

Astuces

De Vim-fr

Cette page a pour but de contenir un paquet de petites découvertes sur le fonctionnement de vim. N'hésitez pas à ajouter les vôtres.

Sommaire

- 1 Mode insertion
- 2 Mode normal
- 3 Pour votre .vimrc
- 4 Autres
- 5 Le mode "Lecture Seul" ou "Read Only"
- 6 La complétion automatique selon le type de fichier

Mode insertion

- C-y Permet de recopier le caractère de la ligne du dessus en mode insertion.
- ^T,^D Permet d'indenter la ligne courante vers la droite, vers la gauche.
- <C-o> Permet, dans le mode insertion, d'exécuter une commande du mode normal (par exemple dd) puis de retourner dans le monde insertion.
- <C-v> Permet de sélectionner en block.
- <C-v>
<Shift
i> Permet de sélectionner en block et d'éditer le bloc sélectionné en mode insertion : on tape le même texte sur toutes les lignes sélectionnées à partir de la position du buffer.

Mode normal

- :mo,:co Permet de déplacer ou de copier les lignes sélectionnées à l'adresse indiquée
- :ce,:ri,:le Permet de ce(ntrer), et de justifier à droite (right) ou à gauche (left) le texte sélectionné
- gi Permet de rentrer en mode insertion à l'endroit où la dernière insertion a été effectuée.
- I,A Permet de rentrer en mode insertion au début de la ligne courante, à la fin de la ligne courante.
- ^,<Entrer>,- Permet d'aller sur le premier caractère de la ligne courante, permet de faire la même chose mais pour la ligne suivante, permet d'aller à la ligne précédente.
- gv Permet de récupérer la précédente sélection visuel que l'on vient de perdre suite à l'application d'une commande ou une autre manipulation.
- zz Permet de centrer la ligne courante sur l'écran.
- ZZ,:wq,:x Permet d'enregistrer puis de fermer vim.
- :<#ligne1>,
<#ligne2>!fmt Formate le texte dans la portée à 75 colonnes.

| | |
|------------------------------------|--|
| <code>`,`</code> | Permet de retourner à l'endroit (ligne + colonne) où l'on était avant un saut (par exemple suite à une recherche). |
| <code>:e!</code> | Recharge le fichier courant. |
| <code>:marks</code> | Permet de lister toutes les marques posées, en indiquant le nom de la marque, la ligne, la colonne, ainsi que le fichier concerné. |
| <code>:%s/\n//,:%s/^\.*\$//</code> | Permet de supprimer toutes les lignes vides. La deuxième solution permet de supprimer également les lignes ne contenant que des espaces, c'est à dire sans caractères <i>visibles</i> mais pas vides non plus. |

Pour votre .vimrc

| | |
|---|--|
| <code>:source ~/.vimrc</code> | Permet de recharger son vimrc sans avoir à relancer vim. Cela marche aussi pour les scripts que vous venez de modifier ou que vous êtes en train de développer. |
| <code>set scrolloff=1000</code> | Permet d'avoir constamment la ligne courante au centre de l'écran. |
| <code>au FileType * setl fo- =cro</code> | Désactive l'autocomplétion pour les lignes de commentaires. |
| <code>inoremap (()<left></code> | Ferme une parenthese, et se place entre les deux. Cf vim.wikia (http://vim.wikia.com/wiki/Automatically_append_closing_characters) pour des solutions plus avancées. |
| <code>autocmd BufNewFile * silent! 0r ~/Templates/:%:e.tpl</code> | Lors de la création d'un nouveau fichier vim ira voir si un fichier <code><code><extension>.tpl</code> existe dans votre <code>\$HOME/Templates/</code> et le chargera. Astuce volé à calico.jeff. |
| <code>abbr </> </<C-x><C-0></code> | Fermeture automatique des tags lors d'édition html lorsque vous tapez " <code></></code> " |
| <code>silent! runtime macros/matchit.vim</code> | Le plugin matchit est intégré au runtime ! (vérifiez dans <code>/usr/share/vim/vim7X/macros</code>) |
| <code>nnoremap n nzz</code> | Lors d'une recherche, en appuyant sur 'n' (correspondance suivante), les correspondances seront centrée dans la fenêtre. |

Autres

| | |
|---|--|
| <code>vim +numéro_de_ligne fichier,vim fichier +numéro_de_ligne,\$</code> | Permet d'ouvrir un fichier de se placer à la ligne indiquée. Le "\$" permet d'indiquer la dernière ligne du fichier. |
| <code>:mksession session.vim,\$ vim -S session.vim</code> | Permet d'enregistrer une session, de charger une session. |
| <code>:set t_Co=256</code> | Permet, dans certains cas, de résoudre certains problèmes de coloration avec certains colorscheme. |
| <code>:!cp % {, .save}</code> | Permet de sauvegarder le fichier courant avec l'extension <code>.save</code> |
| <code>K</code> | Permet d'afficher la documentation de la fonction ou de l'objet sur lequel le curseur est placé. |
| <code>:e .</code> | Ouvre un navigateur de fichiers à partir du répertoire courant. |
| <code>if glob("/path/to/file", 1) != ""</code> | Permet de déterminer si un fichier existe. |

Le mode "Lecture Seul" ou "Read Only"

Lorsque vous éditez un fichier, et que vous ne possédez pas les droits en écriture sur ce fichier, VIM se met en mode "Read Only".

Vous pouvez vous même décider de mettre VIM en mode "Read Only" avec l'option -R.

Lorsque vous avez ouvert un fichier en mode "Lecture Seul", il est tout de même possible d'enregistrer vos modifications avec la commande :

```
:w !sudo tee %
```

le mot de passe root vous sera bien sûr demandé.

La complétion automatique selon le type de fichier

omnifunc permet de compléter des mots clefs selon le code édité. Un `google?q="omnifunc+vim"` donnerait de bon résultats. Mais rapidement, pour php par exemple:

```
autocmd FileType php set omnifunc=phpcomplete#CompletePHP
```

dans `~/.vimrc` devrait le faire.

savoir aussi que

```
set omnifunc=syntaxcomplete#Complete
```

suffit à activer la chose pour tous les langages avec la coloration syntaxique.

Mais a priori (et c'est le cas pour vim70 sous debian lenny/etch (oui, je sais c'est pas récent récent)) ces configurations sont automatiques pour tous les types de fichiers par le mécanisme de l'autoload pour peu que le filetype soit positionné (merci les ftplugins).

Pour plus d'info, essayer le traditionnel `:help omnifunc` et `:help ft-syntax-omni`.

Sinon, tous les fichiers de la complétion se trouvent ici:

```
ls /usr/share/vim/vimcurrent/autoload/*complete.vim
```

comment ça c'est trop long pour une astuce ? rhaaa, jamais content /o\

Récupérée de « <http://www.vim-fr.org/index.php?title=Astuces&oldid=374154> »

Catégorie : Divers

- Dernière modification de cette page le 6 août 2013 à 14:06.