

Mémento pour le shell

Se déplacer dans l’arborescence

<code>cd <i>rep</i></code>	Aller dans le répertoire <i>rep</i> (change directory)
<code>ls [-l]</code>	Affiche le contenu du répertoire de travail (détails avec -l) (list)
<code>ls <i>rep</i></code>	Affiche le contenu d’un répertoire
<code>pwd</code>	Affiche le répertoire de travail actuel (print working directory)

Manipulation de base des fichiers et répertoires

<code>cp <i>fich1 fich2</i></code>	Copier un fichier sous un nouveau nom (copy)
<code>cp <i>fich1 fich2 rep</i></code>	Copier des fichiers à un nouvel emplacement
<code>mv <i>fich1 fich2</i></code>	Renommer un fichier (move)
<code>mv <i>fich1 fich2 rep</i></code>	Déplacer des fichiers vers un nouvel emplacement
<code>rm <i>fich</i></code>	Effacer un fichier (remove)
<code>rmdir <i>rep</i></code>	Effacer un répertoire (doit être vide)
<code>rm -r <i>rep</i></code>	Effacer une arborescence (attention, pas de garde-fou)
<code>mkdir <i>rep</i></code>	Créer un nouveau répertoire <i>rep</i> (make directory)
<code>chmod <i>droits fich</i></code>	Change les droits, les permissions du fichier (change mode)

Interaction avec les fichiers

<code>less <i>fichier</i></code>	Lire un fichier à l’écran (‘h’ pour l’aide, ‘q’ pour quitter)
<code>emacs <i>fichier</i></code>	Éditeur de texte (voir colonne droite)

Aide en ligne

<code>man <i>commande</i></code>	Affiche la page de manuel de la <i>commande</i>
<code>man 3 <i>fonction</i></code>	Affiche la documentation de la <i>fonction</i>
<code>help test</code>	Aide interne de bash, sur les tests

Gestion de votre compte

<code>passwd</code>	Change de mot de passe
<code>quota -v</code>	Affiche le quota disque de l’utilisateur

Contrôler les processus en cours d’exécution

<code>C-c</code>	Interrompt (tue) le programme en cours
<code>C-z</code>	Suspend (endort) le programme en cours
<code>fg ; bg</code>	Relance tâche interrompue (premier/arrière plan) fore/background
<code>top ; ps</code>	Montre (interactivement ou non) les processus actifs
<code>kill <i>pid</i>; killall <i>nom</i></code>	Envoie un signal (tue) un programme
<code>C-s/C-q</code>	Stoppe/relance le défilement de l’affichage (scrolling)

Manipulation avancée des fichiers

<code>find -name “<i>ab*</i>”</code>	Cherche tous les fichiers dont le nom commence par <i>ab</i>
<code>cat <i>fichier</i></code>	Imprime le <i>fichier</i> sur l’écran (pas interactif)
<code>head -n 5 <i>fichier</i></code>	Affiche les 5 première lignes de <i>fichier</i> sur l’écran
<code>tail <i>fichier</i></code>	Affiche la fin de <i>fichier</i> sur l’écran (-f si <i>fichier</i> grandit encore)
<code>grep <i>motif fichier</i></code>	Imprime les lignes du <i>fichier</i> contenant <i>motif</i>
<code>wc <i>fichier</i></code>	Compte les caractères, les mots et les lignes du <i>fichier</i>
<code>file <i>fichier</i></code>	Affiche le type de fichier (exécutable, texte, image, etc)
<code>diff -u <i>fich1 fich2</i></code>	Affiche les différences entre les deux fichiers
<code>touch <i>fichier</i></code>	Créer ou changer la date de modification de <i>fichier</i>

Raccourcis clavier de Bash

C-a	Aller en début de ligne	C-u	Coupe tout à gauche	C-l	Efface l’écran
C-e	Aller en fin de ligne	C-k	Coupe tout à droite	C-d	Quitte bash
TAB	Complétion auto.	C-r	Recherche historique	S-up	Remonte l’écran

