Logiciel

Fichier nanorc exemple

Le fichier

~/.nanorc

```
## Sample initialization file for GNU nano.
##
## Please note that you must have configured nano with --enable-
nanorc
## for this file to be read!  Also note that this file should not
be in
## DOS or Mac format, and that characters specially interpreted by
the
## shell should not be escaped here.
##
## To make sure an option is disabled, use "unset <option>".
##
## For the options that take parameters, the default value is
given.
## Other options are unset by default.
##
## 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, ""')>]}" will
match
## ", ', ), >, ], and }.

## Make the 'nextword' function (Ctrl+Right) stop at word ends
## instead of at beginnings.
# set afterends

## When soft line wrapping is enabled, make it wrap lines at
blanks
## (tabs and spaces) instead of always at the edge of the screen.
# set atblanks

## Automatically indent a newly created line to the same number of
## tabs and/or spaces as the preceding line -- or as the next line
## if the preceding line is the beginning of a paragraph.
set autoindent

## Back up files to the current filename plus a tilde.
# set backup
```

```
## The directory to put unique backup files in.
# set backupdir ""

## Use bold text instead of reverse video text.
# set boldtext

## The characters treated as closing brackets when justifying
paragraphs.
## This may not include any blank characters.  Only closing
punctuation,
## optionally followed by these closing brackets, can end
sentences.
# set brackets ""')>]}"

## Do case-sensitive searches by default.
# set casesensitive

## Constantly display the cursor position in the status bar.  Note
that
## this overrides "quickblank".
# set constantshow

## Use cut-from-cursor-to-end-of-line by default.
# set cutfromcursor
## (The old form, 'cut', is deprecated.)

## Set the line length for wrapping text and justifying
paragraphs.
## If the value is 0 or less, the wrapping point will be the
screen
## width less this number.
# set fill -8

## Remember the used search/replace strings for the next session.
set historylog

## Display line numbers to the left of the text.
# set linenumbers

## Enable vim-style lock-files.  This is just to let a vim user
know you
## are editing a file [s]he is trying to edit and vice versa.
There are
## no plans to implement vim-style undo state in these files.
set locking

## 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.
# set matchbrackets "(<[{)>]}"

## Use the blank line below the title bar as extra editing space.
# set morespace

## 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 mouse

## Switch on multiple file buffers (inserting a file will put it
into
## a separate buffer).
# set multibuffer

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

## Don't display the helpful shortcut lists at the bottom of the
screen.
# set nohelp

## Don't automatically add a newline when a file does not end with
one.
# set nonewlines

## Don't pause between warnings at startup.  Which means that only
the
## last one will be readable (when there are multiple ones).
# set nopauses

## Don't wrap text at all.
set nowrap

## Set operating directory.  nano will not read or write files
outside
## this directory and its subdirectories.  Also, the current
directory
## is changed to here, so any files are inserted from this dir.  A
blank
## string means the operating-directory feature is turned off.
# set operatingdir ""

## Remember the cursor position in each file for the next editing
session.
# set positionlog
```

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

## 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.
# set punct "!.?"

## Do quick status-bar blanking.  Status-bar messages will
disappear after
## 1 keystroke instead of 26.  Note that "constantshow" overrides
this.
# set quickblank

## The email-quote string, used to justify email-quoted
paragraphs.
## This is an extended regular expression.  The default is:
# set quotestr "^([    ]*([#:>|}]|//))+"

## Fix Backspace/Delete confusion problem.
# set rebinddelete

## Fix numeric keypad key confusion problem.
# set rebindkeypad

## Do extended regular expression searches by default.
# set regexp

## Put the cursor on the highlighted item in the file browser;
## useful for people who use a braille display.
# set showcursor

## 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 smarthome

## Use smooth scrolling as the default.
# set smooth

## Enable soft line wrapping (AKA full-line display).
# set softwrap
```

```
## Use this spelling checker instead of the internal one.  This
option
## does not have a default value.
# set speller "aspell -x -c"

## Allow nano to be suspended.
set suspend

## Use this tab size instead of the default; it must be greater
than 0.
set tabsize 4

## Convert typed tabs to spaces.
# set tabstospaces

## Save automatically on exit; don't prompt.
# set tempfile

## Snip whitespace at the end of lines when justifying or hard-
wrapping.
# set trimblanks
## (The old form, 'justifytrim', is deprecated.)

