

Introduction à vim

1. Commandes de base :

1.1 Complétez le tableau ci-dessous :

Mode	Raccourci ou commande	Raccourci alternatif	Description
	h		Se déplacer vers la gauche
	l		Se déplacer vers la droite
	k		Se déplacer vers le haut
	j		Se déplacer vers le bas
commande	i	---	passage en mode insertion
commande	v	---	passage en mode visuel
insertion	Touche echap	---	passage en mode commande
visuel	Touche echap	---	passage en mode commande
commande	V	---	passage en mode visuel (ligne)
	:q	---	quitter
	:q!	---	quitter sans enregistrer
	:w	---	enregistrer le fichier
	:wq	---	enregistrer le fichier et quitter

2. Edition de fichiers textes

2.1 Complétez le tableau ci-dessous :

Mode	Raccourci ou commande	Raccourci alternatif	Description
commande	yy	---	copier la ligne courante
commande	dd	---	couper la ligne courante
commande	p	---	coller après
commande	P	---	coller avant
insertion	ctrl + w	---	effacer le mot précédant
	gg	---	sauter au début de la première ligne du fichier
	G	---	sauter au début de la dernière ligne du fichier
	0	---	sauter au début de la ligne courante
	\$	---	sauter à la fin de la ligne courante
	:n	---	sauter à la ligne n°n
	u	---	undo
	Ctrl + r	---	redo
	:set paste	---	passer en mode collage (paste)
	:set nopaste	---	sortir du mode collage (paste)

2.1 Expliquez l'intérêt du mode « paste »

3. Rechercher / remplacer

3.1 Complétez le tableau ci-dessous :

Mode	Raccourci ou commande	Description
	/ « chaîne recherché »	rechercher une chaîne de caractère
	n	occurrence suivante
	N	occurrence précédente
	:%s/expr1/expr2/g % étant la portée de la recherche Expr1 l'expression à remplacer Expr 2 l'expression dde remplacement	remplacer une expression dans tout le fichier
	:n,ms/expr1/expr2/g Expr1 l'expression à remplacer Expr 2 l'expression dde remplacement n,m de la ligne n à m	Remplacer une expression dans une zone donnée

4. Edition de fichiers multiples

4.1 Expliquez la différence entre buffers et fenêtres

4.2 Complétez le tableau ci-dessous :

Mode	Raccourci ou commande	Description
commande	:ls	lister les buffers actuellement ouverts
commande	:new	ouvrir un nouveau buffer
commande	:bn	passer au buffer suivant
commande	:bp	passer au buffer précédent
commande	:b3 pour le buffer 3 :b4 pour le buffer 4 etc..	passer au buffer numéro <n>
commande	:bd	Fermer le buffer courant
commande	Ctrl +w h	séparer l'écran (horizontalement) en deux fenêtres
commande	Ctrl +w s	séparer l'écran (verticalement) en deux fenêtres
commande	Ctrl +w plus pavé directionnel Exemple : ctrl +w -> pour aller sur la fenêtre de droite	changer de fenêtre
commande	Ctrl +w c	fermer la fenêtre courante

5. Commandes pour le développement, l'édition de code

5.1 Complétez le tableau ci-dessous :

Mode	Raccourci ou commande	Description
	>>	augmenter l'indentation de la ligne
	<<	diminuer l'indentation de la ligne
	==	indenter automatiquement la ligne
	=	indenter automatiquement la selection
	=G en étant au début du fichier	réindenter tout le fichier
	:! Insérer une commande valide	executer une commande shell
	:r fichier par exemple elle insert le contenu de fichier dans notre vim	executer une commande shell sur le fichier courant (ex : wc -l)
	:! La commande voulu > le nom du buffer où on récupère la sortie	executer une commande shell et récupérer la sortie dans le buffer

6. Intégration de plugins avec vundle

Installez, à l'aide de **pathogen**, les plugins suivants :

- nerdcommenter
- vim-airline
- vim-airline-themes
- vim-fugitive
- vim-gitgutter

ainsi qu'au moins deux autres plugins de votre choix, que vous trouvez utiles.

