# 2LADM – TP105 Shells et script Shell

## PARTIE 1 : Gestion des fichiers et répertoires avec un script Bash

**Objectif :**

L'objectif de cet exercice est de créer un script Bash permettant de gérer les fichiers et répertoires sur un système Linux. Ce script doit offrir un menu interactif permettant à l'utilisateur d'effectuer différentes opérations sur les fichiers et répertoires.

Les fonctionnalités à implémenter sont les suivantes :

1. Créer un répertoire.
2. Créer un fichier dans ce répertoire.
3. Écrire du contenu dans le fichier.
4. Afficher le contenu du fichier.
5. Supprimer le fichier et le répertoire.

**Instructions :**

1. **Création du script de base** : Créez un fichier nommé file\_manager.sh.
   * Ajoutez un **shebang** pour indiquer que le script doit être exécuté avec Bash :

|  |
| --- |
| #!/bin/bash |

1. **Ajouter un menu interactif** : Créez un menu qui se répète jusqu'à ce que l'utilisateur choisisse de quitter. Le menu doit afficher les options suivantes :
   * Créer un répertoire
   * Créer un fichier dans un répertoire
   * Écrire du contenu dans un fichier
   * Afficher le contenu du fichier
   * Supprimer le fichier et le répertoire
   * Quitter
2. **Implémenter les fonctionnalités** : Chaque option du menu doit exécuter une fonction spécifique :

a. **Créer un répertoire** :

* + Demander à l'utilisateur de saisir le nom du répertoire.
  + Créer le répertoire avec la commande mkdir.

b. **Créer un fichier dans ce répertoire** :

* + Demander à l'utilisateur de saisir le nom du fichier.
  + Créer le fichier avec la commande touch.

c. **Écrire du contenu dans le fichier** :

* + Demander à l'utilisateur de saisir le contenu à écrire.
  + Utiliser la commande echo ou printf pour écrire dans le fichier.

d. **Afficher le contenu du fichier** :

* + Afficher le contenu du fichier avec la commande cat.

e. **Supprimer le fichier et le répertoire** :

* + Supprimer d'abord le fichier avec rm.
  + Puis supprimer le répertoire avec rmdir ou rm -r.

1. **Tester le script** :
   * Rendre le script exécutable avec la commande chmod +x file\_manager.sh.
   * Exécuter le script avec ./file\_manager.sh et tester chaque fonctionnalité.

**Exemple de script : file\_manager.sh**

|  |
| --- |
|  |

**Explication détaillée du script**

1. **Affichage du menu** :
   * La fonction afficher\_menu affiche les options disponibles.
   * Chaque option correspond à une fonctionnalité distincte.
2. **Création du répertoire** :
   * mkdir est utilisé pour créer un répertoire. Une vérification est effectuée pour s'assurer que le répertoire n'existe pas déjà.
3. **Création du fichier** :
   * Le script demande d'abord le répertoire où créer le fichier, puis utilise touch pour créer le fichier spécifié.
4. **Écriture dans le fichier** :
   * Le script permet d'écrire du contenu dans un fichier avec la commande echo.
5. **Affichage du contenu du fichier** :
   * La commande cat est utilisée pour afficher le contenu du fichier.
6. **Suppression du fichier et du répertoire** :
   * Le fichier est supprimé avec rm et le répertoire avec rmdir. Si le répertoire contient des fichiers, vous pouvez utiliser rm -r à la place de rmdir pour supprimer le répertoire non vide.
7. **Menu interactif** :
   * Le menu continue de s'afficher tant que l'utilisateur ne choisit pas de quitter.

**Test du script**

1. **Rendre le script exécutable** :

|  |
| --- |
|  |

1. **Exécuter le script** :

Copier le code

|  |
| --- |
|  |

1. **Interaction** :
   * Testez les différentes fonctionnalités (création de répertoire, création de fichier, écriture, etc.) en suivant les instructions à l'écran.

**Exemple de sortie :**

|  |
| --- |
|  |

Ce TP permet de maîtriser les bases de la gestion de fichiers et répertoires via un script Bash, tout en intégrant les notions de variables d'environnement et de personnalisation du shell.

## PARTIE 2 : Personnalisation et gestion de l’environnement du shell

**Objectifs :**

* Comprendre la gestion des variables d’environnement (e.g. PATH).
* Créer et personnaliser des fonctions Bash pour automatiser des tâches courantes.
* Mettre à jour les répertoires de base pour de nouveaux utilisateurs.