## Disallow file modification.  Why would you want this in an
rcfile? ;)
# set view

## The two single-column characters used to display the first
characters
## of tabs and spaces.  187 in ISO 8859-1 (0000BB in Unicode) and
183 in
## ISO-8859-1 (0000B7 in Unicode) seem to be good values for
these.
## The default when in a UTF-8 locale:
# set whitespace "»·"
## The default otherwise:
# set whitespace ">."

## Detect word boundaries differently by treating punctuation
## characters as parts of words.
# set wordbounds

## The characters (besides alphanumeric ones) that should be
considered
## as parts of words.  This option does not have a default value.
When
## set, it overrides option 'set wordbounds'.
# set wordchars "<_>."


## Paint the interface elements of nano.  These are examples;
```

```
## by default there are no colors, except for errorcolor.
# set titlecolor brightwhite,blue
# set statuscolor brightwhite,green
# set errorcolor brightwhite,red
# set selectedcolor brightwhite,magenta
# set numbercolor cyan
# set keycolor cyan
# set functioncolor green
## In root's .nanorc you might want to use:
# set titlecolor brightwhite,magenta
# set statuscolor brightwhite,magenta
# set errorcolor brightwhite,red
# set selectedcolor brightwhite,cyan
# set numbercolor magenta
# set keycolor brightmagenta
# set functioncolor magenta


## Setup of syntax coloring.
##
## Format:
##
## syntax "short description" ["filename regex" ...]
##
## 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 filename regexes, and applies to files
that
## don't match any other syntax's filename regexes.
##
## color foreground,background "regex" ["regex"...]
## or
## icolor foreground,background "regex" ["regex"...]
##
## "color" will do case-sensitive matches, while "icolor" will do
## case-insensitive matches.
##
## Valid colors: white, black, red, blue, green, yellow, magenta,
cyan.
## For foreground colors, you may use the prefix "bright" to get a
## stronger highlight.
##
## To use multi-line regexes, use the start="regex" end="regex"
## [start="regex" end="regex"...] format.
##
## If your system supports transparency, not specifying a
background
```

```
## color will use a transparent color.  If you don't want this, be
sure
## to set the background color to black or white.
##
## All regexes should be extended regular expressions.
##
## If you wish, you may put your syntax definitions in separate
files.
## You can make use of such files as follows:
##
## include "/path/to/syntax_file.nanorc"
##
## Unless otherwise noted, the name of the syntax file (without
the
## ".nanorc" extension) should be the same as the "short
description"
## name inside that file.  These names are kept fairly short to
make
## them easier to remember and faster to type using nano's -Y
option.
##
## To include all existing syntax definitions, you can do:
include "/usr/share/nano/*.nanorc"


## Key bindings.
## See nanorc(5) (section REBINDING KEYS) for more details on
this.
##
## The following two functions are not bound to any key by
default.
## You may wish to choose other keys than the ones suggested here.
# bind M-B cutwordleft main
# bind M-N cutwordright main

