

## ┌ TD5 - Configuration du shell ┐

Le shell désigne l'interpréteur de commande qui exécute les commandes tapées dans le terminal. Sur la plupart des systèmes Linux, le shell utilisé est le Bourne-Again SHell, dit bash. Dans ce TD, nous allons voir les différents éléments de la configuration d'un shell bash.

### └ Exercice 1 : Alias ┘

Un alias permet de simplifier l'appel à certaines commandes. La syntaxe est la suivante :

```
alias nom-de-l-alias='commande et ses paramètres'
```

Par exemple, vous pouvez renommer la commande ls pour que l'affichage se fasse en couleur :

```
alias ls='ls -color'
```

Créer les alias suivants :

1. ll : commande ls avec option de listing long
2. la : commande ls avec option pour afficher même les fichiers cachés
3. lla : commande ls (ou un déjà existant) avec les deux options précédentes regroupées
4. cd.. : commande cd pour permettre de se rendre dans le répertoire précédent (dans le nom de l'alias cd., les 4 caractères sont collés)
5. rm : ajouter la commande avec l'option qui force la demande avant suppression
6. cp : ajouter la commande avec l'option qui force la demande avant écrasement d'un fichier déjà existant
7. mv : ajouter la commande avec l'option qui force la demande avant écrasement d'un fichier déjà existant

### └ Exercice 2 : .bashrc ┘

Taper un alias en ligne de commande ne permet pas de le conserver lorsqu'un nouveau terminal est lancé. Il est nécessaire de configurer un fichier nommé `.bashrc` pour cela.

1. Créez un fichier nommé `.bashrc` à la racine de votre compte
2. Éditez le fichier avec un éditeur, par exemple `gedit .bashrc`
3. Ajoutez, ligne par ligne, les alias de l'exercice précédent.
4. Sauvez puis lancez un nouveau terminal. Les alias sont-ils présents ?

### ↳ Exercice 3 : Configuration générique <sup>⌈</sup>

Afin de charger la configuration générique de la Fedora 11, il faut insérer les lignes suivantes au début de votre fichier `.bashrc`.

```
# Source global definitions
if [ -f /etc/bashrc ]; then
    . /etc/bashrc
fi
```

Sauvez puis lancez un nouveau terminal : toute modification du fichier `.bashrc` n'est prise en compte qu'au lancement d'un nouveau shell. Qu'a apporté le chargement de la configuration générique ?

### ↳ Exercice 4 : Variables <sup>⌈</sup>

Sous linux, le système est capable d'échanger ou de récupérer des informations par l'utilisation de variables. Ces variables sont rendues accessibles par le shell. Certaines sont des variables prédéfinies du système d'exploitation (variables d'environnement), d'autres définies par le shell. Il est possible de :

- Afficher une variable : `echo $NOMVAR`
- Définir une variable : `NOMVAR=Valeur`
- Modifier une variable : `NOMVAR=NouvelleValeur`

1. Affichez la variable `HOME` Que représente cette variable ?
2. Affichez la variable `PWD` Que représente cette variable ?
3. Affichez la variable `HOSTNAME` Que représente cette variable ?
4. Affichez la variable `USER` Que représente cette variable ?
5. Affichez la variable `PATH` Que représente cette variable ?
6. Selon vous, quelle(s) variable(s) est-il logique de ne pas modifier ?

### ↳ Exercice 5 <sup>⌈</sup>

Vous allez maintenant modifier la variable `PATH`.

1. Tapez `PATH="nimportequoi"` Pouvez-vous encore taper des commandes ? Fermez le terminal puis relancez un nouveau terminal.
2. Tapez `PATH="nimportequoi":$PATH` puis affichez `PATH`. Que pouvez-vous en conclure ?
3. Exécutez la séquence suivante :
  - `javac -version`
  - `PATH=/usr/local/applications/java/jdk1.6.0_14/bin:$PATH`
  - `javac -version`Que pouvez-vous en conclure ?
4. Ajoutez la commande `matlab (2008b)` à votre `PATH`.
5. Ouvrez un autre terminal et affichez le `PATH`. Que faudrait-il faire pour que le celui-ci soit configuré une fois pour toute ?

6. Modifiez le fichier `.bashrc`. Prettez une attention particulière aux erreurs de frappe ou de recopie!

### ↳ Exercice 6 ↴

Afin de spécifier le proxy (la passerelle pour accéder aux pages internet) il est possible de définir deux variables du shell. Ajoutez les deux lignes suivantes dans le bon fichier, puis testez via un `echo`.

- `export http_proxy='http://proxy.devinci.fr:8080'`
- `export ftp_proxy='http://proxy.devinci.fr:8080'`

### ↳ Exercice 7 ↴

Créez un répertoire `bin` à la racine de votre compte.

1. Que va faire la commande `cd $HOME/bin`?
2. Ajoutez le répertoire `bin` à votre `PATH`.

### ↳ Exercice 8 : Scripts ↴

Nous allons maintenant créer un petit script.

1. Créez un fichier nommé `monenv.sh` dans le répertoire `$HOME/bin`
2. Éditez-le et insérez le texte `echo "Mon environnement"`
3. Afin de pouvoir exécuter le fichier `monenv.sh` donnez à ce fichier les droits en exécution.
4. Testez votre script en tapant `monenv.sh` dans votre terminal.

### ↳ Exercice 9 ↴

Ajoutez la commande `echo "Mon home est: "$HOME` dans le fichier `monenv.sh`. Qu'affiche maintenant votre script ?

### ↳ Exercice 10 ↴

Modifiez le script `monenv.sh` afin qu'il affiche des informations sur votre environnement.

### ↳ Exercice 11 ↴

Ajoutez l'affichage de la date actuelle dans le script `monenv.sh`.