

VIM — CARTE DE RÉFÉRENCE RAPIDE

Mouvement simple

h l k j ... lettre à gauche, droite ; ligne en haut, bas
b w ... mot à gauche, droite
ge e ... fin du mot à gauche, droite
{ } ... début du paragraphe précédent, suivant
() ... début de la phrase précédente, suivante
0 gm ... début, milieu de ligne
^ \$... première, dernière lettre de la ligne
nG ngg ... ligne *n*, par défaut la dernière, première
n% ... pourcentage *n* du document (*n obligatoire*)
n| ... colonne *n* de la ligne courante
% ... autre parenthèse, crochet, commentaire, **#define**
nH nL ... ligne *n* depuis le début, la fin de la fenêtre
M ... milieu de la fenêtre

Insertion & remplacement → mode insertion

i a ... insertion avant, après le curseur
I A ... insertion au début, fin de ligne
gI ... insertion à la première colonne
o O ... nouvelle ligne au dessous, dessus du curseur
rc ... remplace la lettre sous le curseur par *c*
grc ... comme **r**, mais sans changer le formatage
R ... remplace le texte à partir du curseur
gR ... comme **R**, mais sans changer le formatage
cm ... remplace le texte du mouvement *m*
cc ou **S** ... remplace la ligne courante
C ... remplace jusqu'à la fin de la ligne
s ... remplace un caractère et insère
~ ... change la casse et avance d'une lettre
g~m ... change la casse du mouvement *m*
gum gUm ... → minusc., majusc. le texte du mvt. *m*
<m >m ... décale à gauche, droite le texte du mvt. *m*
n<< n>> ... décale *n* lignes à gauche, droite

Suppression

x X ... supprime le caractère sous, avant le curseur
dm ... supprime le texte du mouvement *m*
dd D ... supprime la ligne courante, la fin de la ligne
J gJ ... jointe la ligne courante et suivante, sans espace
:rd↔ ... supprime les lignes d'intervalle *r*
:rdx↔ ... supprime l'intervalle *r* dans le registre *x*

Mode insertion

^c ^vn ... insère le car. *c* littéralement, valeur déc. *n*
^A ... insère le texte précédemment inséré
^@ ... comme **^A** & stoppe l'insertion → mode commande
^Rx ^R Rx ... insère le contenu du reg. *x*, littéralement
^N ^P ... saisie automatique avant, après le curseur
^W ... supprime le mot avant le curseur
^U ... supprime le texte inséré dans la ligne en cours
^D ^T ... décale à gauche, droite d'une tabulation
^Kc₁c₂ ou c₁←c₂ ... saisie le digraphe {*c*₁, *c*₂}
^Oc ... exécute *c* en mode commande temporaire
^X^E ^X^Y ... défilement haut, bas
<esc> ou ^[... abandonne l'édition → mode commande

Copie

"x ... utilise le registre *x* pour la prochaine action
:reg↔ ... affiche le contenu de tous les registres
:reg x↔ ... affiche le contenu du/des registre(s) *x*
ym ... copie le texte de la commande de mouvement *m*
yy ou **Y** ... copie la ligne courante
p P ... colle après, avant la position du curseur
]p [p ... comme **p**, **P**, avec ajustement de l'indentation
gp gP ... comme **p**, **P**, avec le curseur à la fin du texte

Insertion avancée

g?m ... encode en *rot13* le texte de mouvement *m*
n^A n^X ... +*n*, -*n* au nombre sous le curseur
gqm ... justifie le texte du mouvement *m*
:rce w↔ ... centre les lignes *r* à la largeur *w*
:rle i↔ ... aligne à la colonne *i* les lignes *r*
:rri w↔ ... aligne à droite les lignes *r* à la largeur *w*
!mc↔ ... filtre les lignes du mvt. *m* avec la cmd. *c*
n!|c↔ ... filtre *n* lignes avec la commande *c*
:r!c↔ ... filtre l'intervalle *r* avec la commande *c*

Mode visuel

v V ^V ... sélectionne des lettres, lignes, blocs
o ... échange le curseur avec le début de la sélection
gv ... démarre la sélection à la position de la précédente
aw as ap ... sélectionne un(e) mot, phrase, paragraphe
ab aB ... sélectionne un bloc (), un bloc { }

