

[fichier .vimrc](#)

```
" | -----
-----
" | File      : ~/.vimrc
" | By        : sogal
" | -----
-----

" GENERAL OPTIONS :
" -----
-----

set nocompatible      " Ne pas assurer la compatibilité avec l'ancien Vi
set encoding=utf8       " Encodage en utf-8
set mouse=a            " Sélection à la souris
set history=50          " Nombre de commandes dans l'historique

" Montrer les tabulations, espaces insécables et de fin (eol:¶)
set list listchars=nbsp:·,tab:··,trail:·,extends:▶,precedes:◀

set viminfo='20,\\"50 " Options du fichier ~/.viminfo
set backspace=2         " Active la touche Backspace
set wrap
" Wrappe et justifie à 79 caractères avec la touche '@'
map @ {v}! par 79j
" Autorise le passage d'une ligne à l'autre avec les flèches gauche et
droite
set whichwrap=<,>,[,]
" Essaye de garder le curseur dans la même colonne quand on change de
ligne
set nostartofline
" Option de la complétion automatique
set wildmode=list:full
set wildchar=<Tab>
set wildmenu           " Meilleur complétion des commandes dans la ligne
de status
set tabstop=4 shiftwidth=4 expandtab " Pas de vraies tabulations
set softtabstop=4        " Backspace supprime 4 espace d'un coup
set ignorecase          " Les recherches ne sont pas 'case sensitives'...
set smartcase           " ... sauf si le pattern recherché contient des
majuscules
" Le pliage se base sur l'indentation du fichier
set foldmethod=indent
set foldnestmax=3        " 3 niveaux de pliage maxi
set nfoldenable          " Ne pas plier par défaut
set incsearch             " Recherches incrémentales
set hlsearch              " Mettre en surbrillance le mot cherché
set autoread              " Auto lecture si le fichier a été modifié depuis
l'extérieur

" UI :
```

```
" -----
-----
colorscheme molokai " Thème de couleur
syntax enable          " Active la coloration syntaxique
" Contenu de la barre de status
" set statusline=%F\ Type=%y\ %=Pos=%l,%c\ (%p%)\ SLOC=%L\ %P
set laststatus=2        " Barre de status toujours visible
set t_Co=256            " Utiliser 256 couleurs
set textwidth=78         " Largeur maxi du texte inséré
set colorcolumn=80       " Mise en couleur colonne n°80
set showcmd              " Affiche les commandes dans la barre de status
set showmatch             " Affiche la paire de parenthèses
set number                " Afficher les numéros de ligne
set splitright
" set relativenumber     " Mode numérotation des lignes relative
set guifont=Monospace\ 11

" INDENT :
" -----
-----
set smartindent
" Options d'indentation pour un fichier C
set cinoptions=(0
set shiftwidth=4          " Nombre d'espace dans l'indentation

" MAPPING :
" -----
-----
" Remapper pour conserver la sélection
vnoremap < >gv
vnoremap > >gv
" Mapper Y pour copie jusqu'à fin de ligne
map Y y$ 
" U n'est pas utile, en faire un redo (Ctrl - R)
map U <C-r>
" kj est plus rapide à atteindre que <Esc>
inoremap kj <Esc>
cnoremap kj <Esc>
" L sépare une ligne à la position du curseur (inverse de J)
nnoremap L i<Enter><Esc>
" Mapper pour split navigation
nnoremap <C-J> <C-W><C-J>
nnoremap <C-K> <C-W><C-K>
nnoremap <C-L> <C-W><C-L>
nnoremap <C-H> <C-W><C-H>
" Mapper leader key
let mapleader = ","
" ,r passe en mode num ligne relatives
map <leader>r :set relativenumber<CR>
" Folding avec espace
nnoremap <space> za
```

```
" Désactiver la surbrillance du dernier élément recherché en tapant Entrée
nnoremap <CR> :nohlsearch<CR><CR>
" Insérer un # (comment) en début de ligne
nnoremap <C-E> <Esc>I#<Esc>

" POWERLINE :
" -----
-----
"set rtp+=/usr/share/vim/site/plugin/powerline.vim
"python from powerline.vim import setup as powerline_setup
"python powerline_setup()
"python del powerline_setup
"set laststatus=2
"set t_Co=256

" SCROLLING :
" -----
-----
set scrolloff=8           " Conserver 8 lignes visibles en scrollant verticalement
set sidescrolloff=15      " Conserver 15 colonnes visibles en scrollant horizontalement
set sidescroll=1

" ASTUCES :
" -----
-----
" Rouvrir le fichier là où on l'a quitté
au BufReadPost * if line("'"") > 1 && line("'"") <= line("$") | exe
"normal! g'\"'" | endif
" :set scrollbind      " Scroller syncro entre 2 fenêtres

" WRAPPING INTELLIGENT EN FONCTION DU TYPE DE FICHIERS :
