

Editeur VI

Déplacements

Au sein du fichier

<Ctrl><f>	page suivante
<Ctrl>	page précédente
<Ctrl><d>	½ page suivante
<Ctrl><u>	½ page précédente
G	fin de fichier
:<n>	aller à la ligne n
<n>+	avancer de n lignes
<n>-	reculer de n lignes
{	paragraphe précédent
}	paragraphe suivant
z<Enter>	ligne courante en haut de l'écran
z.	ligne courante au milieu de l'écran
<n>z<Enter>	ligne n au milieu de l'écran

Au sein d'une ligne

0	début de ligne
^	début de ligne (pas les espaces)
\$	fin de ligne
w	déplacement un mot à droite
b	déplacement d'un mot à gauche
)	fin de phrase
(début de phrase
%	recherche la parenthèse ou crochet correspondant

[Haut de page](#)

Modifications

Insertion

i	insertion devant le curseur
I	insertion en début de ligne
a	ajout après le curseur

A	ajout en fin de ligne
o	ajout ligne suivante
O	insertion ligne précédente
:r fichier	insert le fichier à la ligne suivante

Suppression

x	caractère courant
<n>x	n caractères
dw	mot courant
<n>dw	n mots
dd	ligne courante
<n>dd	n lignes
d^	depuis le début de la ligne
d\$	jusqu'à la fin de la ligne
:g/xx/d	efface toutes les lignes contenant xx
:g!/xx/d	efface toutes les lignes ne contenant pas xx
:v/xx/d	efface toutes les lignes ne contenant pas xx
:g/^\$/d	efface toutes les lignes vides

Remplacement

r	caractère courant
R	mode remplacement
cw	le mot courant
c\$ ou C	jusqu'à la fin de la ligne
c^	depuis le début de la ligne
s	remplace 1 caractère et passe en insertion

Recherche/Remplacement

Recherche

/xxx	recherche en avant
?xxx	recherche en arrière
/^xxx	recherche en début de ligne
/[xy]zz	recherche xzz ou yzz
/[x-z]yy	recherche xyy à zyy
n	relance la dernière recherche
N	dernière recherche en sens inverse
`^xx	recherche xx en début de ligne
/lin[uU]?x/	recherche linux ou linUx
<AltGr><#>	recherche mot sous le curseur

<Shift><f> recherche mot sous le curseur

Remplacement

:s/xx/yy	yy à la place de xx dans ligne courante
:s/xx/yy/c	idem avec confirmation
:s/xx/yy/g	remplace dans tout le fichier
:s/xx/yy/cg	idem avec confirmation
:g/xx/s//yy/g	yy à la place de xx dans tout le fichier
:g/xx/s//yy/cg	yy à la place de xx avec confirmation dans tout le fichier
:n,m/xx/s//yy/g	yy à la place de xx des lignes n à m
:%s/xx/yy/g	yy à la place de xx dans tout le fichier
:%s/xx/yy/cg	yy à la place de xx avec confirmation dans tout le fichier
&	répète la précédente commande :s

[Haut de page](#)

Buffer

Mémorisation

yy ou Y	mémorisation ligne courante
"xyy ou "xY	idem dans le buffer x
nyy ou nY	mémorisation de n lignes
"xnny ou "xnY	idem dans le buffer x
y\$	mémorisation de la fin de la ligne
mx	marque le début d'un bloc
y'x	mémorise depuis le début du bloc

Si la référence x du buffer est une lettre majuscule, les mémorisations se font en ajout

Utilisation

p	Ajout du buffer après
"xp	Ajout du buffer x après
P	Insertion du buffer avant
"xP	Insertion du buffer x avant

[Haut de page](#)

Astuces

Exécution commande

:g exécute une commande sur les lignes correspondant à une expression régulière.

```
:g/xx/d      efface toutes les lignes contenant xx
:g!/xx/d     efface toutes les lignes ne contenant pas xx
:g/^$/d      efface les lignes vides
```

Supprimer les débuts de lignes

```
:%norm 10x    efface les 10 premières colonnes de toutes les lignes
:5,10 dw      efface le premier mot des lignes 5 à 10
```

