

// lancer vimtutor pour lancer le tuto vim

vim FILENAME<ENTER> (touche Entrée) : ouvrir le fichier "FILENAME" avec vim ; le crée et l'ouvre s'il n'existe pas encore ; exemple : vim file.txt<ENTER> ouvre le fichier "file.txt" avec vim

En bas à droite est indiqué le numéro de la ligne, "," puis le numéro de la colonne (1 caractère = 1 colonne) ; exemple : 20,1 : 20ème ligne, 1ère colonne (curseur juste AVANT le premier caractère)

All signifie que tout le document est affiché à l'écran

Top signifie que l'on se trouve en haut du document

Bot signifie que l'on se trouve en bas du document

50% (par exemple) signifie que l'on se trouve à 50% du document

Naviguer dans vim :

j : flèche du bas, naviguer vers le bas

k : flèche du haut, naviguer vers le haut

h : flèche de gauche, naviguer vers la gauche

h : flèche de droite, naviguer vers la droite

Laisser la touche enfoncée pour naviguer plus vite

Déplacer le curseur avec un compteur :

0 : déplacer le curseur au début de la ligne

\$ : déplacer le curseur à la fin de la ligne

<NUMBER>w (nombre) : déplacer le curseur <NUMBER> mots plus loin ; exemple : 2w déplace le curseur 2 mots plus loin (se place au début du deuxième mot après celui où se trouve le curseur)

<NUMBER>e : déplacer le curseur juste avant le dernier caractère du <NUMBER>ème mot rencontré ; exemples : 1e déplace le curseur juste avant le dernier caractère du mot, 2e déplace le curseur avant le dernier caractère du prochain mot, etc

<ESC> (touche Échap) : revenir au mode Normal (pour pouvoir taper les commandes vim) ; si déjà en mode Normal et qu'une commande a été tapée (même partiellement) mais pas encore validée avec <ENTER>, efface cette commande

:q!<ENTER> : quitter vim sans sauvegarder (toutes les modifications depuis la dernière sauvegarde seront annulées !)

x : effacer (couper) le caractère qui se trouve juste APRÈS le curseur

i : entrer en mode -- INSERT -- (en mode Insertion), pour pouvoir insérer du texte ; le caractère tapé sera écrit juste APRÈS le curseur ; <ESC> pour revenir au mode Normal

A (<SHIFT> (touche Majuscule) + a) : placer le curseur à la fin de la ligne (juste après le dernier caractère) et entrer en mode Insertion

:wq<ENTER> : sauvegarder le fichier et quitter vim (revenir à l'invite de commande)

EFFACER (COUPER, comme <CTRL> + x) DES CARACTERES :

dw : effacer (couper) un mot (effacer jusqu'à la première lettre du prochain mot) ; efface donc aussi l'espace après le mot que l'on veut effacer, le curseur se trouve alors avant la première lettre du prochain mot ;  
ATTENTION : placer le curseur au début du mot (juste avant le premier caractère du mot) que l'on veut effacer avant de taper dw

d<NUMBER>w : effacer (couper) <NUMBER> mots et placer le curseur au début (avant la première lettre) du mot suivant ; exemple : d2w efface les 2 mots suivants et place le curseur au début du 3ème mot suivant

de : effacer (couper) un mot (effacer jusqu'à l'espace après le mot) ; n'efface pas l'espace après le mot que l'on veut effacer, le curseur se trouve alors entre 2 espaces (entre le mot précédent celui effacé et le mot suivant celui effacé) ; on peut alors directement taper un autre mot, qui se trouvera entre ces 2 mots

d\$ : effacer (couper) à partir de la position du curseur jusqu'à la fin de la ligne

dd : effacer (couper) la ligne entière, la remonter, et placer le curseur au début de la ligne suivante ;  
<NUMBER>dd efface <NUMBER> lignes

cw : effacer (couper) un mot (effacer jusqu'à l'espace après le mot) et entrer en mode Insertion

c<NUMBER>w : effacer (couper) <NUMBER> mots et placer le curseur au début (avant la première lettre) du mot suivant et entrer en mode Insertion

ce : modifier jusqu'à la fin du mot (efface (coupe) à partir du curseur jusqu'à la fin du mot et entrer en mode Insertion

c\$ : effacer (couper) à partir de la position du curseur jusqu'à la fin de la ligne et entrer en mode Insertion

cc : effacer (couper) la ligne entière et entrer en mode Insertion

p : insère (colle) le texte précédemment effacé (coupé) avec x, dw, de, d\$ ou dd après le curseur ;

u : annuler la dernière modification du fichier (la dernière commande exécutée, délétion ou insertion) (comme <CTRL> + z ; Annuler) ; en bas à gauche de vim est indiqué combien de changements ont été annulés, quel changement a été annulé (before #<NUMBER> signifie que l'on est revenu à l'état du fichier avant la <NUMBER> modification), et depuis combien de temps en secondes la modification avait été effectuée (ou l'heure de cette modification si plus d'1 minute est passée) ; Si on est revenu à l'état original du document (avant la première modification), il sera affiché "Already a oldest change"

U : faire revenir une ligne à son état original

<CTRL> (touche Contrôle) + r : redo (undo the undo's) (comme <CTRL> + y ; Répéter); annule les annulations (Répète une modification qui a été annulée avec u) ; en bas à gauche de vim est indiqué combien d'annulations ont été annulées, quel changement a été répété (after #<NUMBER> signifie que l'on est revenu à l'état du fichier après la <NUMBER> modification), et depuis combien de temps en secondes la modification avait été effectuée (ou l'heure de cette modification si plus d'1 minute est passée) ; Si on est revenu à l'état final du document (après la dernière modification), il sera affiché "Already at newest change"

