NAME

nano − Nano's ANOther editor, an enhanced free Pico clone

SYNOPSIS

nano [*options*] [[+*line*,*column*] *file*]...

DESCRIPTION

nano is a small, free and friendly editor which aims to replace Pico, the default editor included in the non-free Pine package. On top of 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".

EDITING

Entering text and moving around in a file is straightforward: typing the letters and using the normal cursor movement keys. Commands are entered by using the Control (^) and the Alt or Meta (M-) keys. Typing **^K** deletes the current line and puts it in the cutbuffer. Consecutive **^K**s will put all deleted lines together in the cutbuffer. Any cursor movement or executing any other command will cause the next **^K** to overwrite the cutbuffer. A **^U** will paste the current contents of the cutbuffer at the current cursor position.

When a more precise piece of text needs to be cut or copied, one can mark its start with **^6**, move the cursor to its end (the marked text will be highlighted), and then use **^K** to cut it, or **M-6** to copy it to the cutbuffer. One can also save the marked text to a file with **^O**, or spell check it with **^T**.

The two lines at the bottom of the screen show the most important commands; the built-in help (**^G**) lists all the available ones. The default key bindings can be changed via the .nanorc file -- see **nanorc**(5).

OPTIONS

+*line***,***column*

Places the cursor on line number *line* and at column number *column* (at least one of which must be specified) on startup, instead of the default line 1, column 1.

−A, **−−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.

−B, **−−backup**

When saving a file, back up the previous version of it, using the current filename suffixed with a tilde (~).

−C *directory*, **−−backupdir=***directory*

Make and keep not just one backup file, but make and keep a uniquely numbered one every time a file is saved -- when backups are enabled. The uniquely numbered files are stored in the specified *directory*.

−D, **−−boldtext**

Use bold text instead of reverse video text.

−E, **−−tabstospaces**

Convert typed tabs to spaces.

−F, **−−multibuffer**

Enable multiple file buffers (if support for them has been compiled in).

−G, **−−locking**

Enable vim-style file locking when editing files.

−H, **−−historylog**

Log search and replace strings to *~/.nano/search_history*, so they can be retrieved in later sessions.

−I, **−−ignorercfiles**
Don’t look at the system’s **nanorc** nor at **˜/.nanorc**.

−K, **−−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.

−L, **−−nonewlines**
Don’t add newlines to the ends of files.

−N, **−−noconvert**
Disable automatic conversion of files from DOS/Mac format.

−O, **−−morespace**
Use the blank line below the titlebar as extra editing space.

−P, **−−positionlog**
For the 200 most recent files, log the last position of the cursor, and place it at that position again upon reopening such a file. (The old form of this option, **−−poslog**, is deprecated.)

−Q "*characters***"**, **−−quotestr="***characters***"**
Set the quoting string for justifying. The default is **"^([\t]*[#:>\|}])+"** if extended regular expression support is available, or **"> "** otherwise. Note that **\t** stands for a Tab.

−R, **−−restricted**
Restricted mode: don’t read or write to any file not specified on the command line; don’t read any *nanorc* files nor history files; don’t allow suspending nor spell checking; don’t allow a file to be appended to, prepended to, or saved under a different name if it already has one; and don’t use backup files. This restricted mode is also accessible by invoking **nano** with any name beginning with ’r’ (e.g. "rnano").

−S, **−−smooth**
Enable smooth scrolling. Text will scroll line-by-line, instead of the usual chunk-by-chunk behavior.

−T *number*, **−−tabsize=***number*
Set the size (width) of a tab to *number* columns. The value of *number* must be greater than 0. The default value is 8.

−U, **−−quickblank**
Do quick statusbar blanking. Statusbar messages will disappear after 1 keystroke instead of 25. Note that **−c** overrides this.

−V, **−−version**
Show the current version number and exit.

−W, **−−wordbounds**
Detect word boundaries more accurately by treating punctuation characters as part of a word.