Redirections de flux

<code><i>programme</i> > <i>fichier</i></code>	Sortie du <i>programme</i> redirigée dans <i>fichier</i> .
<code><i>programme</i> < <i>fichier</i></code>	Le programme lit son entrée depuis le <i>fichier</i> .
<code><i>programme</i> >> <i>fich</i></code>	Sortie du <i>programme</i> concaténée à la suite de <i>fich</i> .
<code><i>progA</i> <i>progB</i></code>	Sortie de <i>progA</i> tubée en entrée de <i>progB</i> .
<code><i>prog</i> 2> <i>fichier</i></code>	Le flux 2 (stderr) de <i>prog</i> redirigée dans le <i>fichier</i> .
<code><i>prog</i> 2>> <i>fichier</i></code>	Le flux 2 (stderr) de <i>prog</i> concaténée dans le <i>fichier</i> .
<code><i>prog</i> 2>&1</code>	Flux 2 (stderr) fusionné dans le flux 1 (stdout).
<code><i>prog</i> << marqueur</code>	Entrée standard de <i>prog</i> en place, jusqu’à la ligne marqueur

Guillemets et quotes

<code>\c</code>	Le caractère c, littéralement
<code>‘<i>commande</i>’</code>	Exécute <i>commande</i> et place sa sortie sur la ligne
<code>”blabla”</code>	Garde le blabla après avoir interprété les \$, ‘...’ et \
<code>’blabla’</code>	Garde le blabla sans interpréter quoi que ce soit

Tests pour if [...] (voir aussi help test)

<code>-r fich</code>	Si droit de lecture sur <i>fich</i>	<code>-w fich</code>	Si droit d’écriture sur <i>fich</i>
<code>-x fich</code>	Si droit d’exécution sur <i>fich</i>	<code>-e fich</code>	Si <i>fich</i> existe.
<code>-f fich</code>	Si <i>fich</i> est un fichier standard	<code>-d fich</code>	Si <i>fich</i> est un répertoire.
<code>fA -nt fB</code>	Si fichier <i>fA</i> modifié après <i>fB</i>	<code>-s fich</code>	Si taille supérieure à 0.
<code>A = B</code>	Si les chaînes <i>A</i> et <i>B</i> sont égales	<code>A != B</code>	Si <i>A</i> et <i>B</i> sont différentes
<code>-z CH</code>	Si la chaîne <i>CH</i> est vide	<code>-n CH</code>	Si la chaîne <i>CH</i> n’est pas vide
<code>n1 -eq n2</code>	Si le nombre <i>n1</i> est égal à <i>n2</i>	<code>n1 -ne n2</code>	Si <i>n1</i> \neq <i>n2</i>
<code>n1 -lt n2</code>	Si <i>n1</i> < <i>n2</i> (less than)	<code>n1 -le n2</code>	Si <i>n1</i> \leq <i>n2</i> (less or equal)
<code>n1 -gt n2</code>	Si <i>n1</i> > <i>n2</i> (greater than)	<code>n1 -ge n2</code>	Si <i>n1</i> \geq <i>n2</i> (greater or equal)
<code>e1 -a e2</code>	ET booléen entre expressions	<code>e1 -o e2</code>	OU booléen entre expressions
<code>if [\((-f \$fich1 -a -r \$fich1 \) -o \((-f \$fich2 -a -r \$fich2 \)]</code>			

Gestion d’archives

<code>tar cvfz <i>ar.tgz rep</i></code>	Crée une archive <i>ar.tgz</i> contenant le répertoire <i>rep</i>
<code>tar xvfz <i>arch.tgz</i></code>	Ouvre l’archive <i>arch.tgz</i>
<code>gzip <i>f</i>; gunzip <i>f.gz</i></code>	Compresse / Décompresse un fichier