

# vim Tableau

- Objet : vim en tableaux
- Niveau requis :  
[débutant, avisé](#)
- Commentaires : *Tableaux des commande d'édition vim en console.*
- Débutant, à savoir : [Utiliser GNU/Linux en ligne de commande, tout commence là !](#) 😊
- Suivi :
  - Création par [martin\\_mtl](#) le 11/12/2012
  - Testé par [smolski](#) le 17/10/2012
- Commentaires sur le forum : [C'est ici](#)<sup>1)</sup>

## Commande

1. On accède au mode commande en appuyant sur `Echap`

2. Ces commandes peuvent être répétées.

Par exemple :

```
5Dd supprime 5 lignes.
```



On peut annuler la dernière modification avec la commande « u ».

## Les options courantes

### Remarque :

Souvent, on peut précéder ces commandes d'un nombre qui indiquera combien de fois répéter la commande.

Option	Action
x	efface la lettre située sous le curseur.
r	remplace la lettre située sous le curseur par une lettre au choix.
o	ajoute une nouvelle ligne dw efface le mot courant et le place dans la mémoire tampon.
d\$	efface du curseur jusqu'à la fin de la ligne et place cela dans la mémoire tampon.
dd	efface la ligne entière et la place dans la mémoire tampon.
u	annule la dernière opération.
U	annule les opérations sur toute la ligne.
<code>Ctrl+R</code>	Touches de raccourcis qui annule l'annulation.
.	repète la dernière commande effectuée.
p	place le contenu de la mémoire tampon là où se trouve le curseur.
cw	efface la fin du mot et passe en mode insertion pour corriger.
c\$	efface la fin d'une ligne et passe en mode insertion pour corriger.
<code>Ctrl+G</code>	Touches de raccourci qui indique le chemin du fichier édité en cours.

Option	Action
\	Dans une commande, permet de garder le sens littéral d'une touche active sous vim en la précédant immédiatement. Exemple :%s /mp3\//mp3
Ctrl a	Positionné sur un chiffre en augmente la valeur d'une unité. Précédé d'un chiffre, l'augmentation se fera sur la valeur de ce chiffre
Ctrl x	Positionné sur un chiffre en diminue la valeur d'une unité. Précédé d'un chiffre, la dimunition se fera sur la valeur de ce chiffre

### Commandes de remplacement de texte :

Option	Action
:s/old/new	remplace le premier old par new sur la ligne où se trouve le curseur.
:s/old/new/g	remplace tous les old par new sur la ligne où se trouve le curseur.
:3,7s/old/new/g	remplace tous les old par new de la ligne 3 à 7
:%s/old/new/g	remplace tous les old par new dans tout le fichier
:%s/mp3\//mp3/	remplace mp3/ par mp3 dans tout le fichier ( <i>suppression de la barre oblique finale</i> )

### Saisie

On utilise ces *commandes Saisie* en les précédant des *doubles points (:)* puis en les validant.

Commande	Action
a	Ajout de texte derrière le caractère actif
A	Ajout de texte en fin de ligne
i	Insertion de texte devant le caractère actif
I	Insertion de texte en début de ligne
o	Insertion d'une nouvelle ligne sous la ligne active
O	Insertion d'une nouvelle ligne au-dessus de la ligne active

### Astuce

Insérez des caractères sur plusieurs lignes après la colonne d'un caractère précis

En mode commande.

1. On se place après la colonne à modifier
2. On tape : **Ctrl+V**
3. On descend jusqu'à la dernière ligne à modifier.
4. On tape **Maj+I**
5. On tape ce qu'il faut insérer
6. On tape sur la touche du clavier **Echap**

Et c'est fait.

Exemple :

Je dois ajouter une virgule après le premier chiffre de cette ligne :

91

```
92
93
94
```

J'opère comme indiqué :

1. Se placer après le 9
2. `Ctrl+V`
3. Descendre jusqu'au 94
4. `Maj+I`
5. `,`
6. `Echap`

Et j'obtiens :

```
9,1
9,2
9,3
9,4
```

Pour ajouter une virgule après le chiffre 10 :

```
101
102
103
104
105
```

Je me place après ce chiffre 10 et je fais tout pareil que précédemment.  
J'obtiens alors :

```
10,1
10,2
10,3
10,4
10,5
```

À noter qu'on peut tout aussi bien utiliser la commande sed, par exemple pour le 91 :

```
:7,14s/^\(.\)/\1, /
```

Et pour faire le truc de la ligne 7 à la ligne 14 après le second caractère comme pour 101, on rajoute un point ainsi :

```
:7,14s/^\(..\)/\1, /
```

- `\1` ça veut dire « ce qu'il y avait entre la première paire de parenthèses échappées »

Sacré **matelot** tout de même qu'il en connaît des combines, hein ! 😊

## Quitter

1. La commande **ZZ** quitte et sauve le fichier.
2. On utilise ces *commandes Quitter* en les précédant des *doubles points (:)* puis en les validant.

