NAME

nanorc - GNU nano's rcfile

DESCRIPTION

The *nanorc* file contains the default settings for **nano**, a small and friendly editor. The file should be in Unix format, not in DOS or Mac format. During startup, **nano** will first read the system-wide settings, from **/etc/nanorc** (the exact path might be different), and then the 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 command-line options. Additionally, there are some commands to define syntax highlighting and to rebind keys -- see the two separate sections on those. **nano** reads one command per line.

Options in *nanorc* files take precedence over nano's defaults, and command-line options override *nanorc* settings. Also, options that do not take an argument are unset by default. So using the **unset** command is only needed when wanting to override a setting of the system's *nanorc* file in your own **~/.nanorc**. Options that take an argument cannot be unset.

Below, the *string* parameters need to be enclosed in double quotes. Quotes inside these 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, **')>]}** will match , ',), >,], and }.

The supported commands and arguments are:

set allow_insecure_backup
When backing up files, allow the backup to succeed even if its permissions can't be (re)set due to special OS considerations. You should NOT enable this option unless you are sure you need it.

set autoindent
Use auto-indentation.

set backup
Create backup files by adding a tilde (~) to each file's name.

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

set backwards
Do backwards searches by default.

set boldtext
Use bold instead of reverse video for the titlebar, statusbar, key combos, and selected text. This can be overridden for the first three by setting the options **titlecolor**, **statuscolor**, and **keycolor**.

set brackets *string*
Set the characters treated as closing brackets when justifying paragraphs. This may not include blank characters. Only closing punctuation (see **punct**), optionally followed by the specified closing brackets, can end sentences. The default value is **')>]}**.

set casesensitive
Do case-sensitive searches by default.

set const
Constantly display the cursor position in the status bar.

set cut
Use cut-to-end-of-line by default, instead of cutting the whole line.

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

set functioncolor *fgcolor*,*bgcolor*
Specify the color combination to use for the function descriptions in the two help lines at the bottom of the screen. See **set titlecolor** for more details.

set historylog
Enable the use of **~/.nano/search_history** for saving and reading search/replace strings.

set keycolor *fgcolor*,*bgcolor*
Specify the color combination to use for the shortcut key combos in the two help lines at the bottom of the screen. See **set titlecolor** for more details.

set locking
Enable vim-style lock-files for when editing files.

set matchbrackets *string*
Set the opening and closing brackets that can be found by bracket searches. This may not include blank characters. The opening set must come before the closing set, and the two sets must be in the same order. The default value is (<[{)>]}.

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

set 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. Text can still be selected through dragging by holding down the Shift key.

set multibuffer
Allow inserting files into their own buffers.

set noconvert
Don't convert files from DOS/Mac format.

set nofollow
Don't follow symlinks when writing files.

set nohelp
Don't display the two help lines at the bottom of the screen.

set nonewlines
Don't automatically add a newline to the ends of files.

set nowrap
Don't hard-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 poslog
Save the cursor position of files between editing sessions.

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

set punct *string*
Set the characters treated as closing punctuation when justifying paragraphs. This may not include blank characters. Only the specfified closing punctuation, optionally followed by closing brackets (see **brackets**), can end sentences. The default value is **!.?**.

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

set quiet
nano will not report errors in the *nanorc* file nor ask them to be acknowledged by pressing Enter at startup. If this is used, it should be placed at the top of the file to be fully effective.

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, and > otherwise. Note that 't' stands for a literal Tab character.

set 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 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 regexp
Do extended regular expression searches by default.

set 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 smooth
Use smooth scrolling by default.

set softwrap
Enable soft line wrapping for easier viewing of very long lines.

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

set statuscolor *fgcolor*,*bgcolor*
Specify the color combination to use for the statusbar. See **set titlecolor** for more details.

set suspend
Allow **nano** to be suspended.

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

set tabstospaces
Convert typed tabs to spaces.

set tempfile
Save automatically on exit, don't prompt.

set titlecolor *fgcolor*,*bgcolor*
Specify the color combination to use for the titlebar. Valid color names for foreground and background are: **white**, **black**, **red**, **blue**, **green**, **yellow**, **magenta**, and **cyan**. And either *fgcolor* or ,*bgcolor* may be left out.

set view
Disallow file modification.

set whitespace *string*
Set the two characters used to indicate the presence of tabs and spaces. They must be single-column characters.