## Set this if your Backspace key sends Del most of the time.
# bind Del backspace all
```

Les paramètres

Exemple de fichier d'initialisation pour GNU nano.

Notez que vous devez avoir configuré **nano** avec **–enable-nanorc** pour que ce fichier soit lu !

Notez également que ce fichier ne doit pas être au format DOS ou Mac, et que les caractères spécialement interprétés par le shell ne doivent pas être échappés ici.

Pour désactiver une option, utilisez “unset <option>”.

Pour les options qui acceptent des paramètres, la valeur par défaut est donnée.

Les autres options sont désactivées (unset) par défaut.

Les guillemets à l'intérieur des paramètres de chaîne n'ont pas besoin d'être échappés avec des barres obliques inverses.

Le dernier guillemet double de la chaîne sera traité comme sa fin.

Par exemple, pour l'option “brackets”, “”')>]}“ “correspondra à ## ”, ',), >,], et }.

set afterends
: Fait en sorte que la fonction 'nextword' (Ctrl+→) s'arrête à la fin du mot plutôt qu'au début.

set atblanks
: Lorsque soft line wrapping est activé, fait qu'il délimite les lignes au niveau des blancs (tabulations et espaces) au lieu du bord de l'écran.

set autoindent
: Indente automatiquement une nouvelle ligne avec le même nombre de tabulations et / ou d'espaces que la ligne précédente - ou comme la ligne suivante si la ligne précédente est le début d'un paragraphe.

set backup
: Sauvegarde les fichiers sous le nom de fichier actuel plus un tilde.

set backupdir ””
: répertoire où placer les fichiers de sauvegarde uniques.

set boldtext
: Utiliser du texte en gras au lieu du texte vidéo inversé.

set brackets ””')>]}”
: caractères traités comme des crochets fermants lors de la justification des paragraphes.
Cela ne peut inclure aucun caractère blanc.
Seule la ponctuation fermante, éventuellement suivie de ces crochets fermants, peut terminer les phrases.

set casesensitive
: Effectuez des recherches sensibles à la casse par défaut.

set constantshow
: Affiche en permanence la position du curseur dans la barre d'état.
Notez que ceci remplace «quick blank».

set cutfromcursor
: Utilise par défaut la coupure du curseur à la fin de la ligne.
(L'ancienne forme, «cut», est obsolète.)

set fill -8
: Définit la longueur de ligne pour plier le texte et justifier les paragraphes.
Si la valeur est égale ou inférieure à 0, le point de repliage sera la largeur de l'écran moins ce nombre.

set historylog
: Retenir pour la prochaine session les chaînes de recherche / remplacement utilisées.

set linenumbers
: Affiche les numéros de ligne à gauche du texte.

set locking
: Active les fichiers de verrouillage de style vim.

C'est juste pour informer un utilisateur de vim que vous éditez un ou plusieurs fichiers qu'il essaie d'éditer et vice versa.
Il n'est pas prévu d'implémenter l'état undo (de style vim) dans ces fichiers.

set matchbrackets “(<[{)>]}”
The opening and closing brackets that can be found by bracket searches.
They cannot contain blank characters.
Le premier ensemble doit venir avant le dernier, et les deux doivent être dans le même ordre.

set morespace
Utilise la ligne vide sous la barre de titre comme espace d'édition supplémentaire.

set mouse
Active la prise en charge de la souris, si disponible pour votre système.
Lorsqu'elle est activée, des clics de souris peuvent être utilisés pour placer le curseur, définir la marque (avec un double clic) et exécuter des raccourcis.
La souris fonctionnera dans le système X Window et sur la console lorsque gpm est en cours d'exécution.

set multibuffer
Active plusieurs tampons de fichiers (l'insertion d'un fichier le placera dans un tampon séparé).

set noconvert
Ne pas convertir les fichiers du format DOS / Mac.

set nohelp
N'affichez pas les listes de raccourcis utiles en bas de l'écran.

set nonewlines
Ne pas ajouter pas automatiquement de nouvelle ligne lorsqu'un fichier ne se termine pas par une nouvelle ligne.

set nopauses
Ne pas faire de pause entre les avertissements au démarrage.
Ce qui signifie que seul le dernier sera lisible (lorsqu'il y en a plusieurs).

set nowrap
Ne pas du tout replier le texte.

set operatingdir “”
Définir le répertoire de travail.
nano ne lira ni n'écrira pas de fichiers en dehors de ce répertoire et de ses sous-répertoires.
En outre, le répertoire en cours devient celui-ci, de sorte que tous les fichiers sont insérés à partir de ce répertoire.
Une chaîne vide signifie que la fonction du répertoire de travail est désactivée.

set positionlog
Se souvenir de la position du curseur dans chaque fichier pour la prochaine session d'édition.

set preserve
Conservez les touches XON et XOFF (Ctrl+Q et Ctrl+S).

set punct “!.?”
Les caractères traités comme des signes de ponctuation de fermeture lors de la justification des paragraphes.
Ils ne peuvent pas contenir de caractères blancs.
Seule la ponctuation fermante, éventuellement suivie de crochets fermants, peut terminer les phrases.

set quickblank
Effacer rapidement la barre d'état.

Les messages de la barre d'état disparaîtront après 1 frappe au lieu de 26.
Notez que “constantshow” remplace cela.

