

TP n°4 - Scripts Shell

Exercice 1. La commande `seq`

Écrire un script `wet` qui accepte entre un et trois arguments entiers. Dans le cas où le nombre d'arguments est incorrect, le script affiche un message d'aide à l'utilisation

- avec un argument n , le script affiche les entiers de 1 à n ,
- avec deux arguments m et n , le script affiche les entiers de m à n ,
- avec trois arguments m , k et n , le script affiche les entiers de m à n par pas de k .

Voici un exemple d'utilisation :

```
$ wet 2 5 28
2
7
12
17
22
27
```

Exercice 2. Lignes Tatin

Écrire un script `tatin` qui prend son entrée soit dans un fichier dont le nom est passé en argument, soit sur son entrée standard s'il ne reçoit pas d'argument, et qui écrit sur la sortie les lignes de l'entrée avec tous les mots dans l'ordre inverse de leur apparition dans la ligne. Par exemple :

```
$ ./tatin.sh << EOF
! tarte la de pas C'est
: galactique voyageur du guide le dit le Comme
>> towel. a carry and Panic Don't <<
EOF
C'est pas de la tarte !
Comme le dit le guide du voyageur galactique :
<< Don't panic and carry a towel.
```

Exercice 3. Une histoire de notes.....

Dans le répertoire `notes` que vous obtiendrez en décompressant l'archive `notes.zip`, chaque fichier contient les notes d'un module, par exemple `3IF02.csv`, `3IF04.csv`, etc... Dans chaque fichier, chaque ligne est au format

`nom étudiant; note du module.`

1. Écrire un script shell qui affiche le nom du module suivi par le plus grand nombre d'étudiants.
2. Écrire un script shell qui affiche pour chaque module la note la plus élevée et la note la plus faible.
3. Écrire un script shell qui crée un seul fichier `Total.csv` pour tous les modules. Chaque ligne du fichier est du format
`nom étudiant ; module ; note du module.`
Le fichier `Total.csv` doit être organisé de sorte que toutes les notes d'un même étudiant soient en lignes consécutives.

Exercice 4. Un enseignant de TD dispose d'un fichier `eleves.txt` contenant les prénoms et noms de l'ensemble des étudiants du groupe de TD. Le fichier est de la forme suivante :

```
nom1:prenom1
nom2:prenom2
nom3:prenom3
```

1. Écrire un script shell qui crée un fichier `emails.txt` dans lequel chaque ligne contient l'adresse mail d'un étudiant du fichier `eleves.txt`, sous la forme `prenom.nom@etu.univ-orleans.fr`
2. En utilisant le fichier `emails.txt`, écrire *une commande bash* en une ligne de commande pour afficher la liste des étudiants sous le format

```
prenom1 nom1
prenom2 nom2
prenom3 nom3
```

Exercice 5. On possède un fichier `IP.csv` qui contient les informations sur l'inscription pédagogique des étudiants, où chaque ligne est du format suivant :

```
nom:prenom:inscIF01:inscIF02:inscIF03:inscIF04:inscIF05
```

où pour les cinq modules, `inscIF0i` est `oui` ou `non` pour l'inscription dans le module.
Écrire un script shell qui compte le nombre d'inscriptions pour le module `IF03`.

Exercice 6. Écrire un script shell `listedir.sh` qui permet d'afficher le contenu d'un répertoire en séparant les fichiers et les sous-répertoires. Le nom du répertoire est donné au script en ligne de commande. Exemple d'utilisation :

```
./listedir.sh /etc
affiche :

### fichiers dans /etc
/etc/aliases.db
/etc/auto_home
/etc/auto_master
/etc/bashrc
### repertoires dans /etc
/etc/apache2
/etc/defaults
```