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

SYNTAX HIGHLIGHTING

Coloring the different syntactic elements of a file is done via regular expressions (see the **color** command below). This is inherently imperfect, because regular expressions are not powerful enough to fully parse a file. Nevertheless, regular expressions can do a lot and are easy to make, so they are a good fit for a small editor like **nano**.

A separate syntax can be defined for each kind of file via the following commands:

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 this *str* 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*.

linter *program* [*arg* ...]
For the currently defined syntax, use the given *program* to invoke the linter (this overrides the speller function when defined).

formatter *program* [*arg* ...]
For the currently defined syntax, use the given *program* to automatically reformat text. Useful in certain programming languages (e.g. go).

header *regex* **...**
For the currently defined syntax, add one or more regexes which will be compared against the very first line of the file to be edited, to determine whether this syntax should be used for that file.

magic *regex* **...**
For the currently defined syntax, add one or more regexes which will be compared against the result of querying the **magic** database about the file to be edited, to determine whether this syntax should be used for that file. This functionality only works when **libmagic** is installed on the system and will be silently ignored otherwise.

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. Valid colors for foreground and background are: white, black, red, blue, green, yellow, magenta, and cyan. You may use the prefix bright to get 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 the above commands, from **syntax** to **icolor**.

extendsyntax *str directive* [*arg* ...]
Extend the syntax previously defined as *str* to include new information. This allows you to add a new *color*, *icolor*, *magic*, *header*, or *linter* directive to an already defined syntax -- useful when you want to add to definitions from the system-installed syntax definitions (which are normally not writable).

REBINDING KEYS

Key bindings may be reassigned via the following commands:

bind *key function menu*
Rebinds the key *key* to a new function named *function* in the context of menu *menu* (or in all menus where the function exists by using **all**).

unbind *key menu*
Unbinds the key *key* from the menu named *menu* (or from all menus where it exists by using **all**).

The format of *key* should be one of:

^ followed by an alpha character or the word Space. Example: ^C

M-
followed by a printable character or the word Space. Example: M-C

F followed by a numeric value from 1 to 16. Example: F10

Valid *function* names to be bound include:

help
Invokes the help viewer.

cancel
Cancels the current command.

exit
Exits from the program (or from the help viewer or the file browser).

writeout
Writes the current buffer to disk.

insert
Inserts a file into the current buffer (or into a new buffer when multibuffer is enabled).

whereis
Searches for text in the current buffer -- or for filenames matching a string in the current list in the file browser.

searchagain
Repeats the last search command.

replace
Interactively replaces text within the current buffer.

cut
Cuts and stores the current line (or the marked region).

copytext
Copies the current line (or the marked region) without deleting it.

uncut
Copies the currently stored text into the current buffer at the current cursor position.

mark
Sets the mark at the current position, to start selecting text.

cutrestoffile
Cuts all text from the cursor position till the end of the buffer.

curpos
Shows the current cursor position: the line, column, and character positions.

wordcount
Counts the number of words, lines and characters in the current buffer.

speller
Invokes a spell-checking program (or a linting program, if the current syntax highlighting defines one).

linter
A synonym of **speller** (for when the speller has not been configured).

justify
Justifies the current paragraph.

fulljustify
Justifies the entire current buffer.

indent
Indents (shifts to the right) the currently marked text.

unindent
Unindents (shifts to the left) the currently marked text.

left
Goes left one position (in the editor or browser).

right
Goes right one position (in the editor or browser).

up
Goes one line up (in the editor or browser).

down
Goes one line down (in the editor or browser).

scrollup
Scrolls up one line of text from the current position.

scrolldown
Scrolls down one line of text from the current position.

nextword
Moves the cursor to the beginning of the next word.

prevword
Moves the cursor to the beginning of the previous word.

home
Moves the cursor to the beginning of the current line.

end
Moves the cursor to the end of the current line.

beginpara
Moves the cursor to the beginning of the current paragraph.

endpara
Moves the cursor to the end of the current paragraph.

prevpage
Goes up one screenful.

nextpage
Goes down one screenful.