**Exercice 1 : Gestion des variables d’environnement dans le shell**

**Objectif :**

L'objectif de cet exercice est de comprendre et de manipuler les variables d’environnement utilisées dans le shell, telles que PATH, ainsi que de configurer des variables personnalisées.

**Instructions :**

1. **Définir et afficher une variable d'environnement** :
   * Ouvrir un terminal et définir une nouvelle variable d’environnement appelée MON\_REPERTOIRE pointant vers un répertoire de votre choix.
   * Vérifier si la variable a bien été définie en utilisant la commande echo.
2. **Modifier la variable PATH** :
   * Ajouter un nouveau répertoire au PATH pour que le shell puisse y trouver de nouveaux exécutables. Par exemple, ajoutez ~/bin à votre PATH.
   * Vérifier que la modification a été effectuée correctement avec la commande echo $PATH.
3. **Exporter une variable d’environnement** :
   * Créez une variable USER\_VAR et exportez-la afin qu’elle soit accessible dans les sous-shells.
   * Vérifiez l’exportation avec echo $USER\_VAR.
4. **Supprimer une variable d’environnement** :
   * Utilisez la commande unset pour supprimer la variable USER\_VAR et vérifiez qu’elle n’est plus présente avec echo $USER\_VAR.

**Exemple :**

|  |
| --- |
|  |

**Exercice 2 : Création d’un script Bash pour automatiser les tâches courantes**

**Objectif :**

L'objectif est d'écrire un script Bash qui inclut des fonctions pour gérer des répertoires et des fichiers, ainsi que pour personnaliser l'environnement shell à l'aide de variables d'environnement.

**Instructions :**

1. **Création du script** :
   * Créez un fichier nommé env\_manager.sh.
   * Ajoutez un shebang au début du fichier pour spécifier que le script doit être exécuté avec Bash.
2. **Ajout d'une fonction pour afficher les variables d’environnement** :
   * Créez une fonction afficher\_variables() qui affiche certaines variables d’environnement comme PATH, USER, et MON\_REPERTOIRE.
3. **Création de fonctions pour la gestion des répertoires** :
   * Créez une fonction creer\_repertoire() qui prend en argument un nom de répertoire et crée ce répertoire si il n’existe pas.
   * Créez une fonction supprimer\_repertoire() qui prend en argument un nom de répertoire et le supprime s’il existe.
4. **Ajout d’une fonction pour mettre à jour le fichier de configuration du shell** :
   * Créez une fonction ajouter\_variable\_bashrc() qui ajoute une nouvelle variable d’environnement dans le fichier ~/.bashrc. Vous pourrez l’utiliser pour ajouter un répertoire personnalisé dans le PATH.
5. **Test du script** :
   * Rendez le script exécutable avec la commande ………………….
   * Exécutez-le en utilisant …………………….

**Exemple de code :**

|  |
| --- |
|  |

**Test et exécution :**

1. **Rendre le script exécutable** :

|  |
| --- |
|  |

1. **Exécuter le script** :

|  |
| --- |
|  |

**Exemple d'exécution :**

|  |
| --- |
|  |

**Exercice 3 : Mise à jour des répertoires de base pour les nouveaux utilisateurs**

**Objectif :**

L'objectif est de personnaliser l’environnement des nouveaux utilisateurs en mettant à jour les fichiers de configuration du shell (e.g. ~/.bashrc et ~/.profile) pour inclure des variables d'environnement et des fonctions utiles.

**Instructions :**

1. **Mettre à jour les fichiers de configuration des nouveaux utilisateurs** :
   * Créez un script update\_user\_env.sh qui :
     + Ajoute une variable personnalisée à ~/.bashrc et ~/.profile pour un nouveau compte utilisateur.
     + Ajoute un alias de commande dans ~/.bashrc.
2. **Vérifier que les modifications sont appliquées** :
   * Testez que les variables et alias sont présents lors de la connexion d’un nouveau shell utilisateur.

**CODE :**

|  |
| --- |
|  |

**Rendu =>**

**M’envoyer votre travail à l’adresse** [**micheline.ekoue@ecole-hexagone.com**](mailto:micheline.ekoue@ecole-hexagone.com)

**Object : [B2][Versailles][2LADM] TP105 – Shell**

Nom du fichier **NOMPrenom- TP105.zip ou .tar**

exemple => **EKOUEMicheline-TP105-Shell.zip à l’intérieur**

**le ou les fichiers *EKOUEMicheline \_xxxxxx.sh***