−Y *name*, **−−syntax=***name*
Specify the name of the syntax highlighting to use from among the ones defined in the *nanorc* files.

−c, **−−constantshow**
Constantly show the cursor position. Note that this overrides **−U**.

−d, **−−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.

−h, **−−help**
Show a summary of the available command-line options and exit.

−i, **−−autoindent**
Indent new lines to the previous line's indentation. Useful when editing source code.

−k, **−−cut**
Make the 'Cut Text' command (normally ^K) cut from the current cursor position to the end of the line, instead of cutting the entire line.

−m, **−−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.

−n, **−−noread**
Treat any name given on the command line as a new file. This allows **nano** to write to named pipes: it will start with a blank buffer, and will write to the pipe when the user saves the "file". This way **nano** can be used as an editor in combination with for instance **gpg** without having to write sensitive data to disk first.

−o *directory*, **−−operatingdir=***directory*
Set the operating directory. This makes **nano** set up something similar to a chroot.

−p, **−−preserve**
Preserve the XON and XOFF sequences (^Q and ^S) so they will be caught by the terminal.

−q, **−−quiet**
Do not report errors in the *nanorc* files nor ask them to be acknowledged by pressing Enter at startup.

−r *number*, **−−fill=***number*
Hard-wrap lines at column *number*. If this value is 0 or less, wrapping will occur at the width of the screen less *number* columns, allowing the wrap point to vary along with the width of the screen if the screen is resized. The default value is −8. This option conflicts with **−w** -- the last one given takes effect.

−s *program*, **−−speller=***program*
Use this alternative spell checker command.

−t, **−−tempfile**
Always save a changed buffer without prompting. Same as Pico's **−t** option.

−u, **−−unix**
Save a file by default in Unix format. This overrides nano's default behavior of saving a file in the format that it had. (This option has no effect when you also use **−−noconvert**.)

−v, **−−view**
View-file (read-only) mode.

−w, **−−nowrap**
Disable the hard-wrapping of long lines. This option conflicts with **−r** -- the last one given takes effect.

−x, **−−nohelp**
Don't show the two help lines at the bottom of the screen.

−z, **−−suspend**
Enable the suspend ability.

−$, **−−softwrap**
Enable 'soft wrapping'. This will make **nano** attempt to display the entire contents of any line, even if it is longer than the screen width, by continuing it over multiple screen lines. Since '$' normally refers to a variable in the Unix shell, you should specify this option last when using other options (e.g. 'nano −wS$') or pass it separately (e.g. 'nano −wS −$').

−a, **−b**, **−e**, **−f**, **−g**, **−j**
Ignored, for compatibility with Pico.

INITIALIZATION FILE

nano will read initialization files in the following order: the system's **nanorc** (if it exists), and then the user's **~/.nanorc** (if it exists). Please see **nanorc**(5) for more information on the possible contents of those files.

NOTES

If no alternative spell checker command is specified on the command line nor in one of the *nanorc* files, **nano** will check the **SPELL** environment variable for one.

In some cases **nano** will try to dump the buffer into an emergency file. This will happen mainly if **nano** receives a SIGHUP or SIGTERM or runs out of memory. It will write the buffer into a file named *nano.save* if the buffer didn't have a name already, or will add a ".save" suffix to the current filename. If an emergency file with that name already exists in the current directory, it will add ".save" plus a number (e.g. ".save.1") to the current filename in order to make it unique. In multibuffer mode, **nano** will write all the open buffers to their respective emergency files.

BUGS

Justifications (**^J**) and reindentations (**M-{** and **M-}**) are not yet covered by the general undo system. So after a justification that is not immediately undone, or after any reindentation, earlier edits cannot be undone any more. The workaround is, of course, to exit without saving.

Please report any other bugs that you encounter via https://savannah.gnu.org/bugs/?group=nano.

HOMEPAGE

http://www.nano-editor.org/

SEE ALSO

nanorc(5)

AUTHOR

Chris Allegretta <chrisa@asty.org>, et al (see the files *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).