

# Aide Vim

Amnézic

## Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Présentation . . . . .	3
1.2	Commandes de base . . . . .	3
<b>2</b>	<b>Modifier le document</b>	<b>3</b>
2.1	Se déplacer efficacement . . . . .	3
2.2	Insérer du texte . . . . .	4
2.3	Supprimer . . . . .	4
2.4	Sélectionner du texte . . . . .	4
<b>3</b>	<b>Commandes sur le fichier</b>	<b>4</b>
<b>4</b>	<b>Autres</b>	<b>5</b>
4.1	Les dossiers . . . . .	5
4.2	Recherche . . . . .	6

# 1 Introduction

## 1.1 Présentation

Pour trouver de l'aide sur une commande ou un sujet, il suffit d'être en mode normal et de taper: `:[commande]` ou `:[subject]`. Pour l'utilisation d'une commande dans un certain mode, il suffira de taper: `[mode]_[commande ou sujet]`. Les différents modes sont :

- i (insert mode)
- v (visual mode)
- c (command line mode)

Toutes les commandes seront présentées dans un tableau à deux colonnes, la première avec la commande pour une seule itération et la seconde colonne avec la commande pour  $n$  itérations.

## 1.2 Commandes de base

Se déplacer (les flèches directionnelles fonctionnent aussi) (dans le sens horaire en partant d'en haut): k, l, j et h.

Pour insérer du texte : i[texte]

# 2 Modifier le document

## 2.1 Se déplacer efficacement

Début des mots suivants	w	<i>nw</i>
Fin des mots suivants	e	<i>nge</i>
Début des mots précédents	b	<i>nb</i>
Fin des mots précédents	ge	<i>nge</i>
Aller au début de la ligne	0	
Aller à la fin de la ligne	\$	
Aller au prochain [x]	f[x]	<i>nf[x]</i>
Aller au précédent [x]	F[x]	<i>nF[x]</i>
Aller au caractère ouvrant/fermant correspondant	%	
Aller en haut du document	H	
Aller au milieu du document	M	
Aller en bas du document	L	
Ajouter une ligne au-dessus	maj-o	<i>nmaj-o</i>
Ajouter une ligne en-dessous	o	<i>no</i>
Aller à la ligne	<i>n</i>	<i>:n</i>
Se déplacer à la prochaine occurrence de [texte]	/[texte]	
Aller au suivant	n	
A revoir	mots qui se finissent	par [texte]

## 2.2 Insérer du texte

## 2.3 Supprimer

Des caractères	x	nx
Des mots (se place au début du mot restant)	dw	d <i>n</i> w
Des mots (se place avant le mot restant)	de	d <i>n</i> e
Supprimer le mot courant	daw	<i>n</i> daw
Des lignes depuis la courante	dd	n <i>dd</i>
Supprimer le reste de la ligne	D	
Supprimer la ligne jusqu'au curseur (non-inclus)	d0	
Supprimer le contenu jusqu'à la ligne courante	dgg	
Supprimer le contenu jusqu'à la fin du fichier	dG	
Remplacer un caractère par [x]	rx	<i>r</i> <b>n</b> x
Changer du texte	cw[ <i>texte</i> ]	<i>c</i> <b>n</b> w[ <i>texte</i> ]
Changer le reste de la ligne	C	
Changer toute la ligne	S	

## 2.4 Sélectionner du texte

Pour aider à voir ce que vous faites, vous pouvez vous mettre en mode visuel, simplement en cliquant sur v à partir du mode normal.

Sélectionner <i>n</i> lignes	Vj	V <i>n</i> j
Copier du texte (avec espace final)	yw	y <i>n</i> w
Copier du texte (sans espace final)	ye	y <i>n</i> e
Copier le reste de la ligne courante	y\$	
Copier toute la ligne	Y	
Coller du texte (du texte est supposé copié)	p	