" -----
-----
" Définir wrap et textwidth en fonction du fichier ou de son emplacement
function! SetupEnvironment()
    let l:path = expand('%:p')
    if l:path =~ '/home/sogal/.*'
        setlocal textwidth=0
    elseif (&ft == 'c' || &ft == 'html' || &ft == 'sh' || &ft ==
'txt2tags')
        setlocal textwidth=0
    elseif (&ft == 'mail')
        " Largeur à 72 car. pour les mails
        setlocal textwidth=72
    else
        " Le défaut est à 78 car.
        setlocal textwidth=78
```

```
        endif
endfunction
au BufReadPost,BufNewFile * call SetupEnvironment()

" txt2tags
au BufNewFile,BufRead *.t2t map 2l :w<CR>!txt2tags -t html "%" ;
unoconv -f odt "%:r.html" <CR>
au BufNewFile,BufRead *.t2t map 2h :w<CR>!txt2tags -t html -o
/tmp/txt2tags_apercu.html "%" ; firefox
"file:///tmp/txt2tags_apercu.html" <CR>
au BufRead,BufNewFile *.t2t set tw=78
au BufNewFile,BufRead *.t2t set spell
au BufNewFile,BufRead *.t2t setlocal spell spelllang=fr
au BufNewFile,BufRead *.t2t set ft=txt2tags
au BufNewFile,BufRead *.t2t set noexpandtab

" VUNDLE ET CONFIGURATION DES PLUGINS ET DE LEUR ENVIRONNEMENT :
" -----
-----
set rtp+=~/vim/bundle/Vundle.vim
" L'environnement Vim inclus Vundle et l'exécute au lancement
call vundle#begin()
Plugin 'gmarik/Vundle.vim'

" Ajouter les plugins ci-dessous :
" -----
-----
Plugin 'tmhedberg/SimpylFold'                      " Meilleur folding
let g:SimpylFold_docstring_preview=1                 " Voir la docstring du code
foldé

" Meilleur autocomplétion Python
Bundle 'Valloric/YouCompleteMe'

" Fermer la fenêtre d'autocomplétion après usage
let g:ycm_autoclose_preview_window_after_completion=1

" ,g amène vers la définition de l'élément complété
map <leader>g :YcmCompleter GoToDefinitionElseDeclaration<CR>

Plugin 'klen/python-mode'
Plugin 'scrooloose/syntastic'                         " Vérif syntaxe Python
Plugin 'tpope/vim-fugitive'                           " Git wrapper pour Vim

" Navigateur de fichiers dans une side-bar
Plugin 'scrooloose/nerdtree'
map <leader>t :NERDTreeToggle<CR>                   " ,t affiche l'arborescence
let NERDTreeIgnore=['\.pyc$', '\~$']                  " Ignorer fichier .pyc in
NERDTree

" Barre de status (+ light que powerline)
```

```
Plugin 'vim-airline/vim-airline'
Plugin 'vim-airline/vim-airline-themes'
let g:airline_powerline_fonts = 1
let g:airline_extensions = ['branch', 'tabline']
let g:airline_theme = 'badwolf'

" *TOUS* les plugins sont à ajouter et configurer avant cette ligne !
call vundle#end()                                     " Requis
filetype plugin indent on                            " Requis

" DEMARRAGE AUTO DES PLUGINS :
" -----
-----
" autocmd VimEnter * NERDTree

" PYTHON IDE :
" -----
-----
" Exécuter le script avec '2p'
au BufNewFile,BufRead *.py map rp :w<CR>!python3 "%<CR>
" Vérifier conformité du script à la PEP8 avec '2P'
"au BufNewFile,BufRead *.py map 2P :w<CR>!pep8-python3.4 '%<CR>
" Paraméter le fichier pour conformité à la PEP8
au BufNewFile,BufRead *.py
    \ set tabstop=4 |
    \ set softtabstop=4 |
    \ set shiftwidth=4 |
    \ set textwidth=79 |
    \ set expandtab |
    \ set autoindent |
    \ set fileformat=unix |

"let python_highlight_all=1                                " Highlight du code

" Configuration de python-mode :
let g:pymode = 1
let g:pymode_options = 0
" Rope code helpers
let g:pymode_rope = 1
let g:pymode_rope_lookup_project = 0
let g:pymode_rope_project_root = ""
let g:pymode_rope_completion = 1
let g:pymode_rope_completion_bind = '<C-Space>'
let g:pymode_rope_autoimport_modules = ['os', 'shutil', 'datetime']
" Check du code
let g:pymode_lint = 1
let g:pymode_lint_on_fly = 1
let g:pymode_lint_on_write = 1
let g:pymode_lint_message = 1
" Syntax et indentation
let g:pymode_indent = 1
```

```
let g:pymode_syntax = 1
let g:pymode_syntax_all = 1
" Documentation
let g:pymode_doc = 1
let g:pymode_doc_bind = 'K'
" Support virtualenv
let g:pymode_virtualenv = 1
" Exécuter le code
let g:pymode_run = 0
```

From:
<http://debian-facile.org/> - **Documentation - Wiki**



Permanent link:
<http://debian-facile.org/utilisateurs:sogal:config:vimrc>

Last update: **29/01/2017 15:45**