Commande	Action
q!	quitte sans sauver
q	quitte si le fichier n'a pas été modifié
w	sauve le fichier
wq	sauve et quitte
x	sauve et quitte

## Déplacement

Les déplacements se font en tapant directement les raccourcis indiqués sur le clavier

Commande	Action
h	Vers la gauche
l	Vers la droite
k	Vers le haut
j	Vers le bas
0 (zéro)	Début de ligne
:0	Première ligne
\$	Fin de ligne
:\$	Dernière ligne
w	Mot suivant
b	Mot précédent
f <i>motif</i>	Saut sur le caractère <i>motif</i> indiqué (Exemple "fc" envoie sur le premier caractère "c" suivant)
Ctrl+F	Remonte d'un écran
Ctrl+B	Descend d'un écran
G	Dernière ligne du fichier
NG	Saute à la ligne
:n	Saute à la ligne

## Correction

Commande	Action
x	Efface le caractère sous le curseur
X	Efface le caractère devant le curseur
rc	Remplace le caractère sous le curseur par le caractère "c"
dw	Efface le mot depuis le curseur jusqu'à la fin du mot
d\$	Efface tous les caractères jusqu'à la fin de la ligne
D	Efface tous les caractères jusqu'à la fin de la ligne
dO	Efface tous les caractères jusqu'au début de la ligne
dfc	Efface tous les caractères de la ligne jusqu'au caractère "c"
dG	Efface tous les caractères jusqu'à la dernière ligne, ainsi que la ligne active

Commande	Action
D1G	Efface tous les caractères jusqu'à la première ligne, ainsi que la ligne active
dd	Efface la ligne active

### Astuce :

Commande	Action
gg V G d	En passant ces commandes une par une, cela efface tout le contenu d'un fichier.

1. gg = tu montes à la première ligne,
2. V = tu commences à sélectionner les lignes,
3. G = tu descends jusqu'à la dernière ligne,
4. d = tu effaces tout ce qui a été sélectionné.

## Recherche

Contrairement à un éditeur de texte classique, vi peut rechercher autre chose que des mots simples et fonctionne à l' aide de caractères spéciaux et de critères.

La commande de recherche est le caractère « / ».

1. La recherche démarre du caractère courant à la fin du fichier.
2. Le caractère « ? » effectue la recherche en sens inverse.
3. On indique ensuite le critère, puis .

Exemple :

```
/echo
```

recherche la chaîne "echo" dans la suite du fichier.

4. Quand la chaîne est trouvée, le curseur s'arrête sur le premier caractère de cette chaîne.
5. La commande « n » permet de continuer la recherche dans le sens indiqué au début.
6. La commande « N » effectue la recherche en sens inverse.

Critères - exemple	Action
[FfBb]oule	Foule, foule, Boule, boule
[A-Z]e	Tout ce qui commence par une majuscule avec un e en deuxième position.
[A-Za-Z0-9]	tout ce qui commence par une majuscule, minuscule ou un chiffre
[^a-z]	plage négative tout ce qui ne commence pas par une minuscule
vé.o	le point remplace un caractère, vélo, véto, véro, ...
Au*o	l' étoile est un caractère de répétition, de 0 à n caractères, Auo, Auto, Automoto, ...
.*	l'étoile devant le point, une chaîne quelconque de taille variable
[A-Z][A-Z]*	répétition du motif entre [] de 0 à n fois, recherche d'un mot comportant au moins une majuscule (en début de mot)
^Auto	le ^ indique que la chaîne recherchée devra être en début de ligne
Auto\$	le \$ indique que la chaîne recherchée devra être en fin de ligne

## Remplacement

Pour remplacer du texte, il faut se placer au début de la chaîne à modifier, puis taper l'une des commandes suivantes.



Après ces saisies, il suffit de taper son texte et d'appuyer sur `Echap`.

Commande	Action
<code>cw motif</code>	Remplacement du mot courant
<code>c\$ motif</code>	Remplacement jusqu'à la fin de la ligne
<code>cO motif</code>	Remplacement jusqu'au début de la ligne
<code>cfx motif</code>	Remplacement jusqu'au prochain caractère 'x' dans la ligne courante
<code>c/motif (← Entrée)</code>	Remplacement jusqu'à la prochaine occurrence de la chaîne 'motif'

## Copier-Coller

On utilise la commande « y » (Yank) pour copier du texte, elle-même devant être combinée avec d'autres indications. Pour couper (déplacer), c'est la commande « d ». Pour coller le texte à l'endroit choisi, c'est la commande « p » (derrière le caractère) ou « P » (devant le caractère). Si c'est une ligne complète qui a été copiée, elle sera placée en-dessous de la ligne active.