Annulation, répétition & registres

u U ... annule une commande, restaure la ligne
. ^R ... répète une commande, annule l'annulation
n. ... répète la commande avec le nombre *n*
qc qC ... enregistre, ajoute la frappe au registre *c*
q ... stoppe l'enregistrement
@c ... exécute le contenu du registre *c*
@@ ... répète la commande **@** précédente
:@c↔ ... exécute le registre *c* comme une commande *Ex*
:rg/p/c↔ ... exécute la commande *Ex c*
[sur l'intervalle *r* où le motif *p* correspond

Mouvement complexe

- + ... ligne en haut, bas sur le 1er caractère non blanc
B W ... mot séparé par un espace à gauche, droite
gE E ... fin du mot séparé par un espace à gauche, droite
n_ ... n - 1 lignes en bas sur le 1er caractère non blanc
g0 ... début de ligne d'écran
g^ g\$... premier, dernier caractère de la ligne d'écran
gk gj ... ligne d'écran en haut, bas
fc Fc ... caractère *c* suivant, précédent
tc Tc ... avant le caractère *c* suivant, précédent
; , ... répète le dernier **fFtT** en sens opposé
[[]] ... début de section en arrière, avant
[]] [... fin de section en arrière, avant
[()] ... (,) en arrière, avant
[{]} ... {, } en arrière, avant
[m]m ... début de méthode *Java* en arrière, avant
[#]# ... **#if**, **#else**, **#endif** en arrière, avant
[*]* ... début, fin de bloc **/* */** en arrière, avant

Recherche & substitution

/s↔ ?s↔ ... cherche *s* en avant, arrière
/s/o↔ ?s?o↔ ... idem, avec un décalage *o*
n ou /↔ ... répète en avant la dernière recherche
N ou ?↔ ... répète en arrière la dernière recherche
* ... recherche en arrière, avant le mot sous le curseur
g# g* ... idem, avec les correspondances partielles
gd gD ... définition locale, globale du mot sous le curseur
:rs/f/t/x↔ ... substitue *f* par *t* dans l'intervalle *r*
[*x* : **g**—toutes les occurrences, **c**—confirme
:rs x↔ ... répète la subst. avec de nouveaux *r* & *x*

Caractères spéciaux de recherche

`. ^ $` tout caractère unique, début, fin de ligne
`\< \>` début, fin de mot
`[c1-c2]` un caractère unique dans l'intervalle $c_1..c_2$
`[\^c1-c2]` un caractère unique hors intervalle
`\i \f \I \F` .. un identificateur, mot-clé ; sans chiffres
`\f \p \F \P` .. un fichier, car. imprim. ; sans chiffres
`\s \S` un espace simple, autre espacement
`\e \t \r \b` $\langle esc \rangle$, $\langle tab \rangle$, $\langle \leftarrow \rangle$, $\langle \leftarrow \rangle$
`\= * \+` 0..1, 0..∞, 1..∞ des atomes précédents
`\|` sépare deux branches (\equiv ou)
`\(\)` groupe plusieurs éléments dans un atome
`\& \n` le motif entier, n^{ieme} groupe ()
`\u \l` le caractère suivant en majuscule, minuscule
`\c \C` ignore, respecte la casse

Décalage de recherche

`n ou +n` n lignes en avant en colonne 1
`-n` n lignes en arrière en colonne 1
`e+n e-n` n caractères à droite, gauche de la fin
`s+n s-n` n caractères à droite, gauche du début
`;sc` exécute la recherche sc suivante

Marques et déplacement

`mc` marque la position courante avec $c \in [a..Z]$
`'c 'C` .. va à la marque c , C dans n'importe quel fichier
`'0..9` va à la dernière position en sortie
`' ' "` .. va à la position avant le saut, dernière édition
`' []` va au début, fin du texte préc. manipulé
`:marks \leftarrow` affiche la liste des marques actives
`:jumps \leftarrow` affiche la liste des sauts
`n^0` va au n^{ieme} dernier saut dans la liste
`n^I` va au n^{ieme} premier saut dans la liste

Allocation de touches & abréviation

`:map c e \leftarrow` alloue $c \mapsto e$ en mode normal & visuel
`:map! c e \leftarrow` .. alloue $c \mapsto e$ en mode insertion & cmd.
`:unmap c \leftarrow :unmap! c \leftarrow` supprime l'alloc. pour c
`:mk f \leftarrow` .. sauvegarde les paramètres dans le fichier f
`:ab c e \leftarrow` ajoute l'abréviation pour $c \mapsto e$
`:ab c \leftarrow` liste les abréviations commençant par c
`:una c \leftarrow` supprime l'abréviation pour c