## 3 Commandes sur le fichier

Sauvegarder le fichier	:w
Sortir du fichier sans sauvegarder	:q!
Sortir du fichier et sauvegarder	ZZ
Laisser le fichier en suspend	Ctrl-z (puis fg dans la CLI pour revenir dessus)
Ouvrir les fichiers les uns derrière les autres	vim file1 ... fileN
Aller au fichier suivant	:(w pour sauvegarder)next(! ne pas enregistrer)
Aller au n-ième fichier suivant	:(w pour sauvegarder)nnext(! ne pas enregistrer)
Aller au fichier précédent	:(w pour sauvegarder)previous(! ne pas enregistrer)
Aller au n-ième fichier précédent	:(w pour sauvegarder)nprevious(! ne pas enregistrer)
Aller au premier fichier	:(w pour sauvegarder)first(! ne pas enregistrer)
Aller au dernier fichier	:(w pour sauvegarder)last(! ne pas enregistrer)
Redéfinir les fichiers ouverts	:args newFile1 ... newFileN
Ouvrir les fichiers côte à côte	vim -O file1 ... fileN
Se déplacer au fichier à gauche	ctrl-maj-w h
Se déplacer au fichier à droite	ctrl-maj-w l
(Vim) Ouvrir un nouveau fichier à côté du premier	:vsplit file1 ... fileN
Ouvrir les fichiers empilés	vim -o file1 ... fileN
Se déplacer au fichier au-dessus	ctrl-maj-w k
Se déplacer au fichier en-dessous	ctrl-maj-w j
Ouvrir un nouveau fichier en-dessous du premier	:split file1 ... fileN
Quitter tous les fichiers	qall!
Sauvegarder tous les fichiers	wall
Copier coller du texte dans 2 fichiers différents	:edit file1 sélection du texte et copiage :edit file2 p
Revoir page 74	
Voir le contenu d'un fichier	vim -R file
Passer le fichier en read-only (permanent?)	vim -M file
Créer un nouveau fichier à partir d'un fichier déjà existant	aller dans le fichier déjà existant :edit existingFile (travail sur le nouveau fichier) :saveas newFile
Changer le nom du fichier courant	:file newFileName

## 4 Autres

### 4.1 Les dossiers

Pour "raccourcir" le fichier, il est possible de demander à Vim de créer des dossiers, c'est-à-dire réduire ce qui peut être réduit. Donc si vous avez un fichier avec disons 5 fonctions et que chaque fonction fait 20 lignes, de devoir se déplacer parmi les 100 lignes, il sera possible de se déplacer parmi 5 lignes. Pour cela, il suffit de modifier le vimrc.

Il y a plusieurs méthodes de fold :

- manual : il faut faire manuellement chaque dossier
- indent : crée les dossiers à partir de l'indentation

- syntax : crée les dossiers à partir de la syntaxe du langage du fichier

## 4.2 Recherche

Pour rechercher un caractère ou une suite de caractères dans le fichier, il suffit de faire (en mode normal) `/[texte]`. Le texte est en réalité interprété comme une regex. Donc :

- `/a*` : permet de chercher "", "a", "aa", ...
- `/ab\+` : permet de chercher "ab", "abb", ...
- `/ab\-` : permet de chercher "a" ou "ab"
- `/word1\|...\|wordN` : permet de chercher word1 ou ... ou wordN
- `/\ (ab\)\*` ou `+` ou `-` : permet de faire les mêmes recherches mais sur des groupes de caractères
- `/text` : permet de chercher le texte
- `/text\` : permet de chercher uniquement le texte que l'on veut (par exemple, si on cherche `\i`, `if` et `while` ne seront pas surlignés)
- `/(^ texte)` : permet de chercher tout sauf le texte

Pour ceux qui ont des notions de regex : `*` correspond à `*`, `a\+` correspond à `a+` et `a\-` correspond à `a?`. Les ensembles **évidents** tels que `[a-z]`, `[A-Z]` et `[0-9]` (qui peuvent être combinés d'ailleurs), peuvent aussi être utilisés à la place des caractères ou des mots. D'ailleurs, afin de faciliter la recherche de ces ensembles, il y a les raccourcis suivants (doivent toujours être précédés de `/` pour la recherche):

- `\d` : `[0-9]`
- `\x` : `[0-9a-fA-F]`
- `\s` : caractères espaces
- `\l` : `[a-z]`
- `\u` : `[A-Z]`
- remarques : pour trouver le complémentaire de chaque ensemble, il suffit de faire `\<la lettre en majuscule>`. Par exemple, pour chercher tout sauf les nombres de 0 à 9, on fera `/\D`.