Commande	Action
<code>y</code>	copier du texte
<code>d</code>	couper (déplacer)
<code>p</code>	coller le texte derrière le caractère
<code>P</code>	coller le texte devant le caractère
<code>yy</code>	copier une ligne
<code>5yy</code>	copier cinq lignes

## Tampon (Mémoire)

L'éditeur vi dispose de 26 tampons pour y stocker les données que l'on peut nommer comme on le souhaite. On utilise pour ça le « " ».

Tampon exemple	Action
<code>"m1y5w</code>	copier cinq mots dans la mémoire m1
<code>"m1p</code>	coller le contenu de la mémoire m1 à un endroit donné

## Substitution

La substitution permet de remplacer automatiquement plusieurs occurrences par une autre chaîne.

```
:[1ere ligne, dernière ligne]s/Modèle/Remplacement/[gc]
```

- Les numéros de lignes sont optionnels. Dans ce cas la substitution ne se fait que sur la ligne courante.

- En remplacement des numéros de lignes :
  - « . » détermine la ligne courante
  - « 1 » la première ligne
  - « \$ » la dernière ligne.
- Le modèle est l'un des modèles présenté plus tôt.
- Remplacement est une chaîne quelconque qui remplacera le modèle.
- Par défaut seule la première occurrence est remplacée.
- La lettre « g » indique qu'il faut remplacer toutes les occurrences.
- Avec « c », vi demande une confirmation pour chacune des occurrences.

## Exemple

```
:1,$s/[Uu]nix/UNIX/g
```

Cet exemple remplace, dans tout le fichier, Unix ou unix par UNIX.

## Autres...

On utilise ces *commandes* en les précédant des *doubles points* (:) puis en les validant.

Commande	Action
w Nom_fichier	Sauve le fichier sous Nom_fic, en l' écrasant ou en le créant
1,10w Nom_fichier	Sauve les lignes 1 à 10 dans Nom_fic
r Nom_fichier	Insère le fichier Nom_fic à partir de la ligne courante
! commande	Exécute la commande puis retourne à l'éditeur
r! commande	Exécute la commande et insère le résultat à partir de la ligne courante
f Nom_fichier	Affiche en bas d'écran le nom du fichier, le nombre de ligne et la position actuelle
e Nom_fichier	Le fichier est chargé. Un message indique si le précédent a été modifié
e #	Le dernier fichier chargé est affiché. Permet de commuter entre les fichiers

## Commande set

La commande set permet de configurer l'éditeur.

Commande	Action
set all	affiche l'ensemble des options possibles
set number	affiche les numéros de lignes.
set nonumber	supprime les numéros de lignes.
set nu	affiche les numéros de lignes.
set nonu	supprime les numéros de lignes.
set autoindent	l'indentation est conservée lors d'un retour à la ligne.
set noautoindent	l'indentation n'est pas conservée lors d'un retour à la ligne.
set showmatch	lors de la saisie d'une accolade ou d'une parenthèse de fermeture, celle d'ouverture est affichée un très court instant, puis l'éditeur revient au caractère courant.

Commande	Action
set noshowmatch	lors de la saisie d'une accolade ou d'une parenthèse de fermeture, celle d'ouverture est affichée un très court instant, puis l'éditeur revient au caractère courant.
set showmode	vi affichera une ligne d'état (INPUT MODE).
set noshowmode	vi affichera une ligne d'état (INPUT MODE).
set tabstop=x	définit le nombre de caractères pour une tabulation.

## Tableau d'ensemble

Guide de survie de l'éditeur Vi

Commande	Action
vi fichier1 fichier2	édite les fichiers 1 et 2
i	passage en mode insertion
:	passage en mode commande
Echap	sortir des différents modes
:w	sauvegarde les changements
:q	quitte le document en cours
:q!	quitte sans sauvegarder le document en cours
:wq	enregistre le document en cours et quitte
:f fichier3	enregistre le document en cours sous un autre nom
/test	recherche l'occurrence test et la met en surbrillance
v	passage en mode visualisation pour faire une sélection
d	coupe la sélection
dd	coupe la ligne entière
y	copie la sélection
yy	copie la ligne entière
p	colle après le curseur
a	ajoute après le curseur
A	ajoute après la ligne courante
i	insère avant le curseur
I	insère au début de la ligne courante
o	ouvre une nouvelle ligne en dessous de la ligne actuelle
O (Lettre "o" Maj)	ouvre une nouvelle ligne au-dessus de la ligne courante

## Remerciement

Merci à **artefact** et **melodie** d'avoir offert le tuto vi en pdf ayant insufflé cette mise à jour. 😊

1)

N'hésitez pas à y faire part de vos remarques, succès, améliorations ou échecs !

From:

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

Permanent link:

<http://debian-facile.org/doc:editeurs:vim:tableau>



Last update: **16/11/2019 10:24**