Étiquettes

`:ta t \leftarrow` va à l'étiquette t
`:nta \leftarrow` va à la n^{ieme} nouvelle étiquette
`^T` va à l'étiquette sous le curseur, revient
`:ts t \leftarrow` liste les étiquettes pour sélection
`:tj t \leftarrow` va à l'étiquette ou sélectionne si plusieurs
`:tags \leftarrow` affiche la liste des étiquettes
`:npo \leftarrow :n^T \leftarrow` revient de, va à la n^{ieme} dernière
`:tl \leftarrow` va à la dernière étiquette utilisée
`^W}` :`pt t \leftarrow` .. prévisualise l'étiquette sous le curseur, t
`^W}` .. sépare la fenêtre et montre l'étiq. sous le curseur
`^Wz ou :pc \leftarrow` ferme la fenêtre de prévisualisation

Défilement & multi-fenêtrage

`^E ^Y` défile une ligne en haut, bas
`^D ^U` défile une demi-page en haut, bas
`^F ^B` défile une page en haut, bas
`zt ou z \leftarrow` .. place la ligne courante en haut de la fenêtre
`zz ou z.` .. place la ligne courante au centre de la fenêtre
`zb ou z-` place la ligne courante au bas de la fenêtre
`zh zl` défile un caractère à droite, gauche
`zH zL` défile une demi-fenêtre à droite, gauche
`^Ws ou :split \leftarrow` sépare la fenêtre courante en deux
`^Wn ou :new \leftarrow` créé une nouvelle fenêtre vide
`^Wo ou :on \leftarrow` agrandit la fenêtre courante
`^Wj ^Wk` va à la fenêtre au dessous, dessus
`^Ww ^W^W` .. va à la fenêtre au dessous, dessus (enroule)

Commandes Ex (\leftarrow)

`:e f` édite le fichier f , sauf si changements
`:e! f` édite le fichier f (par défaut celui en cours)
`:wn :wN` .. sauve le fichier et édite le suivant, précédent
`:n :N` .. édite le fichier suivant, précédent dans la liste
`:rw` sauvegarde l'intervalle r dans le fichier en cours
`:rw f` sauvegarde l'intervalle r dans le fichier f
`:rw>>f` ajoute l'intervalle r au fichier f
`:q :q!` quitte et confirme, quitte sans sauvegarder
`:wq ou :x ou ZZ` sauvegarde et quitte
`<up> <down>` rappelle les cmd. comme celle en cours
`:r f` insère le fichier f sous le curseur
`:r! c` insère le résultat de la cmd. c sous le curseur
`:args` affiche la liste des arguments
`:rco a rm a` .. copie, déplace l'interv. r après la ligne a

Intervalles Ex

`,` ; sépare deux lignes, la première ligne
`n` un numéro de ligne absolu n
`.` \$ la ligne courante, la dernière ligne
`% *` le fichier entier, la sélection visuelle
`'t` la position de la marque t
`/p/ ?p?` .. la ligne suivante, précédente où p correspond
`+n -n` $+n$, $-n$ au numéro de ligne précédent

Pliage

`zfm` créé un pli avec le mouvement m
`:rfo` créé un pli sur l'intervalle r
`zd zE` supprime le pli sous le curseur, tous
`zo zc z0 zC` ouvre, ferme un pli, récursivement
`[z]z` début, fin du pli ouvert en cours
`zj zk` début, fin du pli suivant, précédent

Divers

`:sh \leftarrow :!c \leftarrow` .. lance un shell, exécute c dans un shell
`K` affiche l'aide `man` du mot-clé sous le curseur
`:make \leftarrow` lance `make`, lit les erreurs et va à la 1ère
`:cn \leftarrow :cp \leftarrow` affiche l'erreur suivante, précédente
`:cl \leftarrow :cf \leftarrow` liste les erreurs, depuis un fichier
`^L ^G` ré-affiche l'écran, nom de fichier et position
`g^G` montre la position détaillée du curseur
`ga` montre la valeur ASCII du caractère
`gf` ouvre le fichier sous le curseur
`:redir>f \leftarrow` redirige la sortie dans le fichier f
`:mkview [f]` sauve la config. [dans le fichier f]
`:loadview [f]` charge la config. [depuis le fichier f]
`^@ ^K ^_ \ Fn ^Fn` touches inusitées