Vous détaillerez ci-dessous les commandes utilisées pour déployer ces plugins et fournirez l'adresse d'un repo GIT accessible à l'enseignant contenant :

- Un tableau des commandes qui semblent utiles dans ces plugins
- Votre répertoire .vim
- Votre fichier vimrc

La notation portera sur le travail réalisé afin de personnaliser votre vim et sur son originalité par rapport aux tutoriaux disponibles.

Tout d'abord j'installe pathogène avec les commandes suivantes :

```
mkdir -p ~/.vim/autoload ~/.vim/bundle  
curl -LSso ~/.vim/autoload/pathogen.vim https://tpo.pe/pathogen.vim
```

Je crée un fichier .vimrc dans mon /home avec l'instruction `execute pathogen#infect()`



The screenshot shows a terminal window titled "vincent@debian: ~" with a nano 3.2 editor open to the file ".vimrc". The editor contains the text "execute pathogen#infect()". The bottom of the window displays a help footer with various keyboard shortcuts and their functions.

```
vincent@debian: ~  
Fichier  Édition  Affichage  Rechercher  Terminal  Aide  
GNU nano 3.2 .vimrc  
execute pathogen#infect()  
[ Lecture de 1 ligne ]  
^G Aide      ^O Écrire    ^W Chercher  ^K Couper    ^J Justifier ^C Pos. cur.  
^X Quitter   ^R Lire fich.^_ Remplacer  ^U Coller    ^T Orthograp.^ Aller lig.
```

Ensuite on va télécharger les plug-ins avec la commande git clone et le lien https vers le plugin

```
vincent@debian:~/vim/bundle$ git clone https://github.com/tpope/vim-fugitive.git
Clonage dans 'vim-fugitive'...
remote: Enumerating objects: 6509, done.
remote: Total 6509 (delta 0), reused 0 (delta 0), pack-reused 6509
Réception d'objets: 100% (6509/6509), 2.18 MiB | 667.00 KiB/s, fait.
Résolution des deltas: 100% (2968/2968), fait.
vincent@debian:~/vim/bundle$ git clone https://github.com/vim-airline/vim-airline-themes.git
Clonage dans 'vim-airline-themes'...
remote: Enumerating objects: 36, done.
remote: Counting objects: 100% (36/36), done.
remote: Compressing objects: 100% (18/18), done.
remote: Total 7796 (delta 9), reused 23 (delta 6), pack-reused 7760
Réception d'objets: 100% (7796/7796), 1.20 MiB | 1.01 MiB/s, fait.
Résolution des deltas: 100% (4313/4313), fait.
vincent@debian:~/vim/bundle$ git clone https://github.com/preservim/nerdcommenter.git
Clonage dans 'nerdcommenter'...
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 2282 (delta 0), reused 0 (delta 0), pack-reused 2278
Réception d'objets: 100% (2282/2282), 851.67 KiB | 536.00 KiB/s, fait.
Résolution des deltas: 100% (948/948), fait.
vincent@debian:~/vim/bundle$ git clone https://github.com/airblade/vim-gitgutter.git
git : 'https://github.com/airblade/vim-gitgutter.git' n'est pas une commande git
. Voir 'git --help'.
vincent@debian:~/vim/bundle$ git clone https://github.com/airblade/vim-gitgutter.git
Clonage dans 'vim-gitgutter'...
remote: Enumerating objects: 31, done.
remote: Counting objects: 100% (31/31), done.
remote: Compressing objects: 100% (20/20), done.
remote: Total 3647 (delta 11), reused 21 (delta 7), pack-reused 3616
Réception d'objets: 100% (3647/3647), 1.24 MiB | 1.08 MiB/s, fait.
Résolution des deltas: 100% (1962/1962), fait.
vincent@debian:~/vim/bundle$ git clone https://github.com/preservim/nerdtree.git
Clonage dans 'nerdtree'...
remote: Enumerating objects: 68, done.
remote: Counting objects: 100% (68/68), done.
remote: Compressing objects: 100% (66/66), done.
remote: Total 5912 (delta 0), reused 61 (delta 0), pack-reused 5844
Réception d'objets: 100% (5912/5912), 1.80 MiB | 838.00 KiB/s, fait.
Résolution des deltas: 100% (2679/2679), fait.
vincent@debian:~/vim/bundle$ git clone https://github.com/airblade/vim-gitgutter.git
git : 'https://github.com/airblade/vim-gitgutter.git' n'est pas une commande git
. Voir 'git --help'.
vincent@debian:~/vim/bundle$ git clone https://github.com/vim-airline/vim-airline-themes.git
Clonage dans 'vim-airline-themes'...
remote: Enumerating objects: 36, done.
remote: Counting objects: 100% (36/36), done.
remote: Compressing objects: 100% (18/18), done.
remote: Total 7796 (delta 9), reused 23 (delta 6), pack-reused 7760
Réception d'objets: 100% (7796/7796), 1.20 MiB | 1.01 MiB/s, fait.
Résolution des deltas: 100% (4313/4313), fait.
```