firstline
Goes to the first line of the file.

lastline
Goes to the last line of the file.

gotoline
Goes to a specific line (and column if specified).

gototext
Switches from targeting a line number to searching for text.

findbracket
Moves the cursor to the bracket (brace, parenthesis, etc.) that matches (pairs) with the one under the cursor.

prevbuf
Switches to editing/viewing the previous buffer when using multibuffer mode.

nextbuf
Switches to editing/viewing the next buffer when using multibuffer mode.

verbatim
Inserts the next character verbatim into the file.

tab
Inserts a tab at the current cursor location.

enter
Inserts a new line below the current one.

delete
Deletes the character under the cursor.

backspace
Deletes the character before the cursor.

undo
Undoes the last performed text action (add text, delete text, etc).

redo
Redoes the last undone action (i.e., it undoes an undo).

refresh
Refreshes the screen.

suspend
Suspends the editor (if the suspending function is enabled, see the suspendenable entry below).

casesens
Toggles case sensitivity in searching (search/replace menus only).

regexp
Toggles whether searching/replacing is based on literal strings or regular expressions.

backwards
Toggles whether searching/replacing goes forward or backward.

prevhistory
Shows the previous history entry in the prompt menus (e.g. search).

nexthistory
Shows the next history entry in the prompt menus (e.g. search).

flipreplace
Toggles between searching for something and replacing something.

flipexecute
Toggles between inserting a file and executing a command.

flipnewbuffer
Toggles between inserting into the current buffer and into a new empty buffer.

tofiles
Starts the file browser, allowing to select a file from a list.

gotodir
Goes to a directory to be specified, allowing to browse anywhere in the filesystem.

dosformat
When writing a file, switches to writing a DOS format (CR/LF).

macformat
When writing a file, switches to writing a Mac format.

append
When writing a file, appends to the end instead of overwriting.

prepend
When writing a file, 'prepends' (writes at the beginning) instead of overwriting.

backup
When writing a file, creates a backup of the current file.

firstfile
Goes to the first file when using the file browser (reading or writing files).

lastfile
Goes to the last file when using the file browser (reading or writing files).

nohelp
Toggles the presence of the two-line list of key bindings at the bottom of the screen.

constupdate
Toggles the constant display of the current line, column, and character positions.

morespace
Toggles the presence of the blank line which 'separates' the titlebar from the file text.

smoothscroll
Toggles smooth scrolling (when moving around with the arrow keys).

softwrap
Toggles the displaying of overlong lines on multiple screen lines.

whitespacedisplay
Toggles the showing of whitespace.

nosyntax
Toggles syntax highlighting.

smarthome
Toggles the smartness of the Home key.

autoindent
Toggles whether new lines will contain the same amount of whitespace as the preceding line.

cuttoend
Toggles whether cutting text will cut the whole line or just from the current cursor position to the end of the line.

nowrap
Toggles whether long lines will be hard-wrapped to the next line.

tabstospaces
Toggles whether typed tabs will be converted to spaces.

backupfile
Toggles whether a backup will be made of the file to be edited.

multibuffer
Toggles the use of multiple file buffers (if support for them has been compiled in).

mouse
Toggles mouse support.

noconvert
Toggles automatic conversion of files from DOS/Mac format.

suspendenable
Toggles whether the suspend sequence (normally ^Z) will suspend the editor window.

Valid *menu* sections are:

main
The main editor window where text is entered and edited.

search
The search menu (AKA whereis).

replace
The 'search to replace' menu.

replacewith
The 'replace with' menu, which comes up after 'search to replace'.

gotoline
The 'goto line (and column)' menu.

writeout
The 'write file' menu.

insert
The 'insert file' menu.

extcmd
The menu for inserting output from an external command, reached from the insert menu.

help
The help-viewer menu.

spell
The interactive spell checker Yes/no menu.

linter
The linter menu.

browser
The file browser for inserting or writing a file.

whereisfile
The 'search for a file' menu in the file browser.

gotodir
The 'go to directory' menu in the file browser.

all
A special name that encompasses all menus. For **bind** it means all menus where the specified *function* exists; for **unbind** it means all menus where the specified *key* exists.

FILES

/etc/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).