extended regular expression searches by default.

set/unset smarthome

Make the Home key smarter. When Home is pressed anywhere but at the very beginning of non-whitespace characters on a line, the cursor will jump to that beginning (either forwards or backwards). If the cursor is already at that position, it will jump to the true beginning of the line.

set/unset smooth

Use smooth scrolling by default.

set speller *spellprog*

Use spelling checker *spellprog* instead of the built-in one, which calls *spell*.

set/unset suspend

Allow **nano** to be suspended.

set tabsize *n*

Use a tab size of *n* columns. The value of *n* must be greater than 0. The default value is 8.

set/unset tabstospaces

Convert typed tabs to spaces.

set/unset tempfile

Save automatically on exit, don't prompt.

set/unset view

Disallow file modification.

set whitespace *string*

Set the two characters used to display the first characters of tabs and spaces. They must be single-column characters.

set/unset wordbounds

Detect word boundaries more accurately by treating punctuation characters as parts of words.

syntax *str* ["*fileregex*" ...]

Defines a syntax named *str* which can be activated via the `-Y/--syntax` command line option, or will be automatically activated if the current filename matches the extended regular expression *fileregex*. All following **color** and **icolor** statements will apply to *syntax* until a new syntax is defined.

The *none* syntax is reserved; specifying it on the command line is the same as not having a syntax at all. The *default* syntax is special: it takes no *fileregex*, and applies to files that don't match any other syntax's *fileregex*.

color *fgcolor,bgcolor regex* ...

For the currently defined syntax, display all expressions matching the extended regular expression *regex* with foreground color *fgcolor* and background color *bgcolor*, at least one of which must be specified. Legal colors for foreground and background color are: white, black, red, blue, green, yellow, magenta, and cyan. You may use the prefix "bright" to force a stronger color highlight for the foreground. If your terminal supports transparency, not specifying a *bgcolor* tells **nano** to attempt to use a transparent background.

icolor *fgcolor,bgcolor regex* ...

Same as above, except that the expression matching is case insensitive.

color *fgcolor,bgcolor start="sr" end="er"*

Display expressions which start with the extended regular expression *sr* and end with the extended regular expression *er* with foreground color *fgcolor* and background color *bgcolor*, at least one of which must be specified. This allows syntax highlighting to span multiple lines. Note that all subsequent instances of *sr* after an initial *sr* is found will be highlighted until the first instance of *er*.

icolor *fgcolor,bgcolor start="sr" end="er"*

Same as above, except that the expression matching is case insensitive.

include *syntaxfile*

Read in self-contained color syntaxes from *syntaxfile*. Note that *syntaxfile* can only contain **syntax**, **color**, and **icolor** commands.

FILES

SYSCONFDIR/nanorc

System-wide configuration file

~/.nanorc

Per-user configuration file

SEE ALSO

nano(1)

/usr/share/doc/nano/examples/nanorc.sample (or equivalent on your system)

AUTHOR

Chris Allegretta <chrisa@asty.org>, et al (see *AUTHORS* and *THANKS* for details). This manual page was originally written by Jordi Mallach <jordi@gnu.org>, for the Debian system (but may be used by others).