On voit avec bien dans /home/vincent/.vim/bundle les plugins installé.

```
vincent@debian:~/vim/bundle$ ls
nerdcommenter  nerdtree  vim-airline-themes  vim-fugitive  vim-gitgutter
vincent@debian:~/vim/bundle$
```

Git repository :

Création d'un git repository accessible au lien suivant :

https://github.com/sloginn/tp_vim_vincent_tortel

Création d'un dossier où je mettrais tout les fichiers à publier sur le repository.

```
vincent@debian:~$ mkdir tp_git
```

Copie de .vim et .vimrc dedans.

Initialisation du depot avec git init

```
vincent@debian:~$ cp -r .vim ./tp_git/
vincent@debian:~$ cp -r .vimrc ./tp_git/
vincent@debian:~$ cd tp_git
vincent@debian:~/tp_git$ git init
Dépôt Git vide initialisé dans /home/vincent/tp_git/.git/
```

Rajout de .vim et .vimrc dans le depot

Et configuration de l'origin

```
vincent@debian:~/tp_git$ git add .vim .vimrc
warning: dépôt git embarqué ajouté : .vim/bundle/nerdcommenter
astuce: You've added another git repository inside your current repository.
astuce: Clones of the outer repository will not contain the contents of
astuce: the embedded repository and will not know how to obtain it.
astuce: If you meant to add a submodule, use:
astuce:
astuce:         git submodule add <url> .vim/bundle/nerdcommenter
astuce:
astuce: If you added this path by mistake, you can remove it from the
astuce: index with:
astuce:
astuce:         git rm --cached .vim/bundle/nerdcommenter
astuce:
astuce: See "git help submodule" for more information.
warning: dépôt git embarqué ajouté : .vim/bundle/nerdtree
warning: dépôt git embarqué ajouté : .vim/bundle/vim-airline-themes
warning: dépôt git embarqué ajouté : .vim/bundle/vim-fugitive
warning: dépôt git embarqué ajouté : .vim/bundle/vim-gitgutter
vincent@debian:~/tp_git$ git remote add origin https://github.com/sloginn/tp_vim_vincen
t_tortel.git
```

Git commit pour montrer qu'on a fait des modifications au depot.

```
vincent@debian:~/tp_git$ git commit -m ".vim .vimrc"
[master (commit racine) fa06736] .vim .vimrc
 7 files changed, 270 insertions(+)
 create mode 100644 .vim/autoload/pathogen.vim
 create mode 160000 .vim/bundle/nerdcommenter
 create mode 160000 .vim/bundle/nerdtree
 create mode 160000 .vim/bundle/vim-airline-themes
 create mode 160000 .vim/bundle/vim-fugitive
 create mode 160000 .vim/bundle/vim-gitgutter
 create mode 100644 .vimrc
```

Puis on git push pour qu'elle monte sur le dépôt github.

```
vincent@debian:~/tp_git$ git push origin master
Username for 'https://github.com': vincent.tortel@gmail.com
Password for 'https://vincent.tortel@gmail.com@github.com':
Énumération des objets: 7, fait.
Décompte des objets: 100% (7/7), fait.
Compression des objets: 100% (5/5), fait.
Écriture des objets: 100% (7/7), 3.70 KiB | 3.70 MiB/s, fait.
Total 7 (delta 0), réutilisés 0 (delta 0)
To https://github.com/sloginn/tp_vim_vincent_tortel.git
 * [new branch]      master -> master
vincent@debian:~/tp_git$
```

Le dépôt est donc accessible via : https://github.com/sloginn/tp_vim_vincent_tortel