Modifications complexes

```
:2,9norm f"i(^[f:r,$i)
```

- :2,9 on applique de la ligne 2 à la ligne 9 seulement
- norm on applique les commandes usuelles en mode commande
- f" on recherche la première "
- i(on insère une parenthèse ouvrante
- ^[équivalent de la touche **ESC** (on l'obtient avec la séquence **CtrlV** puis **ESC**)
- f: on recherche :
- r, on remplace les : par une virgule
- \$ on va en fin de ligne
- i) on insère une parenthèse fermante

Dos2Unix

Changer tous les <CRLF> par <LF>

```
:%s/<Ctrl><V>/<Ctrl><M>/g
```

Visualiser en mode binaire

VI

```
vi -b <fichier>
```

xxd

```
xxd <fichier>
```

hexdump

```
hexdump -C <fichier>
```

[Haut de page](#)

Macro

Enregistrer une macro

```
q a I , ESC j q
```

- q : enregistrer une macro
- a : nom du registre de la macro
- A partir de là faire les actions à enregistrer. Dans l'exemple (I , ESC j) insertion en début de ligne d'une virgule puis on passe à la ligne suivante.
- q : fin de l'enregistrement de la macro

Exécuter une macro

n fois

Pour exécuter 5 fois la macro du registre a :

```
5 @ a
```

[Haut de page](#)

Quitter / Sauvegarde

```
:w          sauvegarde
:wq         sauvegarde et quitter
:x          sauvegarde et quitter
:w fichier sauvegarde dans fichier
:n,mw fichier sauvegarde dans fichier lignes n à m
```

```
:q          quitter
:q!         quitter sans sauver (abandon)
:e!fichier  abandon puis édition fichier
```

[Haut de page](#)

Commandes diverses

```
.          ré exécute la dernière commande
u          undo : annule l'action précédente
<Ctrl><r>   redo :refait l'action précédente
J          join :concatène la ligne courante et la suivante
:!commande Exécution d'une commande shell
:!!commande Le résultat de la commande est inséré après
:!sh       shell (retour par exit ou <ctrld><d>)
<<         décalage vers la gauche de shiftwidth caractères
>>         décalage vers la droite de shiftwidth caractères
<Ctrl><G>   affiche les infos sur le fichier (nom, nb de
lignes, ...)
```

[Haut de page](#)

Configuration en mode commande

```
:set all    pour visualiser les options
:NoMatchParen inhibition du match parenthesis ne fonctionne pas dans
.vimrc car module pas encore chargé
:DoMatchParen activation du match parenthesis
:set mouse=a activer le copier/coller avec le click droit de la souris
```

[Haut de page](#)

Fichier de configuration

Les options de l'éditeurs sont mémorisées dans le fichier [~/ .exrc](#) ou [~/ .vimrc](#) ou globalement sous [/etc/vim/vimrc](#)

```
colorscheme delek      autre thème pour la coloration syntaxique
set tabstop=4          taille tabulation = 4 caractères
```

```
set shiftwidth=4      taille pour décallage = 4 caractères
set number            affiche les n° de ligne
set autoindent        auto indentation du texte (taper <CTRL><D> en mode
insertion pour revenir en arrière)
set showmode          affiche en bas de l'écran quand on est en mode
insertion
set ic/set noic       recherche indépendante/dépendante de la caste
syn on               coloration syntaxique
let loaded_matchparen = 1 évite le chargement du module qui fait le match
parenthèse
autocmd FileType * setlocal formatoptions-=c formatoptions-=r formatoptions-
=o désactiver l'insertion automatique de commentaires.
```

[Haut de page](#)

Dépannage

Mauvais fonctionnement du copier/coller à la souris

Problème

Quand on essaye de <coller> d'un click de souris, l'éditeur passe dans un mode visuel.

Palliatif

Utiliser la touche <SHIFT> lors du <Coller>

Solution

Le problème vient de l'absence du fichier de configuration. Créer un fichier `~/.vimrc` avec par exemple la ligne de coloration syntaxique pour vim :

```
syn on
```

Penser à créer le fichier `/root/.vimrc` pour les commandes vi passées avec `sudo`

[Haut de page](#)

From:

<https://wiki.iot-acs.fr/> - **Wiki**

Permanent link:

<https://wiki.iot-acs.fr/doku.php?id=all:bibles:linux:vi>

Last update: **2025/02/20 10:47**