set quotestr “^([ ]*([#:>|}]|))+” *: Chaîne d'email-quote, utilisée pour justifier les paragraphes d'email-quote. : Il s'agit d'une expression régulière étendue. : La valeur par défaut est: “^([ ]*([#:>|}]|))+”*

set rebinddelete
Correction du problème de confusion de retour arrière / suppression.

set rebindkeypad
Correction du problème de confusion des touches du pavé numérique.

set regexp
Effectuer par défaut des recherches d'expressions régulières étendues.

set showcursor
Place le curseur sur l'élément en surbrillance dans le navigateur de fichiers ; utile pour les personnes qui utilisent un afficheur braille.

set smarthome
Rend la touche Home plus intelligente.
Lorsque Home est pressé n'importe où sauf au tout début de caractères non blancs sur une ligne, le curseur sautera à ce début (soit vers l'avant, soit vers l'arrière).
Si le curseur est déjà à cette position, il sautera au vrai début de la ligne.

set smooth
Utilisez le défilement fluide par défaut.

set softwrap
Activer le retour à la ligne souple (affichage en ligne complète).

set speller “aspell -x -c”
Utiliser ce vérificateur d'orthographe au lieu de l'interne.
Cette option n'a pas de valeur par défaut.

set suspend
Autoriser nano à être suspendu.

set tabsize 4
Utilisez cette taille de tabulation au lieu de celle par défaut ; doit être supérieur à 0.

set tabstospaces
Convertir les tabulations tapées au clavier en espaces

set tempfile
Enregistrer automatiquement à la sortie ; Ne pas demander.

set trimblanks
Couper les espaces à la fin des lignes lors de la justification ou d repliement des lignes.
(L'ancienne forme «justifytrim» est obsolète.)

set view
Interdire la modification de fichier.

set whitespace “»·”
Les deux caractères à une seule colonne utilisés pour afficher les premiers caractères des tabulations et des espaces.
187 en ISO 8859-1 (0000BB en Unicode) et 183 en ISO-8859-1 (0000B7 en Unicode) semblent être de bonnes valeurs pour ces derniers.
Valeur par défaut dans un environnement local UTF-8 : set whitespace “»·”
Valeur par défaut sinon: set whitespace “>.”

set wordbounds
Détecte les limites des mots différemment en traitant les caractères de ponctuation comme des parties de mots.

set wordchars “<_>.

Les caractères (en plus des caractères alphanumériques) qui doivent être considérés comme des parties de mots.
Cette option n'a pas de valeur par défaut.
Lorsqu'il est défini, il remplace l'option «définir les limites de mots».

Couleurs des éléments d'interface de nano.

Voici sont des exemples :

- set titlecolor brightwhite,blue
- set statuscolor brightwhite,green
- set errorcolor brightwhite,red
- set selectedcolor brightwhite,magenta
- set numbercolor cyan
- set keycolor cyan
- set functioncolor green

par défaut, il n'y a pas de couleurs, sauf pour errorcolor.

Dans le .nanorc de root, vous pouvez utiliser :

- set titlecolor brightwhite,magenta
- set statuscolor brightwhite,magenta
- set errorcolor brightwhite,red
- set selectedcolor brightwhite,cyan
- set numbercolor magenta
- set keycolor brightmagenta
- set functioncolor magenta

Mise en place de la coloration syntaxique.

Format :

syntaxe “short description” [“filename regex” …]

La syntaxe “none” est réservée; le spécifier sur la ligne de commande revient à ne pas avoir du tout de syntaxe.

La syntaxe “default” est spéciale : elle ne prend aucune expression régulière de nom de fichier et s'applique aux fichiers qui ne correspondent à aucune autre expression rationnelle de nom de fichier de syntaxe.

color foreground,background “regex” [“regex”…]
ou icolor foreground,background “regex” [“regex”…]

“color” fera des correspondances sensibles à la casse, tandis que “icolor” fera des correspondances insensibles à la casse.
Couleurs valides: white, black, red, blue, green, yellow, magenta, cyan.

Pour les couleurs de premier plan, vous pouvez utiliser le préfixe «brillant» pour obtenir une mise en évidence plus forte.

Pour utiliser des expressions régulières multilignes, utilisez le format start=“regex” end=“regex” [start=“regex” end=“regex”…].

Si votre système prend en charge la transparence, ne pas spécifier de couleur d'arrière-plan utilisera une couleur transparente.

Si vous ne le souhaitez pas, assurez-vous de définir la couleur d'arrière-plan sur noir ou blanc.

Toutes les expressions régulières doivent être des expressions régulières étendues.

Si vous le souhaitez, vous pouvez placer vos définitions de syntaxe dans des fichiers séparés.

Vous pouvez utiliser ces fichiers comme suit :

```
include "/path/to/syntax_file.nanorc"
```

Sauf indication contraire, le nom du fichier de syntaxe (sans l'extension «.nanorc») doit être le même que le nom de «description courte» à l'intérieur de ce fichier.

Ces noms sont assez courts pour les rendre plus faciles à mémoriser et à taper plus rapidement à l'aide de l'option -Y de nano.

Pour inclure toutes les définitions de syntaxe existantes, vous pouvez mettre :

```
include "/usr/share/nano/*.nanorc"
```

Basé sur « Article » par Auteur.

From:
https://nfrappe.fr/doc-0/ - **Documentation du Dr Nicolas Frappé**

Permanent link:
https://nfrappe.fr/doc-0/doku.php?id=logiciel:systeme:editeurs:nano:nanorc:exemple:start

Last update: **2022/08/13 22:36**