ATTENTION : Si on quitte vim, tout l'historique des modifications sera supprimé, on ne pourra pas revenir en arrière ou en avant !

r<CHARACTER> : remplacer le caractère juste après le curseur par <CHARACTER> (n'importe quel caractère du clavier) ; exemple : rx remplace le caractère juste après le curseur par x

<CTRL> + g : montrer la localisation dans le fichier et le statut du fichier ; le nom du fichier, le numéro de la ligne où se trouve le curseur ("line 529" par exemple), le nombre de lignes du fichier ("of 972" par exemple) là où on se trouve dans le fichier en pourcentage ("--54%--" par exemple), le numéro de la colonne où se trouve le curseur ("col 36" par exemple, juste avant le 36ème caractère)

G (<SHIFT> + g) : aller à la fin du fichier  
gg : aller au début du fichier

<NUMBER>G : aller à la ligne <NUMBER>

/<STRING> : chercher la chaîne de caractères <STRING> dans le fichier, vers le bas du fichier ; le curseur se placera alors au début du mot recherché (s'il le trouve). S'il ne trouve pas, affiche "E486: Pattern not found: <STRING>"

PUIS <n> : chercher la même phrase (chaîne de caractère <STRING>) à nouveau, en partant vers le bas

OU <N> : chercher la même phrase à nouveau, en partant vers le haut

?<STRING> : chercher la chaîne de caractères <STRING> dans le fichier, vers le haut du fichier

PUIS <CTRL> + o : revenir de là d'où l'on vient (là où se situait le curseur avant la dernière recherche réussie) ; répéter pour revenir plus loin

<CTRL> + i : aller vers l'avant (refaire la recherche précédente après un <CTRL> + o)

Placer le curseur sur (, [ ou { (parenthèse, crochet ou accolade ouvrant(e)) puis appuyer sur la touche % : naviguer jusqu'au ), ] ou } correspondant (parenthèse, crochet ou accolade fermant(e)) (et inversement)

:s/old/new<ENTER> : remplacer (substituer) la première occurrence de "old" dans la ligne par "new"

:s/old/new/g<ENTER> : remplacer (substituer) toutes les occurrences de "old" dans la ligne par "new" ; g signifie globally (globalement)

:<NUMBER\_1>,<NUMBER\_2>s/old/new/g<ENTER> : remplacer (substituer) toutes les occurrences de "old" dans la ligne "new" entre les lignes <NUMBER\_1> et <NUMBER\_2>

:%s/old/new/gc<ENTER> : trouver toutes les occurrences de "old" dans tout le fichier, avec un prompt qui demande de valider la substitution par "new" ou non

:!<COMMAND><ENTER> : exécute la commande shell <COMMAND> sans sortir de vim (le résultat de la commande s'affiche à l'écran, mais vim reste ouvert en arrière plan) ; <ENTER> après l'affichage du résultat de la

commande retourne à vim ; exemple : `:!ls<ENTER>` liste les fichiers et dossiers du répertoire courant

`:w <FILENAME><ENTER>` : enregistrer sous "FILENAME" ; enregistre le contenu du fichier ouvert dans ce fichier ; le crée tout d'abord s'il n'existe pas encore ; exemple : `:w file.txt<ENTER>` crée le fichier "file.txt" et enregistre le contenu du fichier ouvert dans ce fichier "file.txt" ; si le fichier existe déjà, utiliser `:w! <FILENAME><ENTER>` pour écraser le fichier avec ce contenu (le contenu du fichier file.txt sera alors perdu et remplacé par le contenu du fichier ouvert !)

`v` : entrer en mode Visual ; permet de surligner (mettre en surbrillance) du texte en déplaçant le curseur ; cela permet d'effectuer certaines actions sur ce texte ; on peut utiliser les raccourcis de navigation (comme `<NUMBER>w`, `<NUMBER>e`, `0` et `$` pour naviguer tout en étant en mode Visual)

Surligner un texte en mode Visual puis `d` : supprimer le texte surligné

Surligner un texte en mode Visual puis `:w <FILENAME><ENTER>` : enregistre le texte sélectionné dans le fichier `<FILENAME>` (le crée s'il n'existe pas) ; REMARQUE : '<,>' apparaîtra en bas à gauche après avoir appuyé sur :

`:r <FILENAME>` : insérer le contenu du fichier `<FILENAME>` là où se trouve le curseur

`:r !<COMMAND>` : afficher le résultat de la commande dans le fichier ; exemple : `:r !ls` écrit le résultat de la commande `ls` là où se trouve le curseur

`o` : insérer une ligne en-dessous du curseur et entrer en mode Insertion

`O` : insérer une ligne au-dessus du curseur et entrer en mode Insertion

`e` : déplacer le curseur au dernier caractère du mot

`a` : déplacer le curseur d'un caractère vers la droite et entre en mode insertion

`R` : entre en mode Remplacer ; ce mode est identique au mode Insertion, mais chaque caractère tapé supprime un caractère existant

Surligner un texte en mode Visual puis `y` : copier le texte surligné

`y<NUMBER>w` : copier `<NUMBER>` mots

`yy` : copier toute la ligne ; ou `y<NUMBER>y` : copier `<NUMBER>` lignes

p : insérer (coller) le texte précédemment copié avec y (ou y<NUMBER>w ou y<NUMBER>y) après le curseur

:set ic : rendre la recherche (avec /) insensible à la casse (ignore case) ; :set noic pour désactiver

:set hls : rendre la recherche surbrillante (highlight search) : surligne en jaune les mots trouvés ; :set nohls pour désactiver

:set is : rendre la recherche incrémentielle (incremental search) : surligne en blanc le mot suivant correspondant à la recherche ; :set nois pour désactiver