



2e année
Bloc 1

Linux
Scripts

Pour cette activité vous utiliserez une machine virtuelle Ubuntu server en SSH. Vous rédigerez les scripts avec "nano".

1 - Rappels

Un script sur linux est un fichier avec une extension sh. Il doit toujours commencer par la ligne `#!/bin/bash`.

1 - Réalisez un script `bonjour.sh` qui affiche "Bonjour".

2 - Réalisez un script `bonjour2.sh` qui prend un paramètre un prénom et qui affiche "Bonjour [prénom]".

Résultat attendu :

```
osboxes@osboxes:~/test$ sh bonjour2.sh Pierre
Bonjour Pierre
```

2 - Les inputs

On peut demander à l'utilisateur qui exécute un script de rentrer une valeur que l'on peut stocker dans une variable avec le mot cle `read` suivi du nom de la variable.

=> testez de créer un script `testRead.sh` qui contient ce code :

```
echo "Bonjour quel est votre prénom ?"
```

```
read prenom
```

```
echo "Bonjour $prenom"
```

=> Réalisez un script `nomPrenom.sh` qui demande à l'utilisateur de saisir son prénom, puis son nom et qui affiche "Bonjour [nom] [Prénom]".

4 - Les fichiers

4.1 - Lecture de fichier

=> Créez un fichier texte `test.txt` qui contient plusieurs lignes.

=> Créez un script `lireFichier.sh` qui contient ce code :

```
while IFS= read -r line; do
```

```
    echo "$line"
```

```
done < test.txt
```

=> Exécutez-le

- Quelle est la valeur de `$line` à chaque itération de la boucle ?

4.2 - Ecriture de fichier

=> Créez un fichier vide `texte.txt`.

- => Créez un script appendText.sh qui contient ce code
- echo "Bonjour">texte.txt
- => exécutez-le
- => affichez texte.txt (commande cat)
 - Que se passe-t-il ?
- => Réexécutez ce code
 - Que se passe-t-il ?
- => Remplacez ">" du script par ">>" et exécutez le script plusieurs fois
 - Que se passe-t-il ?

5 - Gestion de chaînes de caractères

On peut séparer une chaîne de caractère en "morceaux" en utilisant un délimiteur avec la commande cut.

- => Créez un script qui contient le code suivant et exécutez-le :

```
texte = "voici des mots séparés par des virgules"
echo $texte | cut -d" " -f2
```

- => Modifiez -f2 en -f3 et réexécutez

- D'après vous que se passera t-il si on remplace la 2e ligne du script par :
- ```
echo $texte | cut -d"i" -f1
```

- Expliquez ce que vous avez compris du fonctionnement de la commande cut.

## 6 - Exercices

### 6.1 - Fichier nom, prenom

Réalisez un script ajoutNomPrenom.sh qui demande à l'utilisateur de saisir son prénom, puis son nom et qui ajoute une ligne dans un fichier personnes.txt.

Résultat attendu :

Fichier texte avant exécution :

```
osboxes@osboxes:~/test$ cat personnes.txt
Paul,Durand
François,Martin
osboxes@osboxes:~/test$
```

exécution du script :

```
osboxes@osboxes:~/test$ sh ajoutNomPrenom.sh
Quel est votre prénom ?
Pierre
Quel est votre nom ?
Dupont
```

Fichier texte après exécution :

```
osboxes@osboxes:~/test$ cat personnes.txt
Paul,Durand
François,Martin
Pierre,Dupont
osboxes@osboxes:~/test$
```

## 6.2 - Identification

=> Créez un répertoire Activiteldent et rendez-vous dedans.

=> Créez un fichier vide "users.txt"

=> Créez un script addUser.sh qui demande successivement à l'utilisateur son nom, son prénom, son email et son âge puis qui stocke ces infos dans users.txt (séparés par des virgules). (similaire à l'exercice précédent).

=> Créez un script getInfos.sh qui demande à l'utilisateur un nom, et un prénom. Si l'utilisateur est dans le fichier users.txt, affiche :

"[nom] [prénom] a [age] ans, son mail est [mail]".

Si il n'est pas dans le fichier afficher "[nom] [prenom] est inconnu".

## 6.3 - Identification avancée

=> Créez un deuxième script newUser.sh qui prendra en paramètre nom prénom âge et mail et qui créera directement l'utilisateur (sans poser de questions à l'utilisateur).

### Résultat attendu :

Fichier texte avant exécution :

```
osboxes@osboxes:~/test$ cat users.txt
Paul,Durand,paul.durand@gmail.com,24
François,Martin,fmartin@gmail.com,36
osboxes@osboxes:~/test$
```

Exécution du script :

```
osboxes@osboxes:~/test$ sh newUser.sh Martin Dupont martin@toto.fr 45
```

Fichier texte avant exécution :

```
osboxes@osboxes:~/test$ cat users.txt
«Paul,Durand,paul.durand@gmail.com,24
«François,Martin,fmartin@gmail.com,36
Martin,Dupont,martin@toto.fr,45
```

=> Modifier addUser.sh pour qu'il appelle newUser.sh avec les paramètres saisis par l'utilisateur

=> Modifiez getInfos.sh pour que dans le cas où l'utilisateur n'est pas dans la liste il propose de l'ajouter comme ceci :

"[nom] [prenom] est inconnu, voulez-vous l'ajouter ? (o/n)"

Si l'utilisateur répond "o", demande le mail et l'age et appelle newUser.sh avec les bons paramètres.